EUROPEAN JOURNAL OF PURE AND APPLIED MATHEMATICS
2025, Vol. 18, Issue 2, Article Number 6109
ISSN 1307-5543 – ejpam.com
Published by New York Business Global

Classification and Taxonomy of Undeniable Signatures: From Early Designs to Post-Quantum Approaches

Hussah Mohammad Almenderj[1,*], Eddie Shahril Ismail[1]

[1] *Department of Mathematical Sciences, Faculty of Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia*

Abstract. Undeniable signatures are a special kind of digital signature that need the active cooperation of the signer in the verification process, granting properties such as completeness, soundness, unforgeability, and invisibility. Moreover, these schemes techniques allow controlled verification in contrast to traditional signatures, meaning that they are perfect in situations when secrecy and selective disclosure are important. Over the years, undeniable signatures have undergone substantial evolution from the transition from classical encryption such as discrete logarithms to modern algorithms that include lattice-based, hash-based, and code-based post-quantum. In this article, focused on classification and taxonomy of undeniable signature schemes through evolution from early concepts to solutions that are resistant to quantum alterations. Unlike previous surveys, our taxonomy highlights distinctions between classical, convertible, and designated confirmer variants, while also introducing a post-quantum perspective rooted in lattice-based and code-based constructions. We analytically classify key properties for undeniable signature schemes whence inspect their strengths and weaknesses, also offer a methodical framework for comprehending the differences between them. Our contributions comprise a comprehensive taxonomy of undeniable signatures, considering various aspects such as security, efficiency, and practical applicability, as well as insights into potential areas future research paths, especially in terms of post- quantum systems. By displaying our views in this regard, our goal is to provide a more profound comprehension and guide ongoing and effective research in undeniable signature schemes.

2020 Mathematics Subject Classifications: 94A60, 68P25, 68Q12

Key Words and Phrases: Undeniable signatures, Post-quantum cryptography, Zero-knowledge proofs, Signature scheme, The signer, The verifier, public key, message

1. Introduction

Unquestionably, undeniable signatures play a critical part in cryptography by making a distinctive system of control on the verification of signatures. These signatures demand the signer to participate in the verification process unlike traditional digital signatures which can only be verified by someone with the public key. This characteristic makes

[*] Corresponding author.

DOI: https://doi.org/10.29020/nybg.ejpam.v18i2.6109

Email addresses: hissah6200@gmail.com (H. Almenderj), esbi@ukm.edu.my (E. Ismail)

Copyright: © 2025 The Author(s). (CC BY-NC 4.0)

them particularly useful in situations needing sensitive or specialized information, such as secure communications, legal contracts, and digital rights management. The ability to decline or confirm a signature without disclosure its content is perfectly compatible with cryptography that protects privacy.

The idea of undeniable signatures was first created by Chaum and van Antwerpen [1] in 1989 to provide a new way for digital signatures that require the verification step as an interactive process. This was quite a change from more traditional signatures like RSA or DSA that allow anybody who has the public key verify them by themselves. The initial schemes were built around number-theoretic assumptions, especially the discrete logarithm problem (DLP) and factorization which were thought to be secure in practice. These constructions focused on achieving core properties such as undeniability, invisibility, and non-transferability ensuring that they could only be verified with the consent from the signer without having to reveal its signature to third party. As cryptography advanced, researchers began to address the efficiency limitations of early undeniable signature schemes. Protocols with lower communication overhead and faster interactive verification were developed to make them more practical for applications. In addition, this period saw the introduction of zero knowledge proof techniques to enhance security without compromising usability. However, the emergence of quantum computing posed a risk threat to these classical schemes. Many traditional public-key cryptosystems security is compromised due to Shor's algorithm [2] problem and factoring large numbers efficiently. This development spurred a paradigm shift towards post-quantum cryptography, a field of cryptography considered on developing algorithms that remain secure even in the presence of quantum computers. In response, research has expanded into constructing quantum resistant undeniable signature schemes. Several approaches have emerged, counting lattice-based, hash-based, and code-based constructions, each leveraging hard mathematical problems thought to be impervious to both classical and quantum attacks. These post-quantum schemes aim to preserve the core principles of undeniability and invisibility while tackling the difficulties presented by quantum adversaries. As the post-quantum era approaches, the design of undeniable signatures must continue to adapt, balancing efficiency, practicality, and long-term security to meet emerging challenges.

Despite the considerable progress in the field, there is a lack of a unified taxonomy and classification of undeniable signature schemes that encompass both classical and post-quantum designs. So, we need the body of existing literature for providing a comprehensive framework for understanding their relationships, strengths, and weaknesses to evaluate the state of the latest technology systematically and to identify research gaps. A structured classification is essential to bridge this gap and guide the development of future schemes. This paper, we present a structured taxonomy that categorizer undeniable signature schemes into classical and post-quantum frameworks, identifying commonalities and distinctions between them. Also, we provide a detailed comparative analysis of various schemes, considering factors such as security assumptions, efficiency, communication complexity, and resistance to quantum attacks. Furthermore, we identify open challenges and suggest future directions for developing more efficient and secure undeniable signature schemes, particularly in the context of post-quantum cryptography.

2. Related work

As earlier noted, undeniable signature has been discovered by Chaum and van Antwerpen [1]. Afterwards, Chaum [3] himself put up an updated version of the disavowal and confirmation procedure that complies with the zero-knowledge property, this prevents any information from reaching the attacker.However, Fujioka et al. [4] putted the interactive bi-proof system with minimal knowledge and addressing limitations in [3]. Thereafter, convertible undeniable signatures were designed by Chaum et al. [5] , which are dependent on ElGamal's distinctive scheme. These are undeniable signatures but with a few extra touches where the signer changes their signature into a conventional digital signature that can be validated automatically without interaction. Following this Chaum et al. [6] also familiarized the first schemes for undeniable signature that ensure unconditional security for signers. Additionally, Chaum [7] developed traditional digital signatures by require a confided outside party to confirm or deny validity of a signatures. This addresses a limitation of undeniable signatures by enabling non-transferable proofs.In 1994, Okamoto [8] established the definition of designated confirmer signatures and shows that is similar to public key encryption in terms of respect to existence, which is better than the original signature scheme in [6]. However, the signature scheme becomes susceptible to forgery as a result of the conversion following information release. Indeed, Jakobsson [9] highlighted weaknesses in traditional undeniable signature protocol that allow coercion and contrasts these with designated confirmer signatures. Subsequently, Jakobsson et al. [10] presented that ensures only a specified verifier can validate a proof, preventing any third party from verifying it even if the verifier attempts to share it, which selective disclosure enhances privacy and non-transferability. Then, Michels et al. [11] proposed a solution for scheme [5] , but the evidence is inconclusive. In 1996, Damgård and Pedersen [12] presented two new convertible undeniable signature schemes.A modified ElGamal signature scheme is the basis for a convertible signature scheme which is created by Yun and Kim [13]. Stadler and Michels [14] proposed a different scheme of convertible undeniable signature and present evidence of all security properties.This scheme is using discrete logarithm and based on the Schnorr signature scheme [15] . The first RSA-based scheme for undeniable signatures since their 1989 debut by Gennaro et al. [16]. In [17] presented novel ways to design confirmer signatures, a cryptographic tool that combines elements of undeniable and standard digital signatures. By Pointcheval and Okamoto [18] presented a new type of computational problem known as gap problems, which can resolve long-standing security challenges, including an open problem related to Chaum's undeniable signature scheme. An undeniable signature scheme was proposed that works for fully general RSA setting by Galbraith et al. [19]. Then, Galbraith and Mao [20] explicitly defined anonymity for confirmer and undeniable signatures in a multi-user environment, demonstrating that anonymity is intimately connected to invisibility. Also, Goldwasser and Waisbard [21] modified the signature in [8] into designated confirmer signature without requiring random oracle models. Biehl et al. [22] provides a novel system for undeniable signature using the quadrature process, which uses RSA as its basis. In [23] , they provide the first case of an undeniable signature identity using an elliptic curves structure. Monnerat and

Vaudenay [24] introduced a novel approach that generalizes cryptographic challenges by leveraging group homomorphisms. Also, Monnerat and Vaudenay [25] introduced a creative approach to undeniable signatures which the scheme leverages algebraic characters to construct signatures. Ogata et al. [26] explored the vulnerabilities and security dimensions of a Full-Domain adaptation of scheme [7]. Based on bilinear map, Laguillaumie and Vergnaud [27] introduced a novel effective undeniable signature system that is convertible. Also, they introduced a novel undeniable signature scheme that avoids the use of random oracles in [28] , which anonymity of this system stems from a non-standard decisional assumption, while the unforgeability is closely linked to the computational Diffie-Hellman problem. A 3-move verification protocol was created by Heng and Kurosawa [29] , although this protocol satisfies unforgeability and invisibility, it is not transferability. Next, Kurosawa and Takagi [30] introduced a new method that allows a signature to be converted from undeniable to a publicly verifiable from selectively, ensuring enhanced flexibility. Certificate-free undeniable signature scheme was first proposed by Duan [31]. Then, Schuldt and Matsuura [32] focused token soundness on creating a signature scheme that balances flexibility, security, and efficiency. In 2013, Huang and Wang [33] offered a high-quality rebuild for convertible undeniable signature proven without a random oracle. Then, introduced two designated verifier signature schemes that incorporate undeniable properties by Hu et al. [34] . Han et al. [35] improved scheme [10] where the suggested plan is not transferable or delegable. Finally, Loh et al. [36] demonstrated that there are cases where a signature scheme can maintain invisibility without guaranteeing anonymity, especially when it has different signature spaces.

In contrast, there is limited researches on undeniable signatures in post-quantum cryptography. This is because the post-quantum leap came after the classic signatures, so undeniable signatures are relatively complex compared to conventional signatures. Additionally, transitioning classic schemes to post-quantum settings introduces new challenges such as the need for efficient algorithms that can withstand quantum attacks while remaining practical on current hardware. However, the rise of quantum computing needs to explore quantum-resistant alternatives like lattice or code.

In 2012, Li and Wang [37] presented the first working lattice-based undeniable signature system, relying on the random oracle approach for its security. In [38] gave schemes are built on the assumption code-based problem which aim to achieve resistant to quantum attacks and ensures unforgeability. Aguilar-Melchor et al. [39] proposed a new undeniable signature scheme leveraging error-correcting codes for post-quantum security. In [40] introduced a code-based cryptographic approach which leverages quasi-dyadic codes. Jalali et al. [41] represented the inaugural practical implementation of an undeniable signature scheme that is resistant to quantum attacks, utilizing isogeny-based cryptography. Rawal et al. [42] suggested a cryptographic undeniable signature scheme where the security on the hardness of lattice problem. Finally, the first undeniable signature utilizing module lattices was introduced by Dey et al. [43].

3. Background and Preliminaries

3.1. Definition the Undeniable Signature

Four polynomial-time algorithms make up an undeniable signature system as described below.

(i) **KeyGen** (λ)**:** The algorithm uses the security parameter (λ) as an input. Then outputs the pair (S, V) as signature key and validation key respectively.

(ii) **SignGen:** The signature algorithm input the message with the signature key S and outputs the signature σ.

(iii) **Confirmation:** The message and the signature σ are inputs to the confirmation protocol which is an interactive process between the signer and the verifier. Then, its outputs agreeable if the signature σ is valid; if not, they are rejected.

(iv) **Disavowal:** In the same way, disavowal protocol takes as input the message and the signature σ which this protocol too is an interactive process between the signer and the verifier. Similarly, its outputs agreeable if the signature σ is valid; if not, they are rejected.

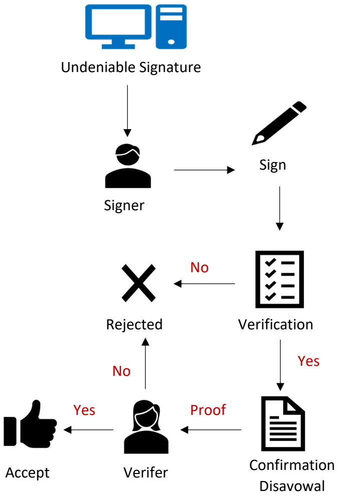

Figure 1: Undeniable Signature.

3.2. Difference Between the Confirmation Protocol and the Disavowal Protocol

The confirmation protocol and the disavowal protocol are two critical components in undeniable signature system, used to verify the validity or invalidity of a given signature while preserving certain security and privacy properties. Their zero-knowledge nature ensures that sensitive information remains secure even during interactive verification processes. By combining these tow protocols, undeniable signature schemes achieve the dual goals of selective verification and robust privacy. Here's how they differ in Table 1:

Table 1: Comparative between the confirmation protocol and the disavowal protocol.

Aspect	Confirmation Protocol	Disavowal Protocol
Purpose	Prove the validity of a signature	Prove the invalidity of a signature
Interaction	Signer confirms validity	Signer disproves validity
Verifier's Goal	Be convinced signature is valid	Be convinced signature is invalid
Non-Transferability	Verifier cannot convince third party	Verifier cannot convince third party
Zero-Knowledge	No information about signature or key revealed	No information about signature or key revealed

3.3. Security Properties of Undeniable Signature

Undeniable signature possesses several key security properties that distinguish from traditional digital signature. These properties ensure the veracity, and privacy of the signed messages. Here, we discuss the main security properties: unforgeability, undeniability, invisibility, and non-transferability.

(i) **Unforgeability:**

Unforgeability is the ability of the signature scheme to prevent unauthorized creation of valid signatures. Meaning that only the real signer can create a valid signature for a given message. This property guarantees that even if an attacker intercepts multiple message-signature pairs, they cannot forge a novel, valid signature on another message.

(ii) **Undeniability:**

Undeniability ensures that the signer cannot deny having signed a message. In undeniable signature schemes, the signer must cooperate in the verification process, making it impossible for them to repudiate their signature.

(iii) **Invisibility:**

Invisibility refers to the property that prevents unauthorized parties from verifying the validity of a signature without the signer's cooperation. This ensures that the signature remains private and invisible to anyone who does not possess the verification protocol.

(iv) **Non-transferability:**

Non-transferability makes sure that the validity of a signature cannot be transferred from one message to another. This property prevents attackers from reusing a valid signature on a different message, maintaining the integrity of the signed message.

4. Design Strategies for Undeniable Signature Schemes

Undeniable signature schemes represent a unique class of digital signatures where it is impossible to publicly confirm the legitimacy of a signature without the signer's participation. These schemes aim to achieve properties such as invisibility (preventing unauthorized verification) and non-transferability (ensuring that even verified signatures cannot be shared as evidence without the signer's consent). To achieve these goals, various design strategies have been employed over the years, combining cryptographic primitives and mathematical tools to ensure both security and efficiency. The key design strategies for undeniable signature schemes include zero-knowledge proofs, trapdoor functions, and homomorphic commitments.

4.1. Zero-Knowledge Proofs

Zero-knowledge proofs [3] are widely used to make sure that no information, apart from the correctness of the signature verification or disavowal, is leaked during the process. Zero-knowledge proofs permit the signer to interactively evidence the validity of the signature without revealing any additional information that could compromise privacy or security. Chaum's [3] early undeniable signature schemes incorporated zero-knowledge proofs to handle both verification and disavowal. Jakobsson [10] explicitly leverage zero-knowledge proofs to achieve verification without revealing additional information.

In the context of undeniable signatures is verification. The signer must supply a proof that the signature is valid without revealing private information, such as the signing key. This is achieved through interactive or non-interactive zero-knowledge proofs, ensuring that the verifier gains confidence in the validity of the signature but learns nothing beyond that. As well as disavowal, if a signature is claimed to be forged, the signer provides a zero-knowledge proof that the signature is invalid. This ensures that even during the disavowal process, no critical information about the signing key or internal structure of the signature scheme is revealed.

Advantages of using zero-knowledge proofs is privacy preservation. Zero-knowledge proofs ensure that private data such as the signing key or message is not disclosed. Also, advantages of using zero-knowledge proofs is security. Zero-knowledge proofs prevent potential attackers from deriving any usable information from the verification/disavowal processes. Furthermore, it has efficiency, such as Sigma protocols or Schorr-like proofs, which to balance computational efficiency and communication complexity.

4.2. Trapdoor Functions

Trapdoor functions play a significant role in undeniable signature schemes by providing computational one direction, which allows efficient signature generation while maintaining the infeasibility of forgery. A trapdoor functions is straightforward to compute in only one direction, but it is hard to reverse them without knowing the precise hidden information (the trapdoor). Early undeniable signature schemes relied on RSA-like trapdoor functions, where signature generation requires knowledge of the private from factoring large numbers like [16, 19]. An additional, post-quantum undeniable signature schemes use lattice-based trapdoor functions due to their quantum resistance like [37, 40]. Also, certain schemes leverage coding theory and the hardness of decoding random linear codes to construct trapdoor functions [39, 40].

Benefits of trapdoor functions is efficiency, where signature generation can be optimized for quick computations, especially with lattice or code-based schemes. As well as, forgery remains infeasible, assuming the trapdoor function's underlying problem is computationally hard. Also, it has scalability, where trapdoor-based designs can often scale to larger key sizes while maintaining security levels.

4.3. Homomorphic Commitments

Homomorphic commitment [24] are cryptographic tools that enable a signer to obligate to a value without disclosing it. Once committed, the signer can later prove certain properties about the value (like validity) without exposing the underlying data. This property is particularly important for invisibility and non-transferability in undeniable signature schemes. Monnerat and Vaudenay [24] demonstrated generic homomorphic undeniable signature schemes, emphasizing their utility in ensuring invisibility.

In invisibility, the signature remains invisible unless the signer explicitly allows it to be verified. Homomorphic commitments ensure that the signature cannot be recognized or transferred without the signer's cooperation, as the committed value is indistinguishable from randomness. In non-transferability, the homomorphic commitment ensures that verification of the signature requires interaction with the signer. Even if a verifier interacts with the signer to confirm validity, they unable to transfer the proof of verification to different party without the signer's involvement. This property protects the signature from unauthorized distribution. Homomorphic operations allow the commitment scheme computations on the committed values without revealing them. This supports more flexible verification protocols.

Benefits of homomorphic commitments is privacy, which they preserve the signer's control over signature visibility. As well as, it has efficiency, where commitments can be combined with zero-knowledge proofs to create lightweight verification protocols. Also, it is flexibility, where homomorphic commitments support advanced cryptographic designs, such as multi-party verification and non-interactive proofs.

The design strategies for undeniable signature schemes zero-knowledge proofs, trapdoor functions, and homomorphic commitments collectively address the unique requirements of undeniability, invisibility, and non-transferability. By leveraging these cryptographic primitives, researchers have built schemes that ensure strong security and practical efficiency. However, some signatures schemes include several strategies such as: Boyar et al. [5] supports zero-knowledge proofs for convertibility properties but used trapdoor for signer control. Gennaro et al. [16] incorporates zero-knowledge based protocols but used RSA-based trapdoors for verification and disavowal. These strategies have evolved through significant contributions from foundational works, so their integration remains essential for advancing undeniable signatures, especially in post-quantum and modern cryptographic contexts.

5. Classification and Taxonomy Framework

Comprehensively understand the development and diversity of undeniable signature schemes, we introduce a structured framework for their classification and taxonomy captures the evolution from classical designs to modern post-quantum proposals, using multiple dimensions to differentiate between the various schemes and their properties. To classify undeniable signatures effectively, we will use the following criteria in Figure 2.

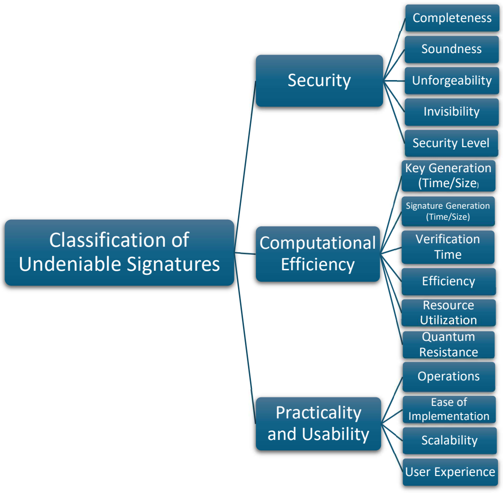

Figure 2: classification of undeniable signature.

5.1. First Undeniable Signature

Chaum and Van Antwerpen's [1] introduced the concept of undeniable signatures, laying the groundwork for this field of research which it the origin of the idea. The schemes [1, 3] relies on the difficulty of solving the discrete logarithm problem (DLP), which ensures that derivate key from the public key is mathematically infeasible. Both verification and disavowal protocols are implemented as interactive zero-knowledge proofs, ensuring that no extra information about the signature is revealed during the process.

(i) **Setup and Key Generation**

Two large prime numbers p and q are chosen such that $p = 2q + 1$. This choice of primes is typical for cryptographic security based on the hardness of certain problems in finite fields. A generator g of the multiplicative group $\mathbb{Z}_p^*$ is chosen such that it has order q. The signer selects a private key x, and the public key g^x used to verify the authenticity of signatures.

(ii) **Signature Generation**

The signer wants to sign a message m. Typically, m might first be hashed to ensure a fixed-length input. The signature s is computed in Chaum's scheme, the actual form of the signature may involve computations similar to standard digital signatures, such as: $s = m^x$. This makes the signature dependent on the message m and the signer's private key x. The choice of s here is critical because s alone cannot be verified without the signer's participation in an interactive protocol.

(iii) **Verification Protocol**

In the undeniable signature phase, an interactive zero-knowledge protocol allows the signer to convince a verifier that s is a valid signature on m without revealing the private key x. The verifier uses the confirmation protocol to check the validity of the signature. This protocol consists of several steps: the signer randomly chooses $r \in \{1, \ldots, q-1\}$ and the verifier computes $m^a g^b$ and sends it to the signer. Then, the signer computes a commitment: $m^a g^{b+q}$, $\left(m^a \cdot g^{b+q}\right)^x$, and it sends to the verifier. Following the verifier reveals a, b to the signer. After verifying the correctness of a and b by computing $m^a g^b$ and comparing it with the message the verifier sent earlier, if they match, then the signer discloses r to the verifier. the verifier reconstructs the message transmitted by the signer by substituting m^x with s and verifies whether s is a valid signature. Lastly, the verifier will verify if the reconstructed message matches the original message sent by the signer. If the checks pass, the verifier is convinced that s is a valid signature of m under the public key y.

(iv) **Disavowal Protocol**

The signature s is not a valid signature for m without revealing the private key x by proving that $s \neq m^x$, so the disavowal protocol is used by both the signer and the verifier. Both parties have agreed upon a secret number k. The signer chooses a random r and the verifier chooses a from group elements as well as an integer c between 0 and k. The verifier computes a commitment $m^c g^a$ and $s^c g^x a$ and sends to the signer, who uses experimental and error to determine c by dividing $(m^c g^a)^x$ by $s^c g^x a$, then and send $blob(r, c)$ to the verifier. If the verifier sees that $s^c g^x a = (m^c g^a)^x$, then the signature is considered legitimate; otherwise, it is deemed forged.

5.2. Convertible Undeniable Signature

Boyar et al. [5] introduced a fascinating cryptographic concept for convertible undeniable signature. These signatures enable converted from undeniable signatures to normal digital signatures by liberating a single bit string. So, provides a flexible signing scheme that can be converted into standard verifiable signatures, offering both undeniable and ordinary signature functionalities as needed.

(i) **Key Generation**

Choose two prime numbers q and p such that q divides $p-1$. Let g be a generator of the group $\mathbb{Z}_p^*$ with an order of q. The signer selects two private integers x and s such that $x, s \in \mathbb{Z}_q^*$, which is secret parameters. To public key computation $y = g^x$ mod p and $u = g^s \mod p$. Only the signer knows (y, u) is the public key and (x, s) is secret key.

(ii) **Signature Generation**

For signing a message m, choose two random values t and k in $\mathbb{Z}_q^*$. Compute signature components: $T = g^t \mod p, r = g^k \mod p$ and $z = k^{-1}(m - rx)(\mod q)$, so the signature on message m is the tuple (T, r, z). This signature generation follows a structure similar to ElGamal [44] and is constructed to secure non-repudiation and undeniability.

(iii) **Verification Protocol**

The values $w = T^T m$ and $v = y^r r^z$ are calculated by the verifier utilizing public information. The validity of the signature is determined by $w^s = v$. Using a zero-knowledge protocol, the signer demonstrates that $\log_w v = \log_g u$. Since the equality relies on discrete logarithms, proving $w^s = v$ ensures that only the signer, who possesses the private key x, could have generated the signature.

(iv) **Disavowal Protocol**

If $w^s \neq v$, the signer could make use a disavowal protocol to prove the signature's invalidity. With zero-knowledge, the signer shows that $\log_w v \neq \log_g u$. This step ensures that a false or tampered signature can be securely disavowed without revealing the private key x.

(v) **Selective Conversion and Total Conversion**

The convertible aspect of this scheme permits the signer to convert an undeniable signature into an ordinary digital signature by disclosing specific information. To convert a signature (T, r, z) into a standard digital signature, the signer can reveal the value of t. The verifier then confirms the signature using $u^t T m \equiv y^r r^z \mod p, t \equiv g^t$ mod p. By releasing the private key s, all signatures created by the signer can be independently verified. For any signature (T, r, z), the verifier checks if $w^z = v$.

5.3. RSA-based Undeniable Signature

Now, we explore into the mathematical details of the RSA-based undeniable signature scheme presented by Gennaro et al. [16] in their 1997 paper.

(i) **Key Generation**

The scheme begins with the setup of parameters, where we selection of two big prime numbers p and q such that p less than q. Both p and q are of the form $p = 2p' + 1$ and $q = 2q' + 1$, where p', q' are also prime numbers. Compute $n = pq$, where

$\varphi(n) = (p-1)(q-1)$. Choose integers e and d such that $ed \equiv 1(\mod \varphi(n))$. Select $w \in Z_n^*$ such that $w \neq 1$ and compute $S_w = w^d \mod n$. The public key is (n, w, S_w) and the private key is (e, d).

(ii) **Signature Generation**

The sign a message m is hashed to a digest $h(m)$. Calculate $S_m = \tilde{m}^d \mod n$. The signature on m is the pair (m, S_m).

(iii) **Confirmation Protocol**

The prover to try to convince the verifier that S_m is a valid signature of m. The verifier selects two random indices i, j from the set $[n]$ and computes $Q = S_m^i S_w^j \mod n$. The verifier sends Q to the prover, then the prover computes $A = Q^e \mod n$. After that, the prover sends A back to the verifier. The verifier checks whether $A = \tilde{m}^i w^j \mod n$. If the equality satisfied, the verifier acknowledges S_m as a valid signature.

(iv) **Denial Protocol**

The verifier randomly chooses b from the range $[k]$, where k is a security parameter chosen to ensure a low probability of error. This determines the index i as $i = 4b$. Let j from the range $[n]$, which introduces another element of randomness. The verifier computes two values $Q_1 = \tilde{m}w^i \mod n, Q_2 = S_m w^j \mod n$, then the verifier sends the tuple (Q_1, Q_2) to the prover. The prover receives it and needs to determine i. To do this, the prover calculates $\frac{Q_1}{Q_2} = w^i$, then tests all possible values of i within the range $[k]$ to identify the one that meets the equation's criteria. If a valid i is found, the prover sets $A = i$, otherwise, the prover terminates. The prover sends A to the verifier. Then, the verifier checks if $A = i$. If the equality satisfied, it means that S_m is not a valid signature, and the verifier acknowledges the denial. If the equality does not hold, the verifier cannot conclude anything about the validity of S_m.

5.4. Code-based Undeniable Signature

A code-based undeniable signature by Aguilar-Melchor et al. [39] is a type of digital signature that leverages coding theory to ensure security. This scheme is particularly interesting because it is resistant to quantum attacks, making it a post-quantum cryptographic solution.

(i) **Key Generation**

The matrix H is generated using a hash function h. The hash function yields pseudo-random values in the field $\mathbb{F}_2^{(n-k)\times n}$, ensuring randomness and eliminating the possibility of maliciously embedding trapdoors. Instead of directly creating H, the signer generates a random seed sd from $\mathbb{F}_2^n$. The matrix H is computed as $H = h(sd)$ where h is a hash function in cryptography that maps the seed to a matrix in $\mathbb{F}_2^{(n-k)\times n}$. By sharing the seed sd, the signer can reproducibly share H with the verifier, maintaining transparency. Run the generic key generation algorithm Gen to obtain a valid key pair (KS_s, KS_p). Here, KS_s is the signing key, and KS_p is the public key. Choose a random value r from the finite field $\mathbb{F}_2^t$. This adds an additional layer of randomness to the key pair. The final key pair consists of the signing key KS_s and the random value r for the private key, and the public key KS_p and the seed sd. So, the output key pair is $(KP_s, KP_p) = (KS_s, r), (KS_p, sd)$.

(ii) **Signature Generation**

The signer first computes an extra key pair (K_s, K_p) with random α and sd using the following steps: he computes $H = h(sd)$ is derived from a hash function h applied to the sd. Then, generate a random word e with Hamming weight w is chosen uniformly from the set S_w^n and compute $s = He^T$. Finally, output the pair $(K_s, K_p) = (e, s)$. Using the extra key pair (K_s, K_p), the signer generates a signature for a message m through the following steps: choose a random value α from $\mathbb{F}_2^t$. Generate the extra key pair $(K_s, K_p) = (e, s)$ with α and sd. Compute $M = h'(m)$ using a hash function h to map the message m into a fixed-length vector in $\mathbb{F}_2^{n\times t}$. Calculate $z = Me^T, x = (s, \alpha \oplus r, sd)$, and $v = Sign(x, KS_s)$, where the result of signing x. Output the signature $\alpha = (x, y, z)$.

(iii) **Verification Protocol**

The key extractor is used by the signer to regenerate the extra key (K_s, K_p) from the signature $\sigma = (x, y, z)$. Verify (x, y) using the public key KS_p. Decompose x into x_1, x_2, x_3. Use extra key generator $(x_2 \oplus r, x_3)$ to compute $K_s = e$ and $k_p = s$ using the random value $x_2 \oplus r$ and the seed $x_3 = sd$. Output the key pair $(K_s, K_p, z) = (e, s, z)$.

(iv) **Confirmation and Disavowal Protocol**

The prover possesses a key pair $(K_s, K_p) = (e, s)$. Both parties compute the public values $H = h(sd)$ and $M = h(m)$ using hash function. The prover chooses a random permutation π over $\{1, \ldots, n\}$ and a random vector $u \in \mathbb{F}_2^n$. The prover sends the following commitments to the verifier: $C_1 = h(\pi \parallel Hu^T), C_2 = h(\pi(u)), C_3 = h(\pi(u+e))$, and $C_4 = h(\pi \parallel Mu^T)$. The verifier sends random challenge $c \in \{0, 1, 2\}$ to prover. Depending on the challenge c, the prover responds as follows:

- $c = 0$, the prover sends π, u,
- $c = 1$, the prover sends $\pi, u + e$,
- $c = 2$, the prover sends $\pi(u), \pi(e)$.

The verifier examines the commitments based on c:

- $c = 0$, verify that C_1, C_2, and C_4 are correctly calculated.
- If $c = 1$, confirmation protocol, verify that C_1 and C_3 are correctly considered and check the validity of $C_4 = com(\pi \parallel M(e + u)^T)$.

  If $c = 1$, disavowal protocol, same checks as above, but ensure $C_4 \neq com(\pi \parallel M(e + u)^T)$.

- $c = 2$, verify that C_2 and C_3 are correctly calculated and check that $wt(\pi(e)) = \omega$.

The verifier accepts the signature if all checks are passed. Otherwise, the verifier rejects.

5.5. Lattice based Undeniable Signature

By Swati Rawal, Sahadeo Padhye, and Debiao He [42] proposes a novel undeniable signature scheme based on lattice cryptography. This scheme offers a secure and efficient alternative to classical schemes with strong resistance to quantum attacks.

(i) **Key Generation**

Run TrapGen, where this function generates the matrix A and the trapdoor T. The matrix A is utilized as the public key, whilst T is kept secret. Extra Key Generation step is performed when a signer is requested to sign a new message by a hash function h is applied to the seed sd to generate the matrix $H \in \mathbb{Z}_q^{(n\times m)}$. Also, choose e from a discrete Gaussian distribution serves as the secret component in this extra key. As well as, compute $S = He^t \mod q$, where becomes part of the public key for the extra key. Finally, outputs public key (A, sd, S) and private key (T, e).

(ii) **Signature Generation**

We start with run Extra Key Generation to generate sd and e. Then, compute $M = h(masseg)$, where h is a hash function. As well as, choose a random value $r \in \{0, 1\}^n$. Compute signature components as: $s_3 = Me^t \mod q, s_1 = h(Sr)$, and $s_2 \longleftarrow SampPre(A, T, \beta, s_1)$, where SampPre is preimage sampling function. It generates a random preimage s_2 such that $As_2 \equiv s_1 \mod q$ and is distributed according to a discrete Gaussian distribution centred around the lattice point defined by A. The final signature is $\sigma = (s_1, s_2, s_3)$.

(iii) **Verification Protocol**

The verification first verifies the validity of the pair (s_1, s_2). Firstly, checks if $s_2 \leq \beta m$. This step ensures that the vector s_2 is within the acceptable range defined by the parameter β and the dimension m. Secondly, verify if $As_2 = s_1 \mod q$. This step checks if the matrix A multiplied by the vector s_2 is congruent to $s_1 \mod q$. If both conditions are satisfied, the signature is considered valid. If either condition fails, the signature is invalid.

(iv) **Confirmation/Disavowal Protocol**

Goal of the protocol is the signer proves to the verifier that the signature s_3 is valid/invalid for the given message. The signer and verifier independently compute $H = h(sd)$ and $M = h(masseg)$. The signer chooses a permutation π over $\{1, 2, \ldots, m\}$ and a random vector $u \in \mathbb{Z}_q^m$. Also, the signer computes the following commitments $C_1 = h(\pi \parallel Hu^t \mod q), C_2 = h(\pi(u)), C_3 = h(\pi(u + e))$, and

$C_4 = h(\pi \parallel Mu^t \mod q)$. These commitments are sent to the verifier. The verifier sends a random challenge $c \in \{0, 1, 2\}$ to the signer. Depending on the challenge c, the signer responds as follows: If $c = 0$, the signer sends π, u. If $c = 1$, the signer sends $\pi, u + e$. If $c = 2$, the signer sends $\pi(u), \pi(e)$. Based on the response, the verifier checks the validity of the commitments: If $c = 0$, verify C_1, C_2, and C_4 were honestly computed. If $c = 1$, verify C_1, C_2,and C_4 were honestly computed. Specifically, check $C_4 = h(\pi \parallel M(u + e)^t - s_3 \mod q)$. If this holds, it serves as a confirmation; otherwise, it is a disavowal. If $c = 2$, verify C_2, C_3, and check the norm $\parallel \pi(e) \parallel \leq \beta\sqrt{m}$. If all the verifies pass, the verifier acknowledges the signature σ as valid. Otherwise, the verifier rejects it.

6. Comparing the Provided Papers

We provide tables highlights how cryptographic advancements, from early designs to post-quantum schemes, progressively address safely requirements. These schemes are compared across multiple dimensions, providing insights into their relative strengths and limitations. Additionally, the comparison tables demonstrate the trade-offs between efficiency, security, and flexibility in undeniable signature schemes. These comparisons underline the continued innovation in designing robust and efficient undeniable signature schemes.

The comparative analysis of the listed papers (see Table 2) reveals the evolution of undeniable signature schemes concerning core safety properties, including undeniability, non-transferability, and zero-knowledge security.

All schemes ensure valid signatures are accepted, which achieve completeness, but [42] and [39] leverage advanced techniques for tighter guarantees in their respective cryptographic frameworks. We note papers [1, 3], [5], and [16] are robust in terms of property soundness that due to advancements in cryptographic protocol, while [42] benefits from lattice-based assumptions offering strong security. Lattice [42] and code-based [39] schemes are the strongest in unforgeability due to their reliance on problems considered challenging even for quantum computers. The schemes [3] and [42] excel in invisibility, with [42] being particularly suitable for post-quantum scenarios, ensuring invisibility even under advanced cryptographic analysis.

The comparative analysis of the listed papers in terms of computational efficiency (see Table 3) highlights the trade-offs between security and performance in undeniable signature schemes.

The scheme [16] is the faster of key generation time due to the efficiency of RSA key generation and have small secret keys. Also, it is relatively faster of signature generation. On the other hand, [42] and [39] require significantly more time and larger keys due to the complexity of lattice-based and code-based construction. As well as, lattice-based [42] and zero-knowledge enhanced [1, 3] and [5] schemes are slower and have larger signature sizes due to their cryptographic primitives. The scheme [16] is efficient, while [42] is the most resource-intensive due to modular arithmetic and lattice structures, followed by classical interactive schemes [1, 3]. Classical schemes like [1, 3] and [16] are resource-efficient compared to lattice-based [42] and cod-based [39] approaches, which require substantial

Table 2: Comparative analysis for the listed papers in terms of safety properties.

The signatures	Completeness	Soundness	Unforgeability	Invisibility	Security Level
[1, 3] Chaum and Antwerpen	Provides a rigorous proof system ensuring that valid signatures are always accepted based zero-knowledge	The zero-knowledge framework ensures sound verification while resisting external biases	Depends on computational hardness with zero-knowledge enhancements for unforgeability but lacks modern robustness techniques	The use of zero-knowledge techniques for ensuring signature invisibility	Moderate classical security
[5] Boyar et al.	Ensures completeness with the added flexibility of convertible properties	Soundness is similar to [3] additionally convertible properties do not compromise soundness	Introduces conversion options, offering a unique attack resistance framework for unforgeability	Maintains invisibility but introduces convertibility, which could potentially expose metadata under certain conditions	High; zero-knowledge improves soundness
[16] Gennaro et al.	Ensures RSA-based signatures are complete; efficient for practical systems	Higher soundness then earlier works due to RSA framework and mathematical formalism	Provides strong unforgeability under standard RSA assumptions	Limited by RSA -based approaches; invisibility depends on secure key management and randomness	High; enhanced security for convertible signatures
[39] Aguilar Melchor et al.	Completeness depends on error-correction mechanisms inherent in code-based cryptography	Soundness guaranteed under the difficulty of decoding random linear codes	Strong unforgeability based on NP-hard decoding problems	Excellent invisibility since code-based schemes inherently obscure the relationship between keys and messages	Moderate; classical RSA-based security
[42] Rawal et al.	Completeness achieved with tight Gaussian sampling and lattice trapdoor construction	Soundness strengthened by lattice-based hardness assumptions	Very strong leveraging post-quantum security against classical and quantum attacks	Superior invisibility due to post-quantum construction and structured Gaussian sampling for enhanced secrecy	High; code-based cryptography is hard to break

computational and memory resources. We noted only [42] fully offer quantum resistance and [39] partially.

The comparative analysis of the listed papers in terms of practicality and usability (see Table 4) reveals significant advancements in making undeniable signature schemes more applicable in real-world scenarios.

Classical and RSA-based schemes rely on arithmetic and modular operations, making them efficient. Code and lattice-based schemes involve complex linear algebra and sampling processes, which demand more resources. RSA-based [16] has the simplest implementation due to its reliance on well-understood modular arithmetic. Also, [16] scale well, even for large systems, while classical interactive schemes [1, 3] and [5] have moderate scalability due to interaction requirements. On the other hand, post-quantum schemes [39] and [42] face scalability challenges because of resource-heavy operations and large key sizes. Simplicity in [16] ensures high usability. Interactive nature and computational overhead in [1, 3], [39], and [42] reduce user experience quality.

Table 3: Comparative analysis for the listed papers in terms of computational efficiency.

The signatures	Key Generation Time/Size	Signature Generation Time/Size	Verification Time	Efficiency	Resource Utilization	Quantum Resistance
[1, 3] Chaum and Antwerpen	Time: Moderate; Size: Small to moderate	Time: Moderate; Size: Moderate	High; requires multiple rounds for interactive zero-knowledge protocol	Moderate; zero- knowledge add complexity	Moderate; zero-knowledge computations increase memory and CPU requirements	None
[5] Boyar et al.	Time: Slightly higher than [1, 3] Size: Small to moderate	Time: Similar to [1, 3]; Size: Moderate to large	High; convertibility validation adds an extra step to standard verification	Moderate; convertible features add complexity	Moderate; to high; increased resources needed for convertibility checks	None
[16] Gennaro et al.	Time: Relatively low Size: Small	Time: Low to moderate Size: Moderate	Low to moderate; RSA-based verification is efficient but still depends on modular arithmetic	High; efficient for classical systems	Moderate; RSA frameworks are resource efficient but require careful parameter management	None
[39] Aguilar Melchor et al.	Time: High Size: Large	Time: Moderate Size: Large	Moderate; verification involves decoding linear codes, which is computationally complex	Low; code-based signing and verification are slow	High; code-based cryptography demands significant storage and processing for large matrices and codes	Partial; relies on NP-hard problem (e.g., decoding)
[42] Rawal et al.	Time: High Size: Large	Time: High Size: Large	High; lattice-based verification involves modular arithmetic and trapdoor dependent computation.	Low to moderate; lattice-based operations are computationally intensive	Very high, lattice-based systems demand extensive memory, computation power, and efficient random	Full; post-quantum secure

Table 4: Comparative analysis for the listed papers in terms practicality and usability.

The signatures	Operation	Ease of Implementation	Scalability	User Experience
[1, 3] Chaum and Antwerpen	Classical modular arithmetic and discrete logarithms. Relies on the hardness of factoring and DL with interactive zero-knowledge protocols	Low to moderate	More complex computations limit scalability	Zero-knowledge proofs increase interaction complexity, making the process less intuitive
[5] Boyar et al.	Classical modular arithmetic and discrete logarithms. Relies on the hardness of factoring and DL with convertible proof mechanisms	Moderate	Scales better than zero-knowledge but less than classical schemes	Convertible signatures improve utility but add some complexity
[16] Gennaro et al.	Modular arithmetic, RSA exponentiation and prime number factorization	High	Highly scalable due to RSA's modular arithmetic	Simple operations and standard key sizes ensure good user experience
[39] Aguilar Melchor et al.	Linear algebra operations on large matrices with decoding algorithms over linear error-correcting codes	Low	Large keys and matrix operations hinder scalability	Resource-intensive operations degrade user experience
[42] Rawal et al.	Matrix multiplications, Gaussian sampling, and modular arithmetic	Low	Computational intensity limits scalability in practical systems	Higher security but at the cost of complex operations impacting usability

7. Future Directions and Open Challenges

The field of undeniable signatures has made significant progress but several challenges and opportunities for innovation remain. These pertain to balancing efficiency with security, addressing post-quantum resilience, and advancing foundational design principles. One of the most persistent challenges in undeniable signature schemes is achieving an optimal balance between efficiency and security. Traditional undeniable signature schemes often rely on large key sizes to guarantee security, especially those based on RSA or discrete logarithms. As these schemes grow, their practicality diminishes, particularly for resource-constrained environment. Reducing key and signature sizes without compromising security remains a critical goal. Techniques like lattice-based cryptography or more compact zero-knowledge proof systems, such as succinct non-interactive zero-knowledge (zk-SNARKs), may offer promising solutions. However, these methods often introduce new complexities, such as increased computational requirements or reliance on less-mature security assumptions. Additionally, signature verification and disavowal protocols in many schemes require multiple rounds of interaction or significant computational power. Simplifying these protocols to reduce computational and communication costs while maintaining security properties such as invisibility and non-transferability is a crucial area for future research. Furthermore, there is a need to develop and adopt lightweight cryptographic primitives tailored for undeniable signatures, particularly for use in loT devices, mobile platforms, and constrained networks. These primitives should ensure robust security while minimizing energy and memory consumption.

As quantum computing advances, may classical cryptographic schemes, including traditional undeniable signatures, are at risk. Designing post-quantum undeniable signature protocols is not just a challenge but a necessity for future-proofing the technology. Lattice-based cryptography has become a prominent contender for post-quantum cryptography because of its robustness against quantum attacks. Incorporating these structures into undeniable signature schemes could provide a strong foundation for quantum-resistant designs. However, ensuring the schemes preserve essential properties like invisibility and undeniability in a post-quantum setting is non-trivial. Undeniability in post- quantum context requires a careful re-examination of interactive proof systems to ensure they remain secure under quantum adversaries. Zero-knowledge proofs, a cornerstone of many schemes, need adaptations to guarantee soundness and zero-knowledge properties against quantum algorithms. Enhancing non-transferability involves creating stronger commitment schemes that resist both classical and quantum attacks. Homomorphic commitments, tailored for post-quantum security, might play a pivotal role in this regard. Beyond adapting existing schemes, there is a need to explore entirely new paradigms and frameworks that align with the capabilities and threats of the quantum era. These protocols must seamlessly integrate post-quantum primitives with the unique requirements of undeniable signatures, including their interaction-heavy verification processes.

While classical security models are well-defined, post-quantum security models for undeniable signatures are still in their infancy. Establishing rigorous models is essential for evaluating and comparing new protocols. As undeniable signature schemes evolve, standardization efforts must focus on interoperability, usability, and performance benchmarks, especially for post-quantum solutions. Adapting schemes to work efficiently in large-scale systems, such as blockchain networks or distributed systems, while maintaining core properties, remains an open challenge. Exploring the use of undeniable signatures in areas such as secure multiparty computation, federated learning, or decentralized identity systems could unlock new applications and requirements.

8. Conclusions

The study of undeniable signature schemes has made significant strides, providing robust mechanisms for authentication with unique properties such as undeniability and non-transferability. This paper examined some the functionality of different undeniable signature schemes from classical to post-quantum designs. This study covered key design strategies, including zero-knowledge proofs, trapdoor function, and homomorphic commitments, highlighting their importance in ensuring the security and efficiency of these schemes. Furthermore, a detailed comparison of schemes has underscored the diversity in approaches and the trade-offs between efficiency and security. By organizing schemes based on their cryptographic primitives, interactive protocols, and application domains, this taxonomy serves as a foundation for identifying gaps and setting directions for future research undeniable signature schemes face several pressing challenges, so we must development the field and ensuring the practical adoption of schemes in real-world systems. While classical schemes rely on problem such as factoring or discrete logarithms, post-quantum schemes must leverage lattice-based, code-based, or hash-based primitives. This transition requires rigorous security proofs and efficient implementations. Balancing the trade-offs between signature size, key size, and computational overhead remains an open problem. By fostering innovation in design strategies, promoting structured classification, and addressing open challenges, the cryptographic community can pave the foundation for a new generation of safe and efficient undeniable signature schemes.

References

[1] David Chaum and Hans Van Antwerpen. Undeniable signatures. In *Advances in Cryptology - CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*. Springer, 1989.

[2] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.

[3] David Chaum. Zero-knowledge undeniable signatures. In *Advances in Cryptology—EUROCRYPT'90: Workshop on the Theory and Application of Cryptographic Techniques Aarhus, Denmark, May 21–24, 1990 Proceedings 9*, pages 458–464. Springer, 1991.

[4] Atsushi Fujioka Tatsuaki Okamoto Kazuo Ohta. Interactive bi-proof systems and undeniable signature schemes. 1998.

[5] Joan Boyar, David Chaum, Ivan Damgård, and Torben Pedersen. *Convertible undeniable signatures.* Springer, 1991.

[6] David Chaum, Eugène van Heijst, and Birgit Pfitzmann. *Cryptographically strong undeniable signatures, unconditionally secure for the signer.* Springer, 1992.

[7] David Chaum. Designated confirmer signatures. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 86–91. Springer, 1994.

[8] Tatsuaki Okamoto. Designated confirmer signatures and public-key encryption are equivalent. In *Advances in Cryptology—CRYPTO'94: 14th Annual International Cryptology Conference Santa Barbara, California, USA August 21–25, 1994 Proceedings 14*, pages 61–74. Springer, 1994.

[9] Markus Jakobsson. Blackmailing using undeniable signatures. In *Advances in Cryptology—EUROCRYPT'94: Workshop on the Theory and Application of Cryptographic Techniques Perugia, Italy, May 9–12, 1994 Proceedings 13*, pages 425–427. Springer, 1995.

[10] Markus Jakobsson, Kazue Sako, and Russell Impagliazzo. Designated verifier proofs and their applications. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 143–154. Springer, 1996.

[11] Markus Michels, Holger Petersen, and Patrick Horster. Breaking and repairing a convertible undeniable signature scheme. In *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, pages 148–152, 1996.

[12] Ivan Damgård and Torben Pedersen. New convertible undeniable signature schemes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 372–386. Springer, 1996.

[13] Sung-Hyun Yun and Tai-Yun Kim. Convertible undeniable signature scheme. In *Proceedings High Performance Computing on the Information Superhighway. HPC Asia'97*, pages 700–703. IEEE, 1997.

[14] Markus Michels and Markus Stadler. Efficient convertible undeniable signature schemes. In *Proc. of 4th annual workshop on selected areas in cryptography (SAC'97)*, pages 231–244, 1997.

[15] Claus-Peter Schnorr. Efficient signature generation by smart cards. *Journal of cryptology*, 4:161–174, 1991.

[16] Rosario Gennaro, Hugo Krawczyk, and Tal Rabin. Rsa-based undeniable signatures. In *Advances in Cryptology—CRYPTO'97: 17th Annual International Cryptology Conference Santa Barbara, California, USA August 17–21, 1997 Proceedings 17*, pages 132–149. Springer, 1997.

[17] Markus Michels and Markus Stadler. Generic constructions for secure and efficient confirmer signature schemes. In *Advances in Cryptology—EUROCRYPT'98: International Conference on the Theory and Application of Cryptographic Techniques Espoo, Finland, May 31–June 4, 1998 Proceedings 17*, pages 406–421. Springer, 1998.

[18] Tatsuaki Okamoto and David Pointcheval. The gap-problems: A new class of problems for the security of cryptographic schemes. In *Public Key Cryptography: 4th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2001 Cheju Island, Korea, February 13–15, 2001 Proceedings 4*, pages 104–118. Springer,

2001.

[19] Steven D Galbraith, Wenbo Mao, and Kenneth G Paterson. Rsa-based undeniable signatures for general moduli. In *Topics in Cryptology—CT-RSA 2002: The Cryptographers' Track at the RSA Conference 2002 San Jose, CA, USA, February 18–22, 2002 Proceedings*, pages 200–217. Springer, 2002.

[20] Steven D Galbraith and Wenbo Mao. Invisibility and anonymity of undeniable and confirmer signatures. In *Topics in Cryptology—CT-RSA 2003: The Cryptographers' Track at the RSA Conference 2003 San Francisco, CA, USA, April 13–17, 2003 Proceedings*, pages 80–97. Springer, 2003.

[21] Shafi Goldwasser and Erez Waisbard. Transformation of digital signature schemes into designated confirmer signature schemes. In *Theory of Cryptography Conference*, pages 77–100. Springer, 2004.

[22] Ingrid Biehl, Sacher Paulus, and Tsuyoshi Takagi. Efficient undeniable signature schemes based on ideal arithmetic in quadratic orders. *Designs, Codes and Cryptography*, 31:99–123, 2004.

[23] Benoît Libert and Jean-Jacques Quisquater. Identity based undeniable signatures. In *Topics in Cryptology–CT-RSA 2004: The Cryptographers' Track at the RSA Conference 2004, San Francisco, CA, USA, February 23-27, 2004, Proceedings*, pages 112–125. Springer, 2004.

[24] Jean Monnerat and Serge Vaudenay. Generic homomorphic undeniable signatures. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 354–371. Springer, 2004.

[25] J Monnerat and S Vaudenay. Undeniable signatures based on characters: How to sign with one bit, pkc'04, lncs 2947, 2004.

[26] Wakaha Ogata, Kaoru Kurosawa, and Swee-Huay Heng. The security of the fdh variant of chaum's undeniable signature scheme. In *Public Key Cryptography-PKC 2005: 8th International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23-26, 2005. Proceedings 8*, pages 328–345. Springer, 2005.

[27] Fabien Laguillaumie and Damien Vergnaud. Time-selective convertible undeniable signatures. In *Topics in Cryptology–CT-RSA 2005: The Cryptographers' Track at the RSA Conference 2005, San Francisco, CA, USA, February 14-18, 2005. Proceedings*, pages 154–171. Springer, 2005.

[28] Fabien Laguillaumie and Damien Vergnaud. Short undeniable signatures without random oracles: The missing link. In *International Conference on Cryptology in India*, pages 283–296. Springer, 2005.

[29] Kaoru Kurosawa and Swee-Huay Heng. 3-move undeniable signature scheme. In *Advances in Cryptology–EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005. Proceedings 24*, pages 181–197. Springer, 2005.

[30] Kaoru Kurosawa and Tsuyoshi Takagi. New approach for selectively convertible undeniable signature schemes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 428–443. Springer, 2006.

[31] Shanshan Duan. Certificateless undeniable signature scheme. *Information Sciences*, 178(3):742–755, 2008.

[32] Jacob CN Schuldt and Kanta Matsuura. An efficient convertible undeniable signature scheme with delegatable verification. In *Information Security, Practice and Experience: 6th International Conference, ISPEC 2010, Seoul, Korea, May 12-13, 2010. Proceedings 6*, pages 276–293. Springer, 2010.

[33] Qiong Huang and Duncan S Wong. Short and efficient convertible undeniable signature schemes without random oracles. *Theoretical Computer Science*, 476:67–83, 2013.

[34] Xiaoming Hu, Xiaojun Zhang, Chuang Ma, Huajie Xu, Jian Wang, and Wenan Tan. A designated verifier signature scheme with undeniable property in the random oracle. In *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pages 960–963. IEEE, 2016.

[35] Shu Han, Mande Xie, Bailin Yang, Rongxing Lu, Haiyong Bao, Jianhong Lin, Hai-Bo Hong, Mian-Xue Gu, and Song Han. A certificateless verifiable strong designated verifier signature scheme. *IEEE Access*, 7:126391–126408, 2019.

[36] Jia-Ch'ng Loh, Swee-Huay Heng, Syh-Yuan Tan, and Kaoru Kurosawa. A note on the invisibility and anonymity of undeniable signature schemes. In *Information Security Applications: 20th International Conference, WISA 2019, Jeju Island, South Korea, August 21–24, 2019, Revised Selected Papers 20*, pages 112–125. Springer, 2020.

[37] S Li and C Wang. An undeniable signature scheme based on lattice. *IJACT Int J Adv Comput Technol*, 4(12):260–267, 2012.

[38] K Preetha Mathew, Sachin Vasant, and C Pandu Rangan. On provably secure code-based signature and signcryption scheme. *IACR Cryptology ePrint Archive*, 2012:585, 2012.

[39] Carlos Aguilar-Melchor, Slim Bettaieb, Philippe Gaborit, and Julien Schrek. A code-based undeniable signature scheme. In *Cryptography and Coding: 14th IMA International Conference, IMACC 2013, Oxford, UK, December 17-19, 2013. Proceedings 14*, pages 99–119. Springer, 2013.

[40] Lei Hua, Mu Han, Shidian Ma, and Xiaolin Feng. An undeniable signature scheme based on quasi-dyadic codes. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 2189–2194. IEEE, 2018.

[41] Amir Jalali, Reza Azarderakhsh, and Mehran Mozaffari-Kermani. Efficient post-quantum undeniable signature on 64-bit arm. In *Selected Areas in Cryptography–SAC 2017: 24th International Conference, Ottawa, ON, Canada, August 16-18, 2017, Revised Selected Papers 24*, pages 281–298. Springer, 2018.

[42] Swati Rawal, Sahadeo Padhye, and Debiao He. Lattice-based undeniable signature scheme. *Annals of Telecommunications*, pages 1–8, 2022.

[43] Kunal Dey, Mansi Goyal, Bupendra Singh, and Aditi Kar Gangopadhyay. An undeniable signature scheme utilizing module lattices. *arXiv preprint arXiv:2410.19220*, 2024.

[44] Demba Sow, Mamadou Ghouraissiou Camara, et al. Provable security of the gener-

alized elgamal signature scheme. *J. Math. Res*, 11(6):1–77, 2019.