Correct, Private, Flexible And Efficient Range Test

Kun Peng[1], Colin Boyd[1], Ed Dawson[1], and Eiji Okamoto,[2]

[1] Information Security Research Centre
IT Faculty, Queensland University of Technology
{k.peng, c.boyd, e.dawson}@qut.edu.au
http://www.isrc.qut.edu.au
[2] University of Tsukuba, Japan

Abstract. In a range test, one party holds a ciphertext and needs to test whether the message encrypted in the ciphertext is within a certain interval range. In this paper, a range test protocol is proposed, where the party holding the ciphertext asks another party holding the private key of the encryption algorithm to help him. These two parties run the protocol to implement the test. The test returns TRUE if and only if the encrypted message is within the certain interval range. If the two parties do not conspire, no information about the encrypted message is revealed from the test except what can be deduced from the test result. Advantages of the new protocol over the existing related techniques are that it achieves correctness, soundness, flexibility, high efficiency and privacy simultaneously.

Keywords: interval range, range test, specialized zero test, correctness, soundness

1 Introduction

In a range test, one party (the *tester*) holds a ciphertext and needs to test whether the message encrypted in the ciphertext is within a certain interval range. This test is frequently required in cryptographic applications like e-auction [1, 15], electronic voting [3, 10, 11, 13], electronic finance [8], group signature [7], publicly verifiable secret sharing [14] and verifiable encryption [2]. The following properties are desired in a range test.

- Correctness: If the encrypted message is in the interval range, the test outputs TRUE.
- Soundness: If the test outputs TRUE, the encrypted message is in the interval range.
- Privacy: No information about the encrypted message is revealed except what can be deduced from the test result.
- Flexibility: The limitation on the range size, encryption format and participants should be as little as possible.

The simplest way to implement a range test is using multiple equality tests linked by "OR" logic to test whether

$$\begin{gathered}\text{the encrypted message equals the first number in the range } \vee \\ \text{the encrypted message equals the second number in the range } \vee \\ \ldots \vee \text{ the encrypted message equals the last number in the range}\end{gathered}$$

without revealing the encrypted message equals which number in the range. This method is called naive range test in this paper. Two special methods, zero knowledge proof of "OR" logic by Cramer *et al* [9] or the verification technique called zero test [17], can be employed to implement naive range test without compromising privacy. These two methods can be flexibly employed so that various ranges (e.g. ranges with very large size), participant models (with or without prover) and encryption formats (even commitment formats) can be used. Although naive range test can be flexible, correct, sound and private, it is very inefficient as its cost is linear in the size of the range. If the ciphertext to test is encrypted in some special encryption format (e.g. encrypted bit by bit), cost of naive range test can be reduced to be linear in the logarithm of the range size. However, ciphertext in practical cryptographic applications (especially when secure computation of ciphertext is needed) cannot be often encrypted in special encryption format. So for the sake of flexibility, naive range test generally needs a cost linear in the size of the range. Even if the special encryption format can be employed to improve efficiency, naive range test is still too inefficient.

Some cryptographic techniques [2, 5, 14, 6, 8] are related to range test. They prove that a committed message is within a certain interval range and are called RPC (range proof of commitment) schemes in this paper. In RPC schemes, a prover with the knowledge of the committed message is needed to give a zero knowledge proof that the message is in the certain interval range. Although RPC schemes are efficient as their cost is independent of the size of the range, they have some drawbacks. Firstly, in many applications like e-auction and e-voting, encrypted messages instead of committed messages are required to be tested. So RPC schemes (especially [5], which requires a certain commitment format) cannot be employed in these applications. Secondly, the message to be tested may be generated by multiple parties and unknown to anybody. For example, in the k^{th}-bid auction [1, 15], the seller has to test whether the number of bids at a price is less than k without revealing the bids. As no single bidder knows the sum, nobody can provide any proof to implement the test. In another example, e-banking, it is required to test whether a sum of money is below a threshold without revealing it while nobody knows the sum as it accumulates multiple dealings. So a prover is not always available. Thirdly, most RPC schemes [2, 14, 6, 8] cannot guarantee correctness and soundness at the same time. The only correct and sound scheme among them is Boudot [5], which is only asymptotically (instead of absolutely) sound. Finally, all the known RPC schemes can work only when the range to test is many magnitudes smaller than the size of the message space of the commitment algorithm.

As the drawbacks of RPC schemes greatly reduce their reliability, flexibility and limit their application, in many circumstances inefficient naive range test has to be employed. So a range test protocol is proposed in this paper, which is much more efficient than naive range test and overcomes the drawbacks of the RPC schemes. In the new range test protocol, two parties are involved: a tester and a (decryption) authority, who can be acted by multiple entities through a threshold key sharing mechanism. The tester holds the ciphertext to test. The private key to decrypt the ciphertext is held by the authority. So the tester asks the authority for help and they run the protocol to implement the test. If the encrypted message is in the certain interval range, the protocol outputs TRUE. If the encrypted message is not in the certain interval range, the protocol outputs FALSE. Namely, the new test protocol is correct and sound. If the two parties do not conspire, no information about the encrypted message is revealed from the test except what can be deduced from the test result. The new protocol is flexible as it accepts ranges of the same magnitude as the size of the message space of the encryption algorithm and does not need any prover with knowledge of the encrypted message. The new protocol is efficient as its computational cost is independent of the range size. This new protocol can overcome the drawbacks of RPC schemes. In the example of k^{th}-bid auction, the seller acts as the tester while an auctioneer acts as the authority to help the seller to determine whether the number of bids at a price is over k without revealing the bids or the number. In the example of e-banking, two servers (neither knowing the sum of the money) act as the tester and authority to test the range of the sum. If the two servers do not conspire, the sum is not revealed.

Two different adversary models are used in this paper to analyse correctness and soundness. In a negatively-malicious model, the adversary does not deviate from the protocol in his attack. In an actively-malicious model, the adversary may attack in any way including deviating from the protocol. Like all the RPC schemes, this paper does not consider CCA (chosen ciphertext attack) model when analysing privacy. As to our knowledge all the secure computation schemes related to range test employ homomorphic encryption or commitment, it is senseless to talk about CCA. Actually, only privacy in CPA (chosen plaintext attack) model is achieved in this paper, while CCA privacy is left as an open question.

The structure of this paper is as follows. Parameters and symbols to be used in the paper are defined in Section 2. In Section 3, a building block, specialized zero test, is designed. In Section 4, three range test protocols are proposed. They are not independent. Instead, each protocol is an optimization of the previous one.

2 Preliminary Work

Parameters, symbols and encryption systems to be used later are described in this section. Two additive homomorphic semantically-secure encryption systems[3] (e.g. modified ElGamal encryption [12, 13]) are needed in this paper. They are called the first encryption system and the second encryption system respectively later in this paper. The ciphertext to test is encrypted in the first encryption system, while the tester holds the ciphertext and the authority holds the private key of the first encryption system. To implement the range test, a second encryption system is set up and its private key is also held by the private key. The public keys of both encryption systems are public, so that both the authority and the tester can use both encryption systems for encryption. The message spaces of the two encryption systems are Z_{p_1} and Z_{p_2} respectively. It is required in this paper that $p_2 \geq 3p_1$ and p_2 is a prime.

Although any additive homomorphic semantically-secure encryption algorithm like Paillier encryption [16] can be employed in the first encryption system, it is suggested to employ the modified ElGamal encryption [12, 13] in the second encryption system so that p_2 is a prime. For simplicity, the modified ElGamal encryption is employed in both encryption systems in this paper. Details about the two (modified ElGamal) encryption systems are as follows where index i stands for the i^{th} encryption system.

- p'_i is the multiplicative modulus in the i^{th} encryption system.
- $< g_i >$ is a cyclic subgroup of $Z^*_{p'_i}$ with generator g_i, which has a prime order p_i.
- The message space in the i^{th} encryption system is Z_{p_i}.
- $x_i \in Z_{p_i}$ is the private key in the i^{th} encryption system. (g_i, y_i) is the public key in the i^{th} encryption system where $y_i = g_i^{x_i} \bmod p'_i$.
- $E_i(m)$ stands for encryption of message m in the i^{th} encryption system: $(g_i^r \bmod p'_i, g_i^m y_i^r \bmod p'_i)$ where r is randomly chosen from Z_{p_i}.
- The product of two ciphertexts $c_1 = (a_1, b_1)$ and $c_2 = (a_2, b_2)$ in the i^{th} encryption system is $(a_1 a_2 \bmod p'_i, b_1 b_2 \bmod p'_i)$. Inversion of a ciphertext $c = (a, b)$ in the i^{th} encryption system is $(a^{-1} \bmod p'_i, b^{-1} \bmod p'_i)$. With multiplication and inversion defined, definition of exponentiation and division is automatically obtained.
- $D_i(c)$, decryption function of ciphertext $c = (a, b)$ in the i^{th} encryption system, is $\log_{g_i} b/a^{x_i}$. Although normally decryption in the modified ElGamal encryption algorithm needs a logarithm search and is not efficient, it is only required to test whether the message is zero or not in any decryption in this paper, which does not need any logarithm search and is very efficient.

[3] An encryption algorithm with message space Z_p and decryption function $D()$ is additive homomorphic if $D(c_1) + D(c_2) = D(c_1 c_2) \bmod p$ for any ciphertexts c_1 and c_2. An encryption algorithm is semantically-secure if given a ciphertext c and two messages m_1 and m_2, such that $c = E(m_i)$ where $i = 1$ or 2, there is no polynomial algorithm to find out i.

Later in this paper, encryption, decryption, ciphertext multiplication, ciphertext inversion and ciphertext exponentiations are computed as described here in this section. The other symbols to be used in this paper are listed in Table 1.

Table 1. Symbols

%	modulus computation
$\|a\|$	the absolute value of an integer a
$[S]$	the size of a set S
$\binom{a}{b}$	the number of possible choices of b elements from a candidate elements

3 A Building Block — Specialized Zero Test

Zero test is a technique to test whether there is at least one null ciphertext (encryption of zero) among multiple ciphertexts. A zero test must be private, namely nothing about the messages encrypted in the ciphertexts can be deduced from the test except whether there is at least one null ciphertext among them. The existing zero test technique (e.g. the so-called complex zero test in [17] or similar technique in [4]) cannot obtain complete privacy as it may reveal some information about the number of null ciphertexts. Fortunately, in this paper it is only desired to test whether there is one null ciphertext among multiple ciphertexts where there is at most one null ciphertext among them. This will be accomplished by modifying the zero test technique from [17] into a new cryptographic primitive: *specialized zero test*, which can achieve complete privacy in the application in this paper. A specialized zero test examines whether there is one null ciphertext among multiple ciphertexts encrypted using the second encryption system described in Section 2 where there is at most one null ciphertext among them. While the zero test technique in [17] is a multiparty protocol, only two parties are involved in the specialized zero test in this paper: a tester A_1 and an authority A_2. A_1 holds ciphertexts $c_1, c_2, \ldots, c_n$ in the second encryption system where there is at most one null ciphertext among them. A_2 holds the private key of the second encryption system. In the specialized zero test A_2 assists A_1 to test whether there is one null ciphertext among $c_1, c_2, \ldots, c_n$. Three properties are desired in specialized zero test.

- Correctness: if there is one null ciphertext in $c_1, c_2, \ldots, c_n$, the test result is TRUE.
- Soundness: if the test result is TRUE, there is one null ciphertext in $c_1, c_2, \ldots, c_n$.
- Privacy: after the test, each party learns only the test result and what can be deduced from it, as long as the authority and the tester do not collude.

The test protocol is denoted as $ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_n\ )$ and described in Figure 1.

1. A_1 chooses $\pi()$, a permutation on $\{1, 2, \ldots, n\}$, and random integers r_i from $Z_{p_2} - \{0\}$ for $i = 1, 2, \ldots, n$. Then he calculates $c'_i = c^{r_i}_{\pi(i)}$ for $i = 1, 2, \ldots, n$. He sends $c'_1, c'_2, \ldots, c'_n$ to A_2.
2. A_2 calculates $d_i = D_2(c'_i)$ for $i = 1, 2, \ldots, n$ one by one until one d_i is found to be zero or all the n ciphertexts are decrypted. A_2 publishes the output of the zero test as follows.

$$ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_n\ ) = \begin{cases} \mathsf{TRUE} & \text{if a zero is found in } d_i \text{ for } i = 1, 2, \ldots, n \\ \mathsf{FALSE} & \text{if no zero in } d_i \text{ for } i = 1, 2, \ldots, n \end{cases} \tag{1}$$

Fig. 1. Specialized zero test

Theorem 1. *The specialized zero test is correct in the negatively-malicious model. More precisely, if nobody deviates from the protocol and there is one zero encrypted in $c_1, c_2, \ldots, c_n$, then*
$ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_n\ ) = \mathsf{TRUE}$.

Proof: As $c'_i = c^{r_i}_{\pi(i)}$ for $i = 1, 2, \ldots, n$ and the encryption algorithm is additive homomorphic, $D_2(c'_i) = D_2(c^{r_i}_{\pi(i)}) = r_i D_2(c_{\pi(i)}) \bmod p_2$ for $i = 1, 2, \ldots, n$. Suppose $D_2(c_j) = 0$ where $1 \leq j \leq n$, then $D_2(c'_{\pi^{-1}(j)}) = r_{\pi^{-1}(j)} \times D_2(c_j) = r_{\pi^{-1}(j)} \times 0 \bmod p_2 = 0$. So there is at least one zero in $D_2(c'_1), D_2(c'_2), \ldots, D_2(c'_n)$. Therefore, $ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_n\ ) = \mathsf{TRUE}$. □

Theorem 2. *The specialized zero test is sound in the negatively-malicious model. More precisely, if nobody deviates from the protocol and $ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_n\ ) = \mathsf{TRUE}$, then there is at least one null ciphertext in $c_1, c_2, \ldots, c_n$.*

Proof: As $c'_i = c^{r_i}_{\pi(i)}$ for $i = 1, 2, \ldots, n$ and the encryption algorithm is additive homomorphic, $D_2(c'_i) = D_2(c^{r_i}_{\pi(i)}) = r_i D_2(c_{\pi(i)}) \bmod p_2$ for $i = 1, 2, \ldots, n$. As $ZM\ (\ A_1,\ A_2 \mid c_1, c_2, \ldots, c_m\ ) = \mathsf{TRUE}$, there is at least one zero encrypted in $c'_1, c'_2, \ldots, c'_n$. Suppose $D_2(c'_j) = 0$ and $1 \leq j \leq n$. Then $r_j D_2(c_{\pi(j)}) = 0 \bmod p_2$. As p_2 is a prime and r_j is chosen from $Z_{p_2} - \{0\}$, $D_2(c_{\pi(j)}) = 0$. Therefore, there is at least one null ciphertext in $c_1, c_2, \ldots, c_n$. □

Theorem 3. *The specialized zero test is private. More precisely, if A_1 and A_2 do not collude, the only knowledge of either of them about $D_2(c_1), D_2(c_2), \ldots, D_2(c_n)$ is the test result.*

Proof: As A_1 has no knowledge of the private key and the encryption algorithm is semantically-secure, nothing about $D_2(c_1), D_2(c_2), \ldots, D_2(c_n)$ is revealed to him if A_2 does not help to decrypt any message. As A_2 does not collude with A_1, A_2 only tells A_1 the test result, which is A_1's only knowledge about $D_2(c_1), D_2(c_2), \ldots, D_2(c_n)$.

Although A_2 has the private key, his knowledge is limited by the ciphertexts sent to him. As A_1 does not collude with him, only $c'_1, c'_2, \ldots, c'_n$ are sent to A_2. So his only knowledge from the test is $D_2(c'_1)||D_2(c'_2)||\ldots||D_2(c'_n)$, which is called his knowledge transcript. Suppose T_1 and T_2 are two knowledge transcripts from two inputs with the same test result. Note that $c'_i = c^{r_i}_{\pi(i)}$, p_2 is a prime and r_i is randomly chosen from $Z_{p_2} - \{0\}$ as A_1 does not collude with A_2. So $D_2(c'_i)$ is distributed uniformly in $Z_{p_2} - \{0\}$ if $D_2(c_{\pi(i)}) \neq 0$ or $D_2(c'_i) = 0$ if $D_2(c_{\pi(i)}) = 0$. So if A_1 does not collude with A_2, when the test result is TRUE, both T_1 and T_2 are uniformly distributed in $\{ \ T \mid T \in \{Z_{p_2}\}^n, T$ contains one $0\}$; when the test result is FALSE, both T_1 and T_2 are uniformly distributed in $(Z_{p_2} - \{0\})^n$. As A_2's knowledge transcripts from any two inputs with the same test result are indistinguishable from each other without A_1's collusion, no information about the input is revealed to A_2 except for the test result without A_1's collusion. □

4 The New Range Test Protocol

In the new range test protocol, given a ciphertext c encrypted in the first encryption system described in Section 2, the tester runs a two-party protocol with the authority to examine whether $D_1(c)$ is in a certain interval range without knowing or revealing $D_1(c)$. In this protocol there is a limitation about the range size: no more than $p_1/5$, which is of the same magnitude as the size of the message space. As p_1 is very large (e.g. 1024 bits long) in any practical encryption algorithm, the range is large enough for normal applications. For simplicity, it is assumed that the range involved in the test is Z_q where $5q \leq p_1$. Note that range test in any consecutive integer range in the message space with a size no more than $p_1/5$ can be easily reduced to a range test in a same-size range starting from zero due to homomorphism of the encryption algorithm. Three range test protocols are designed in this section based on a principle: $m \in Z_q$ if and only if $m\%q = m$, which can be tested by reducing it to multiple simpler tests and repeatedly exploiting homomorphism of the employed encryption algorithms. Firstly, a correct but only partially sound test protocol in the negatively-malicious model — *basic range test* — is described. Then a correct and sound test protocol in the negatively-malicious model, called *precise range test*, is designed based on two basic range tests. Finally, the precise range test is upgraded to *optimized precise range test* through a cut-and-choose mechanism, so that the tester can always get the correct result if he wants even in the actively-malicious model.

4.1 Basic Range Test

The basic range test is an interactive protocol between two parties: the tester and the authority. The tester is denoted as A_1, who possesses a ciphertext c in the first encryption system. The authority is denoted as A_2, who possesses the private keys of the two encryption systems. The basic range test protocol includes three steps. In the first step, m, the message encrypted in c is randomly shared between A_1 and A_2. Namely, A_1 holds random integer m_1, A_2 holds random integer m_2 such that $m = m_1 + m_2 \bmod p_1$. In the second step, A_2 transmits $E_2(m_2)$ and $E_2(m_2\%q)$ to A_1. In the third step, A_1 and A_2 perform a specialized zero test, during which A_1 provides some randomised and shuffled ciphertexts and A_2 decrypts them. The basic range test is denoted as $BR\ (\ A_1,\ A_2 \mid c\ )$ and described in Figure 2, such that

$$BR\ (\ A_1,\ A_2 \mid c\ ) = \begin{cases} \mathsf{TRUE} & \text{if } (3) = \mathsf{TRUE} \\ \mathsf{FALSE} & \text{if } (3) = \mathsf{FALSE} \end{cases}$$

1. A_1 randomly chooses m_1 from Z_{p_1}. He calculates $c_1 = E_1(m_1)$ and sends $c_2 = c/c_1$ to A_2.
2. (a) A_2 calculates $m_2 = D_1(c_2)$.
   (b) A_2 calculates $c'_2 = E_2(m_2)$ and $e_2 = E_2(m_2\%q)$ and sends them to A_1.
3. (a) A_1 calculates $c'_1 = E_2(m_1)$ and $e_1 = E_2(m_1\%q)$.
   (b) A_1 needs to perform the following logic test with the help of A_2:

$$\begin{aligned} D_2(e_1e_2/(c'_1c'_2)) = 0 \ \vee\ D_2(e_1e_2/(c'_1c'_2E_2(q))) = 0 \ \vee\ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0 \\ \vee\ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q - q))) = 0 \ \vee\ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q + q))) = 0 \qquad (2) \end{aligned}$$

   In logic expression (2), either all the five clauses are false or only one of them is true. So the logic test of (2) can be implemented through a specialized zero test:

$$\begin{aligned} ZM\ (\ A_1, A_2 \mid e_1e_2/(c'_1c'_2),\ e_1e_2/(c'_1c'_2E_2(q)),\ e_1e_2/(c'_1c'_2E_2(p_1\%q)), \\ e_1e_2/(c'_1c'_2E_2(p_1\%q - q)),\ e_1e_2/(c'_1c'_2E_2(p_1\%q + q))\ ) \qquad (3) \end{aligned}$$

Fig. 2. Basic range test

Theorem 4. *The basic range test is correct in the negatively-malicious model. More precisely, if nobody deviates from the protocol and* $0 \leq D_1(c) < q$, *the specialized zero test in Formula (3) outputs* TRUE.

Proof: Suppose $D_1(c) = m$. As $0 \leq D_1(c) < q$, $m\%q = m$. There are two important facts.

- As $c = c_1c_2$, $m = m_1 + m_2 \bmod p_1$. So, either (1): $m = m_1 + m_2$ or (2): $m = m_1 + m_2 - p_1$.
- It is always true that either (a): $(m_1 + m_2)\%q = m_1\%q + m_2\%q$ or (b): $(m_1 + m_2)\%q = m_1\%q + m_2\%q - q$.

So the proof is given in four different cases by combining the two possibilities in the first fact, (1) and (2), with the two possibilities in the second fact, (a) and (b): (1a), (1b), (2a) and (2b).

- (1a): According to additive homomorphism of the encryption algorithm

  $$D_2(e_1e_2/(c'_1c'_2)) = D_2(e_1e_2/(E_2(m_1)E_2(m_2))) = D_2(e_1) + D_2(e_2) - (D_2(E_2(m_1)) + D_2(E_2(m_2)) \bmod p_2 = m_1\%q + m_2\%q - (m_1 + m_2) \bmod p_2$$

  According to Condition (1) and Condition (a),

  $$D_2(e_1e_2/(c'_1c'_2)) = (m_1 + m_2)\%q - m \bmod p_2 = m\%q - m \bmod p_2 = 0$$

- (1b): According to additive homomorphism of the encryption algorithm

  $$D_2(e_1e_2/(c'_1c'_2E_2(q))) = D_2(e_1e_2/(E_2(m_1)E_2(m_2)E_2(q))) = D_2(e_1) + D_2(e_2) - (D_2(E_2(m_1)) + D_2(E_2(m_2)) - q \bmod p_2 = m_1\%q + m_2\%q - (m_1 + m_2) - q \bmod p_2$$

  According to Condition (1) and Condition (b),

  $$D_2(e_1e_2/(c'_1c'_2E_2(q))) = (m_1+m_2)\%q+q-m-q \bmod p_2 = m\%q+q-m-q \bmod p_2 = 0$$

- Proof of (2a) and (2b) is similaer to that of (1a) and (1b). Due to space limit, it is provided in Appendix A. Proof in Appendix A illustrates that in the cases of (2a) and (2b), $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0$ or $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q-q))) = 0$ or $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q + q))) = 0$

In summary, it is always true that

$$D_2(e_1e_2/(c'_1c'_2)) = 0 \ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(q))) = 0 \ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0 \\ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q - q))) = 0 \ \vee D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q + q))) = 0$$

As $ZM()$ is correct according to Theorem 1,

$$ZM \ (\ A_1, A_2 \mid e_1e_2/(c'_1c'_2), \ e_1e_2/(c'_1c'_2E_2(q)), \ e_1e_2/(c'_1c'_2E_2(p_1\%q)), \\ e_1e_2/(c'_1c'_2E_2(p_1\%q - q)), \ e_1e_2/(c'_1c'_2E_2(p_1\%q + q)) \ ) \ = \mathsf{TRUE}$$

□

Lemma 1. *If $\sum_{i=1}^{n}(-1)^{m_i}x_i = 0 \bmod p$ and $\sum_{i=1}^{n}|x_i| < p$ where $m_i = 0$ or 1 for $i = 1, 2, \ldots, n$, then $\sum_{i=1}^{n}(-1)^{m_i}x_i = 0$.*

Proof of Lemma 1 is very simple and is not present due to space limitation.

Theorem 5. *The basic range test is partially sound in the negatively-malicious model. More precisely, if nobody deviates from the protocol and the specialized zero test in Formula (3) outputs* TRUE*, then $0 \le D_1(c) < 3q$.*

Proof: As $ZM()$ is sound according to Theorem 2

$$D_2(e_1e_2/(c'_1c'_2)) = 0 \ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(q))) = 0 \ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0 \\ \vee \ D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q - q))) = 0 \ \vee D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q + q))) = 0$$

when

$$ZM\ (\ e_1e_2/(c'_1c'_2),\ e_1e_2/(c'_1c'_2E_2(q)),\ e_1e_2/(c'_1c'_2E_2(p_1\%q)),$$
$$e_1e_2/(c'_1c'_2E_2(p_1\%q-q)),\ e_1e_2/(c'_1c'_2E_2(p_1\%q+q))\ )\ = \mathsf{TRUE}$$

In the following proof $m_1\%q+m_2\%q$ is calculated with the help of homomorphic property $m_1\%q+m_2\%q = D_2(e_1)+D_2(e_2) = D_2(e_1e_2) \bmod p_2$ and under the condition of every clause in Equation (4). Each clause corresponds to a case in the proof, while each case is divided into two sub-cases: either $m = m_1+m_2$ or $m = m_1+m_2-p_1$.

– If $D_2(e_1e_2/(c'_1c'_2)) = 0$, then $D_2(e_1e_2) = D_2(c'_1c'_2) = D_2(E_2(m_1)E_2(m_2)) = m_1+m_2 \bmod p_2$.
  - If $m = m_1+m_2$, then

    $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2 \bmod p_2 = m \bmod p_2$$

    Note that $|m_1\%q|+|m_2\%q|+|m| < 2q+p_1 < p_2$ as $5q \le p_1$ and $p_2 \ge 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m$. Therefore, $m < 2q$.
  - If $m = m_1+m_2-p_1$, then

    $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2 \bmod p_2 = m+p_1 \bmod p_2$$

    Note that $|m_1\%q|+|m_2\%q|+|m|+|p_1| < 2q+2p_1 < p_2$ as $5q \le p_1$ and $p_2 \ge 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m+p_1$, which is impossible as $m_1\%q+m_2\%q < 2q < p_1 < m+p_1$. Therefore, it is impossible that $m = m_1+m_2-p_1$ when $D_2(e_1e_2/(c'_1c'_2)) = 0$.

  So, $m < 2q$.

– If $D_2(e_1e_2/(c'_1c'_2E_2(q))) = 0$, then

  $$D_2(e_1e_2) = D_2(c'_1c'_2E_2(q)) = D_2(E_2(m_1)E_2(m_2)E_2(q)) = m_1+m_2+q \bmod p_2$$

  - If $m = m_1+m_2$, then

    $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+q \bmod p_2 = m+q \bmod p_2$$

    Note that $|m_1\%q|+|m_2\%q|+|m|+|q| < 3q+p_1 < p_2$ as $5q \le p_1$ and $p_2 \ge 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m+q$. Therefore, $m < q$.
  - If $m = m_1+m_2-p_1$, then

    $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+q \bmod p_2 = m+p_1+q \bmod p_2$$

    Note that $|m_1\%q|+|m_2\%q|+|m|+|p_1|+|q| < 3q+2p_1 < p_2$ as $5q \le p_1$ and $p_2 \ge 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m+p_1+q$, which is impossible as $m_1\%q+m_2\%q < 2q < p_1 < m+p_1+q$. Therefore, it is impossible that $m = m_1+m_2-p_1$ when $D_2(e_1e_2/(c'_1c'_2E_2(q))) = 0$.

  So, $m < q$.

– If $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0$, then

$$D_2(e_1e_2) = D_2(c'_1c'_2E_2(p_1\%q)) = D_2(E_2(m_1)E_2(m_2)E_2(p_1\%q)) = m_1+m_2+p_1\%q \bmod p_2$$

- If $m = m_1 + m_2$, then

  $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+p_1\%q \bmod p_2 = m+p_1\%q \bmod p_2$$

  Note that $|m_1\%q| + |m_2\%q| + |m| + |p_1\%q| < 3q + p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q + m_2\%q = m + p_1\%q$. Therefore, $m < 2q$.
- If $m = m_1 + m_2 - p_1$, then

  $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+p_1\%q \bmod p_2 = m+p_1+p_1\%q \bmod p_2$$

  Note that $|m_1\%q| + |m_2\%q| + |m| + |p_1| + |p_1\%q| < 3q + 2p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q + m_2\%q = m + p_1 + p_1\%q$, which is impossible as $m_1\%q + m_2\%q < 2q < p_1 < m + p_1 + p_1\%q$. Therefore, it is impossible that $m = m_1 + m_2 - p_1$ when $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q))) = 0$.

So, $m < 2q$.

– If $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q - q))) = 0$, then

$$D_2(e_1e_2) = D_2(c'_1c'_2E_2(p_1\%q-q)) = D_2(E_2(m_1)E_2(m_2)E_2(p_1\%q-q)) = m_1+m_2+p_1\%q-q \bmod p_2$$

- If $m = m_1 + m_2$, then

  $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+p_1\%q-q \bmod p_2 = m+p_1\%q-q \bmod p_2$$

  Note that $|m_1\%q|+|m_2\%q|+|m|+|p_1\%q|+|q| < 4q+p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m+p_1\%q-q$. Therefore, $m < 3q$.
- If $m = m_1 + m_2 - p_1$, then

  $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+p_1\%q-q \bmod p_2 = m+p_1+p_1\%q-q \bmod p_2$$

  Note that $|m_1\%q| + |m_2\%q| + |m| + |p_1| + |p_1\%q| + |q| < 4q + 2p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q + m_2\%q = m+p_1+p_1\%q-q$, which is impossible as $m_1\%q+m_2\%q < 2q < p_1-q < m + p_1 + p_1\%q - q$. Therefore, it is impossible that $m = m_1 + m_2 - p_1$ when $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q - q))) = 0$.

So, $m < 3q$.

– If $D_2(e_1e_2/(c'_1c'_2E_2(p_1\%q + q))) = 0$, then

$$D_2(e_1e_2) = D_2(c'_1c'_2E_2(p_1\%q+q)) = D_2(E_2(m_1)E_2(m_2)E_2(p_1\%q+q)) = m_1+m_2+p_1\%q+q \bmod p_2$$

- If $m = m_1 + m_2$, then

  $$m_1\%q+m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1+m_2+p_1\%q+q \bmod p_2 = m+p_1\%q+q \bmod p_2$$

  Note that $|m_1\%q|+|m_2\%q|+|m|+|p_1\%q|+|q| < 4q+p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q+m_2\%q = m+p_1\%q+q$. Therefore, $m < q$.

- If $m = m_1 + m_2 - p_1$, then

  $$m_1\%q + m_2\%q = D_2(e_1e_2) \bmod p_2 = m_1 + m_2 + p_1\%q + q \bmod p_2 = m + p_1 + p_1\%q + q \bmod p_2$$

  Note that $|m_1\%q| + |m_2\%q| + |m| + |p_1| + |p_1\%q| + |q| < 4q + 2p_1 < p_2$ as $5q \leq p_1$ and $p_2 \geq 3p_1$. So according to Lemma 1, $m_1\%q + m_2\%q = m + p_1 + p_1\%q + q$, which is impossible as $m_1\%q + m_2\%q < 2q < p_1 < m + p_1 + p_1\%q + q$. Therefore, it is impossible that $m = m_1 + m_2 - p_1$ when $D_2(e_1e_2/(c_1'c_2'E_2(p_1\%q + q))) = 0$.

So, $m < q$.

In summary, it is always true that $m < 3q$. □

Theorem 6. *The basic range test is private. More precisely, if A_1 and A_2 do not collude, the only knowledge of either of them about $D_1(c)$ is the test result.*

Proof: A_1's total knowledge from the basic range test about $D_1(c)$ is the test result as the employed encryption algorithms are semantically secure and only A_2 knows the private key. So A_1's only knowledge about $D_1(c)$ in the basic range test is the test result if A_2 does not collude with him.

Without A_1's collusion, A_2's total knowledge about $D_1(c)$ is m_2 and T, which is his knowledge transcript in the special zero test. So A_2's knowledge transcript in the basic range test is $m_2||T$. Theorem 3 illustrates that T reveals no information except for the test result if A_1 does not collude with A_2. If A_1 does not collude with A_2, m_2 is uniformly distributed in Z_{p_1} and independent of $D_1(c)$ or T. So A_2's knowledge transcript in the basic range test reveals no information about $D_1(c)$ except for the range test result if A_1 does not collude with him. Therefore, without A_1's collusion, A_2's only knowledge about $D_1(c)$ in the basic range test is the test result. □

The largest size of the range in the basic range test, q, is of the same magnitude as p_1. The basic range test is efficient and has a constant cost independent of the range size.

4.2 Precise Range Test

As partial soundness limits the application of the basic range test, it is upgraded to precise range test, which is absolutely sound. More precisely, precise range test outputs TRUE if and only if the encrypted message is in the range. The precise range test of a ciphertext c in the first encryption system is denoted as $PR\ (\ A_1,\ A_2 \mid c\ )$, such that $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE} \Longleftrightarrow 0 \leq D_1(c) < q$. The precise range test of c is described in Figure 3, in which $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$ guarantees $0 \leq D_1(c) < 3q$ while $BR\ (\ A_1,\ A_2 \mid E_1(q-1)/c\ ) = \mathsf{TRUE}$ guarantees $D_1(c) \in \{0, 1, \ldots, q-1\} \cup \{p_1 - 2q + 1, p_1 - 2q + 2, \ldots, p_1\}$. The intersection of the two ranges is Z_q.

1. A_1 prepares two basic range tests $BR\ (\ A_1,\ A_2 \mid c\ )$ and $BR\ (\ A_1,\ A_2 \mid E_1(q-1)/c\ )$.
2. A_1 presents the two basic range tests to A_2 in a random order.
3. A_2 finishes the two basic range tests and tells A_1 whether both basic range tests output TRUE and no more information.
4.

$$PR\ (\ A_1,\ A_2 \mid c\ ) = \begin{cases} \mathsf{TRUE} & \text{if } BR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE} \text{ and} \\ & BR\ (\ A_1,\ A_2 \mid E_1(q-1)/c\ ) = \mathsf{TRUE} \\ \mathsf{FALSE} & \text{otherwise} \end{cases} \tag{4}$$

Fig. 3. Precise range test

Theorem 7. *The precise range test is correct in the negatively-malicious model. More precisely, if nobody deviates from the protocol and* $0 \le D_1(c) < q$*, then* $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$.

Proof: As $0 \le D_1(c) < q$, according to Theorem 4, $BR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$. As $0 \le D_1(c) < q$ and the encryption algorithm is additive homomorphic, $D_1(E_1(q-1)/c) = q - 1 - D_1(c) < q$. So according to Theorem 4, $BR\ (\ A_1,\ A_2 \mid (E_1(q-1)/c\ ) = \mathsf{TRUE}$. Therefore, $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$. □

Theorem 8. *The precise range test is absolutely sound in the negatively-malicious model. More precisely, if nobody deviates from the protocol and* $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$*, then* $0 \le D_1(c) < q$.

Proof: $BR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$ and $BR\ (\ A_1,\ A_2 \mid (E_1(q-1)/c\ ) = \mathsf{TRUE}$ as $PR\ (\ A_1,\ A_2 \mid c\ ) = \mathsf{TRUE}$. So, according to Theorem 5 and additive homomorphism of the encryption algorithm, $0 \le D_1(c) < 3q$ and $(q - 1 - D_1(c))\%p_1 = D_1(E_1(q-1)/c) < 3q$. The fact $(q - 1 - D_1(c))\%p_1 < 3q$ implies $0 \le D_1(c) < q$ or $D_1(c) > p_1 - 2q$. As $5q \le p_1$, the fact $D_1(c) > p_1 - 2q$ implies $D_1(c) \ge 3q$. Therefore, $D_1(c) < 3q \ \ \wedge \ \ (D_1(c) < q \ \vee \ D_1(c) \ge 3q)$. Namely, $0 \le D_1(c) < q$. □

As the employed encryption algorithms are semantically secure and A_1 knows no private key, his total knowledge about $D_1(c)$ is the test result if A_2 does not collude with him. So the precise range test is private to A_1. More precisely, if A_2 does not collude with A_1, A_1's only knowledge about $D_1(c)$ is the test result. Note that the precise range test only employs two basic range tests, so it is not completely private to A_2. According to Theorem 6, A_2's only knowledge in the precise range test are the results of the two basic range tests if A_1 does not collude with him. When the precise range test outputs TRUE, A_2's only knowledge is the result of the precise range test without A_1's collusion as the precise range test outputs TRUE if and only if both basic range tests output TRUE. However, when the precise range test outputs FALSE, A_2 knows whether $-2q < D_1(c) < 3q$. If one basic range test outputs FALSE and the other outputs TRUE, A_2 knows that $-2q < D_1(c) < 3q$. Otherwise, A_2 knows that $3q \leq D_1(c) \leq p_1 - 2q$. So, complete privacy is sacrificed in the precise range test to achieve absolute soundness in the negatively-malicious model.

The largest size of the range in the precise range test, q, is of the same magnitude as p_1. The precise range test is efficient and has a constant cost independent of the range size.

4.3 Optimized Precise Range Test

Correctness and soundness of the precise test cannot be guaranteed in the actively-malicious model. A_2 may deviate from the protocol and return a wrong result to A_1. Moreover, the precise test is not completely private to A_2. In the optimized precise range test A_1 employs a cut-and-choose mechanism to verify correctness of A_2's operation. This cut-and-choose mechanism can also achieve complete privacy against A_2. Precise range test of c is randomly mixed with precise range tests of another random ciphertext. Only A_1 knows which precise range tests are performed on c, while A_2 cannot distinguish the multiple tests. If A_2 attempts to cause an incorrect result, with the help of the cut-and-choose mechanism A_1 can detect A_2's cheating with an overwhelmingly large probability. Moreover, although each precise test is not complete to A_2, he cannot get any information about $D_1(c)$ as he cannot distinguish tests of the two messages. So privacy can be achieved against A_2. The optimized precise range test protocol is described in Figure 4, which guarantees that the tester can always get the correct test result if he wants even in the actively-malicious model.

1. A_1 chooses a security parameter t and randomly divides set $\{1, 2, \ldots, 2t\}$ into four subsets S_1, S_2, S_3 and S_4, such that $[S_1] + [S_2] = [S_3] + [S_4] = t$.
2. A_1 randomly chooses m from Z_q, calculates $\hat{c} = E_1(m)$ and $E(0)$, a probabilistic encryption of zero. Then he repeats for $i = 1, 2, \ldots, 2t$.
   - if $i \in S_1$, A_1 performs $VC_i = PR\ (\ A_1,\ A_2 \mid \hat{c}\ )$ with A_2;
   - if $i \in S_2$, A_1 performs $VC_i = PR\ (\ A_1,\ A_2 \mid E_1(0)/\hat{c}\ )$ with A_2;
   - if $i \in S_3$, A_1 performs $VC_i = PR\ (\ A_1,\ A_2 \mid c\ )$ with A_2;
   - if $i \in S_4$, A_1 performs $VC_i = PR\ (\ A_1,\ A_2 \mid E_1(0)/c\ )$ with A_2.
3. A_1 recognises A_2's honesty if and only if $VC_i =$ TRUE for $i \in S_1$, $VC_i =$ FALSE for $i \in S_2$, VC_i is identical for $i \in S_3$, VC_i is identical for $i \in S_4$ and $VC_i = \neg VC_j$ for $i \in S_3$ and $j \in S_4$. If A_2 is verified to be honest, A_1 accepts VC_i with $i \in S_3$ as the test result.

Fig. 4. Optimized precise range test

Theorem 9. *The probability that a cheating A_2 can pass the verification in the optimized precise range test is no more than $1/\binom{2t}{t}$.*

Proof: Let vc_i denote the result of the i^{th} range test when A_2 acts honestly. Let $CS = \{i : \ 1 \leq i \leq 2t, \ VC_i = vc_i\}$. No matter how A_2 cheats, his malicious behaviour can be classified into three cases: $[CS] < t$, $t < [CS] < 2t$ or $[CS] = t$.

- If $[CS] < t$, $VC_i = \mathsf{TRUE}$ for $i \in S_1$ and $VC_i = \mathsf{FALSE}$ for $i \in S_2$ cannot be satisfied. So A_1 fails in the verification and A_2 is found cheating.
- If $t < [CS] < 2t$, either incorrect precise range test exists in VC_i for $i \in S_1 \cup S_2$ or both correct and incorrect precise range tests exist in VC_i for $i \in S_3 \cup S_4$. So A_1 fails in the verification and A_2 is found cheating.
- If $[CS] = t$, A_2 can pass the verification if and only if $CS = S_1 \cup S_2$. As A_1's input in each precise range test is uniformly distributed, A_2 cannot tell any difference between the precise range tests. Moreover, S_1, S_2, S_3, S_4 are randomly chosen and $\{vc_1, vc_2, \ldots, vc_{2t}\}$ are uniformly distributed in $\{\mathsf{TRUE}, \mathsf{FALSE}\}^{2t}$. So A_2 has no better method to find $S_1 \cup S_2$ other than random guess. Therefore, the probability that $CS = S_1 \cup S_2$ is $1/\binom{2t}{t}$.

Therefore, the only method for a cheating A_2 to pass the verification is to set $CS = S_1 \cup S_2$, the success probability of which is $1/\binom{2t}{t}$. □

Theorem 9 indicates that the tester can get the correct and sound test result in the optimized precise range test with an overwhelmingly large probability if he wants even in the actively-malicious model. Privacy is improved in the optimized precise range test. As A_2 has no idea which precises range tests are performed on c, he cannot get more information about c without A_1's collusion.

The maximum acceptable range is not changed after the test protocol is optimized, so is still of the same magnitude as the message space. Although the cut-and-choose mechanism reduces efficiency, cost of the optimized precise range test is still independent of the range size. As the cutting factor t (which is a small constant number like 20) is often much smaller than the range size, the optimized precise range test is still an efficient solution. So the optimized precise range test can satisfy all the desired properties of range test. Its advantages over the existing related schemes in terms of the desired properties and efficiency are demonstrated in Table 2. In table, cost of general and flexible range test instead of more efficient range test with special encryption format for certain application (costing $O(\log_2 q)$ is listed.

5 Conclusion

A range test protocol is proposed, which can correctly and soundly test whether a ciphertext contains a message in a certain interval range without revealing the message. If the tester wants, he can get the correct test result with an overwhelmingly large probability even in the actively-malicious model. Unlike the existing related techniques, the new protocol is efficient, accepts large enough range size and does not need a prover with knowledge of the message. Open questions are left in regard to security in the actively-malicious model. Can

Table 2. Property comparison

Schemes	Correctness & Soundness	Pri--vacy	Large range	Prover with know--ledge of the message	Format of the message	Cost
naive range test based on [9]	Yes	Yes	Yes	needed	any encryption or commitment	$O(q)$
naive range test based on [17]	Yes	Yes	Yes	not needed	any additive homo--morphic encryption	$O(q)$
[2, 14, 6, 8]	No	Yes	No	needed	commitment	$O(1)$
[5]	asymptotical	Yes	No	needed	certain commitment	$O(1)$
optimized precise range test	Yes	Yes	Yes	not needed	any additive homo--morphic encryption	$O(1)$

correctness, soundness and privacy be achieved simultaneously in the actively-malicious model? Is cut-and-choose inevitable for security of range test in the actively-malicious model?

References

1. Masayuki Abe and Koutarou Suzuki. M+1-st price auction using homomorphic encryption. In *Public Key Cryptology 2002*, volume 2288 of *Lecture Notes in Computer Science*, pages 115–124, Berlin, 2002. Springer-Verlag.
2. Feng Bao. An efficient verifiable encryption scheme for encryption of discrete logarithms. In *the Smart Card Research Conference, CARDIS'98*, volume 1820 of *Lecture Notes in Computer Science*, pages 213–220, Berlin, 1998. Springer-Verlag.
3. Olivier Baudron, Pierre-Alain Fouque, David Pointcheval, Jacques Stern, and Guillaume Poupard. Practical multi-candidate election system. In *Twentieth Annual ACM Symposium on Principles of Distributed Computing*, pages 274–283, 2001.
4. Ian F. Blake and Vladimir Kolesnikov. Strong conditional oblivious transfer and computing on intervals. In *ASIACRYPT '04*, volume 3329 of *Lecture Notes in Computer Science*, pages 515–529, Berlin, 2004. Springer-Verlag.
5. Fabrice Boudot. Efficient proofs that a committed number lies in an interval. In *EUROCRYPT '00*, volume 1807 of *Lecture Notes in Computer Science*, pages 431–444, Berlin, 2000. Springer-Verlag.
6. E. F. Brickell, D. Chaum, I. B. Damgård, and J. van de Graaf. Gradual and verifiable release of a secret. In *Advances in Cryptology - Crypto '87*, volume 293 of *Lecture Notes in Computer Science*, pages 156–166, Berlin, 1987. Springer-Verlag.
7. J Camenisch and M Michels. Separability and efficiency for generic group signature schemes. In *CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 413–430, Berlin, 1999. Springer-Verlag.
8. A Chan, Y Frankel, and Y Tsiounis. Easy come - easy go divisible cash. updated version with corrections. 1998. Available as `http://www.ccs.neu.edu/home/yiannis/`.
9. R. Cramer, I. B. Damgård, and B. Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *CRYPTO '94*, volume 839 of *Lecture Notes in Computer Science*, pages 174–187, Berlin, 1994. Springer-Verlag.

10. Jonathan Katz, Steven Myers, and Rafail Ostrovsky. Cryptographic counters and applications to electronic voting. In *Advances in Cryptology—EUROCRYPT 01*, pages 78–92, 2001.
11. Aggelos Kiayias and Moti Yung. Self-tallying elections and perfect ballot secrecy. In *Public Key Cryptography, 5th International Workshop—PKC 02*, pages 141–158, 2002.
12. Byoungcheon Lee and Kwangjo Kim. Receipt-free electronic voting through collaboration of voter and honest verifier. In *JW-ISC 2000*, pages 101–108, 2000.
13. Byoungcheon Lee and Kwangjo Kim. Receipt-free electronic voting scheme with a tamper-resistant randomizer. In *Information Security and Cryptology, ICISC 2002*, pages 389–406, 2002.
14. Wenbo Mao. Guaranteed correct sharing of integer factorization with off-line shareholders. In *Public Key Cryptography, 1st International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 1998*, volume 1431 of *Lecture Notes in Computer Science*, pages 27–42, Berlin, 1998. Springer.
15. Kazumasa Omote and Atsuko Miyaji. A second-price sealed-bid auction with the discriminant of the p-th root. In *Financial Cryptography 2002*, volume 2357 of *Lecture Notes in Computer Science*, pages 57–71, Berlin, 2002. Springer.
16. P Paillier. Public key cryptosystem based on composite degree residuosity classes. In *EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238, Berlin, 1999. Springer-Verlag.
17. Kun Peng, Colin Boyd, Ed Dawson, and Byoungcheon Lee. An efficient and verifiable solution to the millionaire problem. In *Pre-Proceedings of ICISC 2004*, volume 3506 of *Lecture Notes in Computer Science*, pages 315–330, Berlin, 2004. Springer-Verlag.

A Proof of (2a) and (2b) in Theorem 4

Proof of (2a) and (2b) in Theorem 4 is as follows.

- (2a): According to conditions (2) and (a), $m_1\%q + m_2\%q = (m_1 + m_2)\%q = (m + p_1)\%q$.

  So, (2a) can be divided into two sub-cases: either

  $$\text{(2ai): } m_1\%q + m_2\%q = m\%q + p_1\%q = m + p_1\%q$$

  or

  $$\text{(2aii): } m_1\%q + m_2\%q = m\%q + p_1\%q - q = m + p_1\%q - q$$

  - (2ai): According to additive homomorphism of the encryption algorithm and Condition (2) and Condition (2ai)

    $$\begin{aligned} D_2(e_1e_2E_2(p_1)/(c'_1c'_2E_2(p_1\%q))) &= D_2(e_1e_2E_2(p_1)/(E_2(m_1)E_2(m_2)E_2(p_1\%q))) \\ &= D_2(e_1) + D_2(e_2) + p_1 - (D_2(E_2(m_1)) + D_2(E_2(m_2)) - p_1\%q \bmod p_2 \\ &= m_1\%q + m_2\%q + p_1 - (m_1 + m_2) - p_1\%q \bmod p_2 \\ &= m + p_1\%q + p_1 - (m + p_1) - p_1\%q \bmod p_2 = 0 \end{aligned}$$

- (2aii): According to additive homomorphism of the encryption algorithm and Condition (2) and Condition (2aii)

$$\begin{aligned}
D_2(e_1e_2E_2(p_1)/(c_1'c_2'E_2(p_1\%q-q))) &= D_2(e_1e_2E_2(p_1)/(E_2(m_1)E_2(m_2)E_2(p_1\%q-q))) \\
= D_2(e_1)+D_2(e_2)+p_1-(D_2(E_2(m_1))+D_2(E_2(m_2))-(p_1\%q-q) \bmod p_2 \\
= m_1\%q+m_2\%q+p_1-(m_1+m_2)-(p_1\%q-q) \bmod p_2 \\
= m+p_1\%q-q+p_1-(m+p_1)-(p_1\%q-q) \bmod p_2 = 0
\end{aligned}$$

– (2b): According to conditions (2) and (b), $m_1\%q+m_2\%q=(m_1+m_2)\%q+q=(m+p_1)\%q+q$.
So, (2b) can be divided into two sub-cases: either

$$\text{(2bi): } m_1\%q+m_2\%q=m\%q+p_1\%q=m+p_1\%q+q$$

or

$$\text{(2bii): } m_1\%q+m_2\%q=m\%q+p_1\%q-q=m+p_1\%q$$

- (2bi): According to additive homomorphism of the encryption algorithm and Condition (2) and Condition (2bi)

$$\begin{aligned}
D_2(e_1e_2E_2(p_1)/(c_1'c_2'E_2(p_1\%q+q))) &= D_2(e_1e_2E_2(p_1)/(E_2(m_1)E_2(m_2)E_2(p_1\%q+q))) \\
= D_2(e_1)+D_2(e_2)+p_1-(D_2(E_2(m_1))+D_2(E_2(m_2))-(p_1\%q+q) \bmod p_2 \\
= m_1\%q+m_2\%q+p_1-(m_1+m_2)-(p_1\%q+q) \bmod p_2 \\
= m+p_1\%q+q+p_1-(m+p_1)-(p_1\%q+q) \bmod p_2 = 0
\end{aligned}$$

- (2bii): According to additive homomorphism of the encryption algorithm and Condition (2) and Condition (2bii)

$$\begin{aligned}
D_2(e_1e_2E_2(p_1)/(c_1'c_2'E_2(p_1\%q))) &= D_2(e_1e_2E_2(p_1)/(E_2(m_1)E_2(m_2)E_2(p_1\%q))) \\
= D_2(e_1)+D_2(e_2)+p_1-(D_2(E_2(m_1))+D_2(E_2(m_2))-p_1\%q \bmod p_2 \\
= m_1\%q+m_2\%q+p_1-(m_1+m_2)-p_1\%q \bmod p_2 \\
= m+p_1\%q+p_1-(m+p_1)-p_1\%q \bmod p_2 = 0
\end{aligned}$$