(Convertible) Undeniable Signatures without Random Oracles

Tsz Hon Yuen[1], Man Ho Au[1], Joseph K. Liu[2], and Willy Susilo[1]

[1] Centre for Computer and Information Security Research
School of Computer Science and Software Engineering
University of Wollongong
Wollongong, Australia
{thy738, mhaa456, wsusilo}@uow.edu.au

[2] Department of Computer Science
University of Bristol
Bristol, UK
liu@cs.bris.ac.uk

Abstract. We propose a convertible undeniable signature scheme without random oracles. Our construction is based on Waters' and Kurosawa and Heng's schemes that were proposed in Eurocrypt 2005. The security of our scheme is based on the CDH and the decision linear assumption. Comparing only the part of undeniable signatures, our scheme uses more standard assumptions than the existing undeniable signatures without random oracles due to Laguillamie and Vergnaud.

Keywords: Convertible undeniable signature, random oracle model, pairings

1 Introduction

Standard digital signatures allow universal verification. However in some real world scenarios, privacy is an important issue. In this situation, we may require that the verification of signatures is restricted by the signer. Then, the verification of a signature requires an interaction with the signer. A signer can deny generating a signature that he never signs, but cannot deny one that he signs. The proof by the signer cannot be transferred to convince other verifiers. This concept is known as the "Undeniable Signatures" that was proposed by Chaum and van Antwerpen [11]. Later, Boyar, Chaum, Damgård and Pedersen [6] proposed an extension called "Convertible Undeniable Signatures", that allows the possibility to transform an undeniable signature into a self-authenticating signature. This transformation can be restricted to a particular signature only, or can be applied to all signatures of a signer.

There are many different undeniable signatures with variable features and security levels. These features include convertibility [6, 13, 23, 24], designated verifier technique [16], designated confirmer technique [10, 25], identity based scheme [22], time-selective scheme [21], etc. The security for undeniable signatures is said to be *secure* if it is unforgeable, invisible and the confirmation and disavowal protocols are zero-knowledge. It is believed that the zero-knowledgeness is required to make undeniable signatures non-transferable. However, Kurosawa and Heng [18] suggested that zero-knowledgeness and non-transferability can be separated; and the concept of witness indistinguishability can be incorporated. They proposed another security notion called impersonation attack.

The random oracle model [3] is a popular technique in provable security. However several papers proved that some cryptosystems secure in the random oracle were actually provably insecure when the random oracle was instantiated by any real-world hashing functions [9, 2]. As a result, recently there are many new signature schemes which prove their security without random oracles, such as group signatures [1, 8], ring signatures [12, 4], blind signatures [17], group-oriented signatures [26], undeniable signatures [20], universal designated verifier signatures [28], etc. Nonetheless, some of them introduce new security assumptions that are not well studied, which are the main drawback of some schemes.

Our Contribution. We propose the *first* convertible undeniable signatures without random oracles in pairings. Most of the existing convertible undeniable signatures are proven secure in the random oracle model only [6, 23, 24, 21] [3], except the recent construction in RSA [19].

Most efficient undeniable signatures are proven secure in the random oracle model only. [14] is secure in the random oracle model currently. [4] Recently, Languillaumie and Vergnaud proposed the first efficient undeniable signatures without random oracles [20]. However, their anonymity relies on their *new assumption* DSDH, while their unforgeability relies on the GSDH assumption with the access of a DSDH oracle, which seems to be contradictory. Our proposed variant of undeniable signature is proven unforgeable by the CDH assumption and anonymous by the decision linear assumption. Therefore by removing the protocol for convertible parts, our undeniable signature scheme is the *first* proven secure scheme *without using random oracles* and *without using a new assumption* in discrete logarithm settings.

We extend the security model of [18] to convertible undeniable signatures. We also use the 3-move witness indistinguishable (WI) protocol in [18]. Therefore we incorporate the concept of WI into the convertible undeniable signatures and propose the first 3-move convertible undeniable signatures.

Organization. The next section briefly explains the pairings and some related intractability problems. Section 3 gives the security model and some basic building blocks are given in Section 4. Section 5 gives our construction and security proofs. The paper ends with some concluding remarks.

[3] [13] does not prove the invisibility property. The authors only conjecture the security in section 5.1 and 5.2.

[4] Refer to section 1.1 in [19] for details.

2 Preliminaries

2.1 Pairings and Intractability Problem

Our scheme uses bilinear pairings on elliptic curves. We now give a brief revision on the property of pairings and candidate hard problem from pairings that will be used later.

Let $\mathbb{G}, \mathbb{G}_T$ be cyclic groups of prime order p, writing the group action multiplicatively. Let g be a generator of $\mathbb{G}$.

Definition 1. *A map $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is called a bilinear pairing if, for all $x, y \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$, we have $\hat{e}(x^a, y^b) = \hat{e}(x, y)^{ab}$, and $\hat{e}(g, g) \neq 1$.*

Definition 2 (CDH). *The Computational Diffie-Hellman (CDH) problem is that, given g, $g^x, g^y \in \mathbb{G}$ for unknown $x, y \in \mathbb{Z}_p^*$, to compute g^{xy}.*

We say that the (ϵ, t)-CDH assumption holds in $\mathbb{G}$ if no t-time algorithm has the non-negligible probability ϵ in solving the CDH problem.

Definition 3 (Decision Linear [5]). *The Decision Linear problem is that, given u, u^a, v, v^b, h, $h^c \in \mathbb{G}$ for unknown $a, b, c \in \mathbb{Z}_p^*$, to output 1 if $c = a + b$ and output 0 otherwise.*

We say that the (ϵ, t)-Decision Linear assumption holds in $\mathbb{G}$ if no t-time algorithm has probability over half ϵ in solving the Decision Linear problem in $\mathbb{G}$. The decision linear assumption is proposed in [5] to prove the security of short group signatures. It is also used in [7] and [15] for proving the security of anonymous hierarchical identity-based encryption and obfuscating re-encryption respectively.

3 Undeniable Signature Security Models

In this section we review the security notions and model of (convertible) undeniable signatures. Unforgeability and invisibility are popular security requirement for undeniable signatures. Kurosawa and Heng [18] proposed another security notion called impersonation. We will use the security model of [18], and extend it to convertible undeniable signatures. The changes for convertible undeniable signatures will be given in brackets.

3.1 Security Notions

An (convertible) undeniable signature scheme has the following algorithms:

- **Setup.** On input security parameter 1^λ, outputs public parameters param.
- **Key Generation.** On input public parameters param, outputs a public key pk and a secret key sk.
- **Sign.** On input public parameters param, a secret key sk and a message m, outputs an undeniable signature σ.

- **Confirm/Deny.** This is an interactive protocol between a prover and a verifier. Their common inputs are public parameters param, a public key pk, a message m and a signature σ. The prover's private input is a secret key sk. At the end of the protocol, the verifier outputs 1 if σ is a valid signature of m and outputs 0 otherwise.

  (The following algorithms are for convertible schemes only.)
- **Individual Conversion.** On input public parameters param, a secret key sk, a message m and a signature σ, outputs an individual receipt r which makes it possible to universally verify σ.
- **Individual Verification.** On input public parameters param, a public key pk, a message m, a signature σ and an individual receipt r, outputs $\perp$ if r is an invalid receipt. Otherwise, outputs 1 if σ is a valid signature of m and outputs 0 otherwise.
- **Universal Conversion** On input public parameters param and a secret key sk, outputs an universal receipt R which makes it possible to universally verify all signatures for pk.
- **Universal Verification.** On input public parameters param, a public key pk, a message m, a signature σ and an universal receipt R, outputs $\perp$ if R is an invalid receipt. Otherwise, outputs 1 if σ is a valid signature of m and outputs 0 otherwise.

3.2 Unforgeability

Existential unforgeability against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a simulator $\mathcal{S}$.

1. $\mathcal{S}$ gives the public keys and parameters to $\mathcal{A}$. (For convertible schemes, $\mathcal{S}$ also gives $\mathcal{A}$ the universal receipt R.)
2. $\mathcal{A}$ can query the following oracles:
   - Signing queries: $\mathcal{A}$ adaptively queries q_s times with input message m_i, and obtains a signature σ_i.
   - Confirmation/disavowal queries: $\mathcal{A}$ adaptively queries q_c times with input message-signature pair (m_i, σ_i). If it is a valid pair, the oracle returns a bit $\mu = 1$ and proceeds with the execution of the confirmation protocol with $\mathcal{A}$. Otherwise, the oracle returns a bit $\mu = 0$ and proceeds with the execution of the disavowal protocol with $\mathcal{A}$.
     (For convertible scheme, this oracle is not necessary as the universal receipt is given.)
3. Finally $\mathcal{A}$ outputs a message-signature pair (m^*, σ^*) where m^* has never been queried to the signing oracle.

$\mathcal{A}$ wins the game if σ^* is a valid signature for m^*.

Definition 4. *A (convertible) undeniable signature scheme is (ϵ, t, q_c, q_s)-unforgeable against chosen message attack if there is no t time adversary winning the above game with probability greater than ϵ.*

3.3 Invisibility

Invisibility against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a simulator $\mathcal{S}$.

1. $\mathcal{S}$ gives the public keys and parameters to $\mathcal{A}$.
2. $\mathcal{A}$ can query the following oracles:
   - Signing queries, Confirmation/disavowal queries: same as unforgeability.
   - (For convertible schemes only.) Receipt generating oracle: $\mathcal{A}$ adaptively queries q_r times with input message-signature pair (m_i, σ_i), and obtains an individual receipt r.
3. $\mathcal{A}$ outputs a message m^* which has never been queried to the signing oracle, and requests a challenge signature σ^* on m^*. σ^* is generated based on a hidden bit b. If $b = 1$, then σ^* is generated as usual using the signing oracle, otherwise σ^* is chosen uniformly at random from the signature space.
4. $\mathcal{A}$ can adaptively query the signing oracle and confirmation/disavowal oracle, where no signing query (and receipt generating query) for m^* and no confirmation/disavowal query for (m^*, σ^*) is allowed.
5. Finally $\mathcal{A}$ outputs a guessing bit b'

$\mathcal{A}$ wins the game if $b = b'$. $\mathcal{A}$'s advantage is $Adv(\mathcal{A}) = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 5. *A (convertible) undeniable signature scheme is* $(\epsilon, t, q_c, q_r, q_s)$*-invisible if there is no* t *time adversary winning the above game with advantage greater than* ϵ.

3.4 Impersonation

Impersonation against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a simulator $\mathcal{S}$.

1. $\mathcal{S}$ gives the public keys and parameters to $\mathcal{A}$.
2. $\mathcal{A}$ can query the Signing oracle and Confirmation/disavowal oracle, which are the same as the one in unforgeability.
3. Finally $\mathcal{A}$ outputs a message-signature pair (m^*, σ^*) and a bit b. If $b = 1$, $\mathcal{A}$ executes the confirmation protocol with $\mathcal{S}$. Otherwise $\mathcal{A}$ executes the disavowal protocol with $\mathcal{S}$.

$\mathcal{A}$ wins the game if $\mathcal{S}$ is convinced that σ^* is a valid signature for m^* if $b = 1$, or is an invalid signature for m^* if $b = 0$.

Definition 6. *A (convertible) undeniable signature scheme is* (ϵ, t, q_c, q_s)*-secure against impersonation if there is no* t *time adversary winning the above game with probability at least* ϵ.

Remark: For convertible schemes, if an adversary can forge an individual or universal receipt, he can always convince a verifier in the interactive protocol, by directly giving the receipt to him. Therefore the model of impersonation attack already includes the security notion regarding receipts in convertible schemes.

4 Basic Building Blocks

4.1 Waters Signature Scheme

Waters [27] presented a secure signature scheme based on CDH problem without random oracles. The scheme is summarized as follows:

1. **Gen**. Randomly choose $\alpha \in \mathbb{Z}_p$ and let $g_1 = g^\alpha$. Additionally, choose two random values $g_2, u' \in \mathbb{G}$ and a random n-length vector $\mathsf{U} = (u_i)$, whose elements are chosen at random from $\mathbb{G}$. The public key is $pk = (g_1, g_2, u', \mathsf{U})$ and the secret key is g_2^α.
2. **Sign**. To generate a signature on message $M = (\mu_1, \ldots, \mu_n) \in \{0,1\}^n$, pick $s \in_R \mathbb{Z}_p^*$ and output the signature as $\sigma{=}(g_2^\alpha \cdot (u' \prod_{j=1}^{n} u_j^{\mu_j})^s,\ g^s)$ with his secret key g_2^α.
3. **Verify**. Given a signature $\sigma = (\sigma_1, \sigma_2)$ on message $M = (\mu_1, \ldots, \mu_n) \in \{0,1\}^n$, it outputs 1 if $\hat{e}(g, \sigma_1) = \hat{e}(g_1,\, g_2) \cdot \hat{e}(u' \prod_{i=1}^{n} u_i^{\mu_i},\, \sigma_2)$. Otherwise, it outputs 0.

4.2 WI Protocol

We review the witness indistinguishable (WI) protocol for Diffie-Hellman (DH) tuple and non-DH tuple from [18]. Let $\mathbb{G}$ be an Abelian group with prime order p. Let L be a generator of $\mathbb{G}$. We say that $(L, L^\alpha, L^\beta, L^\gamma)$ is a DH tuple if $\gamma = \alpha\beta \bmod p$. Kurosawa and Heng [18] proposed a WI protocol to prove if (L, M, N, O) is a DH tuple or non-DH tuple using the knowledge of α ($= \log_L M$). For the details of the definition and security model of WI protocol, please refer to [18] for details. We summarize the protocols in table 1 and 2.

	Prover		Verifier
	$c_2, d_2, r \xleftarrow{R} \mathbb{Z}_p$ $z'_1 = L^{d_2}/N^{c_2}$ $z'_2 = M^{d_2}/O^{c_2}$ $z_1 = L^r$		
1	$z_2 = N^r$	$\xrightarrow{z_1, z_2, z'_1, z'_2}$	
2		$\xleftarrow{c}$	$c \xleftarrow{R} \mathbb{Z}_p$
	$c_1 = c - c_2 \bmod p$		
3	$d_1 = r + c_1\alpha \bmod p$	$\xrightarrow{c_1, c_2, d_1, d_2}$	
			$c \stackrel{?}{=} c_1 + c_2 \bmod p$ $L^{d_1} \stackrel{?}{=} z_1 M^{c_1}$ $L^{d_2} \stackrel{?}{=} z'_1 N^{c_2}$ $N^{d_1} \stackrel{?}{=} z_2 O^{c_1}$ $M^{d_2} \stackrel{?}{=} z'_2 O^{c_2}$

Table 1. WI protocol for DH tuple (L, M, N, O)

	Prover		Verifier
	$c_2, d_1', d_2', r, a, b \xleftarrow{R} \mathbb{Z}_p$		
	$A' \xleftarrow{R} \mathbb{G}$ with $A' \neq 1$		
	$z_1' = M^{d_1'}/(O^{d_2'}A'^{c_2})$		
	$z_2' = L^{d_1'}/N^{d_2'}$		
	$A = (N^\alpha/O)^r$		
	$z_1 = N^a/O^b$		
1	$z_2 = L^a/M^b$	$\xrightarrow{A, A', z_1, z_2, z_1', z_2'}$	$A \stackrel{?}{\neq} 1,\ A' \stackrel{?}{\neq} 1$
2		$\xleftarrow{c}$	$c \xleftarrow{R} \mathbb{Z}_p$
	$c_1 = c - c_2 \bmod p$		
	$d_1 = a + c_1\alpha r \bmod p$		
3	$d_2 = b + c_1 r \bmod p$	$\xrightarrow{c_1, c_2, d_1, d_2, d_1', d_2'}$	
			$c \stackrel{?}{=} c_1 + c_2 \bmod p$
			$N^{d_1}/O^{d_2} \stackrel{?}{=} z_1 A^{c_1}$
			$M^{d_1'}/O^{d_2'} \stackrel{?}{=} z_1' A'^{c_2}$
			$L^{d_1}/M^{d_2} \stackrel{?}{=} z_2$
			$L^{d_1'}/N^{d_2'} \stackrel{?}{=} z_2'$

Table 2. WI protocol for non-DH tuple (L, M, N, O)

5 Convertible Undeniable Signature Scheme

5.1 Scheme Construction

In this section, we present our convertible undeniable signature scheme. The scheme consists of the following algorithms.

Setup. Let $\mathbb{G}, \mathbb{G}_T$ be groups of prime order p. Given a pairing: $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. Select generators $g, g_2 \in \mathbb{G}$. Generator $u' \in \mathbb{G}$ is selected in random, and a random n-length vector $\mathsf{U} = (u_i)$, whose elements are chosen at random from $\mathbb{G}$.

Select an integer d as a system parameter. Denote $\ell = 2^d$ and $k = n/d$. Let $H_j : \{0,1\}^n \to \mathbb{Z}_\ell^*$ be collision resistant hash functions, where $1 \leq j \leq k$.

Key Generation. Randomly select $\alpha, \beta', \beta_i \in \mathbb{Z}_p^*$ for $1 \leq i \leq \ell$. Set $g_1 = g^\alpha$, $v' = g^{\beta'}$ and $v_i = g^{\beta_i}$. The public keys are $(g_1, v', v_1, \ldots, v_\ell)$. The secret keys are $(\alpha, \beta', \beta_1, \ldots, \beta_\ell)$.

Sign. To sign a message $m = (m_1, \ldots, m_n) \in \{0,1\}^n$, denote $\bar{m}_j = H_j(m)$ for $1 \leq j \leq k$. The signer picks $r \in_R \mathbb{Z}_p^*$ and computes the signature:

$$S_1 = g_2^\alpha (u' \prod_{i=1}^{n} u_i^{m_i})^r \qquad S_{2,j} = (v' \prod_{i=1}^{\ell} v_i^{\bar{m}_j^i})^r$$

The output signature is $(S_1, S_{2,1}, \ldots, S_{2,k})$.

Confirm/Deny. On input $(S_1, S_{2,1}, \ldots, S_{2,k})$, the signer computes for $1 \leq j \leq k$

$$\begin{aligned} L &= \hat{e}(g, g_2) \\ M &= \hat{e}(g_1, g_2) \\ N_j &= \hat{e}(v' \prod_{i=1}^{\ell} v_i^{\bar{m}_j^i}, g_2) \\ O_j &= \hat{e}(v' \prod_{i=1}^{\ell} v_i^{\bar{m}_j^i}, S_1) / \hat{e}(S_{2,j}, u' \prod_{i=1}^{n} u_i^{m_i}). \end{aligned} \quad (1)$$

We have the 3-move WI protocols of the equality or the inequality of discrete logarithm $\alpha = log_L M$ and $log_{N_j} O_j$ in $\mathbb{G}_T$ shown in table 1 and 2.

Individual Conversion. Upon input the signature $(S_1, S_{2,1}, \ldots, S_{2,k})$ on the message m, the signer computes $\bar{m}_1 = H_1(m)$ and:

$$S_2' = S_{2,1}^{1/(\beta' + \sum_{i=1}^{\ell} \beta_i \bar{m}_1^i)}$$

Output the individual receipt S_2' for message m.

Individual Verification. Upon input the signature $(S_1, S_{2,1}, \ldots, S_{2,k})$ for the message m and the individual receipt S_2', compute $\bar{m}_j = H_j(m)$ for $1 \leq j \leq k$ and check if:

$$\hat{e}(g, S_{2,j}) \stackrel{?}{=} \hat{e}(S_2', v' \prod_{i=1}^{\ell} v_i^{\bar{m}_j^i})$$

If they are not equal, output $\perp$. Otherwise compare if:

$$\hat{e}(g, S_1) \stackrel{?}{=} \hat{e}(g_1, g_2) \cdot \hat{e}(S_2', u' \prod_{i=1}^{n} u_i^{m_i})$$

Output 1 if the above holds. Otherwise output 0.

Universal Conversion. The signer publishes his universal receipt (β', β_1, ..., β_ℓ).

Universal Verification. Upon input the signature $(S_1, S_{2,1}, \ldots, S_{2,k})$ on the message m and the universal receipt $(\beta', \beta_1, \ldots, \beta_\ell)$, check if:

$$v' \stackrel{?}{=} g^{\beta'} \qquad v_i \stackrel{?}{=} g^{\beta_i} \qquad \text{for } 1 \leq i \leq \ell$$

If they are not equal, output $\perp$. Otherwise compute $\bar{m}_j = H_j(m)$ for $1 \leq j \leq k$ and compare if:

$$\hat{e}(g, S_1) \stackrel{?}{=} \hat{e}(g_1, g_2) \cdot \hat{e}(S_{2,j}^{1/(\beta' + \sum_{i=1}^{\ell} \beta_i \bar{m}_j^i)}, u' \prod_{i=1}^{n} u_i^{m_i})$$

Output 1 if the above holds. Otherwise output 0.

5.2 Security Result

Theorem 1. *The scheme is* (ϵ, t, q_s)*-unforgeable if the* (ϵ', t')*-CDH assumption holds in* $\mathbb{G}$*, where*

$$\epsilon' \geq \frac{\epsilon}{4q_s(n+1)}$$
$$t' = t + O\Big(q_s\rho + (n+\ell)q_s\omega\Big)$$

and $H_j : \{0,1\}^n \to \mathbb{Z}_\ell^*$*, where* $1 \leq j \leq k$*, are some collision resistant hash functions and* ρ*,* ω *are the time for an exponentiation in* $\mathbb{G}$ *and an addition in* $\mathbb{Z}_p$ *respectively.*

Proof. Assume there is a (ϵ, t, q_s)-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the CDH problem with probability at least ϵ' and in time at most t'.

$\mathcal{B}$ is given a CDH problem instance (g, g^a, g^b). In order to use $\mathcal{A}$ to solve for the problem, $\mathcal{B}$ needs to simulates a challenger and the oracles for $\mathcal{A}$. $\mathcal{B}$ does it in the following way.

Setup. Let $l_p = 2q_s$. $\mathcal{B}$ randomly selects integer κ such that $0 \leq \kappa \leq n$. Also assume that $l_p(n+1) < p$ for the given values of q_s, and n. It randomly selects the following integers:

- $x' \in_R \mathbb{Z}_{l_p}$; $y' \in_R \mathbb{Z}_p$
- $x_i \in_R \mathbb{Z}_{l_p}$, for $i = 1, \ldots, n$. Let $\hat{X} = \{x_i\}$.
- $y_i \in_R \mathbb{Z}_p$, for $i = 1, \ldots, n$. Let $\hat{Y} = \{y_i\}$.

We further define the following functions for binary strings $\mathfrak{m} = (m_1, \ldots, m_n)$ as follow:

$$F(\mathfrak{m}) = x' + \sum_{i=1}^{n} x_i m_i - l_p\kappa \qquad \text{and} \qquad J(\mathfrak{m}) = y' + \sum_{i=1}^{n} y_i m_i$$

$\mathcal{B}$ randomly picks $\beta', \beta_i \in \mathbb{Z}_p^*$ for $1 \leq i \leq \ell$. Set $v' = g^{\beta'}$ and $v_i = g^{\beta_i}$. $\mathcal{B}$ constructs a set of public parameters as follow:

$$g, \qquad g_2 = g^b, \qquad u' = g_2^{-l_p\kappa + x'} g^{y'}, \qquad u_i = g_2^{x_i} g^{y_i} \ \text{ for } \ 1 \leq i \leq n$$

The signer's public key is $(g_1 = g^a, v', v_1, \ldots, v_\ell)$.

Denote $G(\mathfrak{m}) = \beta' + \sum_{i=1}^{\ell} \beta_i \mathfrak{m}^i$. Note that we have the following equation:

$$u' \prod_{i=1}^{n} u_i^{m_i} = g_2^{F(\mathfrak{m})} g^{J(\mathfrak{m})}, \qquad v' \prod_{i=1}^{\ell} v_i^{\bar{\mathfrak{m}}_j^i} = g^{G(\bar{\mathfrak{m}}_j)} \qquad \text{for } 1 \leq j \leq k$$

where $\bar{\mathfrak{m}}_j = H_j(\mathfrak{m})$ for $1 \leq j \leq k$. All public parameters and universal receipt $(\beta', \beta_1, \ldots, \beta_\ell)$ are passed to $\mathcal{A}$.

Oracles Simulation. $\mathcal{B}$ simulates the oracles as follow:

(*Signing oracle.*) Upon receiving query for message $\mathfrak{m}_i = \{m_1, \ldots, m_n\}$, although $\mathcal{B}$ does not know the secret key, it still can construct the signature by assuming $F(\mathfrak{m}_i) \neq 0 \bmod p$. It randomly chooses $r_i \in_R \mathbb{Z}_p$ and computes the signature as

$$S_1 = g_1^{-\frac{J(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} \big(g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)}\big)^{r_i}, \quad S_{2,j} = (g_1^{-\frac{1}{F(\mathfrak{m}_i)}} g^{r_i})^{G(\bar{\mathfrak{m}}_{i,j})}$$

where $\bar{\mathfrak{m}}_{i,j} = H_j(\mathfrak{m}_i)$ for $1 \leq j \leq k$.

By letting $\tilde{r}_i = r_i - \frac{a}{F(\mathfrak{m}_i)}$, it can be verified that $(S_1, S_{2,1}, \ldots, S_{2,k})$ is a signature, shown as follow:

$$\begin{aligned}
S_1 &= g_1^{-\frac{J(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} (g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)})^{r_i} \\
&= g^{-\frac{aJ(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} (g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)})^{\frac{a}{F(\mathfrak{m}_i)}} (g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)})^{-\frac{a}{F(\mathfrak{m}_i)}} (g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)})^{r_i} \\
&= g^{-\frac{aJ(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} g_2^a g^{\frac{aJ(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} (g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)})^{\tilde{r}_i} \\
&= g_2^a (u' \prod_{j=1}^{n} u_j^{m_j})^{\tilde{r}_i}
\end{aligned}$$

$$S_{2,j} = (g_1^{-\frac{1}{F(\mathfrak{m}_i)}} g^{r_i})^{G(\bar{\mathfrak{m}}_{i,j})} = (g^{r_i - \frac{a}{F(\mathfrak{m}_i)}})^{G(\bar{\mathfrak{m}}_{i,j})} = g^{G(\bar{\mathfrak{m}}_i)\tilde{r}_i} = (v' \prod_{w=1}^{\ell} v_w^{\bar{\mathfrak{m}}_{i,j}^w})^{\tilde{r}_i}$$

$\mathcal{B}$ outputs the signature $(S_1, S_{2,1}, \ldots, S_{2,k})$. To the adversary, all signatures given by $\mathcal{B}$ are indistinguishable from the signatures generated by the signer.

If $F(\mathfrak{m}_i) = 0 \bmod p$, since the above computation cannot be performed (division by 0), the simulator aborts. To make it simple, the simulator will abort if $F(\mathfrak{m}_i) = 0 \bmod l_p$. The equivalence can be observed as follow. From the assumption $l_p(n+1) < p$, it implies $0 \leq l_p\kappa < p$ and $0 \leq x' + \sum_{i=1}^{n} x_i m_i < p$ ($\because x', x_i < l_p$). We have $-p < F(\mathfrak{m}_i) < p$ which implies if $F(\mathfrak{m}_i) = 0 \bmod p$ then $F(\mathfrak{m}_i) = 0 \bmod l_p$. Hence, $F(\mathfrak{m}_i) \neq 0 \bmod l_p$ implies $F(\mathfrak{m}_i) \neq 0 \bmod p$. Thus the former condition will be sufficient to ensure that a signature can be computed without abort.

Output. Finally $\mathcal{A}$ outputs a signature $(S_1^*, S_{2,1}^*, \ldots, S_{2,k}^*)$ for message $\mathfrak{m}^*$. $\mathcal{B}$ checks if $F(\mathfrak{m}^*) = 0 \bmod p$. If not, $\mathcal{B}$ aborts. Otherwise $\mathcal{B}$ computes $\bar{\mathfrak{m}}_1^* = H_1(\mathfrak{m}^*)$ and outputs

$$\frac{S_1^*}{S_{2,1}^{*\;J(\mathfrak{m}^*)/G(\bar{\mathfrak{m}}_1^*)}} = \frac{g_2^a \Big(u' \prod_{i=1}^{n} u_i^{\mathfrak{m}_i^*}\Big)^r}{\Big(v' \prod_{i=1}^{\ell} v_i^{\bar{\mathfrak{m}}_1^{*i}}\Big)^{rJ(\mathfrak{m}^*)/G(\bar{\mathfrak{m}}_1^*)}} = \frac{g_2^a \Big(g^{J(\mathfrak{m}^*)}\Big)^r}{g^{rJ(\mathfrak{m}^*)}} = g^{ab}$$

which is the solution to the CDH problem instance.

Probability Analysis. For the simulation to complete without aborting, we require the following conditions fulfilled:

1. Sign queries on message $\mathfrak{m}_i$ with $F(\mathfrak{m}_i) \neq 0 \bmod p$.
2. Challenge message $\mathfrak{m}^*$ with $F(\mathfrak{m}^*) = 0 \bmod p$.

In order to make the analysis simple, we will bound the probability of a subcase of this event. We define the events A_i, A^* as follow:

$$A^* : F(\mathfrak{m}^*) = 0 \bmod l_p; \qquad A_i : F(\mathfrak{m}_i) \neq 0 \bmod l_p \qquad \text{where } i = 1, \ldots, q_s$$

The probability of $\mathcal{B}$ not aborting is

$$\Pr[\textsf{not abort}] \geq \Pr\left[\bigwedge_{i=1}^{q_s} A_i \wedge A^*\right]$$

The assumption $l_p(n+1) < p$ implies if $F(\mathfrak{m}^*) = 0 \bmod p$ then $F(\mathfrak{m}^*) = 0 \bmod l_p$. In addition, it also implies that if $F(\mathfrak{m}^*) = 0 \bmod l_p$, there will be a unique choice of κ with $0 \leq \kappa \leq n$ such that $F(\mathfrak{m}^*) = 0 \bmod p$. Since κ, x' and x_i's are randomly chosen,

$$\begin{aligned}
\Pr[A^*] &= \Pr[F(\mathfrak{m}^*) = 0 \bmod p \wedge F(\mathfrak{m}^*) = 0 \bmod l_p] \\
&= \Pr[F(\mathfrak{m}^*) = 0 \bmod l_p] \Pr[F(\mathfrak{m}^*) = 0 \bmod p \mid F(\mathfrak{m}^*) = 0 \bmod l_p] \\
&= \frac{1}{l_p} \frac{1}{n+1}
\end{aligned}$$

We have:

$$\Pr\left[\Big(\bigwedge_{i=1}^{q_s} A_i | A^*\Big)\right] \geq 1 - \frac{q_s}{l_p}$$

By letting $l_p = 2q_S$, we have

$$\begin{aligned}
\Pr[\textsf{not abort}] &= \Pr\left[\bigwedge_{i=1}^{q_s} A_i \wedge A^*\right] \\
&= \Pr[A^*] \Pr\left[\bigwedge_{i=1}^{q_s} A_i | A^*\right] \\
&\geq \frac{1}{l_p(n+1)}\Big(1 - \frac{q_s}{l_p}\Big) \\
&= \frac{1}{4q_s(n+1)}
\end{aligned}$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and $O(n+\ell)$ modular addition in $\mathbb{Z}_p$ in the signing stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big(q_s\rho + (n+\ell)q_s\omega\Big)$$

$\square$

Theorem 2. *The scheme is $(\epsilon, t, q_c, q_r, q_s)$-invisible if the (ϵ', t')-decision linear assumption holds in $\mathbb{G}$, where*

$$\epsilon' \geq \epsilon \cdot \frac{1}{4(q_s+1)(n+1)(q_s+q_r)^k} \cdot \Big(1 - \frac{1}{q_s+q_r}\Big)^{(q_s+q_r)k}$$
$$t' = t + O\Big((q_s+q_r)\rho + q_c\tau + \Big(nq_s + \ell\Big)\omega\Big)$$

where $H_j : \{0,1\}^n \to \mathbb{Z}_\ell^*$, where $1 \leq j \leq k$, are some collision resistant hash functions and ρ, τ, ω are the time for an exponentiation in $\mathbb{G}$, an exponentiation in $\mathbb{G}_T$ and an addition in $\mathbb{Z}_p$ respectively, under the assumption that $\ell > q_s + q_r$.

Proof. Assume there is a $(\epsilon, t, q_c, q_r, q_s)$-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the decisional linear problem with probability at least ϵ' and in time at most t'.

$\mathcal{B}$ is given a decisional linear problem instance (u, v, h, u^a, v^b, h^c). In order to use $\mathcal{A}$ to solve for the problem, $\mathcal{B}$ needs to simulates the oracles for $\mathcal{A}$. $\mathcal{B}$ does it in the following way.

Setup. Let $l_p = 2(q_s+1)$. $\mathcal{B}$ randomly selects integer κ such that $0 \leq \kappa \leq n$. Also assume that $l_p(n+1) < p$ for the given values of q_c, q_r, q_s, and n. It randomly selects the following integers:

- $x' \in_R \mathbb{Z}_{l_p}$; $y' \in_R \mathbb{Z}_p$
- $x_i \in_R \mathbb{Z}_{l_p}$, for $i = 1, \ldots, n$. Let $\hat{X} = \{x_i\}$.
- $y_i \in_R \mathbb{Z}_p$, for $i = 1, \ldots, n$. Let $\hat{Y} = \{y_i\}$.

We further define the following functions for binary strings $\mathfrak{m} = (m_1, \ldots, m_n)$ as follow:

$$F(\mathfrak{m}) = x' + \sum_{i=1}^{n} x_i m_i - l_p\kappa \qquad \text{and} \qquad J(\mathfrak{m}) = y' + \sum_{i=1}^{n} y_i m_i - l_p\kappa$$

Then $\mathcal{B}$ randomly picks a set of distinct numbers $\mathcal{S} = \{c_1^*, \ldots, c_s^*\} \in (\mathbb{Z}_\ell^*)^s$. We further define the following functions for any integer $\bar{\mathfrak{m}} \in \mathbb{Z}_\ell^*$

$$G(\bar{\mathfrak{m}}) = \prod_{i \in \mathcal{S}} (\bar{\mathfrak{m}} - i) = \sum_{i=0}^{s} \gamma_i \bar{\mathfrak{m}}^i \qquad \text{and} \qquad K(\bar{\mathfrak{m}}) = \prod_{i=1, i \notin \mathcal{S}}^{\ell} (\bar{\mathfrak{m}} - i) = \sum_{i=0}^{\ell-s} \alpha_i \bar{\mathfrak{m}}^i$$

for some $\gamma_i, \alpha_i \in \mathbb{Z}_p^*$.

$\mathcal{B}$ constructs a set of public parameters as follow:

$$g = u, \qquad g_2 = h, \qquad u' = g_2^{-lk+x'} g^{-lk+y'}, \qquad u_i = g_2^{x_i} g^{y_i} \text{ for } 1 \leq i \leq n$$

The signer's public key is:

$$g_1 = u^a, \qquad v' = v^{\alpha_0} g^{\gamma_0}, \qquad v_i = v^{\alpha_i} g^{\gamma_i} \text{ for } 1 \leq i \leq s, \qquad v_j = v^{\alpha_i}$$

for $s+1 \le i \le \ell$. Note that we have the following equation:

$$u' \prod_{i=1}^{n} u_i^{m_i} = g_2^{F(\mathfrak{m})} g^{J(\mathfrak{m})}, \qquad v' \prod_{i=1}^{\ell-1} v_i^{\bar{\mathfrak{m}}_j^i} = g^{G(\bar{\mathfrak{m}}_j)} v^{K(\bar{\mathfrak{m}}_j)} \qquad \text{for } 1 \le j \le k$$

where $\bar{\mathfrak{m}}_j = H_j(\mathfrak{m})$ for $1 \le j \le k$. All public parameters are passed to $\mathcal{A}$. $\mathcal{B}$ also maintains an empty list $\mathcal{L}$.

Oracles Simulation. $\mathcal{B}$ simulates the oracles as follow:

(*Signing oracle.*) Upon receiving query for message $\mathfrak{m}_i = \{m_1, \ldots, m_n\}$, although $\mathcal{B}$ does not know the secret key, it still can construct the signature by assuming $F(\mathfrak{m}_i) \neq 0 \bmod p$ and $K(\bar{\mathfrak{m}}_{i,j}) = 0 \bmod p$, where $\bar{\mathfrak{m}}_{i,j} = H_j(\mathfrak{m}_i)$ for all $1 \le j \le k$. It randomly chooses $r_i \in_R \mathbb{Z}_p$ and computes the signature as

$$S_1 = g_1^{-\frac{J(\mathfrak{m}_i)}{F(\mathfrak{m}_i)}} \big(g_2^{F(\mathfrak{m}_i)} g^{J(\mathfrak{m}_i)}\big)^{r_i}, \quad S_{2,j} = (g_1^{-\frac{1}{F(\mathfrak{m}_i)}} g^{r_i})^{G(\bar{\mathfrak{m}}_{i,j})} \qquad \text{for } 1 \le j \le k$$

Same as the above proof, $(S_1, S_{2,1}, \ldots, S_{2,k})$ is a valid signature. $\mathcal{B}$ puts ($\mathfrak{m}_i$, S_1, $S_{2,1}$, …, $S_{2,k}$) into the list $\mathcal{L}$ and then outputs the signature $(S_1, S_{2,1}, \ldots, S_{2,k})$. To the adversary, all signatures given by $\mathcal{B}$ are indistinguishable from the signatures generated by the signer.

(*Confirmation/Disavowal oracle.*) Upon receiving a signature $(S_1, S_{2,1}, \ldots, S_{2,k})$ for message $\mathfrak{m}$, $\mathcal{B}$ checks whether $(\mathfrak{m}, S_1, S_{2,1}, \ldots, S_{2,k})$ is in $\mathcal{L}$. If so, $\mathcal{B}$ outputs Valid and runs the confirmation protocol with $\mathcal{A}$, to show that (L, M, N_j, O_j) in equation (1) are DH tuples, for $1 \le j \le k$. Notice that since $\mathcal{B}$ knows discrete logarithm of N_j with base L $(= 1/G(\bar{\mathfrak{m}}_{i,j}))$, it can simulate the interactive proof perfectly.

If the signature is not in $\mathcal{L}$, $\mathcal{B}$ outputs Invalid and runs the disavowal protocol with $\mathcal{A}$. By theorem 1, the signature is unforgeable if the CDH assumption holds. $\mathcal{B}$ runs the oracle incorrectly only if $\mathcal{A}$ can forge a signature. However if one can solve the CDH problem, he can also solve the decision linear problem.

(*Receipt generating oracle.*) Upon receive a signature $(S_1, S_{2,1}, \ldots, S_{2,k})$ for message $\mathfrak{m}$, $\mathcal{B}$ computes $\bar{\mathfrak{m}}_j = H_j(\mathfrak{m})$ for $1 \le j \le k$. If $K(\bar{\mathfrak{m}}_j) \neq 0 \bmod p$ for any j, $\mathcal{B}$ aborts. Otherwise $\mathcal{B}$ outputs $S_2' = S_{2,1}^{1/G(\bar{\mathfrak{m}}_1)}$, which is a valid individual receipt for the signature.

Challenge. $\mathcal{A}$ gives $\mathfrak{m}^* = (\mathfrak{m}_1^*, \ldots, \mathfrak{m}_n^*)$ to $\mathcal{B}$ as the challenge message. Denote $\bar{\mathfrak{m}}_j^* = H_j(\mathfrak{m}^*)$ for $1 \le j \le k$. If $F(\mathfrak{m}_i^*) = 0 \bmod p$, $J(\mathfrak{m}_i^*) \neq 0 \bmod p$ or $G(\bar{\mathfrak{m}}_j^*) \neq 0 \bmod p$ for any j, $\mathcal{B}$ aborts.

Otherwise, $\mathcal{B}$ computes:

$$S_1^* = h^c, \qquad S_{2,j}^* = v^{bK(\bar{\mathfrak{m}}_j^*)/F(\mathfrak{m}_i^*)} \qquad \text{for } 1 \le j \le k$$

and returns $(S_1^*, S_{2,1}^*, \ldots, S_{2,k}^*)$ to $\mathcal{A}$.

Output. Finally $\mathcal{A}$ outputs a bit b'. $\mathcal{B}$ returns b' as the solution to the decision linear problem. Notice that if $c = a + b$, then:

$$S_1^* = g_2^{a+b} = g_2^a (g_2^{F(\mathfrak{m}_i^*)})^{b/F(\mathfrak{m}_i^*)} = g_2^a (u' \prod_{i=1}^{n} u_i^{m_i^*})^{b/F(\mathfrak{m}_i^*)},$$

$$S_{2,j}^* = v^{bK(\bar{\mathfrak{m}_j}^*)/F(\mathfrak{m}_i^*)} = (v' \prod_{i=1}^{\ell} v_i^{\bar{\mathfrak{m}_j}^{i}})^{b/F(\mathfrak{m}_i^*)} \qquad \text{for } 1 \leq j \leq k$$

Probability Analysis. For the simulation to complete without aborting, we require the following conditions fulfilled:

1. Sign queries on message $\mathfrak{m}_i$ with $F(\mathfrak{m}_i) \neq 0 \bmod p$ and $K(\bar{\mathfrak{m}}_{i,j}) = 0 \bmod p$ for all j, where $1 \leq j \leq k$.
2. Receipt generating queries on message $\mathfrak{m}_i$ with $K(\bar{\mathfrak{m}}_{i,j}) = 0 \bmod p$, for all j, where $1 \leq j \leq k$.
3. Challenge message $\mathfrak{m}^*$ with $F(\mathfrak{m}^*) \neq 0 \bmod p$, $J(\mathfrak{m}^*) = 0 \bmod p$ and $G(\bar{\mathfrak{m}_j}^*)$ $= 0 \bmod p$ for all j, where $1 \leq j \leq k$.

In order to make the analysis simple, we will bound the probability of a subcase of this event. We define the events A_i, A^*, B_i, C^*, D^* as follow:

$$\begin{aligned}
A_i &: F(\mathfrak{m}_i) \neq 0 \bmod l_p \qquad \text{where } i = 1, \ldots, q_s \\
A^* &: F(\mathfrak{m}^*) \neq 0 \bmod l_p \\
B_i &: K(\bar{\mathfrak{m}}_{i,j}) = 0 \bmod p \qquad \text{where } j = 1, \ldots, k \text{ and } i = 1, \ldots, q_s + q_r \\
C^* &: J(\mathfrak{m}^*) = 0 \bmod l_p \\
D^* &: G(\bar{\mathfrak{m}_j}^*) = 0 \bmod p \qquad \text{where } j = 1, \ldots, k
\end{aligned}$$

The probability of $\mathcal{B}$ not aborting is

$$\Pr[\mathsf{not\ abort}] \geq \Pr\left[\Big(\bigwedge_{i=1}^{q_s} A_i \wedge A^*\Big) \wedge \Big(\bigwedge_{i=1}^{q_s+q_r} B_i\Big) \wedge C^* \wedge D^*\right]$$

Note that the events $\Big(\bigwedge_{i=1}^{q_S} A_i \wedge A^*\Big)$, $\Big(\bigwedge_{i=1}^{q_s+q_r} B_i\Big)$, C^* and D^* are all independent. The assumption $l_p(n+1) < p$ implies if $J(\mathfrak{m}^*) = 0 \bmod p$ then $J(\mathfrak{m}^*) = 0 \bmod l_p$. In addition, it also implies that if $J(\mathfrak{m}^*) = 0 \bmod l_p$, there will be a unique choice of κ with $0 \leq \kappa \leq n$ such that $J(\mathfrak{m}^*) = 0 \bmod p$. Since κ, y' and y_i's are randomly chosen,

$$\begin{aligned}
\Pr[C^*] &= \Pr[J(\mathfrak{m}^*) = 0 \bmod p \wedge J(\mathfrak{m}^*) = 0 \bmod l_p] \\
&= \Pr[J(\mathfrak{m}^*) = 0 \bmod l_p] \Pr[J(\mathfrak{m}^*) = 0 \bmod p \mid J(\mathfrak{m}^*) = 0 \bmod l_p] \\
&= \frac{1}{l_p} \frac{1}{n+1}
\end{aligned}$$

We have:

$$\Pr\Big[\Big(\bigwedge_{i=1}^{q_s} A_i \wedge A^*\Big)\Big] \geq 1 - \frac{q_s+1}{l_p}$$

$$\Pr\Big[\Big(\bigwedge_{i=1}^{q_s+q_r} B_i\Big)\Big] \geq \Big(1 - \frac{s}{\ell}\Big)^{(q_s+q_r)k}$$

$$\Pr[C^*] = \frac{1}{l_p}\frac{1}{n+1}$$

$$\Pr[D^*] = \Big(\frac{s}{\ell}\Big)^k$$

By letting $l_p = 2(q_s+1), s = \frac{\ell}{q_s+q_r}$, we have

$$\Pr[\mathsf{not\ abort}] \geq \frac{1}{4(q_s+1)(n+1)(q_s+q_r)^k} \cdot \Big(1 - \frac{1}{q_s+q_r}\Big)^{(q_s+q_r)k}$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and $O(n)$ modular addition in $\mathbb{Z}_p$ in the signing stage. There are $O(1)$ exponentiations of $\mathbb{G}_T$ element in the confirm/disavow stage. There are $O(1)$ exponentiations of $\mathbb{G}$ element in the receipt generating stage. There are $O(n+\ell)$ modular addition in $\mathbb{Z}_p$ in both the setup stage and the final stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big((q_s+q_r)\rho + q_c\tau + \Big(nq_s + \ell\Big)\omega\Big)$$

□

Theorem 3. *The scheme is (ϵ, t, q_c, q_s)-secure against impersonation if the $(\epsilon'$, $t')$-discrete logarithm assumption holds in $\mathbb{G}$, where*

$$\epsilon' \geq \frac{1}{2}(1 - \frac{q_s}{2p})(\epsilon - \frac{1}{p})^2$$

$$t' = t + O\Big(q_s\rho + q_c\tau + (n+\ell)q_s\omega\Big)$$

where $H_j : \{0,1\}^n \to \mathbb{Z}_\ell^*$, for $1 \leq j \leq k$, are some collision resistant hash functions and ρ, ω are the time for an exponentiation in $\mathbb{G}$ and an addition in $\mathbb{Z}_p$ respectively.

Proof. (Sketch) Assume there is a (ϵ, t, q_c, q_s)-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the discrete logarithm problem with probability at least ϵ' and in time at most t'. $\mathcal{B}$ is given a discrete logarithm problem instance (g, g^a). The remaining proof is very similar to the proof of theorem 1 and also the proof in [18], so we sketch the proof here.

With 1/2 probability, $\mathcal{B}$ sets $g_1 = g^a$ and hence the user secret key is a. The oracle simulation is the same as the proof in theorem 1, except that $\mathcal{B}$ now knows $b = \log_g g_2$. At the end of the game, $\mathcal{A}$ outputs a message-signature pair (m^*, σ^*) and a bit b. For either $b = 0/1$, $\mathcal{B}$ can extract a with probability 1/2, as shown in [18].

With 1/2 probability, $\mathcal{B}$ sets $v' = g^a$ and hence $\mathcal{B}$ knows the signing key α. $\mathcal{B}$ can simulate the oracles perfectly with α. At the end of the game, $\mathcal{A}$ outputs a message-signature pair (m^*, σ^*) and a bit b. For either $b = 0/1$, $\mathcal{B}$ can extract $a + \sum_{i=1}^{\ell} \beta_i \bar{m_1^*}^i$ with probability 1/2, as shown in [18]. Hence $\mathcal{B}$ can find a.

Probability Analysis. For the simulation to complete without aborting, we require the following conditions fulfilled:

1. Sign queries on message $\mathfrak{m}_i$ with $F(\mathfrak{m}_i) \neq 0 \bmod p$, during the case $g_1 = g^a$.
2. $\mathcal{B}$ can correctly extract a at the end of the game.

Similar to the previous proof, the first case appears with probability at least $1 - \frac{q_s}{2p}$. By Reset Lemma, [18] shows that the latter case appears with probability at least $\frac{1}{2}(\epsilon - \frac{1}{p})^2$. We have

$$\epsilon' \geq \frac{1}{2}(1 - \frac{q_s}{2p})(\epsilon - \frac{1}{p})^2$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and $O(n + \ell)$ modular addition in $\mathbb{Z}_p$ in the signing stage. There are $O(1)$ exponentiations of $\mathbb{G}_T$ element and $O(1)$ modular addition in $\mathbb{Z}_p$ in the confirm/disavow stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big(q_s\rho + q_c\tau + (n + \ell)q_s\omega\Big)$$

□

Remarks. The security of our scheme is related to the length of our signature, as shown in the security theorem. For example, the number of $q_s + q_r$ query and the value of k (the number of blocks) cannot be very large, in order to claim an acceptable security. The number of $q_s + q_r$ query allowed maybe set to 128 and the suitable value of k maybe set to be around 7, to gain a balance between efficiency and security.

6 Conclusion

In this paper, we propose the first convertible undeniable signatures without random oracles in pairings. Comparing with the part of undeniable signatures, our scheme is better than the existing undeniable signatures without random oracles [20] by using more standard assumption in the security proofs. Furthermore, our scheme is particularly suitable for applications that do not require a large number of signing queries.

References

1. G. Ateniese, J. Camenisch, S. Hohenberger, and B. de Medeiros. Practical Group Signatures without Random Oracles. Cryptology ePrint Archive, Report 2005/385, 2005. http://eprint.iacr.org/.
2. M. Bellare, A. Boldyreva, and A. Palacio. An Uninstantiable Random-Oracle-Model Scheme for a Hybrid-Encryption Problem. In *Proc. EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 171–188. Springer-Verlag, 2004.
3. M. Bellare and P. Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In *ACM Conference on Computer and Communications Security*, pages 62–73. ACM Press, 1993.
4. A. Bender, J. Katz, and R. Morselli. Ring Signatures: Stronger Definitions, and Constructions without Random Oracles. In *TCC 2006*, volume 3816 of *Lecture Notes in Computer Science*, pages 60–79. Springer, 2006.
5. D. Boneh, X. Boyen, and H. Shacham. Short Group Signatures. In *Proc. CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 41–55. Springer-Verlag, 2004.
6. J. Boyar, D. Chaum, I. Damgård, and T. P. Pedersen. Convertible Undeniable Signatures. In *Proc. CRYPTO 90*, volume 537 of *Lecture Notes in Computer Science*, pages 189–205. Springer-Verlag, 1991.
7. X. Boyen and B. Waters. Anonymous Hierarchical Identity-Based Encryption (Without Random Oracles). In *Proc. CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 290–307. Springer-Verlag, 2006.
8. X. Boyen and B. Waters. Compact Group Signatures Without Random Oracles. In *Proc. EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 427–444. Springer-Verlag, 2006.
9. R. Canetti, O. Goldreich, and S. Halevi. The Random Oracle Methodology, Revisited. In *Proc. 13th ACM Symp. on Theory of Computing*, pages 209–128. ACM Press, 1998.
10. D. Chaum. Designated Confirmer Signatures. In *Proc. EUROCRYPT 94*, volume 950 of *Lecture Notes in Computer Science*, pages 86–91. Springer-Verlag, 1994.
11. D. Chaum and H. van Antwerpen. Undeniable Signatures. In *Proc. CRYPTO 89*, volume 435 of *Lecture Notes in Computer Science*, pages 212–216. Springer-Verlag, 1989.
12. S. S. Chow, J. K. Liu, V. K. Wei, and T. H. Yuen. Ring Signatures without Random Oracles. In *ASIACCS 06*, pages 297–302. ACM Press, 2006.
13. I. Damgård and T. P. Pedersen. New Convertible Undeniable Signature Schemes. In *Proc. EUROCRYPT 96*, volume 1070 of *Lecture Notes in Computer Science*, pages 372–386. Springer-Verlag, 1996.
14. R. Gennaro, T. Rabin, and H. Krawczyk. RSA-Based Undeniable Signatures. In *Journal of Cryptology, 13(4)*, pages 397–416, 2000.
15. S. Hohenberger, G. Rothblum, A. Shelat, and V. Vaikuntanathan. Securely Obfuscating Re-Encryption. In *TCC 2007*, volume 4392 of *Lecture Notes in Computer Science*, pages 233–252. Springer-Verlag, 2007.
16. M. Jakobsson, K. Sako, and R. Impagliazzo. Designated Verifier Proofs and Their Applications. In *Proc. EUROCRYPT 96*, volume 1070 of *Lecture Notes in Computer Science*, pages 143–154. Springer-Verlag, 1996.
17. A. Kiayias and H.-S. Zhou. Concurrent Blind Signatures without Random Oracles. In *SCN 2006*, volume 4116 of *Lecture Notes in Computer Science*, pages 49–62. Springer-Verlag, 2006.

18. K. Kurosawa and S.-H. Heng. 3-Move Undeniable Signature Scheme. In *Proc. EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 181–197. Springer-Verlag, 2005.
19. K. Kurosawa and T. Takagi. New Approach for Selectively Convertible Undeniable Signature Schemes. In *Proc. ASIACRYPT 2006*, volume 4284 of *Lecture Notes in Computer Science*, pages 428–443, 2006.
20. F. Laguillaumie and D. Vergnaud. Short Undeniable Signatures Without Random Oracles: The Missing Link. In *Proc. INDOCRYPT 2005*, volume 3797 of *Lecture Notes in Computer Science*, pages 283–296. Springer-Verlag, 2005.
21. F. Laguillaumie and D. Vergnaud. Time-Selective Convertible Undeniable Signatures. In *Proc. CT-RSA 2005*, volume 3376 of *Lecture Notes in Computer Science*, pages 154–171. Springer, 2005.
22. B. Libert and J.-J. Quisquater. Identity Based Undeniable Signatures. In *Proc. CT-RSA 2004*, volume 2964 of *Lecture Notes in Computer Science*, pages 112–125. Springer-Verlag, 2004.
23. M. Michels, H. Petersen, and P. Horster. Breaking and Repairing a Convertible Undeniable Signature Scheme. In *CCS '96: Proceedings of the 3rd ACM conference on Computer and Communications Security*, pages 148–152. ACM Press, 1996.
24. M. Michels and M. Stadler. Efficient Convertible Undeniable Signature Schemes. In *Proc. SAC'97*, pages 231–244, 1997.
25. T. Okamoto. Designated Confirmer Signatures and Public Key Encryption are Equivalent. In *Proc. CRYPTO 94*, volume 939 of *Lecture Notes in Computer Science*, pages 61–74. Springer-Verlag, 1994.
26. H. Wang, Y. Zhang, and D. Feng. Short Threshold Signature Schemes Without Random Oracles. In *Proc. INDOCRYPT 2005*, volume 3797 of *Lecture Notes in Computer Science*, pages 297–310. Springer-Verlag, 2005.
27. B. Waters. Efficient Identity-Based Encryption Without Random Oracles. In *Proc. EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer-Verlag, 2005.
28. R. Zhang, J. Furukawa, and H. Imai. Short Signature and Universal Designated Verifier Signature Without Random Oracles. In *Proc. ACNS 2005*, volume 3531 of *Lecture Notes in Computer Science*, pages 483–498. Springer-Verlag, 2005.