Interactive Bi-Proof Systems and Undeniable Signature Schemes

Atsushi Fujioka Tatsuaki Okamoto Kazuo Ohta

NTT Laboratories
Nippon Telegraph and Telephone Corporation
1-2356, Take, Yokosuka-shi, Kanagawa-ken, 238-03 Japan

Abstract

This paper proposes a new construction of the *minimum knowledge undeniable signature scheme* which solves a problem inherent in Chaum's scheme. We formulate a new proof system, the *minimum knowledge interactive bi-proof system*, and a pair of languages, the *common witness problem*, based on the random self-reducible problem. And we show that any common witness problem has the minimum knowledge interactive bi-proof system. A practical construction for undeniable signature schemes is proposed based on such a proof system. These schemes assure signature confirmation and disavowal with the same protocol (or at the same time).

1 Introduction

Digital signatures [DH] are one of the most important concepts of modern cryptography, and have many applications in information security systems.

A new paradigm of signature schemes, *undeniable signatures*, was recently proposed by Chaum *et al.* [CA, Ch], and its properties are different from those of digital signatures. Although an undeniable signature is similar to a digital signature in that it is a number issued by a signer that is related to the signer's public-key and his message, the difference is that an undeniable signature cannot be verified without the cooperation of the signer.

Undeniable signature schemes [CA, Ch] consist of two parts, a confirmation protocol and a disavowal protocol. In the confirmation protocol, a verifier can verify the validity

D.W. Davies (Ed.): Advances in Cryptology - EUROCRYPT '91, LNCS 547, pp. 243-256, 1991.

of a signature by interacting with the signer, and there is no chance that the signer can falsely represent the validity of an invalid signature. If the validity test fails, the verifier can determine if the signature is invalid or the signer is false by the disavowal protocol.

Chaum's scheme [Ch] has a problem, in which two different protocols are necessary for the confirmation and disavowal of the signature. If a dishonest prover, say, Alice claims that her valid signature is not valid, then first the verifier, say, Bob must execute the disavowal protocol to check her claim, then knows that her claim is not true. However, Bob cannot believe that her signature is valid just from this negative result of the disavowal protocol, because Alice may not follow the valid disavowal protocol. So, Bob must execute the confirmation protocol to determine that her signature is valid. Therefore, in the above case, Bob must execute the both protocols to confirm the validity of her signature.

This paper proposes a new undeniable signature scheme which solves the above problem of Chaum's scheme. That is, our scheme assures signature confirmation and disavowal with the same protocol. In other words, executing our scheme once is equivalent to executing both confirmation and disavowal protocols at the same time. Hence, without regard to signer's claim, the verifier can always determine whether a signature is valid or invalid, through executing our scheme only once.

First, in order to construct our undeniable signature scheme, we formulate a class of new proof systems, the *interactive bi-proof systems*, which can exactly determine which of $x \in L_1$ or $x \in L_2$ is a true theorem where L_1 and L_2 are disjoint languages. Roughly speaking, when $x \in L_1$, a prover can prove "x is in L_1", however no prover can prove "x is in L_1" when $x \notin L_1$. On the other hand, when $x \in L_2$, the prover can prove "x is in L_2" with the same protocol, however no prover can prove "x is in L_2" when $x \notin L_2$.

Next, based on the *random self-reducible* problem [TW], we introduce a pair of languages, the *common witness problem*, and show that any common witness problem has the minimum knowledge interactive bi-proof system. Here, the minimum knowledge [GHY] is a variant of zero-knowledge. For example, in a zero-knowledge proof, the prover releases the knowledge such as "$x \in L$", while in a minimum knowledge proof, the prover releases the knowledge such as which one is correct, "$x \in L_1$" or "$x \in L_2$".

Finally, we propose new undeniable signature schemes, which solve the above-mentioned problem of Chaum's scheme, by using these minimum knowledge interactive bi-proof systems for a common witness problem. In addition, several variations of our scheme are discussed in terms of increasing efficiency and useful applications.

2 Interactive Bi-Proof System

First we formulate a new proof system for our undeniable signature scheme.

In interactive proof systems [GMR], a prover has infinite power while the verifier is restricted to probabilistic polynomial time bounded. They interact to perform a proof '$x \in L$' for a language L. When $x \in L$, the proof is accomplished; however, when $x \notin L$, no prover can claim that "x is in L" and such proof is rejected.

This property approximates that of signature schemes. That is, when a signature is valid, the signer can prove it. When, however, the signature is not valid, no signer can prove its validity.

To construct an undeniable signature scheme, we must add a new requirement to the interactive proof system: the verifier can distinguish between $x \notin L$ and the falseness of the signer. The existing interactive proof system does not ensure that the verifier can distinguish between them when proof is rejected. We have, therefore, defined a new proof system, the *interactive bi-proof system*, for a pair of disjoint languages, L_1 and L_2. When $x \in L_1$, a prover can show that "x is in L_1", and when $x \in L_2$, then "x is in L_2". However, no prover can prove that "x is in L_1" when $x \notin L_1$ or "x is in L_2" when $x \notin L_2$.

Definition 2.1 *Let L_1 and L_2 be disjoint languages over $\{0,1\}^*$. Let (P,V) be an interactive protocol. We say that (P,V) is an interactive bi-proof system for (L_1, L_2) if we have the following:*

- *Completeness*
  - *For each k, for sufficiently large $x \in L_1$ given as input to (P,V), V halts and accepts x as "x is in L_1" with a probability of at least $1 - |x|^{-k}$.*
  - *For each k, for sufficiently large $x \in L_2$ given as input to (P,V), V halts and accepts x as "x is in L_2" with a probability of at least $1 - |x|^{-k}$.*

  (The probabilities here are taken over the coin tosses of P and V.)

- *Soundness*
  - *For each k, for sufficiently large $x \notin L_1$, for any ITM P', on input x to (P',V), V halts and accepts x as "x is in L_1" with a probability of at most $|x|^{-k}$.*
  - *For each k, for sufficiently large $x \notin L_2$, for any ITM P', on input x to (P',V), V halts and accepts x as "x is in L_2" with a probability of at most $|x|^{-k}$.*

(The probabilities here are taken over the coin tosses of P' and V.)

Next we define the *minimum knowledgeness* of this proof system.

Definition 2.2 *Let (P,V) be an interactive bi-proof system for (L_1, L_2). We say that (P,V) is minimum knowledge if, given any expected polynomial time probabilistic Turing machine V', there exists another probabilistic Turing machine $M_{V'}$, running in expected polynomial time, such that for all $x \in L_1 \cup L_2$:*

- *$B(x)$ is a probability distribution where $B(x)$ is the output of interactive protocol (P,V) and the distribution probability are taken over the coin tosses of P and V.*
- *$M_{V'}$ has one-time access to an oracle, as follows. Given any input x and auxiliary input h, $M_{V'}$ queries the oracle with input x; the oracle returns a value distributed according $B(x)$.*
- *The ensembles*

$$\left\{ M_{V'}[x,h] \mid x \in L_1 \cup L_2, h \in \{0,1\}^{poly(|x|)} \right\}$$

and

$$\left\{ VIEW_{V'}\{(P,V')[x,h]\} \mid x \in L_1 \cup L_2, h \in \{0,1\}^{poly(|x|)} \right\}$$

are indistinguishable.

(If the ensembles are identical, we say that the bi-proof system is *perfectly* minimum knowledge.)

See [GMR, GHY] for the definitions of *interactive protocol*, *ITM*, *minimum knowledgeness*, *VIEW*, and *indistinguishability*.

3 Interactive Bi-Proof System and Random Self-Reducibility

In this section, we show the essential conditions of the interactive bi-proof system.

First, we explain random self-reducibility [TW].

Definition 3.1 *Let $\mathcal{N}$ be a countably infinite set. For any $N \in \mathcal{N}$, let $|N|$ denotes the length of a suitable representation of N. For any $N \in \mathcal{N}$, let X_N, Y_N be finite sets, and $R_N \subseteq X_N \times Y_N$ be a relation. Let*

$$dom\ R_N = \left\{ x \in X_N \mid (x,y) \in R_N \text{ for some } y \in Y_N \right\}$$

denote the domain of R_N,

$$R_N(x) = \{y \mid (x, y) \in R_N\}$$

the image of $x \in X_N$, *and*

$$R_N(X_N) = \{y \mid (x, y) \in R_N,\ x \in X_N\}$$

the image of R_N. *R is random self-reducible if and only if there is a polynomial time algorithm A that, given any inputs* $N \in \mathcal{N}$, $x \in dom\, R_N$, *and* $r \in \{0,1\}^\omega$, *outputs* $x' = A(N, x, r) \in dom\, R_N$ *satisfying the following three properties.*

R1. If the bits of r are random, uniform and independent, then x' is uniformly distributed over dom R_N.

R2. There is a polynomial time algorithm that, given $N, x, \overline{r}$, *and any* $y' \in R_N(x')$, *outputs* $y \in R_N(x)$. *Here* $\overline{r}$ *is the finite prefix of r consumed in computing* $x' = A(N, x, r)$.

R3. There is a polynomial time algorithm that, given N, x, r, *and any* $y \in R_N(x)$, *outputs some* $y' \in R_N(x')$. *If, in addition, the bits of r are random, uniform, and independent, then y' is uniformly distributed on* $R_N(x')$.

Based on the above problem, we define the following problem.

Definition 3.2 *Let the relation* $R_{(1)}$ *and* $R_{(2)}$ *be random self-reducible. The following pair of languages* (L_R, L_C),

$$L_R = \{(x_1, x_2) \mid \exists y\ [(x_1, y) \in R_{(1),N} \ \wedge\ (x_2, y) \in R_{(2),N}]\},$$

$$L_P = \{(x_1, x_2) \mid \exists y\ [(x_1, y) \in R_{(1),N}]\},$$

$$L_C = \overline{L_R} \cap L_P,$$

are called the common witness problem.

(Note that in this case, L_R and L_C are disjoint, and $L_R \cap L_P = L_R$.)

Now we can obtain the following theorem about the relation of the interactive bi-proof system and the common witness problem.

Theorem 3.3 *Let the relation* $R_{(1)}$ *and* $R_{(2)}$ *be random self-reducible and satisfy the following conditions:*

T1. *For any $y \in R_{(i),N}(X_{(i),N})$, the number of x satisfying $(x,y) \in R_{(i),N}$ is one, and there are probabilistic polynomial time algorithms $B_{(i)}$ that, given N, y, output x satisfying $(x,y) \in R_{(i),N}$ where $i = 1$ and 2.*

T2. *There are probabilistic polynomial time algorithms that, given N, output random pairs $(x,y) \in R_{(i),N}$ with x uniformly distributed over dom $R_{(i),N}$ and y uniformly distributed over $R_{(i),N}(x)$ where $i = 1$ and 2.*

T3. *If $(x,y) \notin R_{(i),N}$, then for any r, $(x',y') \notin R_{(i),N}$ ($i = 1$ and 2) where x' is created from x and r, and y' is created from y and r.*

T4. *$R_{(1),N}(X_{(1),N}) = R_{(2),N}(X_{(2),N})$, and any y' created from y and r on $R_{(1),N}$ is equal to the one created from y and r on $R_{(2),N}$.*

T5. *$\exists y\,[(x_1,y) \in R_{(1),N} \wedge (x_2,y) \in R_{(2),N}] \Rightarrow \forall y\,[(x_1,y) \in R_{(1),N} \Rightarrow (x_2,y) \in R_{(2),N}]$. Let set $F(x_1,x_2,x_1')$ be $\{x_2' \mid \exists r[x_1' = A_{(1)}(N,x_1,r) \wedge x_2' = A_{(2)}(N,x_2,r)]\}$. Then, for any x_1, x_2, and x_1', the number of elements of set $F(x_1,x_2,x_1')$ is at most 1.*

T6. *If there exists a probabilistic polynomial time algorithm that, given x_1, x_2, x_1', x_2' ($x_i \in$ dom $R_{(i),N}, i = 1,2$), determines whether $\exists r\,[x_1' = A_{(1)}(N,x_1,r) \wedge x_2' = A_{(2)}(N,x_2,r)]$ with non-negligible probability, then there exists a probabilistic polynomial time algorithm that, given $(x_1,x_2) \in (L_R, L_C)$, determines $(x_1,x_2) \in L_R$ or $(x_1,x_2) \in L_C$ with overwhelming probability.*

Then, on inputs N and $x = (x_1,x_2)$, there is a minimum knowledge interactive bi-proof system (P,V) for any common witness problem (L_R, L_C).

Sketch of Proof:

We consider the following protocol. Without loss of generality, there exists $(x_1,y_1) \in R_{(1),N}$ from the definition of L_P.

Protocol:

Step 1 Repeat t times from **Step 2** to **Step 7** where $t = O(|x|)$.

Step 2 P generates random numbers r, a, v ($|v| = |x_2| = n$), and calculates X_1, Z,

$$X_1 = A_{(1)}(N, x_1, r),$$

$$Z = BC(v,a).$$

And P sends X_1, Z to V. Here, BC is a bit-commitment function [Na], v is committed bits, and a is random bits used for concealing v. (For simplicity, we write $BC(v,a)$ as $BC(v_1,a_1)\|\cdots\|BC(v_n,a_n)$, where $v = v_1\|\cdots\|v_n$, $a = a_1\|\cdots\|a_n$, and $\|$ denotes concatenation.)

Step 3 V generates random number u ($|u| = |x_2| = n$), and sends u to P.

Step 4 P calculates $q = u \oplus v$ and X_2,

$$X_2 = h_q(A_{(2)}(N, x_2, r)).$$

And P sends X_2 and a, v to V. Here, function h_q is a hard-core predicate or hard-core function shown in Definitions 2 and 3 of [GL], where $|q| = \left|A_{(2)}(N, x_2, r)\right|$ when h_q is a hard-core predicate, and $|q| = 2\left|A_{(2)}(N, x_2, r)\right|$ when h_q is a hard-core function. Hereafter, for simplicity, we will consider h_q a hard-core predicate. Then, $h_q(w) = \sum_{i=1}^{n} w_i q_i \bmod 2$, where $w = w_1 \| \cdots \| w_n$, $q = q_1 \| \cdots \| q_n$, and $|w_i| = |q_i| = 1$ $(i = 1, \ldots, n)$.

Step 5 V checks whether $BC(v, a)$ holds. If it does not hold, V rejects the proof. Otherwise, V calculates $q = u \oplus v$, generates random bit e, and sends it to P.

Step 6 P calculates Y.

$$\begin{cases} Y = r & \text{if } e = 0, \\ Y = y_1' & \text{if } e = 1. \end{cases}$$

Here $y_1' \in R_{(1),N}(x_1')$ ($x_1' = A_{(1)}(N, x_1, r)$) is created from y_1 and r. And P sends it to V.

Step 7 V checks as follows:

- When $e = 0$, V checks the following equations

$$X_1 \stackrel{?}{=} A_{(1)}(N, x_1, Y),$$

$$X_2 \stackrel{?}{=} h_q(A_{(2)}(N, x_2, Y)).$$

If both tests succeed, set this round as "honest" and continue the protocol.

Otherwise V rejects the proof.

- When $e = 1$, V checks the following equations

$$(X_1, Y) \stackrel{?}{\in} R_{(1),N},$$

$$X_2 \stackrel{?}{=} h_q(B_{(2)}(N, Y)).$$

If both tests succeed, set this round as "L_R" and continue the protocol.

If only first test succeeds, set this round as "L_C" and continue the protocol.

Otherwise V rejects the proof.

Step 8 After t rounds, V determines the proof as follows:

- If every round is either "L_R" or "honest", then V accepts as "$x \in L_R$".
- If every round of $e = 0$ is "honest" and $R > 1/3$, then V accepts as "$x \in L_C$", where $R = \#\{\text{"}L_C\text{" round}\}/\#\{\text{"}e = 1\text{" round}\}$. ($\#S$ denotes the number of elements of set S.)
- Otherwise V rejects the proof.

Remark:

hen h_q is a hard-core function (or $|X_2| > 1$), V determines the proof as "$x \in L_C$" as follows: When $|X_2|$ is $O(|x|)$, if every round is either "L_C" or "honest", then V accepts as "$x \in L_C$". When $|X_2|$ is $c = O(1)$, if every round of $e = 0$ is "honest" and $R > 1 - 1/2^c - d$, then V accepts as "$x \in L_C$", where d is a constant.

Consider the completeness and soundness conditions.

- *Completeness*
  - In the case of $x \in L_R$, there exists some y, such that $(x_1, y) \in R_{(1),N} \wedge (x_2, y) \in R_{(2),N}$. Then, it is clear that if P follows the protocol, then both checks in **Step 7** are accomplished. So V accepts as "x is in L_R" with probability 1.
  - On the other hand, when $x \in L_C$, y where $(x_1, y) \in R_{(1),N}$ does not satisfy $(x_2, y) \in R_{(2),N}$. Condition T3 directly implies that P's response Y, cannot satisfy $X_2 = h_q(B_{(2)}(N, Y))$ in **Step 7** with probability 1/2 in each round. So V accepts as "x is in L_C" with overwhelming probability after t rounds repetition.

- *Soundness*
  - In the case of $x \notin L_R$, to cheat V with non-negligible probability, P' must create the messages X_1, X_2 and Y which satisfy both tests in **Step 7**, i.e., $e = 0$ and $e = 1$. When $x \notin L_R$, two cases are considered, $x \in L_C$ or $x \notin L_P$. First we consider $x \in L_C$. In this case, for all y where $(x_1, y) \in R_{(1),N}$, this y must not satisfy $(x_2, y) \in R_{(2),N}$ from the definition of L_C and condition T5. To cheat V, Y that is created from y and r, however, must satisfy $X_2 = h_q(B_{(2)}(N, Y))$ with probability 1. For this, $(X_2, Y) \in R_{(2),N}$ must be satisfied, since q is randomly generated after r is determined. This contradicts condition T3.

On the other hand, when $x \notin L_P$, there is no y where $(x_1, y) \in R_{(1),N}$. So if Y satisfies $(X_1, Y) \in R_{(1),N}$, this contradicts condition T3.

- In the case of $x \notin L_C$, to cheat V with non-negligible probability, P' must create the messages X_1, X_2 and Y which satisfy only first side of tests in **Step 7-2** ($e = 1$) and both tests in **Step 7-1** ($e = 0$). In this case, two cases are also considered, $x \in L_R$ or $x \notin L_P$.

  First we consider $x \in L_R$. In this case, for all y where $(x_1, y) \in R_{(1),N}$, this y must satisfy $(x_2, y) \in R_{(2),N}$ from the definition of L_R and condition T5. To cheat V, Y that is created from y and r, however, must not satisfy $X_2 = h_q(B_{(2)}(N, Y))$ with probability 1/2 in each round. For this, $(X_2, Y) \in R_{(2),N}$ must not be satisfied with at least non-negligible probability. This contradicts the condition T5.

  On the other hand, when $x \notin L_P$, there is no y where $(x_1, y) \in R_{(1),N}$. So if Y satisfies $(X_1, Y) \in R_{(1),N}$, this contradicts condition T3.

- *Minimum knowledgeness*

Then we prove the minimum knowledgeness. First, on input $x \in L_R \cup L_C$, the simulator accesses to an oracle and knows $x \in L_R$ or $x \in L_C$. After that it simulates the view of the history by the 'standard guessing' algorithm.

When $x \in L_R$, it is clear that the simulator can perfectly simulate the view, or that this protocol satisfies perfect minimum knowledgeness.

When $x \notin L_C$, to prove that this protocol is (computationally) minimum knowledge, we must show that (X_1, X_2, q) is (computationally) indistinguishable from (X_1, a, q), where a is a real random number. If BC is a secure bit-commitment function and the prover is honest, then q is a random number. If the common witness problem is not $\mathcal{BPP}$, then a bit-commitment function BC exists [Na, ILL]. From condition T6 and Lemma 1 of [GL], if the common witness problem to be proven in this bi-proof system is not in $\mathcal{BPP}$, then (X_1, X_2) is computationally indistinguishable from (X_1, a). Then, this protocol is (computationally) minimum knowledge. If the common witness problem is in $\mathcal{BPP}$, this protocol is trivially (perfectly) minimum knowledge.

¶

Remark 1:

If the common witness problem is defined over the discrete logarithm problem, then condition T6 holds, because the problem, given g^x, m^x, to check $\exists r$ [$g^{x+r} \wedge m^{x+r}$] is equivalent to the problem to check $\exists x$ [$g^x \wedge m^x$].

Remark 2:

In the above protocol (sequential version), q is generated in each round. However, in the parallel (five round) version of the above protocol [FFS, BMO], the same q is commonly used by all t round messages. So, the communication amount of these parallel versions is much reduced than that of the sequential version.

Remark 3:

The *discrete logarithm* problem and the *graph isomorphism* problem are good examples of problems that satisfy the above conditions.

4 Application to Undeniable Signature Schemes

4.1 Proposed Undeniable Signature Schemes

We apply the interactive bi-proof system directly to an undeniable signature scheme.

From **Theorem 3.3**, if a random self-reducible problem exists, then there exists an interactive bi-proof system. The definition of the interactive bi-proof system is suitable for undeniable signature schemes, so there exists an undeniable signature scheme based on the random self-reducible problem.

We consider the discrete logarithms problem similar to those in [CA, Ch].

- **Center Key Generation**
  - Center generates a large prime number p and selects a primitive root g of field GF(p).
- **Signer Key generation**
  - Signer generates his secret key x, and computes y ($= g^x \bmod p$). He publishes y as his public key.

- **Signature generation**
  - Signer generates signature s of a message m from p and his secret key x, $s = m^x \bmod p$.
- **Signature confirmation and disavowal**
  - Repeat the following procedure t times where $t = O(|x|)$.
  - Signer generates random numbers r, a, v ($|v| = |x_2|$), and calculates

$$X_1 = g^r \cdot y \bmod p,$$

$$Z = BC(v, a).$$

    and sends (X_1, Z) to verifier.
  - Verifier generates random number u ($|u| = |x_2|$), and sends u to the prover.
  - Prover calculates $q = u \oplus v$ and

$$X_2 = h_q(m^r \cdot s \bmod p),$$

    and sends X_2 and a, v to the verifier.
  - Verifier checks whether $BC(v, a)$ holds. If it does not hold, verifier rejects the protocol. Otherwise, verifier calculates $q = u \oplus v$, generates $e \in_R \{0, 1\}$ and sends it to the signer.
  - Signer computes Y ($= r + ex \bmod p - 1$) and sends it to the verifier.
  - Verifier checks as follows:
    - When $e = 0$, verifier checks the following equations

$$X_1 \stackrel{?}{=} g^Y \cdot y \bmod p,$$

$$X_2 \stackrel{?}{=} h_q(m^Y \cdot s \bmod p).$$

      If both tests succeed, set this round as "honest" and continue the protocol.
      Otherwise verifier rejects the protocol.
    - When $e = 1$, verifier checks the following equations

$$X_1 \stackrel{?}{=} g^Y \bmod p,$$

$$X_2 \stackrel{?}{=} h_q(m^Y \bmod p).$$

If both tests succeed, set this round as "valid" and continue the protocol.
If only first test succeeds, set this round as "invalid" and continue the protocol.
Otherwise verifier rejects the protocol.

- After t rounds, verifier determines the validity of (m, s) as follows:
  - If every round is either "valid" or "honest", then verifier accepts as "s is the valid signature of m".
  - If every round of $e = 0$ is "honest" and $R > 1/3$, then verifier accepts as "s is the invalid signature of m", where $R = \#\{$"invalid" round$\}/\#\{$"$e = 1$" round$\}$.
  - Otherwise verifier rejects the protocol.

This protocol satisfies minimum knowledge interactive bi-proof system.

Remark:

haum's confirmation and disavowal protocols are called *zero* knowledge; however, our protocol is called *minimum* knowledge. In both schemes, these words mean that each protocol releases no additional knowledge except that which the prover wants to release. The different point is as follows:

In Chaum's scheme, to prove the validity of a signature, signer Alice claims the validity/invalidity of her signature before using the confirmation/disavowal protocol. To support her claim she then uses the appropriate protocol. In this sequence, the protocols release no additional bit than her claim, so each confirmation/disavowal protocol of his scheme is zero knowledge.

However in our scheme, regardless of the signature's claim, a signer executes the same protocol. Our protocol releases one bit, i.e., the validity or invalidity, so our scheme is minimum knowledge.

4.2 Efficiency

Our scheme is more efficient than Chaum's because our scheme consists of only one protocol. It implies that when this scheme is implemented, the confirmation and disavowal protocol can be done with the same equipment. Furthermore, in this protocol, the number of powering and multiplication operations are smaller than that of the disavowal protocol in Chaum's scheme [Ch].

For even more efficiency, we are proposing two enhancements, one is the *higher degree version* and another is the *parallel version*. Unfortunately these protocols cannot

be proven to satisfy minimum knowledgeness, however, both can decrease the amount of transmission overhead.

To satisfy minimum knowledgeness in the parallel version, the constant round zero knowledge technique shown in [BMO] can be applied to our scheme. This can reduce the round number of the protocol. Moreover, as described in **Remark 2** of **Theorem 3.3**, the parallel version reduce the communication amount as well as the round number, since the same q is commonly used by all t round messages.

5 Conclusion

We have proposed a new proof system, the minimum knowledge interactive bi-proof system, and constructed an undeniable signature scheme using a formulation of the new system.

We have also defined a pair of languages, the common witness problem, based on the random self-reducible problem, and shown that any common witness problem has the minimum knowledge interactive bi-proof system.

A practical undeniable signature scheme was proposed based on such a proof system, in which confirmation and disavowal can be done with the same protocol (or at the same time).

Acknowledgement

The authors wish to thank Toshiya Itoh, Kenji Koyama, Kouichi Sakurai, Hiroki Shizuya, Kazue Tanaka, Yasuyuki Tsukada, and Yuliang Zheng for their valuable comments.

References

[BCC] G. Brassard, D. Chaum, and C. Crépeau, "Minimum Disclosure Proofs of Knowledge", *Journal of Computer and System Sciences*, Vol.37, No.2, pp.156–189 (Oct., 1988).

[BMO] M. Bellare, S. Micali, and R. Ostrovsky, "Perfect Zero-Knowledge in Constant Rounds", Proceedings of 22nd annual ACM Symposium on Theory of Computing, pp.482–493 (May, 1990).

[Ch] D. Chaum, "Zero-Knowledge Undeniable Signatures", in *Advances in Cryptology — EUROCRYPT '90*, Lecture Notes in Computer Science 473, Springer–Verlag, Berlin, pp.458–464 (1991).

[CA] D. Chaum and H. van Antwerpen, "Undeniable Signatures", in *Advances in Cryptology — CRYPTO '89*, Lecture Notes in Computer Science 435, Springer–Verlag, Berlin, pp.212–216 (1990).

[DH] W. Diffie and M. E. Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory, Vol.IT-22, No.6, pp.644–654 (Nov., 1976).

[FFS] U. Feige, A. Fiat, and A. Shamir, "Zero Knowledge Proofs of Identity", Proceedings of 19th annual ACM Symposium on Theory of Computing, pp.210–217 (May, 1987).

[GHY] Z. Galil, S. Haber, and C. Yung, "Minimum-Knowledge Interactive Proofs for Decision Problems", *SIAM Journal on Computing*, Vol.18, No.4, pp.711–739 (Aug., 1989).

[GL] O. Goldreich and L. Levin, "A Hard-Core Predicate for all On-Way Functions", Proceedings of 21st annual ACM Symposium on Theory of Computing, pp.25–32 (May, 1989).

[GMR] S. Goldwasser, S. Micali, and C. Rackoff, "The Knowledge Complexity of Interactive Proof-Systems", Proceedings of 17th annual ACM Symposium on Theory of Computing, pp.291–304 (May, 1985).

[ILL] R. Impagliazzo, L. Levin, and M. Luby, "Pseudo-Random Number Generation from One-Way Functions", Proceedings of 21st annual ACM Symposium on Theory of Computing, pp.12–24 (May, 1989).

[Na] M.Naor, "Bit Commitment Using Pseudo-Randomness", in *Advances in Cryptology — CRYPTO '89*, Lecture Notes in Computer Science 435, Springer–Verlag, Berlin, pp.128–136 (1990).

[TW] M. Tompa and H. Woll, "Random Self-Reducibility and Zero Knowledge Interactive Proofs of Possession of Information", 28th Annual Symposium on Foundations of Computer Science, IEEE, pp.472–482 (Oct., 1987).