Divisible Voting Scheme

Natsuki ISHIDA, Shin'ichiro MATSUO and Wakaha OGATA

April 23, 2003

Abstract

Electronic voting is a prime application of cryptographic tools. Many researches are addressing election or confidence voting in this area. We address a new type of voting scheme "Divisible Voting Scheme," in which each voter has multiple ballots where the number of ballots can be different among the voters. This type of voting is popular, however there is no secure protocol which achieves this type of voting. We first define the divisible voting scheme and show naive protocols based on existing voting schemes. Then we propose two efficient divisible voting schemes. The first scheme uses multisets, the second scheme uses L-adic representation of number of ballots. The total cost for a voter is $O(M^2 \log(N))$ in the first scheme and $O(M \log(N))$ in the second scheme where M is the number of candidates to vote for and N is the number of ballots for a voter.

1 Introduction

An electronic voting system is a prime application of cryptographic tools. Many researchers are studying on it, and many protocols have been proposed. Electronic voting system should assure voters' privacy, as well as the correctness of the result. To achieve the privacy, some protocols such as [1] or [2] use anonymous channels, which is also called MIX-net. On the other hand, it is known that confidence voting can be realized by using homomorphic encryption with multiple servers [3]. In confidence voting, each voter votes "Yes" or "No." [3] showed that the protocol can be extended to a multi-way voting system in which each voter chooses one option from multiple candidates.

In this paper, we focus on another type of voting system in which each voter has multiple ballots. This type of voting is used at a general meeting of stockholders. In such voting systems, there are multiple candidates to vote for and each voter has multiple ballots, where the number of ballots can be different among the voters. Moreover, in some cases, each voter can divide his available ballots into some parts and vote each part for each candidate. We first define such voting system as "divisible voting," and define *secure* divisible voting scheme. Next, we give some naive divisible voting protocols. They are easily obtained from multi-way voting protocol, however, they are not efficient if the number of ballots each voter has is large. Finally, we propose two efficient constructions of divisible voting. Our fist construction uses multisets with a special property, named *divisible multiset.* The second one L-adic representation of the numbers of ballots. Both of our protocols are based on homomorphic encryption, and they do not need MIX-net.

2 Divisible voting

We define divisible voting as follows.

- There are many voters and some voting servers. Let $\{\mathcal{V}_1, \mathcal{V}_2, \ldots\}$ be the set of voters.
- Each voter has multiple ballots. Let $N^{(i)}$ be the number of ballots $\mathcal{V}_i$ has.
- There are multiple candidates to vote for. Let M be the number of candidates.
- Each voter votes some ballots which are taken from his available ballots for all candidates. We denote the number of ballots $\mathcal{V}_i$ votes for j-th candidates by $v_j^{(i)}$. It must be

$$0 \leq v_j^{(i)} \leq N^{(i)} \quad \text{and} \quad \sum_{j=1}^{M} v_j^{(i)} = N^{(i)}. \tag{1}$$

The goal is to output the number of total ballots each candidate obtained without revealing any additional information.

We can consider that divisible voting is a most general model of voting system. When $M = 2$ and $N^{(i)} = 1$ for all i, the system is an ordinal yes/no voting system, in which each voter votes for "yes" or "no". When $M > 2$ and $N^{(i)} = 1$ for all i, the system is a multi-way voting system.

We say that a protocol is a secure divisible voting scheme if it satisfies following requirements.

1. The voting servers correctly output the result of voting, even if some of voting servers are malicious, and any one can verify the validity of the result.

2. Any voter can not vote in excess of the number of his available ballots $N^{(i)}$. That is, Eq. (1) must be satisfied for all i.

3. Any one can not obtain additional information about the choice of each voter from published information. Formally, consider an adversary who corrupts less than t servers and all voters except two target voters. Assume that $(v_j^{(i)}, v_j^{(i')})$ which are choices of two target voters and $(v_j'^{(i)}, v_j'^{(i')})$ which are other choices of them lead same results, then the views the adversary sees in these two cases are indistinguishable.

3 Previous voting protocols based on a homomorphic encryption

Electronic yes/no and multi-way voting has been studied by many researchers, and efficient schemes were already proposed [3]. In this section, we review yes/no voting protocols based on a homomorphic encryption and the extension to multi-way voting.

3.1 Requirements of encryption schemes

Electronic voting protocols based on a homomorphic encryption use a public-key encryption with some desirable conditions. Let E be a public-key probabilistic encryption function. We denote by $E(m)$ the set of encryptions for a plaintext m and by $e \in E(m)$ a particular encryption of m.

We assume that E satisfies following desirable properties.

Homomorphic property. There exists a polynomial time computable operation, $\otimes$, as follows for a large prime q.

$$(e_1 \in E(m_1) \wedge e_2 \in E(m_2)\ ) \implies e_1 \otimes e_2 \in E(m_1 + m_2 \bmod q)$$

For the simplicity, we write

$$\prod_{i=1}^{n} e_i = e_1 \otimes e_2 \otimes \cdots \otimes e_n$$

and

$$e^a = \underbrace{e \otimes e \otimes \cdots \otimes e}_{a}$$

for a positive integer a. Then, if $e_i \in E(m_i)$ then

$$\prod_{i=1}^{n} (e_i)^{a_i} \in E\left(\sum_{i=1}^{n} a_i m_i \bmod q\right).$$

Threshold decryption. For a given ciphertext $e \in E(m)$, any k out of n players can decrypt e along with a zero-knowledge proof of the correctness. However, any $k-1$ out of n players cannot decrypt e.

Non-malleability. A public key cryptosystem is said to be non-malleable [4] if there exists no probabilistic polynomial time adversary such that given a challenge ciphertext e, he can output a different ciphertext e' such that the plaintexts m, m' for e, e' are meaningfully related. (For example, $m' = m + 1$.)

Plaintext membership proof (PMP). There exists an efficient zero-knowledge proof protocol, called plaintext equality test, which proves $e \in E(m)$ for given e and a known plaintext m.

Cramer et al. shows that there exists an efficient construction of witness-indistinguishable proof to prove that

$$e \in E(m_1) \text{ or } e \in E(m_2) \text{ or } \cdots \text{ or } e \in E(m_M)$$

for given ciphertext e and M known messages $m_1, \ldots, m_M$, if there exists a plaintext equality test [5]. In this paper, we denote such a proof system "1-out-of-M plaintext membership proof (PMP)."

ElGamal cryptosystem and Paillier cryptosystem [6] satisfy these properties.

3.2 Overview of yes/no voting scheme

In an yes/no voting scheme, each voter has only one ballot, and he votes it for 'Yes' or 'No'. The protocol is summarized as follows.

1. In advance, n voting servers share a decryption key of a public-key probabilistic encryption function E.

2. $\mathcal{V}_i$ sets $v^{(i)} = 1$ if he wants to vote for Yes, otherwise $v^{(i)} = 0$, and opens $e^{(i)} \in E(v^{(i)})$. Then he proves that $e^{(i)} \in E(1)$ or $e^{(i)} \in E(0)$ using a 1-out-of-2 PMP.

3. After the voting period,

$$e^{(sum)} \triangleq \prod_i e^{(i)}$$

is computed. This computation can be done by every one. Then, n servers jointly decrypt $e^{(sum)}$ and obtain $v^{(sum)}$. $v^{(sum)}$ is the number of voters who voted $v^{(i)} = 1$.

In the above protocol, homomorphism of E assures that

$$e^{(sum)} \in E\left(\sum_i v^{(i)}\right)$$

and then

$$v^{(sum)} = \sum_i v^{(i)}.$$

Decryption is done correctly and public verifiably since it is done by k-out-of-n threshold manner. Then the first condition of security is satisfied.

The second condition is assured, since each voter has to pass a 1-out-of-2 PMP.

If ElGamal encryption is used as a homomorphic encryption, the ciphertext of no-vote is (g^r, y^r) and that of yes-vote is (g^r, gy^r), where y is a public key and r is a random number. Then the message the servers actually obtain is

$$v^{(sum)} = g^{\sum_i v^{(i)}}.$$

Since $\sum_i v^{(i)}$ is not more than the number of voters, servers can get $\sum_i v^{(i)}$.

3.3 Multi-way voting

There are two ideas of construction of a multi-way voting scheme. In the first one, we call *combined-type*, each voter opens only one ciphertext. In the other one, *separate-type*, he opens one ciphertext for one candidate, then totally M ciphertexts. (M is the number of the candidates to vote for.)

Combined-type. Let $N^{(sum)}$ be the number of voters. To get a multi-way voting for M candidates, we use M specified messages, $(m_1, \ldots, m_M)$ such that $m_j = (N^{(sum)} + 1)^{j-1}$. A voter who votes for $\hat{\jmath}$-th candidate publishes a ciphertext of $m_{\hat{\jmath}}$, and proves that the ciphertext is valid by using a 1-out-of-M PMP. The tallying is done as same as yes/no-voting. The plaintext $v^{(sum)}$ does not specify the result explicitly, but

$$v^{(sum)} = T_1 + (N^{(sum)} + 1)T_2 + \cdots + (N^{(sum)} + 1)^{M-1} T_M,$$

where T_j's are the results of the election. Since $T_j < N^{(sum)} + 1$, T_j's are determined uniquely.

Separate-type. A voter who votes for $\hat{\jmath}$-th candidate prepares M ciphertexts as follows.

$$e_j \in \begin{cases} E(0) & \text{if } j \neq \hat{\jmath} \\ E(1) & \text{if } j = \hat{\jmath} \end{cases}$$

Then opens them and proves that each ciphertext is valid by using a 1-out-of-2 PMP and $\prod_{j=1}^{M} e_j \in E(1)$. In this case, the tallying is very simple. The servers combine ciphertexts for each candidate and decrypt them. The obtained plaintexts are the results of voting.

Comparison of two types. We compare the two types of multi-way voting protocols briefly. Column "#ciphertext" in Table 1 shows the number of ciphertexts each voter has to compute and publish. The next column, "PMP," shows the number x such that each voter performs 1-out-of-x PMP for each ciphertext. (For example, in separate-type protocol, each voter performs 1-out-of-2 PMP for each ciphertext.) In general, a PMP costs a few times of computation and communication costs of sending a ciphertext. In this paper, we assume that the cost of a PMP equals τ times of that of computing and sending a ciphertext[1]. Furthermore, we assume that a 1-out-of-x PMP costs x times of a PMP. Then, the total costs are estimated such as column "Total."

Column "#decryption" shows the number of ciphertexts the servers have to decrypt jointly.

4 Naive divisible voting protocols

Before showing efficient protocols, we give some naive solutions. It is easy to show they are secure protocols.

[1] For example, $1 < \tau < 2$ for ElGamal encryption.

Table 1: Comparison of multi-way voting protocols

Type	#ciphertext	PMP	Total	#decryption
Combined	1	M	$\tau M + 1$	1
Separate	M	2	$(2\tau + 1)M$	M

Iteration-type. We can easily realize divisible voting in such a way that $\mathcal{V}_i$ performs a multi-way voting procedure $N^{(i)}$ times independently. If the multi-way voting protocol is secure, then the divisible voting protocol is also secure. We can use both combined-type one and separate-type one.

In such protocols, the communicational and computational costs of voter $\mathcal{V}_i$ are $N^{(i)}$ times of those in the multi-way voting protocol.

Modified separate-type. A voter who votes v_j ballots for j-th candidate prepairs M ciphertexts as follows.

$$e_j \in E(v_j)$$

Then opens them and proves that each ciphertext is valid by using a 1-out-of-$(N^{(i)} + 1)$ PMP and $\prod_{j=1}^{M} e_j \in E(N^{(i)})$.

Comparison. Table 2 shows the comparison of naive protocols. From this table, we can say that iteration of separate-type is less efficient than others, and the total cost is $O(MN^{(i)})$ for all types.

Table 2: Comparison of naive protocols

Type	#ciphertext	PMP	Total	#decryption
Iteration+combined	$N^{(i)}$	M	$(\tau M + 1)N^{(i)}$	1
Iteration+separate	$MN^{(i)}$	2	$(2\tau + 1)MN^{(i)}$	M
Modified separate	M	$N^{(i)} + 1$	$M(\tau N^{(i)} + \tau + 1)$	M

5 Efficient divisible voting schemes

In the naive protocols, each voter's communication and computation costs are $O(MN^{(i)})$ where $N^{(i)}$ is the number of ballots he has. When $N^{(i)}$ is large, it is not efficient.

In this section, we propose two efficient constructions of secure divisible voting scheme.

5.1 Divisible voting scheme using multisets

The first construction uses multisets of positive integers, called *divisible multiset*, which satisfies special conditions to reduce the number of iterations of iteration-type naive protocols.

Definition 1 *For positive integers M and N, we call a multiset X an (M, N)-Divisible Multiset (DM) if for all sets of non-negative integers $(x_1, \dots, x_M)$ such that $\sum_{i=1}^{M} x_i = N$, there exist multisets $X_1, \dots, X_M \subseteq X$ such that*

$$\left(\bigsqcup_{1 \le i \le M} X_i = X \right) \wedge \left(\forall i \in \{1, \dots, M\} : \sum_{a \in X_i} a = x_i \right). \qquad (2)$$

Here we define $\sqcup$ as multiset union such that when $A = \{a_1, \dots, a_n\}$ and $B = \{b_1, \dots, b_m\}$, $A \sqcup B = \{a_1, \dots, a_n, b_1, \dots, b_m\}$.

For example, $X = \{\underbrace{1, 1, \dots, 1}_{N}\}$ is an (M, N)-DM for any M. For a given set of integers $(x_1, \dots, x_M)$ such that $\sum_i x_i = N$, let $X_i = \{\underbrace{1, 1, \dots, 1}_{x_i}\}$. Then X_i's satisfy Eq. (2).

For another example,

$$X = \{1, 2, 4, 8, 16, 32, 37\}$$

is a $(2, 100)$-DM. For $x_1 = 41, x_2 = 59$,

$$X_1 = \{1, 8, 32\}, \quad X_2 = \{2, 4, 16, 37\}$$

satisfy Eq. (2).

It is clear that $\sum_{a \in X} a = N$ if X is an (M, N)-DM.

Here we consider the following protocol. For a publicly known $(M, N^{(i)})$-DM $X = \{a_1, a_2, \dots, a_u\}$, voter $\mathcal{V}_i$ has u bundles of ballots such that the l-th bundle consists of a_l ballots. So, totally he has $N^{(i)}$ ballots. He votes each bundle (instead of each ballot) for one of M candidates using multi-way voting procedure. He can vote u bundles for different candidates, but can not divide ballots in a bundle and vote for multiple candidates. The tallying is done as same as the multi-way voting protocol, but a bundle of a_l ballots is weighted a_l times.

The property of DM assures that the voter can vote $v_j^{(i)}$ ballots for j-th candidate for all $(v_1^{(i)}, \ldots, v_M^{(i)})$. Therefore, the above system is a divisible voting protocol. (In fact, if $X = \{1, 1, \ldots, 1\}$ and $u = N^{(i)}$, the above protocol is identical to the iteration-type naive protocol.)

In this protocol, the number of iterations $\mathcal{V}_i$ has to do is the number of elements of the divisible multiset. Next, we show that we can construct an (M, N)-DM that includes fewer elements. Then we can save the number of iterations.

Theorem 1 *For any positive integer N, we can construct a $(2, N)$-DM that includes $\lfloor \log_2 N \rfloor + 1$ elements.*

Theorem 2 *For any positive integer N and any integer $M (\geq 3)$, we can construct an (M, N)-DM that includes at most $(M-1)\left(\left\lfloor \log_2 \left\lceil \frac{N}{M} \right\rceil \right\rfloor + 1\right) + 1$ elements.*

The proofs of these theorems are shown in Appendix.

5.2 Improvement of modified separate-type

The second efficient construction can be considered as an improved one of modified separate-type naive protocol.

Assume that a voter wants to vote v_j ballots for j-th candidate. In modified separate-type naive protocol, the voter encrypts v_j for all j and proves that Eq. (1) holds using 1-out-of-$(N^{(i)} + 1)$ PMP. In the second construction, he first writes v_j with L-adic representation for some $L (\leq N^{(i)} + 1)$. Let $(b_{j,t}, \ldots, b_{j,1}, b_{j,0})$ be the L-adic representation, that is, $b_{j,l} \in \{0, 1, \ldots, L-1\}$ and $v_j = \sum_{l=0}^{t} L^l \ b_{j,l}$, where $t = \lfloor \log_L N^{(i)} \rfloor$. Then he encrypts $b_{j,l}$ for all j and l and proves that $b_{j,l} \in \{0, 1, \ldots, L-1\}$ holds using 1-out-of-L PMP.

Concrete protocol is described as follows.

1. In advance, n voting servers share a decryption key of a public-key probabilistic encryption function E.
2. Let $(b_{j,t}^{(i)}, \ldots, b_{j,1}^{(i)}, b_{j,0}^{(i)})$ be a L-adic representation of $v_j^{(i)}$ $(1 \leq j \leq M)$. For each j, $\mathcal{V}_i$ computes $(e_{j,t}^{(i)}, \ldots, e_{j,1}^{(i)}, e_{j,0}^{(i)})$ where
$$e_{j,l}^{(i)} \in E(b_{j,l}^{(i)}),$$

then opens $(e_{j,t}^{(i)}, \ldots, e_{j,0}^{(i)})$, and proves that $e_{j,l}^{(i)}$ is valid for all l using 1-out-of-L PMP. In addition, he shows that

$$\prod_{j=1}^{M} \prod_{l=0}^{t} \left(e_{j,l}^{(i)}\right)^{L^l} \in E(N^{(i)}).$$

It is enough for the voter by publishing a random number which is used to encrypt to show above fact.

3. After the voting period,

$$e_j^{(sum)} \triangleq \prod_i \prod_{l=0}^{t} \left(e_{j,l}^{(i)}\right)^{L^l}$$

is computed. Next n servers jointly decrypt $e_j^{(sum)}$ and obtain $v_j^{(sum)}$. $v_j^{(sum)}$ is the number of ballots j-th candidate got.

It is clear that

$$e_j^{(sum)} \in E\left(\sum_i \sum_{l=0}^{t} L^l \ b_{j,l}^{(i)}\right) = E\left(\sum_i v_j^{(i)}\right).$$

Then

$$v_j^{(sum)} = \sum_i v_j^{(i)}.$$

5.3 Efficiency

Table 3 shows the comparison of efficient protocols. Similar to Table 1 and 2, column "Total" shows the total complexity of each type when cost of one PMP is τ times of that of computing and sending one plaintext, and one 1-out-of-x PMP equals x times of one PMP. "DM+combined" and "DM+separate" denote the first efficient construction in which combine-type and separate-type multi-way voting protocol is used, respectively. "L-adic separate" denotes the second construction.

Comparing with Table 2, our constructions are more efficient than naive protocols.

The first construction with a separate-type multi-way voting protocol is clearly inferior to the others.

Figure 1 shows that the total costs of L-adic separate construction for $L = 1, 2, 3, 4, 5, 10$ where $\tau = 1$ and $M = 5$. It shows that $L = 3$ or $L = 4$

Table 3: Comparison of cost for a voter

Type	Total
DM+combined	$(\tau M+1)\left((M-1)\left(\left\lfloor\log_2\left\lceil\frac{N^{(i)}}{M}\right\rceil\right\rfloor+1\right)+1\right)$
DM+separate	$(2\tau+1)M\left((M-1)\left(\left\lfloor\log_2\left\lceil\frac{N^{(i)}}{M}\right\rceil\right\rfloor+1\right)+1\right)$
L-adic separate	$(\tau L+1)M\left(\lfloor\log_L N^{(i)}\rfloor+1\right)$

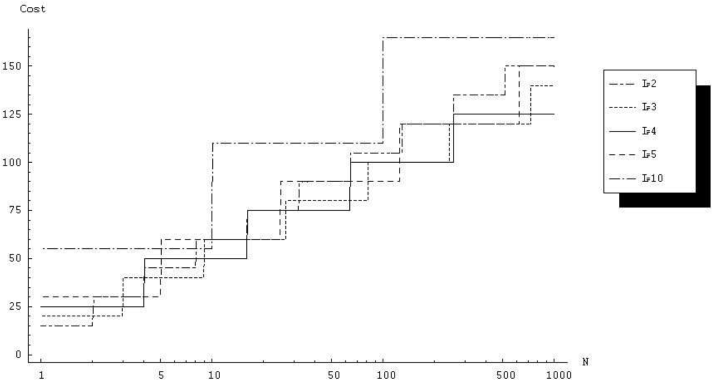

Figure 1: Cost of L-adic construction where $\tau = 1$ and $M = 5$

is most efficient in most cases. Figure 2 shows that the total cost of 4-adic separate construction and DM+combined construction where $\tau = 1, L = 4$ and $M = 5$. It shows that DM+combined are more efficient only when $N = 4, 5$, otherwise 4-adic separate construction is more efficient.

Next, look at the efficiency of tallying. Table 4 shows the number of decryption procedures and maximal number of plaintexts obtained by the decryption. Here we omit the first construction with separate-type.

Table 4: Comparison of tallying

Type	#decrypt	Message space
DM+combined	1	$(N^{(sum)}+1)^M$
4-adic separate	M	$N^{(sum)}$

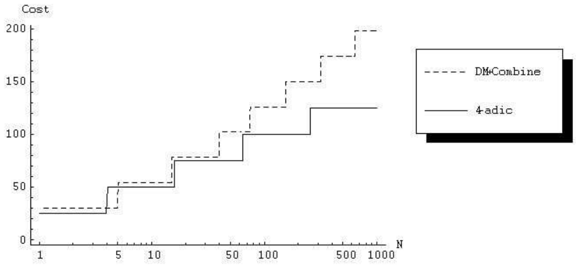

Figure 2: Cost comparison between L-adic construction and DM+combined where $\tau = 1, L = 4$ and $M = 5$

If Paillier cryptosystem is used, T_j's are easily computed from $T_1 + (N+1)T_2 + \cdots + (N+1)^{M-1}T_M$. Therefore, the first construction is more efficient.

If we use ElGamal scheme, large message space makes the tallying costly. Then the second construction is more efficient.

References

[1] C. Park, K. Itoh and K. Kurosawa, "Efficient Anonymous Channel and All/Nothing Election Scheme," In Proc. of EUROCRYPT '93, pp. 248–259 (1993).

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

[3] R. Cramer, R. Gennaro, and B. Schoenmakers, "A Secure and Optimally Efficient Multi-Authority Election Scheme," Proc. of Eurocrypt '97, LNCS 1233, pp. 103–118 (1997).

[4] D. Dolev, C. Dwork, M. Naor, "Non-malleable cryptography," Proc. of STOC '91, pp. 542–552 (1991).

[5] R. Cramer, I. Damgard, and B. Schoenmakers, "Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols," Proc. of Eurocrypt '94, LNCS 839, pp. 174–187 (1994).

[6] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," Proc. of Eurocrypt '99, LNCS 1592, pp.223–238 (1999).

A Proofs of Theorem 1

When $N = 1$, a multiset $X = \{1\}$ is obviously a $(2,1)$-DM. We will prove in $N \geq 2$. For any N, following equation holds.

$$\begin{aligned} N &= \left(2^{\lfloor \log_2 N \rfloor} - 1\right) + \left(N - 2^{\lfloor \log_2 N \rfloor} + 1\right) \\ &= \left(\sum_{i=0}^{\lfloor \log_2 N \rfloor - 1} 2^i\right) + \left(N - 2^{\lfloor \log_2 N \rfloor} + 1\right) \end{aligned}$$

Note that $N - 2^{\lfloor \log_2 N \rfloor} + 1 \geq 1$ holds from $2^{\lfloor \log_2 N \rfloor} \leq N$. Let define X as

$$X = \left\{2^i \mid 0 \leq i \leq \lfloor \log_2 N \rfloor - 1\right\} \sqcup \left\{N - 2^{\lfloor \log_2 N \rfloor} + 1\right\}.$$

Then $\sum_{a \in X} a = N$. Hereafter, we prove X is a $(2,N)$-DM.

Let x_1, x_2 be integers which are not negative and satisfy $x_1 + x_2 = N$. Here we does not lose generality when we assume that $0 \leq x_1 \leq x_2 \leq N$. Then

$$x_1 \leq 2^{\lfloor \log_2 N \rfloor} - 1 \tag{3}$$

holds.

When $x_1 = 0$, $x_2 = N$, let

$$X_1 = \phi\ ,\ X_2 = X.$$

Then eq. (2) holds.

When $x_1 \geq 1$, let $\left(\beta_{\lfloor \log_2 x_1 \rfloor}, \ldots, \beta_0\right)$ be a binary representation of x_1 and

$$\begin{aligned} X_1 &= \left\{2^i \mid 0 \leq i \leq \lfloor \log_2 x_1 \rfloor, \beta_i = 1\right\}, \\ X_2 &= X - X_1. \end{aligned}$$

From eq. (3), $\lfloor \log_2 x_1 \rfloor \leq \lfloor \log_2 N \rfloor - 1$. Thus,

$$X_1 \subseteq \left\{2^i \mid 0 \leq i \leq \lfloor \log_2 N \rfloor - 1\right\} \subset X.$$

From

$$\sum_{a \in X_1} a = \sum_{i=0}^{\lfloor \log_2 x_1 \rfloor} \beta_i 2^i = x_1$$

and

$$\sum_{a \in X_2} a = \sum_{a \in X} a - \sum_{a \in X_1} a = N - x_1 = x_2,$$

eq. (2) holds.

B Proofs of Theorem 2

For any positive integer N,

$$N = \sum_{i=1}^{M} \left(\left\lfloor i\frac{N}{M} \right\rfloor - \left\lfloor (i-1)\frac{N}{M} \right\rfloor \right)$$

holds. Let N_i $(1 \leq i \leq M)$ be

$$N_i = \left\lfloor i\frac{N}{M} \right\rfloor - \left\lfloor (i-1)\frac{N}{M} \right\rfloor .$$

Then N_i is a non-negative integer and N_M is a positive integer.

Lemma 1

$$\left\lceil \frac{N}{M} \right\rceil - 1 \leq \left\lfloor \frac{N}{M} \right\rfloor \leq N_i \leq \left\lceil \frac{N}{M} \right\rceil . \tag{4}$$

(Proof) When we assume $N_i \leq \left\lfloor \frac{N}{M} \right\rfloor - 1$, $\left\lfloor i\frac{N}{M} \right\rfloor \leq \left\lfloor (i-1)\frac{N}{M} \right\rfloor + \left\lfloor \frac{N}{M} \right\rfloor - 1$. It is contradict from $\left\lfloor i\frac{N}{M} \right\rfloor > i\frac{N}{M} - 1$ and $\left\lfloor (i-1)\frac{N}{M} \right\rfloor + \left\lfloor \frac{N}{M} \right\rfloor - 1 \leq (i-1)\frac{N}{M} + \frac{N}{M} - 1 = i\frac{N}{M} - 1$. Thus

$$\left\lfloor \frac{N}{M} \right\rfloor \leq N_i. \tag{5}$$

If we assume $\left\lceil \frac{N}{M} \right\rceil + 1 \leq N_i$, $\left\lfloor (i-1)\frac{N}{M} \right\rfloor + \left\lceil \frac{N}{M} \right\rceil + 1 \leq \left\lfloor i\frac{N}{M} \right\rfloor$. It is contradict from $\left\lfloor (i-1)\frac{N}{M} \right\rfloor + \left\lceil \frac{N}{M} \right\rceil + 1 > \left((i-1)\frac{N}{M} - 1 \right) + \frac{N}{M} + 1 = i\frac{N}{M}$ and $\left\lfloor i\frac{N}{M} \right\rfloor \leq i\frac{N}{M}$. Thus

$$N_i \leq \left\lceil \frac{N}{M} \right\rceil . \tag{6}$$

From eq. (5) and (6),

$$\left\lceil \frac{N}{M} \right\rceil - 1 \leq \left\lfloor \frac{N}{M} \right\rfloor \leq N_i \leq \left\lceil \frac{N}{M} \right\rceil . \tag{7}$$

Q.E.D.

If $N_i = 0$ $(1 \leq i \leq M-1)$, let $Y_i = \phi$, otherwise let Y_i be a $(2, N_i)$-DM constructed by using Theorem 1. Let $Y_M = \{N_M\}$. Let X be

$$X = \bigsqcup_{1 \leq i \leq M} Y_i.$$

From lemma 1 and Theorem 1,

$$|X| = \sum_{i=1}^{M-1} |Y_i| + |Y_M| \leq (M-1)\left(\left\lfloor \log_2 \left\lceil \frac{N}{M} \right\rceil \right\rfloor + 1\right) + 1$$

holds.

From now, we prove that X is an (M, N)-DM. Let x_i $(1 \leq i \leq M)$ be a non-negative integer which fulfills $\sum_{i=1}^{M} x_i = N$. Without loss of generality, we assume that $0 \leq x_1 \leq x_2 \leq \cdots \leq x_M \leq N$. We show that following equation holds when $j = M-1$ by using mathematical induction.

$$\begin{aligned} &\exists X_1 \subseteq X \ \cdots \ \exists X_j \subseteq X \\ &\quad \bigsqcup_{1 \leq i \leq j} X_i \subseteq \bigsqcup_{1 \leq i \leq j} Y_i \ \wedge \ \forall i \in \{1, \ldots, j\} : \sum_{a \in X_i} a = x_i \end{aligned} \tag{8}$$

At first, we show that eq. (8) holds when $j = 1$. If $N_1 + 1 \leq x_1$, then $\sum_{i=1}^{M} x_i \neq N$ from $N = M\frac{N}{M} < M\left(\left\lfloor \frac{N}{M} \right\rfloor + 1\right) = M(N_1 + 1) \leq \sum_{i=1}^{M} x_i$. Thus $x_1 \leq N_1$.

When $N_1 = 0$, eq. (8) holds with $X_1 = \phi = Y_1$ from $x_1 = 0$. When $N_1 \geq 1$ there exists multiset $X_1 \subseteq Y_1$ which satisfies $\sum_{a \in X_1} a = x_1$ for any x_1 $(0 \leq x_1 \leq N_1)$, because Y_1 is a $(2, N_1)$-DM. Thus eq. (8) holds.

Next we show that if eq. (8) holds in the case of $j = k$, then it holds in the case of $j = k+1$, where $1 \leq k \leq M-2$. It is clear that $\sum_{i=1}^{k+1} x_i \leq \sum_{i=1}^{k+1} N_i$ from $0 \leq x_1 \leq \cdots \leq x_M \leq N$ and the definition of N_i. Thus, we must show only that there exists multiset

$$X_{k+1} \subseteq \bigsqcup_{1 \leq i \leq k+1} Y_i - \bigsqcup_{1 \leq i \leq k} X_i = \left(\bigsqcup_{1 \leq i \leq k} Z_i\right) \sqcup Y_{k+1}$$

which satisfies $\sum_{a \in X_{k+1}} a = x_{k+1}$ for any x_{k+1}. Note that $0 \leq x_{k+1} \leq \sum_{i=1}^{k+1} N_i - \sum_{i=1}^{k+1} x_i$. Here let multiset Z_i be

$$Z_i = \begin{cases} \phi & i = 0 \\ Y_i \setminus \bigsqcup_{1 \leq l \leq k} X_l & 1 \leq i \leq k \\ Y_{k+1} & i = k+1. \end{cases}$$

When $x_{k+1} = \sum_{i=1}^{k+1} N_i - \sum_{i=1}^{k} x_i$ holds, let

$$X_{k+1} = \bigsqcup_{1 \leq i \leq k+1} Y_i - \bigsqcup_{1 \leq i \leq k} X_i,$$

then eq. (8) holds from

$$\sum_{a\in X_{k+1}} a \;=\; \sum_{i=1}^{k+1}\sum_{a\in Y_i} a - \sum_{i=1}^{k}\sum_{a\in X_i} a \;=\; \sum_{i=1}^{k+1} N_i - \sum_{i=1}^{k} x_i \;=\; x_{k+1}$$

and

$$\bigsqcup_{1\le i\le k+1} X_i = \bigsqcup_{1\le i\le k+1} Y_i.$$

When $x_{k+1} < \sum_{i=1}^{k+1} N_i - \sum_{i=1}^{k} x_i$, there exists d $(0 \le d \le k)$ which satisfies

$$\sum_{i=0}^{d}\sum_{a\in Z_i} a \;\le\; x_{k+1} \;\le\; \sum_{i=0}^{d+1}\sum_{a\in Z_i} a - 1 \;=\; \sum_{i=0}^{d}\sum_{a\in Z_i} a + \sum_{a\in Z_{d+1}} a - 1.$$

Then,

$$0 \le \left(x_{k+1} - \sum_{i=0}^{d}\sum_{a\in Z_i} a\right) \le \sum_{a\in Z_{d+1}} a - 1.$$

On the other hand,

$$\sum_{a\in Z_{d+1}} a - 1 \le \sum_{a\in Y_{d+1}} a - 1 = N_{d+1} - 1$$

from $Z_{d+1} \subseteq Y_{d+1}$. Then.

$$N_{d+1} - 1 \le \left\lceil \frac{N}{M} \right\rceil - 1 \le N_{k+1}$$

from Lemma 1. So

$$0 \le \left(x_{k+1} - \sum_{i=0}^{d}\sum_{a\in Z_i} a\right) \le N_{k+1}$$

holds. Therefore there exists $Y'_{k+1} \subseteq Y_{k+1}$ which satisfies

$$\sum_{a\in Y'_{k+1}} a = \left(x_{k+1} - \sum_{i=0}^{d}\sum_{a\in Z_i} a\right),$$

because Y_{k+1} is a $(2, N_{k+1})$-DM.

Here, let

$$X_{k+1} = \left(\bigsqcup_{0\le i\le d} Z_i\right) \cup Y'_{k+1}.$$

Then eq. (8) holds from

$$\begin{aligned}
\sum_{a\in X_{k+1}} a &= \sum_{i=0}^{d}\sum_{a\in Z_i} a + \sum_{a\in Y'_{k+1}} a \\
&= \sum_{i=0}^{d}\sum_{a\in Z_i} a + \left(x_{k+1} - \sum_{i=0}^{d}\sum_{a\in Z_i} a \right) \\
&= x_{k+1}
\end{aligned}$$

and

$$\begin{aligned}
X_{k+1} &= \left(\bigsqcup_{0\le i\le d} Z_i \right) \cup Y'_{k+1} \\
&\subseteq \left(\bigsqcup_{1\le i\le k} Z_i \right) \cup Y_{k+1} \\
&= \left(\bigsqcup_{1\le i\le k} Y_i - \bigsqcup_{1\le i\le k} X_i \right) \cup Y_{k+1},
\end{aligned}$$

$$\bigsqcup_{1\le i\le k+1} X_i \subseteq \bigsqcup_{1\le i\le k+1} Y_i.$$

Consequently, there exist $X_1, \ldots, X_{M-1}$ which satisfy eq. (8) when $j = M-1$.

Finally, set

$$X_M = X - \bigcup_{1\le i\le M-1} X_i,$$

then eq. (2) holds from

$$\sum_{a\in X_M} a = \sum_{a\in X} a - \sum_{i=1}^{M-1}\sum_{a\in X_i} a = N - \sum_{i=1}^{M-1} x_i = x_M.$$