社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

分割可能な複数票電子投票方式

松尾真一郎[†,††]　尾形わかは[††]

† 株式会社 NTT データ セキュリティ事業部
〒 212-0058 神奈川県川崎市幸区鹿島田 890-12 新川崎三井ビル West Tower 15F
†† 東京工業大学 大学院 理工学研究科
〒 152-8550 東京都目黒区大岡山 2-12-1
E-mail: †matsuosn@nttdata.co.jp, ††wakaha@crypt.ss.titech.ac.jp

あらまし　電子投票は暗号プロトコルの研究における有望な応用形態の 1 つと考えられており，これまでにも数多くの研究がなされている．既存の方式のほとんどは，一般的な選挙を主な用途として，投票者 1 人が 1 票を投じる形態の電子投票を実現したものであった．一方，現実の世界では，株主総会の投票や，持ち点を複数の選択肢に分配するアンケートのように，各投票者が個別の複数の票を持ち，場合によっては複数の選択肢に分配して投票を行うケースが存在する．既存の投票方式は，これらの場合に対して現実的ではない．本稿では，各投票者がそれぞれ異なる複数の票を持ち，複数の候補に分配して投票を行うことができる，新たな電子投票方式を提案する．本稿では，提案投票方式に対する要求条件を示し，次にプロトコルの提案を行い，提案プロトコルがこの要求条件を満たすことを示す．このプロトコルでは，各投票者の投票内容のプライバシは守られ，また，各投票者は不正な投票を行うことができない．提案プロトコルはゼロ知識証明が含まないため，他の投票方式に比べて実用的である．
キーワード　電子投票，多候補者選挙，プライバシ，秘密分散，準同型性

Divisible Voting Scheme

Shin'ichiro MATSUO[†,††] and Wakaha OGATA[††]

† NTT DATA Corporation, Security Business Division
Shin-Kawasaki Mitsui Bldg. West Tower 15F, 890-12, Kawshimada, Saiwai-ku, Kawasaki-Shi,
Kanagawa, 212-0058, Japan.
†† Tokyo Institute of Technology
2-12-1, O-okayama, Meguro-ku, Tokyo, 152-8550, Japan
E-mail: †matsuosn@nttdata.co.jp, ††wakaha@crypt.ss.titech.ac.jp

Abstract　Electronic voting is considered as one of major applications of cryptographic protocols and many researches are conducted in this area. Most of existing schemes are mainly for election in which each voter has one ballot. However, there are many cases in which each voter has multiple ballots, for example general meeting of stockholders. Existing schemes are not practical for these cases. In this paper, we propose a new voting scheme in which each voter has multiple ballots and votes divisible ballots for multiple candidates. We give some requirements for divisible voting schemes and show that our protocol fulfills them. In our protocol, privacy of each voter's choice holds and any voters cannot vote irregular ballots. Our protocol contains no interactive proof system. Thus, it is more efficient than other voting schemes.
Key words　Secure electronic voting, multi-way election, privacy, secret shaing and homomorphism

1. はじめに

1.1 背　　景

現在，電子マネー，電子入札，電子投票などの暗号プロトコルを利用した多くのサービスがインターネット上で利用されようとしている．とりわけ，インターネットを通じた電子投票は，誰もが時間と時を選ばずに投票を行えるという点で非常に有益であると考えられている．そのため，セキュアな電子投票方式

に関する数多くの研究がこれまでになされており，また多くの実験システムが作られ，試行実験が行われている．

これまで提案されている電子投票方式は，Yes か No を選択するような信任投票，定められた複数の候補者から１名を選び投票する多候補者選挙，任意の投票内容を投票可能な投票などを実現するものであった。それらのほとんどは，各投票者はそれぞれ１つの票を持つ単票投票であった。本稿では，各投票者が複数の票を持つ複数票投票を扱う．このような投票は一般的で，株主総会における投票がよい例であり，各投票者の持つ票数は投票者によって異なる場合もある．複数票投票では、投票者が選んだ１つの選択肢に全ての票を投ずる形式と，票を複数の選択肢に分割する形態がある．後者の分割可複数票投票は，例えば，持ち点を優先順位に基づいて複数の選択肢に割り振るようなアンケートなどでよく用いられる．

分割可複数票に対する電子投票プロトコルは，既存の電子投票方式を用いて構成することができるが、プライバシーが確保されない，効率が悪いなどの欠点を持つ．

1.2 関 連 研 究

既存の電子投票方式は用いられる暗号技術により２つに大別できる．

- 電子的でない投票の形式をそのまま電子化することによって実現するタイプ．各投票者が投票内容をメッセージとして，匿名通信路によって送信し，匿名性を持たせた後に個々の票を開票して集計をするものである．このタイプは，投票者の送信した各票が正しく集計されていることを保証するために用いられる技術によって，さらに２つに分類することができる．
  - MIX-net に基づく方式 [8], [11], [15], [17], [19]
  - ブラインド署名に基づく方式 [7], [13], [14]
- 暗号化された個々の票を開票（復号化）することなく，暗号方式の特性を生かして集計結果のみを得るタイプ．暗号方式の準同型性に基づく方式 [2]〜[6], [9], [10], [18] と呼ばれる．

準同型性に基づく方式は，Yes，No 形式の投票に有効な方式であり，この拡張によって複数の候補から 1 人の候補者を選ぶ多候補者選挙も実現されている [4], [10]．このタイプの電子投票は，各投票が Yes，No のどちらを表しているかを証明するためにゼロ知識証明を利用しており，そのため投票者にとって計算コスト，通信コストが高い．

MIX-net は，全てのメッセージが正しく処理されていることを公開検証可能な匿名通信路である．MIX-net に基づくタイプの方式は，任意の投票内容を扱うことができ，投票者の通信コストも低い．しかし，MIX-net の中で，MIX サーバが正しく動作していることをゼロ知識証明を用いて証明しなければいけないため，大規模な選挙には向かないという特徴がある．

ブラインド署名に基づく方式では，各投票者は投票内容に対するブラインド署名を要求し，それを匿名通信路を用いて集計サーバに送信する．ブラインド署名を用いることにより，集計サーバの不正に対して異議を唱えることができるため，MIX-net のような公開検証性をもつ (処理の重い）匿名通信路を必要としない．しかし，各投票者はプロトコルの全てのフェーズに参加する必要がある．プロセスを完了しない投票者が存在すると，投票結果に影響が出てしまう．このタイプの電子投票方式に使われる匿名通信路の性能改善の研究がこれまでに数多くなされている [1].

一方、複数票投票に対しては，準同型性を用いた電子投票方式が提案されている．しかし，この方式は分割不可能な投票を扱っており，分割可投票への拡張はなされていない [20].

金融向けの暗号プロトコルの研究分野では，McKenzie らが電子株券を実現するプロトコルを提案している [12]．このプロトコルでは，電子株券を構築する方法や，株主の権利を保障する方法が提案されている．また，このプロトコルでは，匿名性とセキュリティを守りながら投票をする方法についても述べている．しかし，複数票の投票に対して，現実的な方法は示されていない．

1.3 本論文の結果

本論文では，まず最初に分割可複数票投票を実現する電子投票プロトコルとして Divisible Voting Scheme（DVS）を定義する．次に，Yes と No の２つの選択肢を持つ DVS の一方式を提案する．提案方式では，線形性を持った秘密分散を用いることによってゼロ知識証明などの証明システムを不要としているため，現実的であると言える．その後，このプロトコルを一般化し，任意の数の選択肢を持つ DVS の構成方法を示す．さらに，分割不可な複数票投票への応用についても述べる．

2. 分割可能な電子投票方式

2.1 定　　義

u 通りの選択肢 $A_1, \dots A_u$ に対する Divisible Voting Scheme（以後 DVS と記す）を以下のように定義する．

DVS は，投票サーバと，n 人の投票者 $V_1, \cdots, V_n$ によって構成されるマルチパーティープロトコルである．投票者 V_i は x_i 票を持っているとする．投票者は持ち票を分割して $A_1, \dots A_u$ に投票する．ここで，$A_1, \dots A_u$ に対する票数をそれぞれ $v^{(i)}_{A_1}(\geqq 0), \dots, v^{(i)}_{A_u}(\geqq 0)$ と記述する．すると，全ての i に対して $\sum_{j=1}^{u} v^{(i)}_{A_j} = x_i$ が成り立つ．投票サーバは，すべての j に対して $\sum_i v^{(i)}_{A_j}$ を求める．

ここで，以下の条件を満たすとき，DVS はセキュアであると言う．

健全性: いかなる投票者 V_i も，自分の持ち票数 x_i を超える投票はできない．すなわち，

$$0 \leqq v^{(i)}_{A_j} \leqq x_i (1 \leqq i \leqq n, 1 \leqq j \leqq u)$$

$$\sum_{j=1}^{u} v^{(i)}_{A_j} = x_i (1 \leqq i \leqq n)$$

完全性: 健全性を満たす全ての投票に対して，投票サーバはすべての j に対して集計結果 $\sum^{(i)}_{A_j}$ を正しく出力する．

プライバシ: 投票サーバと投票者は最終結果 $\sum_i v^{(i)}_{A_j} (1 \leqq j \leqq u)$ 以外のいかなる情報も得ない．ただし，各投票者 V_i の持ち票数 x_i と，各 V_i が投票に参加したかどうかは公開でもよい．

2.2 既存投票方式を適用した場合の問題点

既存の電子投票方式を用いて DVS を実現するためには以下

の2つの方法が考えられる．

最初の方法は，任意の投票が可能な匿名通信路を利用する方式を用いる方法である．ここでは，簡単のため，2つの選択肢 A_1, A_2 を持つ投票を考える．この場合，投票者は $v_{A_1}^{(i)}, v_{A_2}^{(i)}$ を暗号化して送信する．全ての暗号化された $v_{A_1}^{(i)}, v_{A_2}^{(i)}$ は，シャッフルされた後，復号され，最終的に結果が計算される．ここで，次のような場合を考えよう．$x_1 = \max_i x_i$ であるとき，V_1 が $v_{A_1}^{(1)} = x_1, v_{A_2}^{(1)} = 0$ の票を投じたとする．このとき，集計者はこの投票が V_1 のものであることが推測できる．よって，この方法はセキュアな DVS とは言えない．

2番目の方法は，既存の単票投票方式において1人の投票者が投票作業を複数回繰り返すという方法である．V_i が x_i 票を持っている場合，V_i は1票の投票作業を x_i 回繰り返すとこによって x_i 票の投票を行う．使用される既存の単票投票方式が安全であれば，この方式で安全な DVS が得られる．しかし，この方式は効率的ではない．

準同型性を利用したタイプの投票方式を用いる場合，各投票者は送信したデータが正しい投票であることを証明しなければならず，このためにゼロ知識証明のような証明システムを利用する必要がある．つまり V_i は x_i 回のゼロ知識証明を実行する必要がある．一般的にゼロ知識証明は，多大な計算コストと通信コストを必要とする．

既存の単票投票方式としてシャッフルベースの投票方式を利用する場合は，MIX-net の入出力の数が（投票者数でなく）投票総数に等しくなり，やはり効率的ではない．

3. 提案プロトコル

簡単のため、この章では Yes と No の2つの選択肢を持つ DVS について，これを実現する効率的な方式を提案する．

3.1 システムモデル

ここでは，以下のシステムを仮定する．n 人の投票者 $V_i (1 \leqq i \leqq n)$ が存在する．各投票者は，x_i 票を持ち，Yes に $v_Y^{(i)}$ 票，No に $v_N^{(i)}$ 票を投じたいとする．

$$v_Y^{(i)} \geqq 0, v_N^{(i)} \geqq 0$$

かつ，

$$x_i = v_Y^{(i)} + v_N^{(i)}$$

が成立しているときに，$(v_Y^{(i)}, v_N^{(i)})$ は有効であると言う．

各投票のプライバシを確保するために，線形性を持った秘密分散を利用して，集計機能を m 台の集計サーバに分散させる．ここで，$m \geqq 3t+1$ とし，m 台の集計サーバのうち不正を行うサーバは高々t 台とする．V_i は，投票内容 $v_Y^{(i)}, v_N^{(i)}$ を複数のサーバに分散させる．集計は全投票者からのシェアの総和を計算し，投票内容の総和を復元することによって実現される．

多くの既存の投票方式では，投票の有効性を保証するためにゼロ知識証明などを利用しているが，これは効率的ではない．そこで，投票の有効性を保証するために，新たなサーバ S_R と公開掲示板 BBS を想定する．S_R は準備段階にのみ動作し，一度投票が始まると，プロトコルには一切参加しない．また，S_R は任意の集計サーバや投票者とは結託しないことと，BBS の内容は誰でも参照できるが，BBS の情報の追加・変更は S_R のみが可能であり，S_R が一度全ての情報を入力した後は，一切のデータを変更しないことを仮定する．

3.2 主なアイディア

集計サーバ S_j $(1 \leqq j \leqq m)$ は共同で Yes 用のカウンタ C_Y と No 用のカウンタ C_N を保持する．これらのカウンタ値は，Shamir の $(t+1, m)$ しきい値分散法 [16] により分散されている．また S_j はリスト L_j を持つ．L_i にはすでに投票を終えた投票者が含まれていて，ある投票者が票を全ての S_j に送信した時に各 S_j は L_i を使って同じ投票者による投票が行われていたかどうかをチェックできる．

サーバ S_R は各投票者の持ち票数を管理している．すなわち，全ての i に対して，(V_i, x_i) のペアを知っている．S_R は，全ての V_i のための投票データを作成し，V_i に配る．また，S_j は投票の正当性を検証を行うために必要なデータを作る．S_R は，全ての i に対して，$v_Y^{(i)}, v_N^{(i)}$ を知ることはできない．

公開掲示板 BBS は，全ての可能な投票データのシェアのハッシュ値を記録する．また，BBS に記録されているデータはプライバシの保護のためにシャッフルされている．ある投票者が投票データを全ての S_j に送信したとき，S_j は，BBS に記録されたデータを用いて，投票データの正当性を検証できる．

提案プロトコルは準備フェーズ，投票フェーズ，集計フェーズの，3つのフェーズから構成される．準備フェーズでは，S_R が各投票者に対して投票可能な全ての投票データを作り，投票者に送信する．その後，S_R は，不正投票を検出するためのデータを作成し，シャッフルして BBS に記録する．全ての投票者は，BBS を参照して，シャッフルが正しくされていることと，投票データが正しく作られていることをチェックする．投票フェーズでは，各投票者は，自身の投票内容に沿った投票データを選び，全ての S_j に送信する．S_j は二重投票がないことと，投票データが正しいことを BBS を使いチェックする．このチェックが正しく終了した場合のみ，S_j はそれぞれのカウンタをアップデートする．集計フェーズでは，全ての集計サーバが協力して，Yes と No の総数を，秘密分散の復元プロトコルを利用して復元する．

3.3 プロトコル詳細

3.3.1 準備フェーズ

準備フェーズでは，以下のようなプロトコルを実行する（図1）．

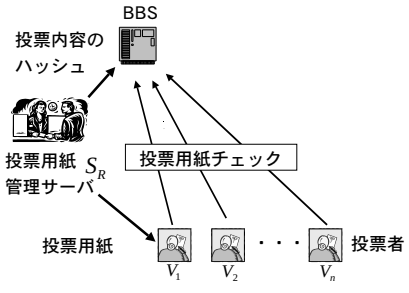

図1 準備フェーズ

(Step1). 全ての集計サーバ S_j は，協力して C_Y, C_N を 0 にセットする．つまり，各サーバ S_j は，

$$C_{Y,j} = 0$$

$$C_{N,j} = 0$$

を実行する．次に，各 S_j は，投票を終えた投票者の名前を含むリスト L_j を空リストとして作る．

(Step2). 全ての $i(1 \leqq i \leqq n)$ について，S_R は以下を実行する．

全ての $j(1 \leqq j \leqq m)$，$k(1 \leqq k \leqq x_i)$ について，サーバ S_j に対する k のシェア $s_{Y,j}^{(i,k)}$ と，S_j に対する $x_i - k$ のシェア $s_{N,j}^{(i,k)}$ を作成する．

その後，すべての投票データのシェアを RSA 暗号のような確定的な暗号アルゴリズムで暗号化し，

$$ES_j^{(i,k)} = Enc\langle PkS_j\rangle(s_j^{(i,k)})$$

を全ての k, j について得る．ここで $s_j^{(i,k)} = (s_{Y,j}^{(i,k)} || s_{N,j}^{(i,k)})$，$||$ を結合を表し，PkS_j は S_j の公開鍵であり，$Enc\langle Pk\rangle(m)$ は平文 m を公開鍵 Pk で暗号化した結果の暗号文を表す．

次に，自身の秘密鍵 SkS_R を用いて，各 k について $ES_j^{(i,k)}$ の電子署名を作成する．すなわち，各 k について，

$$Sig^{(i,k)} = Sig\langle SkS_R\rangle(ES_1^{(i,k)} || \cdots || ES_m^{(i,k)})$$

を計算する．ここで，$Sig\langle Sk\rangle(m)$ は，秘密鍵 Sk を用いて作成した m の電子署名を表す．

(Step3). 全ての $i(1 \leqq i \leqq n)$ について，S_R は以下を実行する．

以下のハッシュ値を計算する．

$$h_j^{(i,k)} = H(s_j^{(i,k)})(0 \leqq k \leqq x_i, 1 \leqq j \leqq m)$$

ここで，H は衝突困難な一方向性ハッシュ関数とする．続いて，ランダム置換 π を選び，各 j について $\{h_j^{(i,0)}, \cdots, h_j^{(i,x_i)}\}$ を π を使ってシャッフルする．

$$HS_j^{(i)} = (h_j^{(i,\pi(0))}, \cdots, h_j^{(i,\pi(x_i))})$$

最後に $HS_j^{(i)}$ を BBS に記録する．

(Step4). S_R はセキュアな通信路を通じて，

$$(s_1^{(i,k)}, \cdots, s_m^{(i,k)}), (ES_1^{(i,k)}, \cdots, ES_m^{(i,k)}), Sig^{(i,k)}(0 \leqq k \leqq x_i)$$

を V_i に送信する．V_i はこれらが正しく作られていることを以下のように確認する．

全ての k, j について，$H(s_j^{(i,k)})$ を計算し，BBS に記録されている $HS_j^{(i)}$ について，$H(s_j^{(i,k)})$ の集合が $HS_j^{(i)}$ の集合と等価であり，$HS_j^{(i)}$ がシャッフルされていることを確認する．次に，全ての k について，V_i は，$(s_{Y,1}^{(i,k)}, \cdots, s_{Y,m}^{(i,k)})$ が k のシェアであり，$(s_{N,1}^{(i,k)}, \cdots, s_{N,m}^{(i,k)})$ が，$x_i - k$ のシェアであることを確認する．さらに $s_j^{(i,k)}$ を PkS_j で暗号化し，

$$ES_j^{(i,k)} \stackrel{?}{=} Enc\langle PkS_j\rangle(s_j^{(i,k)})$$

を全ての k, j について検証する．

3.3.2 投票フェーズ

投票フェーズでは，以下のようなプロトコルを実行する（図 2）．

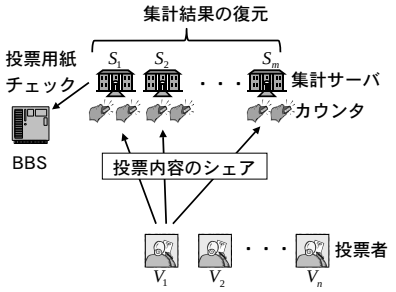

図 2 **投票フェーズ**

(Step1). $k = v_Y^{(i)}$ とする．V_i は $(ES_1^{(i,k)}, \cdots, ES_m^{(i,k)})$ と $Sig^{(i,k)}$ を全ての S_j に送信する．各 S_j は，L_j が V_i を含むかどうかをチェックし，もし，L_j が V_i を含んでいれば，その投票は棄却する．続いて，S_j は，$Sig^{(i,k)}$ の検証を行う．そして，各 S_j は，$ES_j^{(i,k)}$ を秘密鍵を利用して復号する．

$$s_{Y,j}^{(i,k)} || s_{N,j}^{(i,k)} = s_j^{(i,k)} = Dec\langle SkS_j\rangle(ES_j^{(i,k)})$$

ここで，$Dec\langle Sk\rangle(c)$ は，暗号文 c を秘密鍵 Sk を用いて復号した結果得られる平文を表す．S_j は，BBS から $HS_j^{(i)}$ を取得し，平文のハッシュ値が $HS_j^{(i)}$ に含まれることをチェックする．

$$H(s_j^{(i,k)}) \stackrel{?}{\in} HS_j^{(i)}$$

以上の全てのチェックが完了したら，S_j は，V_i を L_j に加える．

(Step2). 全ての S_j は，カウンターを以下のようにアップデートする．

$$C_{Y,j} := C_{Y,j} + s_{Y,j}^{(i,k)}$$

$$C_{N,j} := C_{N,j} + s_{N,j}^{(i,k)}$$

3.3.3 集計フェーズ

全ての S_j は，協力してカウンタ C_Y, C_N を，シェア $C_{Y,j}, C_{N,j}$ から復元する．$m \geqq 3t+1$ であるため，t 台以下のサーバの不正があったとしても，集計サーバは誤り訂正を利用して C_Y, C_N を正しく復元することができる．C_Y は Yes の票の総数，C_N は No の票の総数を表す．

3.4 セキュリティ評価

ここでは，提案プロトコルがセキュアな DVS であることを示す．

健全性: 投票権がない投票者が投票データを集計者に送信した場合，BBS に存在する $(V_i, HS_j^{(i)})$ のペアの検索によって，この投票を棄却することができる．また，投票者 V_i が投票データを複数回 S_j に送信した場合，S_j は L_j が V_i が含むかどうかをチェックすることで，その投票を棄却することができる．したがって，権利のない投票者による投票者はすべて棄却されるため，以下では権利のある投票者の投票についてのみ考える。

S_R によって作られたシェアは各 j について暗号化され，有

効な $(v_Y^{(i)}, v_N^{(i)})$ についてまとめた上で署名が付与されている．シェアの各ペアは投票内容が有効であるように計算されている．よって，S_R によって署名が付与されている投票データはすべて有効な投票である．集計サーバは，投票フェーズで投票データの署名を検証するため，投票者は有効でない投票を行うことができない．

完全性: 投票の集計結果は，秘密分散を用いて m 台のサーバに分散されている．準備フェーズの Step1 で，全てのサーバは分散カウンタを 0 にセットしている．少なくとも $2t+1$ 台のサーバはカウンタ $C_{Y,j}, C_{N,j}$ のアップデートを，Shamir の秘密分散の線形性を用いて正しく行っている．よって，正しい Yes と No の票の集計結果は，秘密分散の復元と誤り訂正プロトコルによって復元される．よって，提案プロトコルは完全性を持っている．

プライバシ: 全ての集計サーバ S_j のカウンタの値は，$(t+1, m)$ しきい値法で分散されている．よって，最大 t 台のサーバが結託したとしても，カウンタの値を得ることはできない．BBS で公開されている $HS_j^{(i)}$ は S_R によってシャッフルされているので，S_j は $k = v_Y^{(i)}$ を，$ES_j^{(i.k)}, HS_j^{(i)}$ から知ることはできない．

V_i は全ての S_j のカウンタにアクセスすることはできない．よって，V_i は S_R と V_i の通信路がセキュアである限り，V_i' から S_j に送られた $ES_j^{(i',k')}$ から，$v_Y^{(i')}$ を知ることはできない．また，使用している公開鍵暗号プロトコルがセキュアで，S_R によるシャッフルが秘密である限り，BBS に記録されている $HS_j^{(i)}$ と $ES_j^{(i,k)}$ から $k = v_Y^{(i)}$ を，任意の投票者が知ることはできない．

3.5 性能評価

ここでは，提案プロトコルの計算コストと通信コストの評価を行う．

準備フェーズ: 準備フェーズでは，S_R は，全ての投票者 V_i についてシェアと電子署名を作成する．ここで，X を，投票総数とすると，S_R は $2m(X+n)$ 個のシェアを作成し，$m(X+n)$ 回の暗号化を実行する．その後 S_R は，$X+n$ 個の電子署名を作成し，$m(X+n)$ 回のハッシュ計算を行う．

以上の計算が完了した後，S_R はこれらのハッシュ値を BBS に登録する．ハッシュ値のビット長を l_h とすると，BBS に登録されるデータサイズは $ml_h(X+n)$ となる．ここで，$l_h = 128, n = 1{,}000, X = 100{,}000, m = 4$ とすると，このデータサイズは $6.5MB$ となる．また S_R は，mn 回の置換を行う．

その後，S_R はシェアとその暗号文，電子署名を全ての投票者 V_i に対して送信する．ここで，シェアの（最大）ビット長を l_b，電子署名のビット長と l_s，暗号文のビット長を l_C とする．すると，S_R と各投票者 V_i との間の通信サイズは $(x_i+1)(l_s+2ml_b+2ml_c)$ となる．ここで，l_b を 32 ビット，$l_s = l_c = 1024$，x_i の平均値を x_i とすると，通信サイズの平均値は $120KB$ となる．

最後に V_i は投票データの正当性をチェックする．V_i は $x_i m$ 回のハッシュ計算を行い，その後 BBS から $x_i ml_h$ ビットのデータを取得する．前述の例を当てはめると $6KB$ となる．また，Step4 で，V_i は $m(x_i+1)$ 回の暗号化を行う．

投票フェーズ: Step1 で，各投票者 V_i は，全ての S_j にデータを送信する．各 S_j に対する通信データサイズは，$ml_c + l_s$ ビットであり，データサイズの合計は $m(ml_c+l_s)$ となる．これは，前述の例を当てはめると $2.6KB$ となる．

各 S_j は 1 回だけ電子署名の検証を行い，その後 2 回の復号と 1 回のハッシュ計算を行う．続いて，S_j は BBS 上の x_i 個の要素からの検索を行う．最後に S_j は 2 個のカウンターのアップデートを行うが，これは加算となる．

集計フェーズ: 集計フェーズでは，全てのサーバが協力して秘密分散の復元プロトコルを実行するのみである．

ここで，提案プロトコルでは，S_R を加えることにより，ゼロ知識証明を必要としないため．非常に効率的である．

提案プロトコルでは，準備フェーズの Step4 で，暗号化された投票データの有効性を検証するために，各投票者は $m(x_i+1)$ 回のシェアの暗号化を行う．PDA のような計算能力の低いデバイスを用いている場合，この演算は実用的ではない．もし，S_R がシェアを正しく計算すると仮定するならば，いくつかの計算は必要がなくなる．この場合，各投票者は，シェアからの投票内容を復元と，シェアの暗号化を省略することができる．よって，準備フェーズにおいて，$m(x_i+1)$ 回の暗号化を省略することが可能となる．

4. 一般化とその他の電子投票形態への応用

4.1 一 般 化

前章で提案したプロトコルは，2 つの選択肢だけでなく，より多くの選択肢を持つ一般的な DVS に拡張することが可能である．ここでは，u 通りの選択肢 $A_1, \cdots, A_u$ があり，各投票者 V_i が x_i 票を持つとする．V_i は，$x_i = \sum_{s=1}^{u} v_{a_s}$，かつ $0 \leqq v_{a_s} \leqq x_i (1 \leqq s \leqq u)$ を満たすように，x_i 票を $(v_{a_1}, \ldots, v_{a_u})$ に分割し、これを投票することができる．

このような投票を実現するためには，各集計サーバが S_j が u 個の分散カウンタを持つようにする．準備フェーズでは，S_R は，3.3 のプロトコルの $s_{k,j}^{(Y,i)}$ と $s_{k,j}^{(N,i)}$ の代わりに，$k_1 + \ldots + k_u = x_i$ かつ $0 \leqq k_1, \ldots, k_u \leqq x_i$ を満たすような $s_{(1,j)}^{(i,k_1,\ldots,k_u)}, \ldots, s_{(u,j)}^{(i,k_1,\ldots,k_u)}$ を，全ての $j, k_1, \ldots, k_u$ に対して作成する．この点以外は，一般化プロトコルは基本プロトコルと同様に構成できる．

この場合，S_R が V_i に対して作成するシェアの数は $um\sum_{w=0}^{x_i}(u-1)^{(x_i-w)}$ 個となる．

4.2 部分的な棄権が可能な投票

一般的な投票では，棄権票が存在する．つまり，$v_Y^{(i)} + v_N^{(i)} \leqq x_i$ となるような投票も有効とする場合がある．このような投票を実現するためには，カウンタの数を $u = 3$ として，追加されたカウンタを棄権票のためのカウンタとして，S_R が投票内容に応じたシェアを作成すればよい．

4.3 分割不可な複数票投票

提案プロトコルは，分割不可複数票投票に応用することがで

きる．たとえば、2つの選択肢 yes と no を持つ分割不可複数票投票は，x_i 票を持つ投票者 V_i が Yes に x_i 票を投じるか，No に x_i 票を投じるかのどちらかを選ぶタイプの投票である．つまり，

$$(v_Y^{(i)}, v_N^{(i)}) = (x_i, 0)$$

または

$$(v_Y^{(i)}, v_N^{(i)}) = (0, x_i)$$

が有効な投票となる．

このような投票を実現するには，S_R は以下の 2 つのデータのシェアを作成すればよい．

$$(x_i, 0), (0, x_i)$$

この場合，プロトコルは分割可能な投票よりも非常に効率的になる．

4.4 多候補者選挙

多候補者選挙は，投票者が u 個の候補から 1 つの候補のみを選び出すような，一般的な形式の投票である．このような投票を実現するためには，集計サーバ S_j は，u 種類の分散カウンタ $C_{1,j}, \cdots, C_{u,j}$. を持つ．

そして，S_R は以下のデータのシェアを作成する．

$$(1, 0, \cdots, 0), (0, 1, \ldots, 0), \cdots, (0, \cdots, 0, 1)$$

ここで，各シェアの元となる秘密は，1 つだけが 1 で残りが全て 0 となるように構成される．

5. ま と め

本稿は，分割可能な投票（Divisible Voting Scheme: DVS）のモデルと，実用的な実現プロトコルを示した．提案プロトコルでは，線形性を持つ秘密分散を利用している．このプロトコルでは，投票内容のシェアを作り，シェアのハッシュ値のシャッフルを計算するサーバ S_R の存在を仮定している．このサーバの存在により，ゼロ知識証明が不要となり，提案プロトコルは既存の電子投票方式に比べて効率的になっている．提案プロトコルは S_R が集計サーバや投票者と結託しない限り，健全性，完全性，プライバシを保っている．さらに，本論文では提案プロトコルの一般化を示し，分割不可の複数投票，多候補者選挙にも応用できることを示した．

文　献

[1] M. Abe, Universally Verifiable Mix-net with Verification Work Independent of the Number of Mix-servers," In Proc. of Eurocrypt'98, pp.437-447.

[2] J. Benaloh and D. Tuinstra, "Receipt-Free Secret-Ballot Elections," STOC94, 99.544-553.

[3] J. Benaloh and M. Yung, "Distributing the Power of a Government to Enhance the Privacy of Voters," Proc. of PDOC86, pp.52-62.

[4] R. Cramer, R. Gennaro and B. Schoenmakers, "A Secure and Optimally Efficient Multi-Authority Elections," In Proc. of EUROCRYPT'97, pp.103-118.

[5] J. Cohen and M. Fischer, "A Robust and Verifiable Cryptographically Secure Election Scheme," FOCS 85, pp.372-382.

[6] R. Cramer, M. Franklin, B. Schoenmakers and M. Yung, "Multi-Authority Secret-Ballot Elections with Linear Work," In Proc. of EUROCRYPT'96, pp.72-82.

[7] A. Fujioka, T. Okamoto, K. Ohta, "A Practical Secret Voting Scheme for Large Scale Elections," In Proc. of AUSCRYPT '92, pp3244-251.

[8] J. Furukawa and K. Sako, "An Efficient Scheme for Proving a Shuffle," In Proc. of CRYPTO2001, pp.368-387

[9] K.R. Iversen, "A Cryptographic Scheme for Computerized General Elections," In Proc. of CRYPTO'91, pp.405-419.

[10] H. Kikuchi, J. Nakazato and S. Nakanishi, "Oblivious Ciphertext Demultiplexer and Multi-way Election," In Proc. of SCIS2002.

[11] M. Michels, O. Horster, "Some Remarks on a Receipt-Free and Universally Verifiable Mix-Type Voting Scheme," In Proc. of ASIACRYPT'96, 99.125-132.

[12] P. McKenzie and J. Sorensen, "Anonymous Investing," In Proc. of Financial Cryptography '99, Lecture Notes in Computer Science, Vol.1648, pp.212-229, Anguilla, BWI, February, 1999.

[13] T. Okamoto, "An Electronic Voting Scheme," Proc. of IFIP96, Advanced IT Tools.

[14] M. Ohkubo, F. Miura, M. Abe, A. Fujioka and T. Okamoto, "An improvement on a Practical Secret Voting Scheme," In Proc. of ISW'99, pp.255-234.

[15] C. Park, K. Itoh and K. Kurosawa, "Efficient Anonymous Channel and All/Nothing Election Scheme," In Proc. of EUROCRYPT'93, pp.248-259.

[16] A. Shamir, "How to Share a Secret," Communications of the ACM, Vol.22, No.11, pp.612-613, 1979.

[17] K. Sako, "Electronic Voting Scheme Allowing Open Objection to the Tally," IEICE Trans. on Fundamentals, Vol.E77-A, No.1 January 1994, pp.24-30.

[18] K. Sako and J. Kilian, "Secure Voting Using Partially Compatible Homomorphisms," In Proc. of CRYPTO'94. pp.411-424.

[19] K. Sako and J. Kilian, "Receipt-Free Mix-Type Voting Scheme," In Proc of EUROCRYPT'95, pp.393-403.

[20] 税所，齋藤，鈴木，土井，辻井，"準同型性暗号を利用した 1 人複数投票可能な電子投票方式", コンピュータセキュリティシンポジウム 2002 論文集，IPSJ Symposium Series, Vol.2002, No.16, pp.467-472