

# MIX-net を用いた電子投票

Electronic Voting Schemes using MIX-net

阿部 正幸<sup>†1</sup>

Masayuki ABE

藤岡 淳<sup>†1</sup>

Atsushi FUJIOKA

星野 文学<sup>†1</sup>

Fumitaka HOSHINO

大久保美也子<sup>†2</sup>

Miyako OOKUBO

鈴木幸太郎<sup>†1</sup>

Koutarou SUZUKI

## あらまし

本論文では、電子投票方式において匿名性を保つために重要な役割を果たすMIX-netについて概説し、それを用いた電子投票方式について述べる。MIX-netについては、長さ不変性を満たすMIX-netと、全体検証可能性を満たすMIX-netを提案する。また、投票の買収・脅迫を防止するために重要な無証拠性について概説し、これを実現する電子投票方式を提案する。

## Abstract

This paper explains MIX-net that is a scheme for cryptographically realizing of anonymous channels, and discusses anonymous electronic voting schemes using MIX-net. Two MIX-nets are proposed: one satisfies length invariance and the other satisfies universal verifiability. Also proposed is a voting scheme that achieves receipt-freeness, which prevents vote buying or coercing.

## 1 まえがき

ここ数年、急速に発展したインターネットを中心としたネットワーク社会の到来が一般社会へ大きな影響を見せ始めており、その影響範囲は、電子商取引に代表されるような経済的な活動から、社会的・政治的なものへと広がりつつある。例えば、今年(2000年)3月にアリゾナ州民主党大統領選挙予備戦でインターネットを用いた投票が正式な投票手段として利用されたことは、その象徴的な例であり<sup>(1)</sup>、7月の沖縄サミットでも電子投票のデモが行われている。

ネットワークを利用した電子化された投票方式、すなわち、電子投票では、記名投票に関しては、たとえ電子化されたものであっても、通常の認証技法を用いれば、容易に実現できる。しかし、無記名投票を電子化した場合には大きな課題が存在する。それは、いかにして投票者のプライバシーを保護するか、すなわち、匿名性(anonymity)の保証という点である。以下、本論文では無記名電子投票を扱う。

電子投票方式において、匿名性を保証する手段としては、

様々な方式が提案されているが、実用性の面から現実解として有力なものは、現在、以下の2方式であろう<sup>(注1)</sup>。

- ・ブラインド署名<sup>\*1</sup>を用いたもの
- ・MIX-netを用いたもの

ブラインド署名を用いる方式は、物理的な匿名通信路<sup>(注2)</sup>の存在を仮定しており、匿名通信路とブラインド署名を組み合わせることで、無記名投票を実現している。その代表的な方式がFOO方式<sup>(2)~(5)</sup>である。このFOO方式は、単純な構造、現実の選挙との類似性から理解が容易であり、世界中で実装され、試用されている(FOO方式の詳細については、5.1節において述べる)。

<sup>(注1)</sup> これら電子投票の詳細については文献(3)および(28)を参照されたい。

<sup>(注2)</sup> 実際には、物理的な仮定である匿名通信路をMIX-netで実現することも多い。

\*1 ブラインド署名

文書作成者と署名作成者がいるプロトコルを実行することにより、①文書作成者はある文書に対する署名作成者のデジタル署名を得て、②署名作成者にはその文書内容が不明となる、のような署名。電子現金や電子投票などに用いられる。

<sup>†1</sup> NTT情報流通プラットフォーム研究所 Information Sharing Platform Laboratories, NTT

<sup>†2</sup> NTT東日本法人営業本部 Business Communications Headquarters, NTT EAST

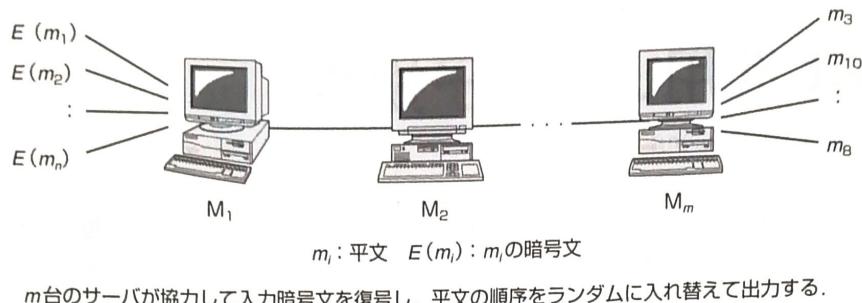


図1 MIX-netの概念図

FOO方式の実装例として有名なものはSt. LuisのWashington大学で実装された“Sensus”であり、これは、アメリカ国内ではフリーソフトウェアとして配布されている<sup>(6)(7)</sup>。また、MITでもJavaを用いた方式(EVOXと呼ばれている)が実装されている<sup>(8)</sup>。

また、ドイツのOsnabrück大学でも、Sensusを基に実装したプログラムを学生議会の選挙に用いている<sup>(9)</sup>。この選挙はドイツの法律で規定されたものであり、また、電子投票の実施についても州の承認も受けているとのことである。これを根拠に、彼らは、この投票を世界初の合法的なインターネット投票であると主張している<sup>(注3)</sup>。

日本国内でも、NTTと5大学(中央大学、東海大学、東京大学、横浜国立大学、早稲田大学(五十音順))との間で、インターネットを用いた電子投票の共同実験が行われている。

一方、匿名性を保証する通信路であるMIX-netを用いて電子投票を行う方式も実用性において有力である。MIX-netとはMIXと呼ばれる複数のサーバが連係して、送信者(投票者)と通信内容(投票内容)の関連性を取り除くことで、匿名性を保証するものである。

本論文では、まず、2章において、プライバシを保護するための仕掛けであるMIX-netについて述べ、どのように匿名性が保証されるかについて概説する。次に、MIX-netの実用性と安全性に着目し、それぞれの点において有利な方式として、3章で、長さが不变なMIX-netを、4章で、全体検証可能なMIX-netを提案する。また、5章において、広い意味でのプライバシとして、票の買収や投票者に対する脅迫を防止する仕組み、無証拠性(receipt-freeness)について概説し、これを実現する手法を述べる。

## 2 MIX-net

インターネットのように匿名性を有しないネットワーク上で、匿名通信路を実現する手段の1つが、1981年UC Berkeley

<sup>(注3)</sup> これ以前にPrinceton大学でも実施されたようである<sup>(29)</sup>。

のD.Chaumによって提案されたMIX-netである<sup>(10)(11)</sup>。MIX-netは当初から、電子メールの交換や電子投票などトラッキング解析を避ける必要のあるアプリケーション中の使用を目的として提案された。

MIX-netは $m$ 個のサーバ $M_1, \dots, M_m$ が図1のように直列的に記名通信路で接続されたシステムである。

MIX-netへの入力は、暗号文のリストであり、出力は各入力暗号文を復号して得られる平文をランダムに並べ替えたリストである。復号鍵と、並べ替えの順序が漏洩しないかぎり、この2つのリストから、どの平文がどの暗号文に対応するものであるかを言い当てることはできない。したがって、各暗号文が別々のユーザによって作成されたものと考えれば、どの平文がどのユーザのものかを判断することができず、集団の中で区別がつかないという意味での匿名性が保たれることになる。

公開鍵暗号を用いてこれを実装する場合について述べる。各サーバは公開鍵と秘密鍵を持つものとし、サーバ $M_i$ の公開鍵によってメッセージ $msg$ を暗号化する手順を $\epsilon_i^{asy}(msg)$ と書く。ここでは確率暗号<sup>\*2</sup>、すなわち、1つの平文に対応する暗号文が複数存在するような公開鍵暗号方式を用いるものとする。

各ユーザは、送信すべき平文 $msg$ を $\epsilon_1^{asy}(\epsilon_2^{asy}(\dots \epsilon_m^{asy}(msg)\dots))$ のように、各サーバの公開鍵を用いて多重に暗号化し、 $M_1$ へ送付する。

$M_1$ は、暗号化された入力文を複数のユーザから受信した後、各入力暗号文を $M_1$ の秘密鍵で復号することによって $C' = \epsilon_2^{asy}(\dots \epsilon_m^{asy}(msg)\dots)$ を得る。 $M_1$ はそれぞれの入力文から得られた $C'$ の順序をランダムに置換し、 $M_2$ へ送付する。確率暗号では、1つの平文が複数の暗号文に対応しうるので、 $M_2$ へ送付された各 $C'$ が $M_1$ への入力のどのメッセージに関する復号結果であるのかを判別できない。以下、 $M_2, \dots, M_m$ も同様の処理を繰り返す。結局、最後のサーバまでに各暗号文は $m$ 回復号され、 $m$ 重に暗号化されたメッセージ $msg$ が復号されることになる。

<sup>\*2</sup> 確率暗号

1つの平文に対して複数の暗号文が存在するような暗号系。暗号化手順中で乱数を用いているため、同じ平文を複数回暗号化しても同一の暗号文になりにくいので、辞書攻撃等に強い。平文の長さに対して、暗号文が2~3倍程度の長さになる方式が多い。

$M_m$  は、復号されてランダムに置換されたメッセージのリストを公開する。少なくとも 1 つのサーバが置換の順序を秘密にしておくことにより、各利用者が  $M_i$  へ入力した入力暗号文と、 $M_m$  が output したメッセージとの関連は隠蔽されるので、この MIX-net は匿名通信路として機能する。

以上のような MIX-net を用いると、簡易な電子投票システムを構築できる。説明を簡略化するため、公開掲示板を利用できるものとする。公開掲示板は、あらかじめ決められた利用者が決められた場所にデータを書き込むことができ、誰もそのデータを消去したり上書きしたりできないという機能を提供するサーバと考えてよい。まず、各投票者は投票内容を平文  $msg$  とする。前述の手順に従って多重に暗号化した暗号文を公開掲示板へ書き込む。その際、選挙管理者は、投票者が本人であることと、まだ投票済みでないことを確認する。投票者の本人性はデジタル署名や、パスワードなどを用いた本人認証方法を適用すればよい。投票締切時刻を過ぎたら、掲示板に投票された暗号文のリストを MIX-net へ入力し、対応する平文  $msg$  のリストを得、これを掲示板で公開する。開票者は、公開された平文（投票内容）を集計する。

上記のような構成は、MIX サーバが故障、あるいは意図的な不正を行った場合に弱い（頑健でない）方式である。例えば、最後尾の MIX サーバは、復号結果とは全く異なる平文を出力して集計結果を意のままにすることができる。したがって、投票結果の正当性を検証する手段が必要である。最も単純な検証方法として、各投票者が、投票文に乱数を含ませておき、最終的に公開された平文のリスト中に、その乱数を含む投票文があることを確認するという方法がある。このように、各投票者が自分の投票が公開された平文のリスト中に含まれているという事実だけを納得できるような検証方法を提供する投票方式を個別検証可能（individually verifiable）な方式と呼ぶ。大規模な投票では、実際に全投票者が投票締切後に検証を行うことはないと考えられるので、個別検証可能な方式では、MIX サーバによる不正が本当にかかったことを保証することはできない。より高度な検証性を提供する方式として、全体検証可能（universally verifiable）な方式がある<sup>(12)</sup>。これは、各 MIX サーバが正しく動作したことをゼロ知識証明<sup>\*3</sup>を用いて任意の第三者に証明する方式である。このような方式では、各投票者が検証を行う必要はなく、選挙監視団体の代表など、任意の監視者が、開票後にいつでも検証を行うことができる。従来、そのようなゼロ知識証明を行うことは膨大な計算量、通信量を必要とするため、現実的ではないと考えられていたが、近

年のセキュリティ技術の進歩により、数万人程度の投票ならば、現実的な処理時間で実現することができる方式が提案されている。4 章で効率的に全体検証が可能な方式を提案する。

MIX-net を用いた電子投票の特徴の 1 つは、最終的に投票内容が開示されるため、記述できる投票内容に制約がないことである（投票内容が Yes/No に制限されるような電子投票方式も多い<sup>(13)~(15)</sup>）。そのため、MIX-net は電子投票以外にも、アンケートや一般的な匿名通信路の代替に適用できる。ところが、公開鍵暗号を用いているため、送信できる平文の長さは、最後のサーバの公開鍵の長さ程度に制限されてしまう。長い平文を扱う効率的な方法は、公開鍵暗号と共通鍵暗号を組み合わせた Hybrid 暗号を用いることである。Hybrid 暗号は、平文を共通鍵暗号で暗号化し、そこで用いた共通鍵を公開鍵暗号で暗号化する方式と考えてよい。暗号化された平文と暗号化された共通鍵の対が Hybrid 暗号の暗号文と見なされる。単純に Hybrid 暗号を MIX-net に適用すると、ユーザは、共通鍵暗号の共通鍵を、サーバの数だけ多重に暗号化して送らねばならないので、ユーザがつくる多重化された暗号文の長さが MIX サーバの数に比例して増加するという欠点が生じる。3 章では、Hybrid 暗号を用いた MIX-net において、ユーザのつくる暗号文の長さが MIX サーバの数に依存しない、長さ不变（length-invariant）な方式を提案する。

### 3 暗号文の長さが不变な Hybrid MIX-net

本章では長いメッセージを効率よく通信できる MIX-net を提案する<sup>(16)</sup>。提案方式（以下、方式 1 と呼ぶ）は公開鍵暗号による鍵配達とその鍵による共通鍵暗号を組み合わせた Hybrid 型の MIX-net である。方式 1 はさらに、暗号文の長さが MIX-net のサーバ数によって増加しないという特徴（長さ不变性）を持つ。

MIX-net のサーバ数を  $m$  台とし、 $M_i$  を  $i$  番目の MIX サーバとする。また、入力暗号文作成者（ユーザ）の数を  $n$  とし、 $U$  で代表させる。

$p, q$  を  $q \mid p - 1$  を満たす大きな素数とする。 $G_q$  は  $Z_p^*$  の位数  $q$  の部分群とし、 $g$  を  $G_q$  の元とする。これら 3 つ組  $(p, q, g)$  は公開パラメータとする。以下本論文では、特に記述のない限り、これらの記号を用い、また、すべての算術は  $Z_p$  上で行うものとする。

<公開パラメータ>  $(p, q, g)$

$(\epsilon^{sym}, D^{sym}, KSPC, MSPC, CSPC)$  を共通鍵暗号とする。ここで、 $KSPC$  は鍵空間、 $MSPC$  はメッセージ空間、 $CSPC$  は暗号文空間を示す。 $\epsilon^{sym}$  および  $D^{sym}$  はそれぞれ、暗号化アルゴリズム、復号アルゴリズムを示す。 $\epsilon_K^{sym}(x)$  は平文  $x$  を鍵  $K$  で暗号化

\*3 ゼロ知識証明

ある情報がある性質を満たすことを、その性質が満たされるという事実以外の一切の情報を漏らさずに証明すること、およびその証明手法。例えば、パスワードを秘密にしたまま、パスワードを知っているという事実のみを相手に納得させることができる。

した出力値であり、 $D_K^{\text{sym}}(x)$ は平文  $x$  を鍵  $K$  で復号した出力値である。

方式1に用いる共通鍵暗号 ( $\varepsilon_K^{\text{sym}}$ ,  $D_K^{\text{sym}}$ ,  $KSPC$ ,  $MSPC$ ,  $CSPC$ ) は、 $MSPC = CSPC$  であるとする。例えば、Triple DES 暗号を CFB や OFB モードで使用した場合にこのような性質が得られる。

$H : G_q \rightarrow KSPC$  を一方向性ハッシュ関数とする。 $\Pi_n$  を  $n$ 次の置換  $\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$  全体の集合とする。

[サーバの鍵生成] 各サーバは  $i = 1, \dots, m$  の順に次の手順を実行する。サーバ  $M_i$  は、秘密鍵  $(a_i, x_i) \in {}_v Z_q^2$  を選び、 $h_i := h_{i-1}^{a_i}(h_0 := g)$ ,  $y_i := h_i^{x_i}$  として、公開鍵  $(y_i, h_i)$  を計算して公開する。

<サーバ  $M_i$  の秘密鍵>  $(a_i, x_i)$

<サーバ  $M_i$  の公開鍵>  $(y_i, h_i)$

[ユーザの暗号化] ユーザ  $U$  は、メッセージ  $msg \in MSPC$  に対する暗号文

$$C_0 := (G_0, E_0) = (g^r, \varepsilon_{K_1}^{\text{sym}}(\dots \varepsilon_{K_m}^{\text{sym}}(msg) \dots))$$

を生成する。ただし、 $r \in {}_v Z_q$ 、および  $K_i = H(y_i^r)$  とする。また、 $\varepsilon_{K_1}^{\text{sym}}(\dots \varepsilon_{K_m}^{\text{sym}}(msg) \dots)$  は共通鍵  $K_i = H(y_i^r)$  ( $i = 1, \dots, m$ ) を用いて、多重に暗号化した暗号文である。

[サーバ  $M_i$  の動作] サーバ  $M_i$  への入力は  $n$  個の暗号文からなるリスト  $L_{i-1}$  である（ただし、 $L_0$  は各ユーザが暗号化した MIX-net への  $n$  個の入力暗号文  $C_0$  である）。

まず、リスト  $L_{i-1}$  中の各暗号文  $C_{i-1}$  (=  $(G_{i-1}, E_{i-1})$ ) について以下の計算を行う。

$$K := H(G_{i-1}^{a_i x_i})$$

$$C_i := (G_{i-1}^{a_i}, D_K^{\text{sym}}(E_{i-1}))$$

（最終サーバ  $M_m$  は  $C_m = D_K^{\text{sym}}(E_{m-1})$  とする。）

上記の手順で得られた  $n$  個の  $C_i$  からなるリストの各エントリを、ランダムに選んだ置換  $\pi_i \in {}_v \Pi_n$  に従って置換してできるリストを  $L_i$  として出力する。

すべての  $M_i$  が正常に動作した場合、すべての  $i = 1, \dots, m$  において、 $L_{i-1}$  中のすべてのエントリ  $(G_{i-1}, E_{i-1})$  に対し

$$G_{i-1}^{a_i x_i} = g^{r a_i \dots a_i x_i} = y_i^r$$

が成り立ち、よって、 $H(G_{i-1}^{a_i x_i}) = K_i$  となるので、各サーバは正しい共通鍵を得ることができる。したがって、出力リスト  $L_m$  は入力リスト  $L_0$  中の各暗号文を復号して得られる平文  $msg$  からなる。

方式1の通信量は、送信メッセージ  $msg$  を共通鍵暗号で多段に暗号化したもの  $E_i$  と固定長の鍵情報  $G_i$  のそれぞれのビット

数の和があるので、暗号文の長さがサーバ数に依存せず、送信メッセージのビット数に公開鍵  $p$  のビット数を加えたものとなることが分かる。

また、方式1のユーザの計算量はサーバの数に依存し、その増加に伴い線形に増加してしまうが、 $q$  のビット数を小さくできるため、サーバ数が実用的な範囲（11台以下）においては、むしろ従来法（例えば、PIK法<sup>(17)</sup>）よりも計算量が少なくなる。

方式1の安全性については、ユーザ、サーバ両者に対して受動的（passive）<sup>\*4</sup>な攻撃者、および、ユーザに対して能動的（active）<sup>\*5</sup>でありサーバに対して受動的である攻撃者に対する安全性を、整数論の問題に帰着することにより、数学的に証明している。

#### 4 全体検証可能な MIX-net

本章では、置換網（permutation network）を利用して、比較的少ない計算量で全体検証を可能とする MIX-net の構成法について紹介する。以下では、全体検証の技術についてのみ記述するが、検証結果が不正となった場合に不正を行ったサーバを排除し、正常な出力を回復するための技術も開発されている<sup>(18)(19)</sup>。

システムの安全性を決めるパラメータとして、 $m$  台の MIX サーバのうち、許容できる不正サーバの台数の上限値  $t$  を  $t < m/2$  となるように定める（多数決による不正者の排除と正常な出力の回復のため、不正者の上限  $t$  を全体の半数未満とする必要がある）。サーバ  $M_i$  は秘密鍵  $x_i$  と、対応する公開鍵  $y_i (= g^{x_i})$  を持つ。MIX-net 全体の公開鍵を  $y (= y_1 \cdot y_2 \cdots y_m)$  とし、 $y$  を公開してすべての投票者が利用できるようとする。

MIX-net への入力は、El Gamal 暗号で暗号化された暗号文のリストとする。すなわち、平文  $msg$  に対して乱数  $s$  を選び、暗号文  $(M, G)$  を  $(M, G) = (msg \cdot y^s, g^s)$  のように作成する。

以下で紹介する MIX-net は 2 段階の処理で構成される。

[ランダム化置換段階] 各サーバが入力暗号文を、同じ平文を持つ別のランダムな暗号文となるように変換し、その順序をランダムに入れ替える段階

[復号段階] 各サーバがランダム化置換段階の処理の正当性を相互に検証し、正しいと結論した後、ランダム化された暗号文を復号する段階

以下、各段階について説明する。

\*4 受動的な攻撃

定められた通信手順を逸脱せずに、受信したデータのみから、（秘密情報の入手などの）攻撃を行うこと。安全なプロトコルは、少なくとも受動的な攻撃は防止できなければならない。

\*5 能動的な攻撃

定められた通信手順を逸脱し、不正なデータを送信するなどして、（秘密情報の入手や誤った結果を出力させるなどの）攻撃を行うこと。能動的な攻撃は、受動的な攻撃よりも強力な攻撃である。安全なプロトコルは、受動的な攻撃だけでなく、能動的な攻撃も防止できることが望ましい。

#### 4.1 ランダム化置換段階

まず、ランダム化置換段階の基礎となる置換網について説明する。置換網とは、 $\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$ なる任意の置換 $\pi$ が与えられ、入力 $(I_1, \dots, I_n)$ を置換して $(I_{\pi(1)}, \dots, I_{\pi(n)})$ を出力する回路あるいは通信網である。以下、本論文では説明を容易にするため、 $n$ を2のべき数とする。いま、2入力の置換網を考え、これを特に「切換ゲート」と呼ぶことにする。切換ゲートは、入力端子 $(I_0, I_1)$ と出力端子 $(O_0, O_1)$ を持ち、制御信号 $b$ に従って、 $b = 0$ のとき $(O_0 = I_0, O_1 = I_1)$ 、 $b = 1$ のとき $(O_0 = I_1, O_1 = I_0)$ のように出力を切り換えるゲートである。1つの切換ゲートによる遅延を1と考えると、以下の定理が成り立つ。

**定理**  $n = 2^k$ なる任意の $n$ に対して、 $n(k - 1) + 1$ 個の切換ゲートからなる、遅延時間 $2k - 1$ の $n$ 入力置換網が存在する。

証明および、置換網の具体的な構成方法は文献(20)を参照することとし、図2に4入力の置換網を例示する。5つの切換ゲートの設定により、任意の置換が行えることが分かる。図2中の $SG_1$ と $SG_2$ のように、同じ遅延で入力が届くゲートの組を段と呼ぶ。ただし、 $SG_5$ の上にある固定された結線部分も遅延を生じると考える。そのような部分を直結ゲートと呼ぶ。また、置換網の最終段にあるゲートを出力ゲートと呼ぶ（後述のように置換網が複数接続されている場合は、その最後の置換網の最終段にあるゲートを指す）。

いま、置換網の各切換ゲートにおいて、2つの入力文をそれぞれ別の値にランダムに変換するとともに、その順序をランダムに入れ換えて出力することによって、入力と出力の対応を隠蔽することを考える。El Gamal 暗号による1つの暗号文 $(M, G) = (msg \cdot y^s, g^s)$ と同じ平文を持つ別の暗号文 $(\tilde{M}, \tilde{G})$ に変換するには、乱数 $r$ を用いて、 $(\tilde{M}, \tilde{G}) = (My^r, Gg^r)$ とすればよい。このとき、 $(\tilde{M}, \tilde{G}) = (My^r, Gg^r) = (msg \cdot y^{s+r}, g^{s+r})$ が成り立

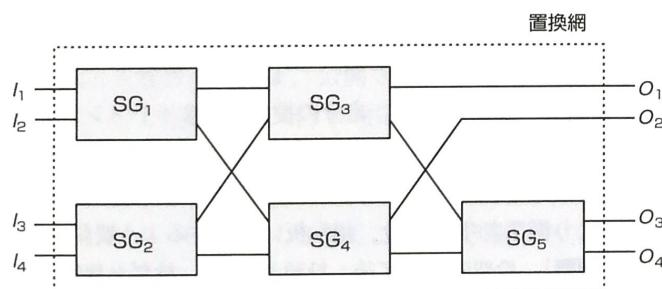


図2 4入力の置換網

(注4) 2つのメッセージと、対応する2つの暗号文があり、暗号文を順不同で与えられたとき、メッセージと暗号文の対応を判別することが困難であること。

つため、 $(M, G)$ と $(\tilde{M}, \tilde{G})$ は同じ平文を持つ異なる暗号文となる。El Gamal 暗号は識別困難(indistinguishable)<sup>(注4)</sup>なので、切換ゲートの入出力対応 $b$ を判定することは困難になる。

次に、切換ゲートは、処理が正しく行われたことを、使用した乱数 $r$ を漏洩することなく、ゼロ知識証明によって証明する（具体的な方法は後述）。このように、ランダム化およびランダム切換を行う切換ゲートからなる置換網を、ランダム化置換網と呼ぶことにする。

上記のランダム置換網を $m$ 台のサーバが順次実行することにより、直ちに検証可能なMix-netを構成することもできるが、ここでは、許容すべき不正サーバの数 $t$ 台に対して、 $O(mk)$ の効率を得る構成を説明する。まず、図3のような $t + 1$ 個のランダム置換網の直列接続を考える。ここで、 $m$ 台のサーバから $t + 1$ 台の代表サーバを選び、各代表サーバに図中の1つのランダム置換網の処理を実行させる。 $i$ 番目のランダム置換網を担当する代表サーバは、1番目から $i - 1$ 番目までの代表サーバによる置換網の処理を検証し、正しければ、自分の処理を実行する。この構成によって、不正サーバの数が $t$ 以下ならば、どのサーバが不正であっても、少なくとも1つのランダム置換網は正常なサーバによって実行されるため、入出力のどのような対応も実現でき、その対応は秘密に保たれる。したがって、ランダム化置換段階が正常に終了すれば、匿名性を保つことができる。

各サーバは、まず、置換 $\pi \in \Pi_n$ をランダムに選び、 $\pi$ を実現するように各切換ゲートの制御信号 $b$ を設定する。次に、各切換ゲートにおけるランダム化とランダム切換を以下のように実行する。切換ゲートへの入力暗号文を $I_i = (M_i, G_i)$ ,  $i = 0, 1$ とする。まず、乱数 $r_0, r_1 \in Z_q$ を選び、

$$O_b := (\tilde{M}_b, \tilde{G}_b) := (M_0 y^{r_b}, G_0 g^{r_b}) \quad (1)$$

$$O_{\bar{b}} := (\tilde{M}_{\bar{b}}, \tilde{G}_{\bar{b}}) := (M_1 y^{r_{\bar{b}}}, G_1 g^{r_{\bar{b}}}) \quad (2)$$

とする。次に、この処理を正しく行ったことを証明するため、

$$\log_y \tilde{M}_0 / M_0 = \log_g \tilde{G}_0 / G_0$$

$$\log_y \tilde{M}_1 / M_1 = \log_g \tilde{G}_1 / G_1$$

が成り立つ、または

$$\log_y \tilde{M}_0 / M_1 = \log_g \tilde{G}_0 / G_1$$

$$\log_y \tilde{M}_1 / M_0 = \log_g \tilde{G}_1 / G_0$$



図3 ランダム化置換網の接続

が成り立つ、ことをゼロ知識証明によって示す。証明プロトコルは、2つの離散対数が等しいことを証明する Chaum-Pedersen のプロトコル<sup>(21)</sup>と、2つの命題の少なくとも一方が成り立つことを証明する Cramer 等のプロトコル<sup>(22)</sup>を組み合わせて行う。以下にその証明プロトコル SP<sub>1</sub> を示す。

[SP<sub>1</sub>-1] 証明者（サーバ）は乱数  $w_0, w_1, z_{\bar{b}, 0}, z_{\bar{b}, 1}, e_{\bar{b}}$  を  $Z_q$  より選び、 $i = 0, 1$  について、

$$T_{b, i} := y^{w_i}$$

$$W_{b, i} := g^{w_i}$$

$$T_{\bar{b}, i} := y^{z_{\bar{b}, i}} (\tilde{M}_i / M_{\bar{b} \oplus i})^{e_{\bar{b}}}$$

$$W_{\bar{b}, i} := g^{z_{\bar{b}, i}} (\tilde{G}_i / G_{\bar{b} \oplus i})^{e_{\bar{b}}}$$

を計算し（⊕は排他的論理和を示す）、検証者に送付する。

[SP<sub>1</sub>-2] 検証者は  $c \in Z_q$  を選び、証明者へ送る。

[SP<sub>1</sub>-3] 証明者は、 $e_b := c - e_{\bar{b}} \pmod q$  を求め、 $i = 0, 1$  について  $z_{b, i} := w_i - e_b t_i \pmod q$  を計算し、検証者へ  $e_0, e_1, z_{0, 0}, z_{0, 1}, z_{1, 0}, z_{1, 1}$  を送付する。

[SP<sub>1</sub>-4] 検証者は、まず、 $e_b + e_{\bar{b}} = c \pmod q$  を確認し、

$$T_{i, j} = y^{z_{i, j}} (\tilde{M}_i / M_j)^{e_i} \quad (3)$$

$$W_{i, j} = g^{z_{i, j}} (\tilde{G}_i / G_j)^{e_i} \quad (4)$$

が  $i, j = 0, 1$  について成り立つならば True を出力する。そうでなければ False を出力する。

上記の構成により、各切換ゲートは以下の性質を満たすことが証明できる。

- ・プロトコル SP<sub>1</sub> で、正しい検証者が True を出力するならば、その切換ゲートの入力の復号結果と出力の復号結果は（順不同で）一致する。
- ・切換ゲートの入出力対応を  $1/2$  より有意に大きな確率で判定することは困難である。

#### 4.2 復号段階

前述のランダム化置換段階が終了すると、掲示板上には、El Gamal 暗号の暗号文  $\{\tilde{M}_i, \tilde{G}_i\}_{i=1,\dots,n}$  が現れる。この各暗号文を入力として、以下を実行する。

[P<sub>2</sub>-1] 各  $M_j$  は、 $D_{ji} := \tilde{G}_i^{x_j}$  を公開し、 $\log_g y_j = \log_{\tilde{G}_i} D_{ji}$  であることをゼロ知識証明で証明する。

[P<sub>2</sub>-2] 検証者は証明を検証する。次に、 $i = 1, \dots, n$  に対して

$$msg_i := \tilde{M}_i / \prod_{j=1}^m D_{ji}$$

を計算して平文  $msg_i$  を得る。

## 5 投票の買収・脅迫と無証拠性

今まで述べてきたように、MIX-net を用いることにより投票の匿名性を保つことができる。しかし、それだけでは公正な投票を実現するのに十分ではない。たとえ匿名性を満たしていたとしても、以下のような投票の買収（vote buying）・投票の脅迫（vote coercing）が行われると、投票の公正さが脅かされてしまう。

[投票の買収] 投票者は買収者の指定した投票行動を行い、その投票行動の“証拠”を買収者に見せ、買収者から報酬を受け取る。

[投票の脅迫] 投票者は脅迫者の指定した投票行動を行い、その投票行動の“証拠”を脅迫者に見せるよう、脅迫者から脅迫される。

そこで、投票の買収・脅迫を防ぎ、投票の公正さを担保するため、投票者が投票行動の“証拠”を残せないという性質が必要である。この性質を、投票の無証拠性（receipt-freeness）という<sup>(23)</sup>。無証拠性を満たす電子投票方式として、いくつかの方法が提案されている<sup>(13)(24)~(26)</sup>。

以下ではまず、FOO 方式を紹介し、この方法が“証拠”を残してしまうことを示す。次に、この方式を改良した方式<sup>(25)(26)</sup>を提案し、これが無証拠性を満たしていることを示す。

### 5.1 FOO 方式と投票の買収・脅迫

FOO 方式は、コミットメント<sup>\*6</sup>とブラインド署名とを組み合わせて用いており、以下の4段階で構成されている。

[1 認証] 投票者  $U_i$  は、秘密の乱数  $r_i$  を生成し、自分の投票  $v_i$  のコミットメント  $C_{r_i}(v_i)$  を生成する。そして、ブラインド署名を用いて、自分のコミットメントを明かすことなく、選挙管理者 A の署名  $\sigma_i = S_A(C_{r_i}(v_i))$  をもらう。

[2 投票] 投票者  $U_i$  は、コミットメントと署名  $(C_{r_i}(v_i), \sigma_i)$  を、匿名通信路により開票者 T に送り、投票する。開票者 T は、署名を検証し、コミットメントと署名を掲示板に公開する。

[3 確認] 投票者  $U_i$  は、自分の投票のコミットメントと署名が掲示板に公開されていることを確認する。もし掲示板にない場合は、コミットメントと署名を、再度匿名通信路により開票者 T に送り、掲示板に公開するよう要請する。

[4 開票] 投票時間終了後、投票者  $U_i$  は、投票と秘密の乱数  $(v_i, r_i)$  を、匿名通信路により開票者 T に送る。開票者 T

#### \*6 コミットメント

ビット列に対して計算される値で、①その値から元のビット列は分からず、②その値を変えずに元のビット列を変更することはできない、という性質を持つもの。あるビット列に対してコミットメントを計算し公開することを、そのビット列をコミットするという。あるコミットメントに対して元のビット列を公開しそのコミットメントと整合性があることを示すことを、そのコミットメントを開示するという。

は、 $(v_i, r_i)$ によりコミットメント  $C_{r_i}(v_i)$ を検証し、投票と秘密の乱数 $(v_i, r_i)$ を掲示板に公開する。

投票者は、ブラインド署名を用いることにより、選挙管理者Aに対して匿名性を保つことができ、匿名通信路を用いることにより、開票者Tに対して匿名性を保つことができる。また、投票には選挙管理者の署名がついているので、開票者は投票結果を改ざんすることはできない。しかし、投票者U<sub>i</sub>は、投票と秘密の乱数 $(v_i, r_i)$ を示し、コミットメント  $C_{r_i}(v_i)$ を検証することにより、自分の投票内容を証明できる（せざる得ない）ので、投票の買収（脅迫）が可能である。したがって、投票と秘密の乱数が“証拠”となってしまい、無証拠性を満たさない。

## 5.2 無証拠性を満たす方式

FOO方式において無証拠性を達成するために、投票 $v_i$ のコミットメントを使う代りに、投票 $v_i$ の落し戸付きコミットメント<sup>(27)</sup>を用いた方式（方式3と呼ぶ）を提案する。方式3<sup>(25)</sup>は、以下の4段階で構成されている。

- [1 認証] 投票者U<sub>i</sub>は、秘密の乱数 $r_i$ を生成し、自分の投票 $v_i$ の落し戸付きコミットメント  $C_{r_i}(v_i)$ を生成する。そして、ブラインド署名を用いて、自分のコミットメントを明かすことなく、選挙管理者Aの署名 $\sigma_i = S_A(C_{r_i}(v_i))$ をもらう。
- [2 投票] 投票者U<sub>i</sub>は、開票者Tに、コミットメントと署名 $(C_{r_i}(v_i), \sigma_i)$ を匿名通信路により送り、さらに、投票と秘密の乱数 $(v_i, r_i)$ を盗聴不能匿名通信路により送る。開票者Tは、コミットメントと署名を検証しそれを掲示板に公開する。

- [3 確認] 投票者U<sub>i</sub>は、自分の投票のコミットメントと署名が掲示板に公開されていることを確認する。もし掲示板にない場合は、コミットメントと署名を、再度匿名通信路により開票者Tに送り、掲示板に公開するよう要請する。

- [4 開票] 投票時間終了後、開票者Tは、投票 $v_i$ をランダムな順番で掲示板に公開し、ゼロ知識証明により秘密の乱数 $r_i$ を秘匿したまま、公開されているコミットメント  $C_{r_i}(v_i)$ と投票 $v_i$ が1対1に対応していることを証明する。ここで、落し戸付きコミットメントとは、落し戸の秘密鍵を知らない場合は（普通のコミットメントと同様に）一意にコミットしたメッセージにしか開くことができないが、落し戸の秘密鍵を知っている場合はコミットしたメッセージとは違うメッセージを開くことができる、特別なコミットメントである。この性質を利用すると、以下のように、開票者Tによる投票結果の改ざんを防ぎつつ、投票者U<sub>i</sub>に対しては投票の買収・脅迫を防止することができる。

[コミット] 投票者U<sub>i</sub>は、秘密鍵 $\alpha_i \in Z_q$ を生成し、公開鍵 $G_i = g^{\alpha_i} \in G_q$ を計算する。投票する際に、投票者U<sub>i</sub>は秘

密の乱数 $r_i \in Z_q$ を生成し、自分の投票 $v_i \in Z_q$ の落し戸付きコミットメント  $C_{r_i}(v_i) = (g^{v_i} G_i^{r_i}, G_i) \in (G_q)^2$ を生成し、それに選挙管理者の署名をもらう。

[秘密鍵 $\alpha_i$ を知らない開票者による開示] 秘密鍵 $\alpha_i$ を知らない開票者Tは、 $C_{r_i}(v_i)$ を $v_i$ のコミットメントとしてしか開示できない。

[秘密鍵 $\alpha_i$ を知っている投票者による開示] 秘密鍵 $\alpha_i$ を知っている投票者U<sub>i</sub>は、 $v_i + \alpha_i r_i = v'_i + \alpha_i v'_i \in Z_q$ なる $v'_i, r'_i \in Z_q$ を求めることができ、 $C_{r_i}(v_i)$ を $v_i, v'_i$ のどちらのコミットメントとしても開示することができる。

開票者Tは、コミットメントを正しい投票内容 $v_i$ としてしか開けず、したがって正しい投票内容 $v_i$ としてのゼロ知識証明しかできないため、投票内容を改ざんすることはできない。また、誰でも、公開された証明を検証することにより、コミットメントと投票内容が1対1で対応していることを確認することができる。

一方、乱数 $r_i$ は公開されないので、公開されたコミットメントと投票内容の個別の対応自体は秘匿されており、さらに、投票者U<sub>i</sub>はコミットメントを別の投票内容 $v'_i$ として開くことができるため、投票内容を $v'_i$ と偽ることができる。このため、買収者は投票者の投票内容の証明を信頼せず、投票者は脅迫者への投票内容の証明を偽ることができ、投票の買収・脅迫を防止することができる。

また、投票者U<sub>i</sub>は[2 投票]の時点で、投票と秘密の乱数 $(v_i, r_i)$ を開票者Tに送るため、投票後に投票内容 $v_i$ を変更することはできない<sup>(注5)</sup>。

しかし、この方法では次のような買収・脅迫を許してしまうため、無証拠性を達成できない。

・買収・脅迫者は秘密鍵 $\alpha_i$ と公開鍵 $G_i = g^{\alpha_i}$ を生成し、投票者U<sub>i</sub>にその $G_i$ を用いて投票させる。投票者U<sub>i</sub>は秘密鍵 $\alpha_i$ を知らないため、コミットメントを一意にコミットしたメッセージに開示することしかできないため、買収・脅迫者を偽ることができず、コミットメントが投票内容の“証拠”となってしまう。

これを防ぐためには、投票者が秘密鍵 $\alpha_i$ を知っていることを示すよう修正すればよい。そこで、次のように修正した、無証拠性を満たす方式（方式3' と呼ぶ）を提案する。方式3'<sup>(26)</sup>では、以下のように投票・開票を行う。

・投票者U<sub>i</sub>は、秘密鍵 $\alpha_i$ を知っていることを示すため、鍵をN個に秘密分散<sup>\*7</sup>  $\alpha_i = \sum_{j=1}^N \alpha_{i,j}$ して、鍵の秘密分散 $\alpha_{i,j}$

<sup>(注5)</sup> 開票者Tによる、投票の途中結果の漏洩の危険性はある。

\*7 密钥分散

秘密情報をいくつかの部分情報に分け、①その部分情報のうちある決められた数以上の部分情報が集まると元の秘密情報を復元できるが、②決められた数以下の部分情報からは元の秘密情報を復元できない、という性質を持つ情報分散法。

を盗聴不能匿名通信路により鍵登録者  $R_j$  に送る。鍵登録者  $R_j$  は、 $G_{i,j} = g^{\alpha_{i,j}}$  を公開する。開票者  $T$  は、 $G_i = \prod_{j=1}^N G_{i,j}$  を検証し、投票者が秘密鍵  $\alpha_i$  を知っていることを確認する。

この電子投票方式は無証拠性を満たしており、投票の買収・脅迫を防ぎ公正な投票を実現することができる。

## 6 あとがき

電子投票方式において匿名性を保つために重要な役割を果たす MIX-net について概説し、それを用いた電子投票方式について述べた。MIX-net については、長さ不变性を満たす MIX-net と、全体検証可能性を満たす MIX-net を提案した。また、投票の買収・脅迫を防止するために重要な無証拠性について概説し、これを実現する電子投票方式を提案した。

## 文 献

- (1) “初のネット投票がアリゾナ州で開始,” 朝日新聞夕刊, 2000年3月8日。
- (2) 太田：“単一の選挙管理者を用いた電子投票方式,” 昭和63年信学会春季全国大会, A-294, 1988.
- (3) 藤岡・藤崎・岡本：“電子投票方式,” NTT R&D, Vol. 44, No. 10, pp. 939-976, 1995.
- (4) A. Fujioka, T. Okamoto, and K. Ohta: “A Practical Secret Voting Scheme for Large Scale Elections,” Advances in Cryptology —AUSCRYPT’92, LNCS 718, Springer-Verlag, Berlin, pp. 244-251, 1993.
- (5) M. Ohkubo, F. Miura, M. Abe, A. Fujioka, and T. Okamoto: “An Improvement on a Practical Secret Voting Scheme,” Proceedings of ISW’99, LNCS 1729, Springer-Verlag, Berlin, pp. 225-234, 1999.
- (6) L. F. Cranor and R. K. Cytron: “Design and Implementation of a Practical Security-Conscious Electronic Polling System,” WUCS-96-02, Department of Computer Science, Washington University, St. Louis, 1996.
- (7) L. F. Cranor and R. K. Cytron: “Sensus: A Security-Conscious Electronic Polling System for the Internet,” Proceedings of the Hawaii International Conference on System Sciences, 1997.
- (8) M. A. Herschberg: “Secure Electronic Voting Over the World Wide Web,” Master Thesis, Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1997.
- (9) “2.2.2000 \* Fist Legal Vote Via Internet \*,” <http://www.internetwahlen.de/englisch.html>.
- (10) D. Chaum: “Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms,” Communications of the ACM, Vol. 24, No. 2, pp. 84-88, 1981.
- (11) B. Pfitzmann and A. Pfitzmann: “How to Break the Direct RSA Implementation of MIXes,” Advances in Cryptology —EUROCRYPT’89, LNCS 434, Springer-Verlag, Berlin, pp. 373-381, 1989.
- (12) M. Abe: “Universally verifiable mix-net with verification work independent of the number of mix-servers,” Advances in Cryptology—EUROCRYPT’98, Vol. 1403 of LNCS, pp. 437-447, Springer-Verlag, 1998.
- (13) R. Cramer, M. Franklin, B. Schoenmakers, and M. Yung: “Multi-authority secret-ballot elections with linear work,” Advances in Cryptology —EUROCRYPT’96, Vol. 1070 of LNCS, pp. 72-83, Springer-Verlag, 1996.
- (14) K. Sako and J. Kilian: “Receipt-Free Mix-type Voting Scheme,” Proceedings of EUROCRYPT’95, LNCS 921, pp. 393-403, 1995.
- (15) R. Cramer, R. Gennaro, and B. Schoenmakers: “A Secure and Optimally Efficient Multi-Authority Election Scheme,” Advances in Cryptology—EUROCRYPT’97, Vol. 1233 of LNCS, pp. 103-118, Springer-Verlag, 1997.
- (16) 大久保：“暗号文の長さが不变な Hybrid Mix,” 2000年暗号と情報セキュリティシンポジウム, SCIS2000, B29, 2000.
- (17) C. Park, K. Itoh, and K. Kurosawa: “All/Nothing Election Scheme and Anonymous Channel,” Advances in Cryptology —EUROCRYPT’93, LNCS 765, Springer-Verlag, Berlin, pp. 248-259, 1994.
- (18) M. Abe: “Mix-networks on Permutation Networks,” Advances in Cryptology -Asiacrypt’99, Vol. 1716 of LNCS, pp. 258-273, Springer-Verlag, 1999.
- (19) 星野・阿部：“Permutation Network を利用したより効率的な Mix-net,” 2000年暗号と情報セキュリティシンポジウム, SCIS2000, B28, 2000.
- (20) A. Waksman: “A permutation network,” Journal of the Association for Computing Machinery, 15(1), pp. 159-163, 1968.
- (21) D. L. Chaum and T. P. Pedersen: “Wallet databases with observers,” Advances in Cryptology—CRYPTO’92, Vol. 740 of LNCS, pp. 89-105, Springer-Verlag, 1993.
- (22) R. Cramer, I. Damgård, and B. Schoenmakers: “Proofs of partial knowledge and simplified design of witness hiding protocols,” Advances in Cryptology —CRYPTO’94, Vol. 839 of LNCS, pp. 174-187, Springer-Verlag, 1994.
- (23) J. Benaloh and D. Tuinstra: “Receipt-Free Secret-Ballot Elections,” Proceedings of STOC’94, pp. 544-553, 1994.
- (24) M. Jakobsson, K. Sako, and R. Impagliazzo: “Designated Verifier Proofs and Their Applications,” Proceedings of EUROCRYPT’96, LNCS 1070, pp. 143-154, 1996.
- (25) T. Okamoto: “An Electronic Voting Scheme,” Proceedings of IFIP’96, Advanced IT Tools, pp. 21-30, 1996.
- (26) T. Okamoto: “Receipt-Free Electronic Voting Schemes for Large Scale Elections,” Proceedings of 5th Secure Protocol, LNCS 1361, pp. 25-35, 1997.
- (27) G. Brassard, D. Chaum, and C. Crépeau: “Minimum Disclosure Proofs of Knowledge,” Journal of Computer and System Sciences, Vol. 37, No. 2, pp. 156-189, 1988.
- (28) 岡本：“図解 暗号と情報セキュリティ,” 日経BP社, 1998.
- (29) L. Cranor: “Electronic Voting Hot List,” <http://www.ccrc.wustl.edu/~lorracks/sensus/hotlist.html> (June, 2000).

## 著者紹介

### 阿部 正幸

NTT 情報流通プラットフォーム研究所研究主任

平成 4 年入社。主に、情報セキュリティの構築技術の研究開発に従事。現在、デジタル署名、電子投票方式等の研究開発に従事。

平成 2 年東京理科大学工学部電気工学科卒業。4 年同大学院電気工学科修士課程修了。

IACR・電子情報通信学会会員。

### 藤岡 淳

NTT 情報流通プラットフォーム研究所主任研究員

平成 2 年入社。主に、情報セキュリティの研究に従事。現在、ネットワーク・セキュリティの研究開発に従事。

昭和 60 年東京工業大学工学部電気・電子工学科卒業。62 年同大学院電気・電子工学専攻修士課程修了。平成 2 年同大学院電気・電子工学専攻博士後期課程修了。同年工学博士（同大学）。

電子情報通信学会・情報処理学会会員。

平成 5 年電子情報通信学会業績賞、小林記念特別賞受賞。

### 星野 文学

NTT 情報流通プラットフォーム研究所社員

平成 10 年入社。主に、高速公開鍵暗号の実装および応用の研究に従事。現在、全体検証可能な電子投票の実用化に従事。

平成 8 年東京大学工学部物理工学科卒業。10 年同大学院工学系研究科物理工学専攻修士課程修了。

### 大久保美也子

NTT 東日本 法人営業本部社員（元 NTT 情報流通プラットフォーム研究所社員）

平成 9 年入社。主に、情報セキュリティの研究に従事。現在、公開鍵暗号を中心とした、暗号プロトコルの研究に従事。

平成 7 年信州大学工学部電気電子工学科卒業。9 年同大学院工学系研究科電気電子工学専攻修士課程修了。

### 鈴木幸太郎

NTT 情報流通プラットフォーム研究所社員

平成 11 年入社。主に、情報セキュリティの研究開発に従事。

平成 6 年東京大学理学部数学科卒業。8 年同大学院数理科学研究科修士課程修了。11 年同大学院数理科学研究科博士課程修了。同年数理科学博士（同大学）。