Statistical Zero Knowledge Protocols to Prove Modular Polynomial Relations

Eiichiro FUJISAKI and Tatsuaki OKAMOTO

NTT Laboratories,
1-1 Hikarinooka, Yokosuka-shi, 239 Japan
Email: {fujisaki, okamoto}@sucaba.isl.ntt.co.jp

Abstract. This paper proposes a bit commitment scheme, $BC(\cdot)$, and *efficient* statistical zero knowledge (in short, SZK) protocols in which, for any given multi-variable polynomial $f(X_1, .., X_t)$ and any given modulus n, prover $\mathcal{P}$ gives $(I_1, .., I_t)$ to verifier $\mathcal{V}$ and can convince $\mathcal{V}$ that $\mathcal{P}$ knows $(x_1, .., x_t)$ satisfying $f(x_1, .., x_t) \equiv 0 \pmod{n}$ and $I_i = BC(x_i)$, $(i = 1, .., t)$. The proposed protocols are $O(|n|)$ times more efficient than the corresponding previous ones [Dam93, Dam95, Oka95]. The (knowledge) soundness of our protocols holds under a computational assumption, the intractability of a modified RSA problem (see Def.3), while the (statistical) zero-knowledgeness of the protocols needs no computational assumption. The protocols can be employed to construct various practical cryptographic protocols, such as fair exchange, untraceable electronic cash and verifiable secret sharing protocols.

1 Introduction

1.1 Problem

In many cryptographic protocols, a party often wants to prove something related to his secret while concealing his secret from the others. Such relations are often specified by modular polynomials and bit commitments are very useful in such protocols. This paper focuses on the following problem: for given multi-variable polynomial $f(X_1, .., X_t)$ and modulus n, a party (prover) $\mathcal{P}$ gives $(I_1, .., I_t)$ to another party (verifier) $\mathcal{V}$ and convinces $\mathcal{V}$ that $\mathcal{P}$ knows $(x_1, .., x_t)$ satisfying $f(x_1, .., x_t) \equiv 0 \pmod{n}$ and $I_i = BC(x_i)$, $(i = 1, .., t)$, without revealing the values, $x_1, .., x_t$.

This problem is indeed raised on many cryptographic protocols. In fair exchange and contract signing protocols based on RSA signatures [Dam93, Dam95], (n, e) is the public-key of the RSA scheme, $f(x) = x^e - m$ and $I = BC(x)$. After proving that $\mathcal{P}$ knows x satisfying the relations, $\mathcal{P}$ releases x bit by bit using I. In untraceable off-line electronic cash protocols, restricted blind signatures [Bra95, Oka95] play an important role, where, for instance, $f(x_1, x_2, x_3) = (x_1 x_2) x_3^e$ and $I_1 = BC(x_1)$ and $I_2 = BC(x_2)$. After proving that $\mathcal{P}$ knows (x_1, x_2, x_3) satisfying the relations, $\mathcal{V}$ stores I_1 and I_2. If $\mathcal{P}$ double-spends a coin, $\mathcal{V}$ can get (x_1, x_2) from $x_1 x_2$ as evidence of double-spending (See [Oka95] for more details). If f is a polynomial and n is a prime for Shamir's secret sharing scheme, some protocols related to secret sharing such as (publicly) verifiable secret sharing [CGMA85, Ped91, Sta96] can be interpreted as this type of problem.

1.2 Previous Works

Theoretically, the general construction of protocols to solve the above-mentioned problem has been already given assuming that a secure bit commitment scheme exists. This is derived from the results of zero knowledge proof for NP-language [GMRa89, GMW86, BCC86] and converting them to proof of knowledge [FFS88, TW87, BG92]. Depending on the types of the underlying bit commitment schemes, there exist two different results: namely, computational ZK (CZK) for interactive proof (IP) and perfect ZK (PZK) for argument (computationally-sound proof). However, those protocols are very inefficient in general.

In 1993, Damgård proposed the first efficient protocol to solve the problem with a specific form for constructing a *fair exchange and contract signing* protocol [Dam93, Dam95]. He proposed the protocols in which prover $\mathcal{P}$ can convince verifier $\mathcal{V}$ that he knows s of bit commitment $BC(s)$ and that it is a Rabin signature ($s = m^{1/2} \bmod n$) or a RSA signature ($s = m^{1/e} \bmod n$), for a message m. The protocols are PZK computationally-sound proof of knowledge systems (PZK arguments of *knowledge*). Those protocols essentially consist of some primitives: a bit commitment scheme and three protocols, which correspond to the *basic*, *comparing*, and *mod-multi* protocols in this paper. His *basic* protocol is the protocol in which $\mathcal{P}$ proves to $\mathcal{V}$ that secret s is in a given range $[a, b)$ and the *comparing* and *mod-multi* protocols are compositions of the *basic* protocol. It is easy to construct a PZK argument of knowledge for any multi-variable modular polynomial (f, n) based on these primitives.

In 1995, Okamoto showed another application of the problem above. He constructed an RSA-type *restricted blind signature* for his *untraceable off-line electronic cash* [Oka95] by using similar primitives: a bit commitment scheme and three protocols, which are essentially equivalent to those of Damgård's except for the bit commitment scheme.

Unfortunately, both of their protocols are not so efficient, because $\mathcal{V}$, in their *basic* protocols, needs to request $\mathcal{P}$ to open one of the commitments, BC(t) or BC(t+s), *many times* (the so called *cut-and-choose* method).

1.3 Results

This paper gives a more efficient solution to the problem above than previous ones. We first propose primitives, a bit commitment scheme and four (statistical) *witness indistinguishable* (WI) protocols (See [FS90] for WI). Then we construct, by using these primitives, statistical zero-knowledge protocols (SZK argument of knowledge) in which, for any given multi-variable polynomial $f(X_1, .., X_t)$ and any given modulus n, prover $\mathcal{P}$ gives $(I_1, .., I_t)$ to verifier $\mathcal{V}$ and can convince $\mathcal{V}$ that $\mathcal{P}$ knows $(x_1, .., x_t)$ satisfying $f(x_1, .., x_t) \equiv 0 \pmod{n}$ and $I_i = BC(x_i)$, $(i = 1, .., t)$ without revealing any additional information. The proposed protocols are $O(|n|)$ times more efficient than the corresponding protocols in [Dam93, Dam95, Oka95], because our protocols do not need to confirm that a secret is in any range nor to execute any (single-bit based) *cut-and-choose* method. At the same time, the communication complexity of our protocols is $O(|n|)$ times less than those of [Dam93, Dam95] and [Oka95]. Although a set-up procedure for the parameter of the underlying bit-commitment is necessary and plays an essential role to satisfy the zero-knowledgeness of our protocols, the procedure can be done separately before the main parts of the protocols in pre-processing and can be shared by repeated execution of the main parts. (Similarly, a set-up procedure is also necessary in [Dam93, Dam95].)

A computational assumption, the intractability of a modified RSA problem (defined in Def.3), is necessary to prove the (knowledge) soundness regarding $(x_1, .., x_t)$ in our protocols, while no computational assumption is required for (statistical) zero-knowledgeness. In addition, any poly-time bounded prover $\mathcal{P}^*$can open the bit commitment in any different ways with negligible probability under the factoring assumption.

These protocols can be employed to construct various practical cryptographic protocols such as fair exchange, untraceable electronic cash and some protocols regarding secret sharing. In Section 5, we demonstrate how to employ the proposed protocols to construct the *fair exchange and contract signing* protocol.

2 Definitions and Assumptions

This section mainly defines the factoring assumption and the modified RSA problem and its assumption; the modified RSA problem is a little different from the well-known RSA problem at the point that a cracking algorithm, A, can on input (N, Y) choose a convenient exponent, e (≥ 2), to output (X, e) such that $X \equiv \sqrt[e]{Y} \pmod{N}$ (Of course, it is less intractable than the *factoring* problem since a cracking algorithm, A, which can factor N, can solve the modified RSA problem of N). The validity (soundness) of the *whole* protocols against $\mathcal{P}$ can be guaranteed under Assumption 4 while the validity of the commitment against $\mathcal{P}$ can be guaranteed under Assumption 2.

Definition 1. $f(n)$ is **negligible** in n if, for any constant c, there exists a constant, N, such that $f(n) < (1/n)^c$ for any $n > N$. $f(n)$ is **non-negligible** in n if, there exits constants c and N such that $f(n) > (1/n)^c$ for any $n > N$. f(n) is **overwhelming** in n if, for any constant c, there exists a constant, N, such that $f(n) > 1 - (1/n)^c$ for any $n > N$.

Assumption 2. (Factoring Assumption) A probabilistic polynomial-time generator Δ_1 exists which on input $1^{|N|}$ outputs composite N where N is a composite of two prime numbers, P and Q, such that for any probabilistic polynomial-size circuit family, A, the probability that A can factor N is negligible. The probability is taken over the random choices of Δ_1 and A.

Definition 3. Modified RSA problem is, for given (N, Y), finding X and e ($e \geq 2$), such that $Y \equiv X^e \pmod{N}$, where N is the composite of two prime numbers, P and Q.

Assumption 4. (Modified RSA Assumption) A probabilistic polynomial-time generator Δ_2 exists which on input $1^{|N|}$ outputs (N, Y) such that for any probabilistic polynomial-size circuit family A, the probability that A can solve the modified RSA problem is negligible. The probability is taken over the random choices of Δ_2 and A.

In this paper, we use the following symbols. "$\alpha \in_R S$" means uniformly choosing a random element, α, from a set, S. Let Z_N be a residue class ring modulo N, and Z_N^* the reduced residue class group. Other symbols and definitions will be set as needed.

3 Bit Commitment and WI protocols

In this section, we propose a bit commitment scheme, and four WI protocols. The suitable parallel executions of those protocols, for any multi-variable polynomial $f(X_1, .., X_t)$ and any modulus n, can construct an WI protocol over the relation $((I_1, .., I_t, I_{t+1}), (x_1, r_1, .., x_t, r_t, y, r_{t+1}))$ such that $y \equiv f(x_1, .., x_t) \pmod{n}$ (For simplicity, we often call the protocol WI protocol to confirm $y \equiv f(x_1, .., x_t) \pmod{n}$). We show later, as an example, a WI protocol to confirm $y \equiv ax^5 + b \bmod n$.

3.1 Bit Commitment Scheme

Our proposed commitment statistically reveals to the verifier no information of secret s in $BC(s)$ and holds computational validity against the prover. The validity of the commitment is guaranteed if the factoring assumption (Assumption 2) holds true. The commitments are given by

$$BC_{b_0}(s, r) = {b_0}^s b_1^r \bmod N \text{ or } BC_{b_0}(s, r_1, r_2) = {b_0}^s b_1^{r_1} b_2^{r_2} \bmod N.$$

Here, (N, b_0, b_1, b_2) is a set of system parameters given by verifier $\mathcal{V}$ or authority (i.e. trusted third party).

To set the system parameters, verifier $\mathcal{V}$ (or authority) executes the following procedure:

[Set-up procedure]

1. $\mathcal{V}$ generates large primes, P and Q, including odd prime divisors, p and q, such that $p = (P-1)/2$, $q = (Q-1)/2$, and $p \neq q$).
2. $\mathcal{V}$ finds at random $g_p \in G_p \backslash \{1\}$, and $g_q \in G_q \backslash \{1\}$, where G_p, G_q are subgroups of the order p, q in Z_P^*, Z_Q^* respectively (The complexity of finding g_p and g_q is comparable to that of finding generator elements of Z_P^* and Z_Q^*).
3. $\mathcal{V}$ computes, $b_0 \in Z_N^*$, by using the Chinese Remainder Theorem, such that $b_0 = g_p \bmod P$ and $b_0 = g_q \bmod Q$ (b_0 is a generator element of G_{pq}).
4. $\mathcal{V}$ finds at random $\alpha, \beta \in Z_{pq}^*$ and sets $b_1 = {b_0}^\alpha \bmod N$ and $b_2 = {b_0}^\beta \bmod N$.
5. $\mathcal{V}$ sends (N, b_0, b_1, b_2) to prover $\mathcal{P}$. Then $\mathcal{V}$ proves that he knows α, α^{-1}, β, and β^{-1} such that $b_1 = {b_0}^\alpha \bmod N$, and $b_2 = {b_0}^\beta \bmod N$ in the zero knowledge manner (that is, the orders of b_0, b_1,and b_2 are equivalent).

In the bit-commitment phase, $\mathcal{P}$ sends to $\mathcal{V}$, $BC_{b_0}(x, r) = {b_0}^x {b_1}^r \bmod N$ or $BC_{b_0}(x, r_1, r_2) = {b_0}^x {b_1}^{r_1} {b_2}^{r_2} \bmod N$ where $x \in [\,0,\ N)$ is a secret and $r, r_1, r_2 \in [\,0,\ 2^m N)$ are auxiliary random numbers.

Lemma 5. (Indistinguishability) *If $m = O(|N|)$, $BC_{b_0}(x, r)$ and $BC_{b_0}(x, r_1, r_2)$ statistically reveal no information of x to $\mathcal{V}$.*

The following results show that the validity (security) of these commitments are guaranteed if the factoring assumption (Assumption 2) holds true.

Lemma 6. (Miller) *Let $N = p_1^{v_1} \cdots p_m^{v_m}$ be the prime factorization of the odd integer N. Let $\lambda(N) = \mathrm{lcm}\{p_1^{v_1 - 1}(p_1 - 1), .., p_m^{v_m - 1}(p_m - 1)\}$ (the Carmichael λ-function) and L be a multiple of $\lambda(N)$ (i.e., $\lambda(N) | L$). There exists a probabilistic polynomial-time algorithm M which, on input (N, L), can output the factorization of N with non-negligible probability in $|N|$. (Note: N is given by Δ_1 and the probability is taken over the coin tosses of Δ_1 and M.)*

The proof of Lemma 6 is implied by Theorem 2 in [Mil76].

Definition 7. (Generator Δ_{BC}) Let Δ_{BC} be a probabilistic polynomial-time algorithm which on input $1^{|N|}$ outputs (N, b_0, b_1, b_2) where the distribution of N is equal to that of Δ_2 and (b_0, b_1, b_2) is generated by the **Set-up procedure** of the bit commitment scheme.

Theorem 8. (Validity against $\mathcal{P}^*$) *If Assumption 2 holds true, there exists no probabilistic polynomial-time algorithm $\mathcal{P}^*$ which, on input (N, b_0, b_1), given by Δ_{BC}, can output (s_1, r_1) and (s_2, r_2), with non-negligible probability in $|N|$, where $(s_1, r_1) \neq (s_2, r_2)$ and $b_0{}^{s_1} b_1^{r_1} \equiv b_0{}^{s_2} b_1^{r_2} \pmod{N}$. (Note: the probability is taken over the coin tosses of Δ_{BC} and $\mathcal{P}^*$.)*

Sketch of Proof:

The proof is by contradiction. Assuming that a probabilistic polynomial-time algorithm $\mathcal{P}^*$ can output (s_1, r_1) and (s_2, r_2), with non-negligible probability, then we can construct a probabilistic poly-time algorithm M which can factor N with non-negligible probability. Let $s = s_1 - s_2$, and $r = r_2 - r_1$. The algorithm $\mathcal{P}^*$ above can be replaced by the algorithm which, on input (N, b_0, b_1), can output

$$b_0{}^s b_1{}^r \equiv 1 \pmod{N}, \tag{1}$$

where $(s, r) \neq (0, 0)$. In addition, by Lemma 6, the algorithm M can be replaced by the algorithm which on input N outputs L' such that $\lambda(N) | L'$.

The strategy of M is the following:

Algorithm M

1. Input N generated by Δ_{BC} to M.
2. M picks $b_0 \in_R Z_N$ and $\alpha \in_R (0, 2^k N)$ $(k = O(|N|))$, then computes $b_1 = b_0{}^{\alpha} \bmod N$.
3. M inputs (N, b_0, b_1) to $\mathcal{P}^*$.
4. If $\mathcal{P}^*$ returns (s, r), go the next step, otherwise M halts.
5. M outputs $L = 2(r - \alpha s)$ if $L \neq 0$, otherwise halts.

The algorithm M can output L with non-negligible probability.

When M picks b_0 uniformly in Z_N in Step 2, the probability that the order of b_0 is pq is non-negligible because $\frac{\varphi(pq)}{\#Z_N} = \frac{(p-1)(q-1)}{(2p+1)(2q+1)} \approx \frac{1}{4}$, where $\varphi(\cdot)$ is the Eulerian function and $\varphi(pq)$ is the number of generators of G_{pq}. This means that the distribution of a non-negligible fraction (about 1/4) of (N, b, b_1)'s picked by M is indistinguishable from those generated by Δ_{BC}. $\mathcal{P}^*$ therefore outputs (s, r) with non-negligible probability in Step 4. In Step 5, the probability of $L \neq 0$ is non-negligible. This is because even infinite power $\mathcal{P}^*$ can only know $\alpha_0 = \alpha \bmod pq$. Therefore, if α is uniformly picked in $[0, 2^k N)$, the probability of $L \neq 0$ is non-negligible. From equation (1), $L \equiv 0 \pmod{pq}$. This means that

$$L = 2kpq = k\lambda(N), \tag{2}$$

where $k \neq 0$, $\lambda(N) = \mathrm{lcm}(P - 1, Q - 1)$. □

Corollary 9. (Validity against $\mathcal{P}^*$) *If Assumption 2 holds true, there exists no probabilistic polynomial-time algorithm $\mathcal{P}^*$ which, on input (N, b_0, b_1, b_2), given by Δ_{BC}, can output $(t, u, v) \neq (0, 0, 0)$ such that $b_0^t b_1^u b_2^v \equiv 1 \pmod{N}$, with non-negligible probability.*

If base b_0 is clear, we use the expressions $BC(s, r)$ and $BC(s, r_1, r_2)$. If auxiliary parameters are not important, we use just $BC(s)$.

3.2 Basic Protocol

Let $R^{(1)}_{(N,b_0,b_1)} := \{(I, (x, r)) | I = BC_{(N,b_0,b_1)}(x, r)\}$. The basic protocol is (statistical) *witness indistinguishable* (WI) over the relation $R^{(1)}_{(N,b_0,b_1)}$ and convinces $\mathcal{V}$ that $\mathcal{P}$ knows (x, r) such that $I = BC_{(N,b_0,b_1)}(x, r)$.

[Basic Protocol]

1. $\mathcal{V}$ executes with $\mathcal{P}$ the *set-up procedure* for parameter (N, b_0, b_1, b_2).
2. $\mathcal{P}$ sets $I = BC_{(N,b_0,b_1)}(x, r)$ and sends it to $\mathcal{V}$.
3. $\mathcal{P}$ chooses w_1^0, $w_1^1 \in_R [\ 0,\ 2^{2m}N)$ and sets w_2^0, w_2^1 by $w_2^0 = w_1^0 - 2^{2m}N$ and $w_2^1 = w_1^1 - 2^{2m}N$. $\mathcal{P}$ picks four elements, $w_{i,j}^2$'s $\in_R [\ 0,\ 2^m N)$, then computes $t_{i,j} = BC(w_i^0, w_j^1, w_{ij}^2)$, where $1 \leq i, j \leq 2$.
4. $\mathcal{P}$ sends to $\mathcal{V}$, four unordered commitments, $t_{i,j}$'s.
5. $\mathcal{V}$ picks a challenge $c \in_R [\ 0, 2^m)$ and sends it to $\mathcal{P}$.
6. $\mathcal{P}$ sets $X = cx + w_i^0$ and $R = cr + w_j^1$ such that $X, R \in [\ 0,\ 2^{2m}N)$, and sends to $\mathcal{V}$, the pair, (X, R, $w_{i,j}^2$).
7. $\mathcal{V}$ checks there exists a t_{ij} such that $BC(X, R, w_{ij}^2) \equiv t_{i,j} I^c \pmod{N}$.

The completeness is obvious, since when $X = cx + w_i^0$ and $R = cr + w_j^1$ (if $\mathcal{P}$ is honest, there exists X, $R \in [\ 0,\ 2^{2m}N)$), the left-hand side in the verification equation is equal to the right-hand, because

$$b_0^X b_1^R b_2^{w_{i,j}^2} \equiv b_0^{cs+w_i^0} b_1^{cr+w_j^1} b_2^{w_{i,j}^2} \equiv b_0^{w_i^0} b_1^{w_j^1} b_2^{w_{i,j}^2} I^c \pmod{N}.$$

Lemma 10. (Soundness) *Under Assumption 4, there exists a probabilistic poly-time algorithm M such that, for any probabilistic poly-time algorithm $\mathcal{P}^*$, if probabilistic interactive algorithm $(\mathcal{P}^*, \mathcal{V})$ accepts with non-negligible probability in $|N|$, then M with Δ_{BC} and $\mathcal{P}^*$ as oracles can extract (x, r) satisfying $I = BC_{(N,b_0,b_1)}(x, r)$ with overwhelming probability in $|N|$ where I is given by $\mathcal{P}^*$ as output. The success probability of $(\mathcal{P}^*, \mathcal{V})$ is taken over the coin tosses of $\mathcal{P}^*$ and $\mathcal{V}$ (including Δ_{BC}), while the success probability of M over those of Δ_{BC}, $\mathcal{P}^*$ and M.*

The proof of the soundness is given in Appendix A.

Lemma 11. (Witness Indistinguishable) *If $m = O(|N|)$ and $x, r \in [\ 0,\ 2^m N)$, the basic protocol is statistically witness indistinguishable over $R^{(1)}_{(N,b_0,b_1)}$.*

3.3 Checking Protocol

The following protocol is considered as a kind of the basic protocol. However, since it is also utilized in the mod-multi protocol and in Subsection 3.6, we state it as a different one. Let $R^{(2)}_{(N,b_0)} := \{(I,\gamma)|I = b_0^{\gamma} \bmod N\}$. The checking protocol is WI over the relation $R^{(2)}_{(N,b_0)}$ and convinces $\mathcal{V}$ that $\mathcal{P}$ can know γ such that $I = {b_0}^{\gamma} \bmod N$.

[Checking Protocol]

1. $\mathcal{V}$ executes with $\mathcal{P}$ a *set-up procedure* for (N, b_0, b_1).
2. $\mathcal{P}$ sets $I = {b_0}^{\gamma} \bmod N$ and sends it to $\mathcal{V}$.
3. $\mathcal{P}$ chooses $w_1^0 \in_R [\ 0,\ 2^{2m}l)$ and sets w_2^0 by $w_2^0 = w_1^0 - 2^{2m}l$. $\mathcal{P}$ picks two elements, w_i^1's $\in_R [\ 0,\ 2^m N)$, then computes $t_i = BC_{b_0}(w_i^0, w_i^1)$, where $1 \leq i \leq 2$ and $l := \max[b-a, N]$.
4. $\mathcal{P}$ sends to $\mathcal{V}$, two unordered commitments, t_i's.
5. $\mathcal{V}$ picks a challenge $c \in_R [\ 0, 2^m)$ and sends it to $\mathcal{P}$.
6. $\mathcal{P}$ sets $X := c(\gamma - a) + w_i^0 \in [\ 0,\ 2^{2m}l)$, and sends to $\mathcal{V}$, the pair, (X, w_i^1).
7. $\mathcal{V}$ checks there exists a t_{ij} such that $BC(X, w_i^1) \equiv t_i(Ib_0^{-a})^c \pmod N$.

The following results are easily obtained by the properties of the basic protocol.

Lemma 12. (Soundness) *Under Assumption 4, there exists a probabilistic algorithm M such that, for any probabilistic poly-time algorithm $\mathcal{P}^*$, if probabilistic interactive algorithm $(\mathcal{P}^*,\mathcal{V})$ accepts with non-negligible probability in $|N|$, then M with Δ_{BC} and $\mathcal{P}^*$ as oracles can extract γ satisfying $I = {b_0}^{\gamma} \bmod N$ with overwhelming probability in $|N|$ where I is given by $\mathcal{P}^*$ as output. The success probability of $(\mathcal{P}^*,\mathcal{V})$ is taken over the coin tosses of $\mathcal{P}^*$ and $\mathcal{V}$ (including Δ_{BC}), while the success probability of M over those of Δ_{BC}, $\mathcal{P}^*$ and M.*

Lemma 13. (Witness Indistinguishable) *If $m = O(|N|)$ and $\gamma \in [\ a, b\ )$, the checking protocol is statistically witness indistinguishable over $R^{(2)}_{(N,b_0)}$.*

3.4 Comparing Protocol

Let $R^{(3)}_{(N,b_0,b_1,a)} := \{((I_1,I_2),(x,r_1,r_2))|I_1 = BC_{b_0}(x,r_1), I_2 = BC_a(x,r_2)\}$. The comparing protocol is WI over the relation $R^{(3)}_{(N,b_0,b_1,a)}$, in which $\mathcal{P}$ can convince $\mathcal{V}$ that he knows (x,r_1,r_2) such that $I_1 = BC_{b_0}(x,r_1)$ and $I_2 = BC_a(x,r_2)$.

[Comparing Protocol]

1. $\mathcal{V}$ executes with $\mathcal{P}$ the *set-up procedure* for parameters (N, b_0, b_1, b_2).
2. $\mathcal{P}$ sets $I_1 = BC_{b_0}(x,r_1)$ and $I_2 = BC_a(x,r_2)$, and sends them to $\mathcal{V}$.
3. $\mathcal{P}$ computes, for $1 \leq i,j \leq 2$, $t_{ij} = BC_{b_0}(w_i^0, w_j^1, w_{ij}^2)$ and $u_{ij} = BC_a(w_i^0, \eta_j^1, \eta_{ij}^2)$.
4. $\mathcal{P}$ sends to $\mathcal{V}$, four unordered pairs, $(t_{i,j}, u_{i,j})$'s.
5. $\mathcal{V}$ picks a $c \in_R [\ 0, 2^m)$ and sends it to $\mathcal{P}$.
6. $\mathcal{P}$ sets $X := cx + w_i^0$, $R_1 := cr_1 + w_j^1$, and $R_2 := cr_2 + w_k^2$ such that $X, R_1, R_2 \in [\ 0,\ 2^{2m}N)$. $\mathcal{P}$ then sends to $\mathcal{V}$, the pair, $(X, R_1, R_2, w_{i,j}^2, \eta_{i,k}^2)$.

7. $\mathcal{V}$ checks that there exists a pair ($t_{i,j}$, $u_{i,k}$) such that

$$BC_{b_0}(X, R_1, w_{ij}^2) \equiv t_{i,j} I_1^c \pmod{N} \text{ and } BC_a(X, R_2, \eta_{ij}^2) \equiv u_{i,k} I_2^c \pmod{N}.$$

If $\mathcal{V}$ sets a new base a, he has to convince $\mathcal{P}$ that there exists an α such that $a = {b_0}^\alpha \bmod N$ before executing this protocol, but in many cases, a is set by $\mathcal{P}$ as $a := I_1$. Note that in the case of $a := I_1$, $\mathcal{P}$ can show $\mathcal{V}$ $I_2 = BC_{b_0}(x^2, r_1 x_1 + r_2)$. This means $\mathcal{P}$ can convince $\mathcal{V}$ that commitments, $BC_{b_0}(x, r_1)$ and $BC_{b_0}(y, r_2)$, satisfy $y = x^2$.

The following results are easily obtained by the properties of the basic protocol.

Lemma 14. (Soundness) *Under Assumption 4, there exists a probabilistic algorithm M such that, for any probabilistic poly-time algorithm $\mathcal{P}^*$, if probabilistic interactive algorithm $(\mathcal{P}^*, \mathcal{V})$ accepts with non-negligible probability in $|N|$, then M, with Δ_{BC} and $\mathcal{P}^*$ as oracles, can extract (x, r_1, r_2) with overwhelming probability in $|N|$, where (I_1, I_2) is given by $\mathcal{P}^*$ as output, and $I_1 = {b_0}^x b_1^{r_1} \bmod N$, $I_2 = a^x b_1^{r_2} \bmod N$. The success probability of $(\mathcal{P}^*, \mathcal{V})$ is taken over the coin tosses of $\mathcal{P}^*$ and $\mathcal{V}$ (including Δ_{BC}), while the success probability of M over those of Δ_{BC}, $\mathcal{P}^*$ and M.*

Lemma 15. (Witness Indistinguishable) *If $m = O(|N|)$ and $x_1, r_1, x_2, r_2 \in [\,0,\, 2^m N)$, the comparing protocol is statistically witness indistinguishable over $R^{(3)}_{(N,b_0,b_1)}$.*

3.5 Mod-Multi Protocol

Let $R^{(4)}_{(N,b_0,b_1)}$:={ $((I_1, I_2, I_3), (x_1, r_1, .., x_3, r_3)) \mid I_i = BC(x_i, r_i),\ x_3 \equiv x_1 x_2 \pmod{n}$ }. The mod-multi protocol is WI over the relation $R^{(4)}_{(N,b_0,b_1)}$ (We call it a mod-multi protocol to confirm $x_3 \equiv x_1 x_2 \pmod{n}$). In the mod-multi protocol, $\mathcal{P}$ can convince $\mathcal{V}$ that he knows $(x_1, x_2, x_3, r_1, r_2, r_3)$ such that $x_3 \equiv x_1 x_2 \pmod{n}$, where $I_1 = BC_{b_0}(x_1, r_1)$, $I_2 = BC_{b_0}(x_2, r_2)$ and $I_3 = BC_{b_0}(x_3, r_3)$.

[Mod-Multi Protocol]

1. $\mathcal{V}$ executes with $\mathcal{P}$ the *set-up procedure* and sends to $\mathcal{P}$, parameters (N, b_0, b_1, b_2).
2. $\mathcal{P}$ sets $I_1 = BC_{b_0}(x_1, r_1)$, $I_2 = BC_{b_0}(x_2, r_2)$, and $I_3 = BC_{b_0}(x_3, r_3)$, and sends them to $\mathcal{V}$.
3. $\mathcal{P}$ sets $I_2^1 = BC_{I_2}(x_1, r_4) = BC_{b_0}(x_1 x_2, r_2 x_1 + r_4)$, and $I_d = BC_{b_0}(d, r_d)$ where $d = (x_3 - x_1 x_2)/n$.
4. $\mathcal{P}$ executes in parallel with $\mathcal{V}$ the *comparing* protocol for (I_1, I_2^1) and the three *basic* protocols for I_2, I_3, and I_d.
5. $\mathcal{P}$ computes $\gamma = (r_2 x_1 + r_4 + r_d n) - r_3$, and executes with $\mathcal{V}$ the *checking* protocol for $b_1^\gamma = I_3 (I_2^1 I_d^n)^{-1} \bmod N$ and range $[\,-2^m N,\ 2^{2m+1} N)$ over the relation $R^{(2)}_{(N,b_1)}$.

In this protocol, $\mathcal{P}$ executes one comparing protocol, three basic protocols, and one checking protocol for b_1^γ in parallel. (in the case of $x_1 = x_2$ the number of the basic protocols is reduced to two). This protocol is also WI.

Lemma 16. (Soundness) *Under Assumption 4, there exists a probabilistic algorithm M such that, for any probabilistic poly-time algorithm $\mathcal{P}^*$, if probabilistic interactive algorithm $(\mathcal{P}^*,\mathcal{V})$ accepts with non-negligible probability in $|N|$, then M, with Δ_{BC} and $\mathcal{P}^*$as oracles, can extract $(x_1, r_1, .., x_3, r_3)$ with overwhelming probability in $|N|$, where (I_1, I_2, I_3) is given by $\mathcal{P}^*$as output, $I_i = BC(x_i, r_i)(i = 1, \ldots, 3)$ and $x_3 \equiv x_1 x_2 \pmod{n}$. The success probability of $(\mathcal{P}^*,\mathcal{V})$ is taken over the coin tosses of $\mathcal{P}^*$ and $\mathcal{V}$ (including Δ_{BC}), while the success probability of M over those of Δ_{BC}, $\mathcal{P}^*$ and M.*

Sketch of Proof:

From lemma 10, if $(\mathcal{P}^*,\mathcal{V})$ has non-negligible success probability, we can construct a probabilistic poly-time knowledge extractor M, which extracts $(x_1, r_1, .., x_3, r_3)$ and d such that $I_i = BC(x_i, r_i)$ and $x_3 \equiv x_1 x_2 + dn \pmod{pq}$. Then the probability of $x_3 \neq x_1 x_2 + dn$ is negligible. If it is non-negligible, we can construct an algorithm M', with poly-time bounded $\mathcal{P}^*$as an oracle, which can factor N given by Δ_{BC} with non-negligible probability in $|N|$. This is a contradiction. M' indeed extract L such that $L = 2(x_3 - x_1 x_2 - dn)$ $(= 2kpq = k\lambda(N))$ where $\lambda(N) = \mathrm{lcm}(P-1, Q-1)$. By Lemma 6, this contradicts Assumption 2 and thereby contradicts Assumption 4. Consequently, M extracts (x_1, x_2, x_3, d) such that $x_3 \equiv x_1 x_2 \pmod{n}$ with overwhelming probability in $|N|$. □

Lemma 17. (Witness Indistinguishable) *If $m = O(|N|)$ and $x_1, r_1, .., x_3, r_3 \in [\,0,\, 2^m N)$, the mod-multi protocol is statistically witness indistinguishable over $R^{(4)}_{(N,b_0,b_1)}$.*

3.6 WI protocol to Confirm $y \equiv ax^5 + b \pmod{n}$

We show, as an example, a WI protocol to confirm $y \equiv ax^5 + b \bmod n$.

Let $[x_1, x_2; x_3]$ be the mod-multi protocol to confirm $x_3 \equiv x_1 x_2 \pmod{n}$ and let $[x_1; x_2]$ be the mod-multi protocol to confirm $x_2 \equiv x_1^2 \pmod{n}$.

Prover $\mathcal{P}$ sets $(I_y, I_x, I_d, I_1, I_2, I_3)$ as $(BC_{b_0}(y, r_y),\ BC_{b_0}(x, r),\ BC_{b_0}(d, r_d),\ BC_{b_0}(t_1),\ BC_{b_0}(t_2),\ BC_{b_0}(t_3, r_3))$ where $d = \frac{y-(at_3+b)}{n}$, $t_1 = x^2 \bmod n$, $t_2 = x^4 \bmod n$ and $t_3 = x^5 \bmod n$. $\mathcal{P}$ executes with $\mathcal{V}$ the two basic protocols for I_y and I_d and the three mod-multi protocols, $[x; t_1]$, $[t_1; t_2]$, and $[x, t_2; t_3]$, in parallel. $\mathcal{P}$ then executes with $\mathcal{V}$ a checking protocol for b_1^γ and range $[\,-2^m N,\ 2^{2m+1} N)$, where $\gamma = ar_3 + r_d n - r_y$ and $b_1^\gamma \equiv I_3^a b_0{}^b (I_y I_d)^{-1} \pmod{N}$.

4 Statistical Zero Knowledge Protocol

In this section, we state the main results of this paper. As mentioned above in Section 3, for any multi-variable polynomial $f(X_1, .., X_t)$ and any modulus n, we can construct a statistical WI protocol to prove that $\mathcal{P}$ knows $(x_1, r_1, .., x_t, r_t, y, r_{t+1})$ such that $I_i = BC(x_i, r_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(y, r_{t+1})$, and $y \equiv f(x_1, .., x_t) \pmod{n}$. This WI protocol can be transformed to the following statistical zero knowledge (SZK) protocol.

Here, we define some terminology. Let f be a multi-variable polynomial. Let $\mathcal{S} := \{(f, n) |\ \exists (x_1, .., x_t) \in Z_n^t$ s.t. $f(x_1, .., x_t) \equiv 0 \bmod n\}$. We can assume, without loss of generality, coefficients of f, number of variables in f, i.e. t, and

parameters (N, b_0, b_1, b_2) are related to modulus n regarding their size (that is, the size of them is $O(|n|)$).

The SZK protocol is constructed as follows:

[SZK Protocol]
common input: (f, n).
output: $I_1, .., I_t$, N, b_0, b_1, and the remaining conversation of $(\mathcal{P}, \mathcal{V})$.
knowledge of $\mathcal{P}$ **:** $(x_1, r_1, .., x_t, r_t)$ such that $I_i = BC_{(N,b_0,b_1)}(x_i, r_i)$ $(i = 1, .., t)$ and $f(x_1, .., x_t) \equiv 0 \pmod{n}$.

1. $\mathcal{V}$ executes with $\mathcal{P}$ a *set-up procedure* for (N, b_0, b_1, b_2).
2. $\mathcal{P}$ sets $I_i := BC_{b_0}(x_i, r_i)$ $(i = 1, .., t)$, $I_{t+1} := BC_{b_0}(0, r_{t+1})$, and sends them to $\mathcal{V}$.
3. $\mathcal{P}$ executes, with $\mathcal{V}$, the WI protocol mentioned above to prove that $\mathcal{P}$ knows $x_1, r_1, .., x_t, r_t$, and y, r_{t+1} such that $I_i = BC(x_i, r_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(y, r_{t+1})$, and $y \equiv f(x_1, .., x_t) \pmod{n}$ where $y = 0$.
4. $\mathcal{P}$ sends r_{t+1} to $\mathcal{V}$.
5. $\mathcal{V}$ checks that $I_{t+1} \equiv BC(0, r_{t+1}) \pmod{N}$.

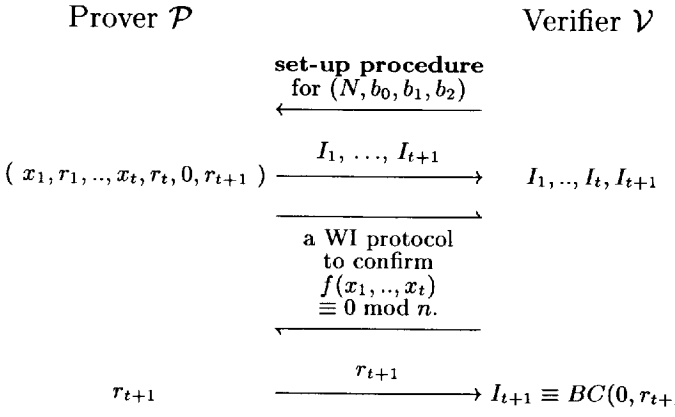

Fig. 1. The SZK protocol that convinces $\mathcal{V}$ that $\mathcal{P}$ knows $(x_1, .., x_t)$ satisfying $f(x_1, .., x_t) \equiv 0 \pmod{n}$ and $I_i = BC(x_i)$ $(i = 1, .., t)$.

Theorem 18. (Soundness) *Under Assumption 4, there exists a probabilistic poly-time algorithm M such that, for any probabilistic poly-time algorithm $\mathcal{P}^*$ and for any input $(f, n) \in \mathcal{S}$, if probabilistic interactive algorithm $(\mathcal{P}^*, \mathcal{V})$ accepts on input (f, n) with non-negligible probability in $|n|$, then M, with Δ_{BC} and $\mathcal{P}^*$ as oracles, can extract $(x_1, r_1, \ldots, x_t, r_t)$ with overwhelming probability in $|n|$, where $I_i = BC(x_i, r_i)$ and $f_t(x_1, .., x_t) \equiv 0 \bmod n$. The success probability of $(\mathcal{P}^*, \mathcal{V})$ is taken over the coin tosses of $\mathcal{P}^*$ and $\mathcal{V}$ (including Δ_{BC}), and the success probability of M over those of Δ_{BC}, $\mathcal{P}^*$ and M.*

Sketch of Proof:

Assume that ($\mathcal{P}^*$,$\mathcal{V}$) has non-negligible success probability. The sketch of the proof is as follows: M executes, with $\mathcal{P}^*$, the set-up procedure for parameters (N, b_0, b_1, b_2) given by Δ_{BC}, in which M should convince $\mathcal{P}^*$ that he knows α, α^{-1}, β, and β^{-1} such that $b_1 = {b_0}^{\alpha} \bmod N$, and $b_2 = {b_0}^{\beta} \bmod N$. Instead of using the values, α, α^{-1}, β, and β^{-1}, M can execute the set-up procedure using the resetable simulation technique for $\mathcal{P}^*$ because the set-up procedure is a zero-knowledge system of $(M, \mathcal{P}^*)$. After M completes the set-up procedure, $\mathcal{P}^*$ sends to M, $I_1, .., I_t$, and I_{t+1} to start a WI protocol. Note that this protocol has (knowledge) soundness over the relation $((I_1, .., I_t, I_{t+1}), (x_1, r_1, .., x_t, r_t, 0, r_{t+1}))$ such that $I_i = BC(x_i, r_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(0, r_{t+1})$, and $f(x_1, .., x_t) \equiv 0 \bmod n$. Therfore M can extract from $\mathcal{P}^*$ desirable witnesses, (x_1, r_1, ..., x_t, r_t, 0, r_{t+1}). □

Theorem 19. (Zero Knowledge) *Let $m = O(|n|)$. There exists a probabilistic algorithm M which runs in expected polynomial time such that, for any $\mathcal{V}^*$, and for any common input $(f, n) \in \mathcal{S}$, the view of $\mathcal{V}^*$ is statistically indistinguishable from the output of M with $\mathcal{V}^*$ as an oracle.*

Sketch of Proof:

Let M be an expected poly-time algorithm allowed to use $\mathcal{V}^*$ as an oracle. M can extract α and α^{-1} from $\mathcal{V}^*$ in the set-up procedure. Let $L := \alpha\alpha^{-1} - 1$. Note that the order of b_0, b_1, and b_2 divides L. Next, M chooses $x'_1, r'_1, .., x'_t, r'_t, r_{t+1} \in_R [\,0,\ 2^m N)$ and sets I'_1, ..., I'_t and $I_{t+1} := BC(0, r_{t+1})$. M computes $y := f(x'_1, .., x'_t) \bmod n$ and $r'_{t+1} := r_{t+1} - \alpha^{-1} y \bmod L$. Note that $I_{t+1} = BC(0, r_{t+1}) = BC(y, r'_{t+1})$. M executes with $\mathcal{V}^*$ a (statistical) WI protocol over the relation $((I'_1, .., I'_t, I_{t+1}), (x'_1, r'_1, .., x'_t, r'_t, y, r'_{t+1}))$ such that $I'_i = BC(x'_i, r'_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(y, r'_{t+1})$, and $y \equiv f(x_1, .., x_t) \pmod n$. Finally, M sends r_{t+1} to $\mathcal{V}^*$.

Here the distribution of (I_1, ..., I_t, I_{t+1}) such that $I_i = BC(x_i, r_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(0, r_{t+1})$ and $f(x_1, .., x_t) \equiv 0 \pmod n$ and that of (I'_1, ..., I'_t, I_{t+1}) such that $I_i = BC(x'_i, r'_i)$ $(i = 1, .., t)$, $I_{t+1} = BC(y, r'_{t+1})$ and $y \equiv f(x_1, .., x_t) \pmod n$ are statistically indistinguishable. In addition, for common input (I'_1, ..., I'_t, I_{t+1}) the protocols with witness (x_1, r_1, ..., x_t, r_t, 0, r_{t+1}) and with witness (x'_1, r'_1, ..., x'_t, r'_t, y, r'_{t+1}) are statistically indistinguishable. Therefore the view of $\mathcal{V}^*$ is also statistically indistinguishable from the output of $M^{\mathcal{V}^*}$. □

Example 1. Suppose that $f(X) \equiv X^e - m \pmod n$. $\mathcal{P}$ can prove, in the statistical zero knowledge manner, that he knows s such that $f(s) \equiv 0 \pmod n$ and $BC(s)$.

Remark. Although the set-up procedure is described in the first step of the proposed SZK protocol, the procedure can be executed in an off-line manner before the remaining protocol begins. In addition, the set-up procedure can be shared by repeated execution of the main protocol. The zero-knowledgeness is still guranteed even if the set-up procedure is shared by repeated execution of the main protocol between $\mathcal{P}$ and $\mathcal{V}$.

5 Application to *Fair Exchange and Contract Signing*

We propose a gradual release protocol to realize *fair exchange and contract signing*. We modify our commitments into *bit releaseable* commitments like those of [Dam93, Dam95] for our gradual release protocol. The protocol is as follows:

$\mathcal{V}$ executes with $\mathcal{P}$ a *set-up procedure* and they hold parameters (N, b_0, b_1, b_2) in common. $\mathcal{P}$ and $\mathcal{V}$ set l such that $|s| < l$ and compute $b_1' := b_1^{2^l} \bmod N$ ($\mathcal{V}$ should prove that he knows $(2^l)^{-1} \bmod pq$ in the ZK manner to show that b_1 and b_1' have the same order) (**set-up phase**). $\mathcal{P}$ sends to $\mathcal{V}$, $(m, BC_{b_0,b_1'}(s, r_1), BC_{b_0,b_1'}(0, r_2))$. For parameters (N, b_0, b_1', b_2), $\mathcal{P}$ executes, with $\mathcal{V}$, the protocol to confirm that $BC_{b,b_1'}(s, r_1)$ and $BC_{b,b_1'}(0, r_2)$ satisfy the relation $s^e - m \equiv 0 \pmod{n}$, where (e, n) are RSA (or Rabin) public-key. $\mathcal{P}$ then open the commitment $BC_{b,b_1'}(0, r)$ (**confirming phase**). $\mathcal{P}$ releases the secret s bit by bit from the least-significant bit (LSB). Let s_k be the remaining secret of s after k bit release. $\mathcal{P}$ opens the LSB of s_k by revealing $X_{k+1} = b_0^{\frac{s_k - i}{2}} b_1^{r 2^{l-k-1}} \bmod N$. $\mathcal{V}$ can know the LSB of s_k by checking $X_k \equiv X_{k+1}^2 b_0^i \pmod{N}$ (**bit by bit release phase**).

6 Efficiency

We compare our protocols with those in [Dam95] from the view points of computational and communication complexity. In [Dam95], the commitment is defined by the form $BC(s, r) = g^s r^{2^l} \bmod N$. As our *comparing* and *mod-multi* protocols are constructed in a similar manner to those in [Dam95], it is enough to compare our *basic* protocol with that in [Dam95]. Our *comparing* protocol is composed of at most two *basic* protocols and our *mod-multi* protocol consists of three *basic*, a *comparing*, and a *checking* protocols. Therefore those in [Dam95] have nearly the same construction. We assume below that $m = |N| = |n| = |c|$.

We estimate the computational complexity of the both *basic* protocols from the number of modular multiplications. In our *basic* protocol, $\mathcal{P}$ needs to compute four auxiliary parameters, t_{ij}'s ($t_{ij} = b_0^{w_i^0} b_1^{w_j^1} b_2^{w_{ij}^2}$), and $\mathcal{V}$ needs to check the verification $t_{ij} I^c = b_0^X b_1^R b_2^{w_{ij}^2}$, where $|w_i^0| = |w_j^1| = |X| = |R| = |2^{2m} N| = 3m$ and $|w_{ij}^2| = |2^m N| = 2m$. $\mathcal{P}$ and $\mathcal{V}$ both need $O(m)$ modular multiplications of N (about $32m$, $9m$ respectively). In Damgård's *basic* protocol, $\mathcal{P}$ needs to compute $2m$ auxiliary parameters, t_i's ($t_i = g^{w_i^0} {w_i^1}^{2^l}$), and $\mathcal{V}$ needs to check m verifications, $t_i I = g^X R^{2^l}$, where $|w_i^0| = |w_i^1| = |X| = 3m$ and $l = 2m$. $\mathcal{P}$ and $\mathcal{V}$ both need $O(m^2)$ modular multiplications of N (about $6m^2$, $3m^2$ respectively). Accordingly, our protocol is about $O(m)$ times more efficient than Damgård's.

The amount of communication in our *basic* protocol is $O(m)$ bits since $4|t_{ij}| + |c| + |X| + |R| + |w_{ij}^2| = 8m$ while that of [Dam95] is $O(m^2)$ bits since $m \cdot (2|t_i| + |X| + |R|) = 4m^2$. Hence, the communication complexity of ours is also $O(m)$ times less than that of Damgård's.

Comparing our protocols with those in [Oka95], the modulus size of Okamoto's bit commitment, $BC(s, r) = g^s G^r \bmod p$, should be at least twice ours. Hence,

our protocol is about $O(m^3)$ times more efficient than [Oka95]. The communication complexity of ours is also $O(m)$ times less than that of [Oka95].

7 Conclusions

We have proposed a bit commitment scheme, $BC(\cdot)$, and related statistical zero knowledge (SZK) protocols in which, for any given multi-variable polynomial $f(X_1, .., X_t)$ and any given modulus n, prover $\mathcal{P}$ gives $(I_1, .., I_t)$ to verifier $\mathcal{V}$ and can convince $\mathcal{V}$ that $\mathcal{P}$ knows $(x_1, .., x_t)$ satisfying $f(x_1, .., x_t) \equiv 0 \pmod{n}$ and $I_i = BC(x_i)$, $(i = 1, .., t)$. The proposed protocols are $O(|n|)$ times more efficient than the corresponding previous ones [Dam93, Dam95, Oka95]. The (knowledge) soundness of our protocols holds under a computational assumption, the intractability of the modified RSA problem, while the (statistical) zero-knowledgeness of the protocols needs no computational assumption. We have also shown the applications of fair exchange and contract signing by using the proposed protocol.

References

[BCC86] G.Brassard, D.Chaum, and C.Crépeau, "Minimum Disclosure Proofs of Knowledge," Journal of Computer and System Sciences, Vol.37, pp.156-189 (1988)

[BG92] Bellare, M. and Goldreich, O., "On Defining Proofs of Knowledge", Proceedings of Crypto 92, pp.390–420 (1992).

[Bra95] Brands, S., "Restrictive Blinding of Secret-Key Certificates", Proceedings of Eurocrypt 95, pp.231–247 (1995).

[CDS94] Cramer, R., Damgård, I. and Schoenmakers, B., "Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols", Proc. of Crypto'94, LNCS, Springer, pp.174-187 (1994)

[CGMA85] Chor, B., Goldwasser, S., Micali, S. and Awerbuch, B., "Verifiable Secret Sharing and Achieving Simultaneity in the Presence of Faults", Proc. of FOCS, pp.383-395 (1985).

[Dam93] Damgård, I., "Practical and Provably Secure Release of a Secret and Exchange of Signatures," Proceedings of Eurocrypt 93 (1993).

[Dam95] Damgård, I., "Practical and Provably Secure Release of a Secret and Exchange of Signatures," vol. 8 pp.201–222, Journal of CRYPTOLOGY(1995).

[FFS88] U.Feige, A.Fiat and A.Shamir, "Zero Knowledge Proofs of Identity," Journal of Cryptology, Vol. 1, pp.77-94 (1988).

[FS90] U.Feige, and A.Shamir, "Witness Indistinguishable and Witness Hiding Protocols," Proc. of STOC90.

[GMRa89] Goldwasser, S., Micali, S., and Rackoff, C., "The knowledge complexity of interactive proof systems", SIAM J. Comput., vol.18, pp.186-208 (1989).

[GMW86] O.Goldreich, S.Micali, and A.Wigderson, "Proofs that Yield Nothing But their Validity and a Methodology of Cryptographic Protocol Design," Proc. FOCS, pp.174-187 (1986)

[Mil76] Miller, G.L., "Riemann's Hypothesis and Tests for Primality", Journal of Computer and System Sciences 13, 300-317 (1976).

[Oka95] Okamoto, T., "An Efficient Divisible Electronic Cash Scheme", Proceedings of Crypto 95, pp.438-451 (1995).

[Ped91] Pedersen, T. P., "Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing", Proceedings of Crypto 91, pp. 129–140 (1992).

[Sta96] Stadler, M., "Publicly Verifiable Secret Sharing", Proc. of Eurocrypt'96, LNCS 1070, Springer, pp.190-199 (1996)

[TW87] Tompa, M., and Woll, H., "Random Self-Reducibility and Zero-Knowledge Interactive-Proofs of Possession of Information", Proc. FOCS, pp 472–482 (1987).

A Proof of Lemma 10

Sketch of Proof:
The top level strategy of knowledge extractor M is as follows:

Protocol:

Step 1 M inputs $1^{|N|}$ to Δ_{BC} and gets parameter (N, b_0, b_1, b_2).

Step 2 M executes, with $\mathcal{P}^*$, the set-up procedure for parameters (N, b_0, b_1, b_2), in which M should convince $\mathcal{P}^*$ that he knows α, α^{-1}, β, and β^{-1} such that $b_1 = {b_0}^{\alpha} \bmod N$, and $b_2 = {b_0}^{\beta} \bmod N$. Instead of using the values, α, α^{-1}, β, and β^{-1}, M can execute the set-up procedure using the resetable simulation technique for $\mathcal{P}^*$ because the set-up procedure is a zero-knowledge system of $(M, \mathcal{P}^*)$.

Step 3 M can extract $(t_{i,j}, c, X, R, w^2_{i,j})$ and $(t_{i,j}, c', X', R', w^2_{i,j})$ for the same $t_{i,j}$, by using $\mathcal{P}^*$ as an oracle.

Step 4 M outputs $(\frac{\Delta X}{\Delta c}, \frac{\Delta X}{\Delta c})$ as a witness of I, where $\Delta c := c - c'$, $\Delta X := X - X'$ and $\Delta R := R - R'$.

We explain Step 3 and Step 4.

Consider Step 3. Let $\epsilon_{i,j}$ be the success probability of $(\mathcal{P}^*, \mathcal{V})$ with the conversation, $(t_{i,j}, c, X, R, w^2_{i,j})$. Note that at least one of $\epsilon_{i,j}$'s is non-negligible if $(\mathcal{P}^*, \mathcal{V})$ accepts with non-negligible probability. Then M can find two different pairs for a $t_{i,j}$ in expected polynomial time in $|N|$. Indeed, the following strategy succeeds with overwhelming probability (See also [FFS88]):

1. For any (i, j), do the following steps.
2. Probe $O(1/\epsilon)$ random entries in $H_{i,j}$ (Here $H_{i,j}$'s are boolean matrices and each $H_{i,j}$'s rows corresponds to all possible states α of RP and its columns correspond to all possible choices c of RV, where the RP is $\mathcal{P}^*$'s random tape, and the RV is $\mathcal{V}$'s random tape.).
3. If find the first $(t_{i,j}, c, X, R, w^2_{i,j})$ which $(\mathcal{P}^*, \mathcal{V})$ accepts, then probe $O(1/\epsilon)$ random entries along the same row in order to find $(t_{i,j}, c', X', R', w^2_{i,j})$ which $(\mathcal{P}^*, \mathcal{V})$ accepts.

In Step 4, $(t_{i,j}, c, X, R, w^2_{i,j})$ and $(t_{i,j}, c', X', R', w^2_{i,j})$ satisfy that $X \equiv cx + w^0_i \bmod pq$, $X' \equiv c'x + w^0_i \bmod pq$, $R \equiv cr + w^1_j \bmod pq$, and $R' \equiv e'R + w^1_j \bmod pq$. Therefore,

$$\Delta X \equiv \Delta c \cdot x \pmod{pq} \quad \text{and} \quad \Delta R \equiv \Delta c \cdot r \pmod{pq}. \tag{3}$$

M can obtain x and r only ΔX and ΔR dividing by Δc respectively, with overwhelming probability in $|N|$ under Assumption 4.

Let $\alpha_0 \in Z_{pq}$ such that $b_1 = b_0^{\alpha_0} \bmod N$. Let $d := gcd(\Delta c, \Delta X + \Delta R\alpha_0)$. From (3), the following relation holds

$$\Delta X + \Delta R\alpha_0 \equiv \Delta c(x + r\alpha_0) \pmod{pq}. \tag{4}$$

Here we replace, without loss of generality, $\mathcal{P}^*$ with the poly-time bounded machine which, on input (N, b_0, b_1, b_2) given by Δ_{BC}, outputs $(I, \Delta c, \Delta X, \Delta R)$ with overwhelming probability in $|N|$. We consider a poly-time bounded algorithm M' using $\mathcal{P}^*$ as an oracle in the following:

Algorithm M'

1. inputs (N, C) generated by Δ_2 to M'.
2. M' picks $b_2 \in_R Z_N$, $\alpha \in_R (0, 2^k N)$ (k is a constant.), then computes $b_1 = C^\alpha \bmod N$.
3. M' inputs (N, C, b_1, b_2) to $\mathcal{P}^*$.
4. If $\mathcal{P}^*$ returns $(I, \Delta c, \Delta X, \Delta R)$, go to the next step, otherwise M halts.
5. M' outputs $(I^Y C^Z \bmod N, \frac{\Delta c}{d})$ and halts, where Y and Z are integers such that
$$\frac{\Delta X + \Delta R\alpha}{d} Y + \frac{\Delta c}{d} Z = 1.$$

Note that $C \equiv C^{\frac{\Delta X + \Delta R\alpha}{d} Y + \frac{\Delta c}{d} Z} \equiv I^{\frac{\Delta c}{d} Y} C^{\frac{\Delta c}{d} Z} \equiv (I^Y C^Z)^{\frac{\Delta c}{d}} \pmod{N}$.

If $d \neq \Delta c$, M' is a machine, with $\mathcal{P}^*$ as an oracle, which can solve the modified RSA problem with non-negligible probability. It contradicts Assumption 4. Therefore $d = \Delta c$, namely $\Delta c | (\Delta X + \Delta R\alpha)$. Moreover, $(\Delta c, \Delta X, \Delta R)$ must satisfy that $\Delta c | \Delta X$ and $\Delta c | \Delta R$ to hold $d = \Delta c$. Let $\alpha = \alpha_0 + \xi pq$. From $d = \Delta c$,

$$\frac{\Delta X + \Delta R\alpha}{\Delta c} = \frac{\Delta X + \Delta R\alpha_0 + \Delta R\xi}{\Delta c}.$$

As even an infinite power $\mathcal{P}^*$ can never know ξ, The condition of $\Delta c | \Delta X$ and $\Delta c | \Delta R$ has to be held to satisfy that of $d = \Delta c$.

Thus, M can extract (x, r) with overwhelming probability in $|N|$.

□