Quantum Resistant Digital Signatures Constructed from a Nonstandard Discrete Logarithm Problem over Finite Fields

Tuan Nguyen Kim [1,*], Luu Hong Dung [2], Hoang Duc Tho [3], and Ha Nguyen Hoang [4]

[1] Phenikaa School of Computing, Phenikaa University, Ha Dong, Hanoi, Vietnam
[2] Faculty of Information Security, Le Quy Don Technical University, Northern Tu Liem, Hanoi, Vietnam
[3] Faculty of Information Security, Vietnam Academy of Cryptography Techniques, Thanh Tri, Hanoi, Vietnam
[4] University of Sciences, Hue University, Hue, Vietnam
Email: tuan.nguyenkim@phenikaa-uni.edu.vn (T.N.K.); luuhongdung@lqdtu.edu.vn (L.H.D.);
thohd@actvn.edu.vn (H.D.T.); nguyenhoangha@hueuni.edu.vn (H.N.H.)
*Corresponding author

***Abstract*—The rise of quantum computing is putting tremendous pressure on existing cryptographic systems, particularly digital signature schemes based on Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography. Quantum algorithms such as Shor and Grover have demonstrated the ability to severely weaken traditional security assumptions, highlighting the urgent need to develop new quantum-resistant digital signature schemes. Rather than relying on standard approaches such as lattice-based or multivariate-based cryptography, this paper explores an alternative direction: leveraging a new nonlinear exponentiation problem defined over finite fields, whose mathematical structure is designed to render Shor's algorithm inapplicable and to reduce the effectiveness of Grover's algorithm. Based on this newly proposed hard problem, we introduce multiple digital signature schemes, each with a distinct structure in its signing and verification algorithms. Although these schemes differ in operational mechanisms, they all maintain correctness, remain secure against classical attacks, offer strong quantum resistance, and are fully compatible with existing Public Key Infrastructure system. Through both theoretical analysis and performance evaluation, we demonstrate that diversifying digital signature constructions from a single underlying hard problem is not only feasible but also offers practical advantages: it allows selecting a design best suited for specific application environments while maintaining post-quantum security. This result opens a promising new path toward the development of flexible, efficient, and long-term secure digital signature schemes for the post-quantum era.**

***Keywords*—post-quantum digital signature, new hard problem, shor's algorithm, grover's algorithm, Public Key Infrastructure (PKI), non-standard assumption**

I. Introduction

The continuous advancement of quantum computing is creating a major turning point for modern cryptography. Security assumptions that once served as the solid foundation for traditional digital signature schemes such as Rivest-Shamir-Adleman (RSA) [1], Elliptic Curve Digital Signature Algorithm (ECDSA) [2], Elgamal [3], etc., are becoming increasingly obsolete under the influence of quantum algorithms. Specifically, Shor's algorithm enables solving the Integer Factorization Problem (IFP) and the Discrete Logarithm Problem (DLP) in polynomial time, rendering most group-based cryptosystems ineffective; meanwhile, Grover's algorithm significantly reduces the complexity of brute-force attacks on hash functions and symmetric ciphers. As large-scale quantum computers become a reality, many existing security systems will no longer be safe, including the widely deployed Public Key Infrastructure (PKI) [4].

In response to this threat, the cryptographic community has heavily invested in the development of Post-Quantum Digital Signature schemes [5]. Among them, lattice-based schemes (e.g., Dilithium [6], Falcon [7]) and hash-based schemes (e.g., SPHINCS+ [8]) are being standardized by NIST due to their strong quantum resistance. However, these schemes still face several practical challenges, including large key and signature sizes, computational complexity [9, 10], and difficulties integrating with existing infrastructures due to incompatibility with traditional PKI structures. These limitations have created a gap between post-quantum security theory and real-world deployment, especially in resource-constrained environments such as IoT, embedded systems, and high-speed digital services.

Instead of following the well-established directions mentioned above [11], this paper approaches the problem from a different perspective: it proposes a new hard problem [12], defined over a Finite Field (FF) or on an Elliptic Curve (EC), in which the generator element is kept secret to neutralize exploitation by Shor's algorithm and reduce the effectiveness of Grover's algorithm. Unlike existing PQC schemes [13, 14], which typically rely on a fixed mathematical problem to construct a single scheme,

Manuscript received July 19, 2025; revised August 12, 2025; accepted September 10, 2025; published January 9, 2026.

doi:10.12720/jcm.21.1.1-11

we demonstrate that the proposed hard problem can serve as the foundation for multiple digital signature schemes, each with independent signing and verification algorithms tailored for different requirements in terms of performance, size, or compatibility.

The results of this research not only clarify the practical applicability of the new hard problem but also open up a flexible design path for post-quantum digital signatures, where a single mathematical foundation can give rise to diverse schemes tailored to varied deployment requirements. The following sections present the relationship between the proposed hard problem and the limitations of Shor and Grover, as well as how to prevent quantum attacks through careful group structure design.

II. The Threats Posed By Shor and Grover

The emergence of quantum computers not only transforms the landscape of modern computation but also places the entire current cryptographic infrastructure at risk of collapse. In this section, we examine two representative quantum algorithms, Shor's algorithm and Grover's algorithm, which lie at the heart of this threat.

However, unlike many current studies that attempt to build new schemes based on quantum-resistant structures such as lattice-based or code-based cryptography, our approach takes a different path: we design a new class of mathematical hard problems that cannot be reduced to the attack models used by Shor, while also minimizing the impact of Grover.

A. Shor's Algorithm and Its Implications for Classical Prolems

Shor's algorithm [15], introduced in 1994, marked a turning point in cryptography by being the first to solve two problems that were once considered the "backbone" of classical cryptography, IFP and DLP, in polynomial time on a quantum computer. This shattered the long-standing belief in the "hardness" assumptions that underpin the security of signature schemes such as RSA, DSA, and ECDSA.

Specifically, Shor's algorithm uses the Quantum Fourier Transform (QFT) to find the period of a number-theoretic function, a key step in breaking both IFP and DLP. While classical algorithms like GNFS or Pollard's rho solve IFP or DLP in sub-exponential or quasi-polynomial time, Shor's algorithm achieves a solution in $O((logN)^3)$ time, rendering cryptographic systems based on large integers or cyclic groups highly vulnerable once scalable quantum computers become a reality.

Therefore, new digital signature schemes must ensure that there is no reduction from their underlying problem to a period-finding problem, in order to prevent Shor-style attacks. In this research, we propose a new class of hard problems defined over prime finite fields, where the generator element, typically exposed in traditional systems, is kept secret. This approach breaks a necessary condition for applying the quantum Fourier transform, thereby allowing the proposed scheme to remain resilient against Shor's algorithm.

B. Grover's Algorithm and the Limits of Brute-Force Search

Grover's algorithm [16], introduced in 1996, does not solve structured mathematical problems like Shor's algorithm, but it proves to be extremely effective in reducing the time required for brute-force attacks. With Grover, a search space of size N requires only $\sqrt{N}$ queries to locate the desired element, significantly lowering the cost of attacks on components such as hash functions, symmetric ciphers, and even private keys if their length is insufficient.

In this context, the use of popular hash functions such as SHA-256 or SHA-3 in digital signature schemes is also affected: the effective security of an n bit hash function is reduced to approximately $2^{n/2}$. This makes preimage attacks and collision finding easier than originally expected. For example, with SHA-256, an attacker needs only 2^{128} trials instead of 2^{256}, compelling us to be more careful in choosing the output length and compression functions when designing signature schemes.

The distinctive feature of our approach is this: rather than replacing the entire cryptographic structure with bulky post-quantum systems, we make lightweight adjustments to key parameters, such as increasing the output length or iteration rounds of the hash function. This balances the impact of Grover's algorithm while keeping the overall design simple and fully compatible with existing PKI infrastructure.

In summary, the rapid advancement of quantum computing poses a serious threat to existing cryptographic systems, particularly those based on DLP and IFP. Although many post-quanta signature schemes have been proposed, they still exhibit significant limitations. Therefore, it is both urgent and promising to explore new approaches, especially those that construct quantum-resistant digital signature schemes without relying on traditional post-quantum algorithms, while remaining fully compatible with existing PKI systems.

III. Related Works

In the current trend of constructing quantum-resistant digital signature schemes, most existing research focuses on applying well-established post-quantum hard problem families such as Lattice-based problems (e.g., SVP, LWE [17]), Multivariate Quadratic (MQ) [18], Code-based (e.g., McEliece) [19], Hash-based [20], and Isogeny-based approaches. However, these directions often come with challenges such as large key sizes, low computational efficiency, or limited compatibility with traditional PKI.

An alternative research direction that has attracted growing attention is to leverage variants of DLP or, with the goal of constructing quantum-resistant signature schemes that retain compatibility with existing systems. Although these approaches may not fall within the scope of standard post-quantum cryptography, they introduce new types of hard problems that are sufficiently complex to serve as the foundation for efficient and quantum-resistant signature schemes. Notable works in this line of research include:

- Dobraunig *et al.* [21] proposed SPHINCS-256, a post-quantum secure digital signature scheme based on hash trees, while maintaining compatibility with traditional PKI infrastructures. Although it does not rely on lattice- or code-based assumptions, SPHINCS still achieves high quantum security.
- Hülsing *et al.* [22] introduced qTesla-F, a signature scheme that exploits special properties from a DLP variant defined over sparse matrix structures. Although not part of standardized post-quantum cryptography, this scheme demonstrates good performance and integration capabilities with existing PKI systems.
- Koblitz and Menezes [23] proposed using elliptic curve DLP variants combined with structure-oriented hash functions to build a new security layer that resists Shor's algorithm. They particularly emphasized selecting group structures that preserve compatibility with traditional deployment environments.
- Tuan [24] and his collaborators proposed a novel digital signature scheme based on the Inverse Nonlinear Multiplicative Problem, an extended form of classical DLP. This scheme showed promising performance on embedded systems and quantum resistance thanks to its nonlinear and hard-to-invert structure.

These studies have opened a new direction in the design of post-quantum digital signatures that avoid reliance on standardized post-quantum algorithms, thereby laying the groundwork for further research, such as the present paper, which explores newly proposed non-standard hard problems to construct digital signature schemes that are not only quantum-resistant but also fully compatible with existing PKI infrastructures.

IV. Proposed New Difficult Problem

In this section, we introduce a new class of computationally hard problems designed to serve as the foundation for constructing quantum-resistant digital signature schemes. This problem is formulated as an enhanced variant of the traditional DLP [25] and the Elliptic Curve (ECDLP) [26], with a crucial distinction: a key parameter (such as g in DLP or G in ECDLP) is kept secret.

Hiding this parameter not only breaks the conventional analytical structure but also neutralizes classical solution techniques based on discrete logarithms. As a result, the proposed problem becomes significantly harder than the standard DLP or ECDLP. We present the formal definition of the problem, analyze its mathematical hardness, and demonstrate that it cannot be efficiently solved by either classical or quantum algorithms, including Shor's and Grover's algorithms.

A. Proposed Hard Problem over a Prime Finite Field

Based on the traditional DLP over a prime finite field F_p, we propose two new variants of hard problems that offer a higher level of security. The core idea is to conceal the generator parameter g (which is the generator of the multiplicative group F_p^* and is publicly known in classical DLP).

When this parameter is hidden or replaced by a secret value, such as the private key x, the problem loses its familiar algebraic structure, thereby becoming significantly harder to solve. In the simplest case, if the generator g is replaced by a secret value x, then the new problem defined over the prime finite field can be stated as follows:

Type 1: Nonlinear Monotonic Exponentiation Problem: Given a prime number p, for every positive integer y in $\mathbb{F}_p$, find an integer x that satisfies the equation:

$$a^x \equiv x^b \, mod \, p \tag{1}$$

Type 2: Nonlinear Polynomial Exponentiation Problem: Given a prime number p, for every pair of integers (a, b) in $\mathbb{F}_p \times \mathbb{F}_p$, find an integer x such that:

$$a^x \equiv x^b \, mod \, p. \tag{2}$$

Type 3: Two-Dimensional Cross Exponentiation Problem (Nonlinear): Given a prime number p, for every pair $(y_1, y_2) \in \mathbb{F}_p \times \mathbb{F}_p$, find two integers $(x_1, x_2) \in \mathbb{F}_p \times \mathbb{F}_p$ such that:

$$\begin{cases} y_1 \equiv {x_1}^{x_2} \, mod \, p \\ y_2 \equiv {x_2}^{x_1} \, mod \, p^{\cdot} \end{cases} \tag{3}$$

All three newly proposed hard problem variants exhibit strong non-linearity, which makes them resistant to linearization through conventional techniques and unsuitable for classical DLP-solving methods. This indicates their high potential for cryptographic applications, particularly in constructing digital signature schemes with enhanced security levels.

None of the three variants can be efficiently solved using traditional algorithms designed for DLP over finite fields, such as the Baby-Step Giant-Step algorithm, Pollard's Rho for DLP, or the Index Calculus method. Specifically:

- For Type 1: The function x^x is nonlinear with respect to x, and it is not a fixed-base exponentiation function. Therefore: Pollard's Rho algorithm cannot be applied, as it relies on the linearity of modular exponentiation; Baby-Step Giant-Step is not applicable, because the algorithm has no way to precompute a table for the x^x function; Index Calculus cannot be used, since the function cannot be decomposed into base elements as required by the method. Moreover, computing the inverse of the function $x^x \, mod \, p$ is mathematically extremely difficult, as it lacks an exploitable group structure.
- For Type 2: This is an equation involving the interference of two exponential functions: On the left-hand side, a^x follows the form of traditional discrete logarithm; On the right-hand side, x^b is a polynomial form. In this equation, x appears both in the exponent and the base, so it cannot be isolated on one side as in the traditional DLP. Therefore, known DLP-solving algorithms cannot be applied. Furthermore, since both sides of the equation depend on x in fundamentally different and nonlinear ways, there is no method to transform it into simple additive or multiplicative

groups. In other words, the group structure is unclear, making it impossible to reduce the problem to a cyclic group DLP, a fundamental requirement for many DLP-based algorithms.

- For Type 3: Although this type appears superficially similar to the traditional DLP due to its use of modular exponentiation, the cross-dependent and nonlinear relationship between the two variables x_1 and x_1 makes it impossible to transform it into a classical DLP form. Specifically, traditional algorithms exploit the one-dimensional structure of the DLP, that is, solving for the exponent x in the equation $y = g^x \, mod \, p$, given y and g. However, Type 3 is a system of two nonlinear equations, creating intertwined and asymmetric dependencies between the two unknowns, which prevents any straightforward transformation into a linear or basic logarithmic form where classical algorithms could be applied.

The absence of a fixed generator, as well as the lack of a linear relationship between the exponent and the remaining components, renders operations like "conversion to discrete logarithm" or "variable separation" meaningless. Moreover, no standard technique exists to reduce this system of equations to a single variable without losing critical information. As a result, Form 3 constitutes a fundamentally different hard problem compared to the standard DLP, and it cannot be solved using any existing classical algorithms for DLP over finite fields.

In summary, both forms of the proposed new hard problem exhibit strong non-linearity, lack familiar algebraic structures, and cannot be reduced to traditional DLPs. This exceptional complexity opens new possibilities for use as the cryptographic foundation of post-quantum digital signature schemes, where both Shor's and Grover's algorithms become ineffective.

B. Post-Quantum Resistance of the Proposed Hard Problems

We evaluate the post-quantum resistance of the newly proposed hard problems based on their immunity to two representative quantum algorithms: Shor's and Grover's.

- Shor resistance of Type 1 ($y = x^x \, mod \, p$): Shor's algorithm can only solve hidden-variable problems of the form $g^x \equiv y$, where the base ggg is fixed. In this case, both the base and the exponent are x, and the function $f(x) = x^x$ lacks a group structure or a usable period for QFT (Quantum Fourier Transform), rendering Shor ineffective.
- Shor resistance of Type 2 ($a^x \equiv x^b \, mod \, p$): This nonlinear "exponential-polynomial" equation cannot be reduced to the standard form g^x. Since there exists no quantum transformation capable of exploiting a hidden period in this type, Shor's algorithm is inapplicable to this problem.
- Shor resistance of Type 3 $\begin{pmatrix} y_1 \equiv {x_1}^{x_2} \, mod \, p \\ y_2 \equiv {x_2}^{x_1} \, mod \, p \end{pmatrix}$: This is a nonlinear, two-dimensional cross-exponentiation problem, lacking a pure group structure. It does not fall within the class of Abelian problems like the standard DLP or other homomorphic group problems. Shor's algorithm, which leverages periodicity in finite Abelian groups (especially for classical DLP), cannot be directly applied to Type 3. Specifically, solving the system in Type 3 would require a quantum algorithm to search over the entire space of pairs (x_1, x_2), which forms a nonlinear, two-dimensional search space with no standard group structure. Currently, there is no known effective quantum method to transform Type 3 into a periodic group structure suitable for Shor's algorithm. These highlights Type 3's potential resistance to Shor-based quantum attacks, indicating structural-level post-quantum security.
- In theory, Grover's algorithm reduces the time complexity of brute-force search from $O(N)$ to $O(\sqrt{N})$, applicable to so-called "black-box" problems, where no specialized quantum algorithm exists to solve the problem faster. The hard problems proposed in this paper, including: Type 1 (Nonlinear Monotonic Exponentiation); Type 2 (Nonlinear Exponential-Polynomial), and Type 3 (Two-Dimensional Cross Exponentiation); all exhibit high nonlinearity, lack group structure, and do not possess clear mathematical forms that would enable exploitation by quantum algorithms more efficient than Grover.

Therefore, in the worst-case scenario where no specific quantum algorithm exists, these problems would only be affected by Grover's algorithm, meaning the security level would degrade by half the size of the search space. Specifically, Grover can reduce the complexity of brute-force search over the set $x \in [1, p-1]$ from $O(p)$ to $O(\sqrt{N})$, which is still exponential complexity. This implies that if the original key space provides 128 bit security, under Grover's impact, the effective security would drop to approximately 64 bits. To compensate, one can increase the key size accordingly (e.g., from 128 to 256 bits) to maintain the desired security level. Therefore, if ppp is chosen sufficiently large (e.g., ≥ 256 bits), the proposed hard problems remain secure against Grover's algorithm. For instance, if $p \approx 2^{256}$, Grover would require 2^{128} steps, which remains infeasible for practical attacks.

In summary, all three proposed problem types can effectively resist Grover's algorithm through parameter size scaling, while still maintaining acceptable runtime performance.

From the three proposed hard problem variants, it can be observed that they all represent nonlinear, monotonic, and multidimensional extensions of the classical DLP. These problems not only preserve strong one-wayness but also naturally increase computational complexity through nonlinear exponents and symmetric, interleaved structures among the variables. Preliminary evaluations of resistance against classical algorithms, such as Baby-Step Giant-Step, Pollard's Rho, and Index Calculus, as well as quantum algorithms like Shor and Grover, indicate that these problems have strong potential as the foundation for both traditionally secure and post-quantum secure cryptographic schemes. In the following chapters, these problems will be applied to construct new digital signature schemes that aim to ensure correctness, security, and performance, with a vision to eventually replace traditional DLP-based schemes in modern computing environments.

V. Construction of Post-Quantum Digital Signature Schemes Based on the Proposed Non Standard DLP Problem

In this section, we present the construction of two new digital signature schemes based on the proposed non-standard exponentiation problems over a prime finite field. Specifically, the first scheme is built upon Problem Type 1, the Nonlinear Monotonic Exponentiation Problem, while the second is based on Problem Type 3, the Nonlinear Polynomial Exponentiation Problem, which involves two private keys and two public keys. Both schemes follow the standard three-phase model of a digital signature scheme (key and parameter initialization, signature generation, and signature verification) and are designed with the goal of achieving strong security against both classical and quantum attacks. Furthermore, the post-quantum resistance, including robustness against Shor's and Grover's algorithms, is thoroughly analyzed for each scheme in this section.

A. First Scheme (DSS-5.1)

1) Key and parameter generation algorithm for DSS-5.1

The public/private key pair of the end-user is generated by the Key Generation algorithm, based on a set of domain parameters, which includes a pair of prime numbers p and q satisfying $q|(p-1)$. These domain parameters can be generated according to standards such as ISO/IEC 14888-3, FIPS 186-4, or GOST R34.10-94.

The proposed scheme is constructed based on Problem Type 1 of the newly introduced hard problems, and unlike traditional DLP-based systems where the private key is chosen directly, here the private key x is computed from an element $\alpha \in Z_p^*$.

The Key Generation Algorithm of DSS-5.1 is described as follows:

```
Algorithm DSS-5.1a:
Input: $L_p, L_q$.
Output: $p, q, x, y$.
    [1]. generate $p, q$: $len(p) = L_p, len(q) = L_q, q|(p-1)$
    [2]. select $\alpha$: $1 < \alpha < p$
    [3]. $x \leftarrow \alpha^{\frac{p-1}{q}} \mod p$: If $(x = 1)$ then goto [2]
    [4]. $y \leftarrow x^{-(x)^{-1}} \mod p$: If $(y = 1)$ then goto [2]
    [5]. return $(p, q, x, y)$
```

where $len(.)$ denotes function that calculates the length (in bits) of an integer, L_p and L_q denote the lengths (in bits) of prime numbers p and q, p and q are system parameters or domain parameters, and x and y denote the private and public keys of the signer.

Note that:

- Not choosing the secret key x directly, but instead computing x through α, where $\alpha \in Z_p^*$, offers several advantages in both security and the mathematical structure of the scheme: (i) This construction ensures that the secret key x lies within the subgroup generated by α, meaning the secret key space is constrained by a controllable structure with high randomness, rather than being chosen arbitrarily. This reduces the risk of generating weak or degenerate keys; (ii) The use of the exponentiation function $\alpha^{\frac{p-1}{q}}$ leverages the structure of the order-q subgroup in Z_q^*, a common technique in DLP-based schemes. This enhances resistance against structural analysis attacks or weak-key exploitation techniques; and (iii) Theoretically, deriving x from the element α rather than selecting it directly makes the key generation process less dependent on the quality of the random number generator. It is well known that low-quality randomness sources are frequently exploited in real-world attacks.
- In the Type 1 hard problem, the expression $x^x \mod p$ is used to construct a nonlinear one-way function with high computational complexity, thereby forming the foundation for the problem's security. However, when designing a digital signature scheme, directly using the formula $y = x^x \mod p$ to compute the public key can lead to relationships that are easily traceable in reverse, especially if information such as the value of x or parts of the signature are leaked. Moreover, the expression $x^x \mod p$ may result in repetitions or collisions for certain values of x, potentially compromising the uniqueness and unpredictability of the public key.
- Instead, the formula $y \leftarrow x^{-(x)^{-1}} \mod p$ is chosen to enhance nonlinearity and further complicate the attacker's ability to infer the private key. This expression introduces two layers of nonlinearity: one involving an inverse, and another involving an exponentiation of that inverse, significantly increasing the difficulty of decryption or reverse-engineering. This is a key distinction that allows the scheme to achieve a higher level of security, while effectively leveraging the structure of the Type 1 hard problem without directly copying its full expression into the public key formula.

Moreover, this approach introduces a level of independence between the hard problem and the public key: Although the scheme is constructed based on the Type 1 hard problem, modifying the public key generation formula helps decouple the direct dependency between the problem and the key entity. This enhances the overall security because the attacker cannot easily exploit the solution of the hard problem to attack the signature scheme.

2) Signature generation algorithm for message M

Assume (r, s) is the digital signature for the message M. The first component of the signature, r, is computed using the following formula:

$$r = (x)^{(x \times h)^{-1} \times \left(1 + k^{-(x-1)} \times x^{-(k-1)}\right)} (k)^{k^{-x} \times x^{-(k-1)}} \mod p \tag{4}$$

where k is a random integer selected in the range $(1, q)$, h is the representative value (hash value) of the message M, generated by the hash function $H(.)$ as $h = H(M)$.

The second component of the signature, s is calculated as follows (where $n = p \times q$):

$$s = r \times k^{h} \times x^{k} \; mod \; n \tag{5}$$

The signature generation algorithm of the DSS-5.1 scheme is described as follows:

```
Algorithm DSS-5.1b:
Input: $p, q, x, M$.
Output:$(r, s)$.
    [1]. select $\beta$: $1 < \beta < p$
    [2]. $k \leftarrow \beta^{\frac{p-1}{q}} \; mod \; p$: If ($k = 1$) then goto [1]
    [3]. $h \leftarrow H(M)$
    [4]. $r \leftarrow (x)^{(x \times h)^{-1} \times (1 + k^{-(x-1)} \times x^{-(k-1)})} \times$
              $(k)^{k^{-x} \times x^{-(k-1)}} \; mod \; p$
    [5]. $n \leftarrow p \times q$
    [6]. $s \leftarrow r \times k^{h} \times x^{k} \; mod \;\; n$
    [7]. return $(r, s)$
```

where M denotes the message to be signed, with $M \in \{0,1\}^{\infty}$; $H(.)$ is hash function defined as $H :\{0,1\}^{*} \mapsto Z_{h}$, and the message value satisfies $q < M < p$.

3) Signature verification algorithm for message M

The signature verification algorithm of the scheme is based on the following assumption:

$$(y)^{(s \; mod \; q)} \times (s \; mod \; p)^{r} \; mod \; p = (r)^{(s \; mod \; q) \times h + r} \; mod \; p \tag{6}$$

That is, if the message M and the signature (r, s) satisfy Eq. (6), then the signature is considered valid, and the message is verified as authentic and intact. Conversely, if Eq. (6) is not satisfied, the signature is considered forged, and the message is rejected in terms of authenticity and integrity.

The signature verification algorithm of the DSS-5.1 scheme is described as follows:

```
Algorithm DSS-5.1c:
Input: $p, q, y, M, (r, s)$.
Output: TRUE/FALSE.
    [1]. $h \leftarrow H(M)$
    [2]. $a \leftarrow (y)^{(s \; mod \; q)} \times (s \; mod \; p)^{r} \; mod \; p$
    [3]. $b \leftarrow (r)^{(s \; mod \; q) \times h + r} \; mod \; p$
    [4]. if ($a = b$) then return (True)
            else return (False)
```

where M denotes the message and (r, s) denotes the signature to be verified. If the result is True, the integrity and origin of M are confirmed; otherwise, if the result is False, the origin and integrity of M are rejected.

4) Proof of correctness of the DSS-5.1 scheme

What needs to be proved here is:

If $a = (y)^{(s \; mod \; q)} \times (s \; mod \; p)^{r} \; mod \; p$ (7) and $b = (r)^{(s \; mod \; q) \times h + r} \; mod \; p$ (8) then: $a = b$.

Indeed, if the signature and message to be verified are not forged, from Eqs. (4)−(7) we will have:

$$\begin{aligned} a &= (y_1)^{(s \; mod \; q)} \times (s \; mod \; p)^{r} \; mod \; p \\ &= (x)^{-x^{-1} \times r \times k^{h} \times x^{k}} \times (r \times k^{h} \times x^{k})^{r} \; mod \; p \\ &= (x)^{-r \times k^{h} \times x^{k-1}} \times (k)^{h \times r} \times (x)^{k \times r} \times (r)^{r} \; mod \; p \end{aligned} \tag{9}$$

From Eqs. (5−8) we get:

$$\begin{aligned} b &= (r)^{(s \; mod \; q) \times h + r} \; mod \; p \\ &= (r)^{(s \; mod \; q) \times h} \times (r)^{r} \; mod \; p \\ &= (r)^{r \times h \times k^{h} \times x^{k}} \times (r)^{r} \; mod \; p \\ &= \left((x)^{-x^{-1} \times h^{-1}} \times (x)^{k^{-(x-1)} \times x^{-k} \times h^{-1}} \times (k)^{k^{-h} \times x^{-k}}\right)^{r \times h \times k^{h} \times x^{k}} \times (r)^{r} \; mod \; p \\ &= (x)^{-r \times k^{h} \times x^{k-1}} \times (x)^{r \times k} \times (k)^{r \times h} \times (r)^{r} \; mod \; p \end{aligned} \tag{10}$$

From Eqs. (9)−(10) we have: $a = b$.

Thus, the correctness of the scheme has been proved.

B. Second Scheme (DSS-5.2)

1) Key and parameter generation algorithm for DSS-5.2

In the second scheme, which is constructed based on Problem Type 3 of the proposed hard problem, the signer must generate a pair of private keys consisting of two components (x_1, x_2), along with the corresponding public key pair (y_1, y_2) , and standard domain parameters including a large prime p and a smaller prime q such that $q|(p-1)$. The selection and generation of these domain parameters are performed according to current recommendations from international standards such as ISO/IEC 14888-3, FIPS 186-4, or GOST R34.10-94, to ensure compatibility and security in real-world application environments.

The DSS-5.2a key generation algorithm below describes the detailed initialization procedure. Here, L_q and L_p denote the bit lengths of q and p, respectively. The algorithm ensures that the generator elements are chosen randomly, without repetition, and that the resulting public key values have strong cryptographic strength, while avoiding degenerate values that may lead to private key information leakage.

```
Algorithm DSS-5.2a:
Input: $L_q, L_p$.
Output: $p, q, x_1, x_2, y_1, y_2$.
    [1]. generate $p, q, len(p) = L_p; len(q) = L_q; q|(q-1)$
    [2]. select $a_1$:$1 < a_1 < p$
    [3]. $x_1 \leftarrow (a_1)^{\frac{p-1}{q}} \; mod \; p$ if ($x_1 = 1$) then goto [2]
    [4]. select $a_2$: $1 < a_2 < p$
    [5]. $x_2 \leftarrow (a_2)^{\frac{p-1}{q}} \; mod \; p$: If ($x_2 = 1$) then goto [4]
    [6]. $y_1 \leftarrow (x_1)^{x_2}$ mod$p$: If ($y_1 = 1$) then goto [2]
    [7]. $y_2 \leftarrow (x_2)^{-x_1}$ mod$p$: If ($y_2 = 1$) then goto [2]
    [8]. return $(p, q, x_1, x_2, y_1, y_2)$
```

where $len(.)$ denotes the function that calculates the length (in bits) of an integer; L_q and L_p denote the lengths (in bits) of prime numbers p and q; p and q are domain parameters; and x_1, x_2, y_1, y_2 denote the private and public keys of the signer.

Re-selecting values when the result is 1 is intended to avoid degenerate cases in which some intermediate values during the signing process become overly simple (e.g., equal to 1), potentially allowing an attacker to exploit this to establish a direct relationship with the private key, thereby leaking key information or enabling key recovery.

2) Signature generation algorithm for message M

Assume we need to generate a digital signature (r, s) for the message M. The first component of the signature, r, is calculated using the following formula:

$$r = \left((x_1)^{x_2+h\times(x_1)^{-h}\times(x_2)^{-k}} \times (x_2)^{(x_1\times h+k)\times(x_1)^{-h}\times(x_2)^{-k}}\right) \ mod\,p \qquad (11)$$

where k is a random integer selected in the range $(1, q)$ and h is the hash value representing the message M, computes as $h = H(M)$.

The second component of the signature, s, is calculated as follows $(n = p \times q)$:

$$s = r \times (x_1)^h \times (x_2)^k \ mod\,n \qquad (12)$$

The signature generation algorithm of the proposed scheme is described in detail as follows (see DSS-5.2b).

where M denotes the message to be signed, with $M \in \{0,1\}^\infty$; $H(.)$ is a hash function defined as $H: \{0,1\}^* \mapsto Z_h$, where $q < h < p$; h is the representative value (hash value) of M, computed as $h = H(M)$ and k is a randomly chosen value in the range $(1, q)$.

Algorithm DSS-5.2b:

```
Input: $p, q, x_1, x_2, M_2$
Output: $(r, s)$
    [1]. generate k: $1 < k < q$
    [2]. $h \leftarrow H(M)$
    [3]. $r \leftarrow \left((x_1)^{x_2+h\times(x_1)^{-h}\times(x_2)^{-k}} \times (x_2)^{(x_1\times h+k)\times(x_1)^{-h}\times(x_2)^{-k}}\right) mod\,p$
    [4]. $n \leftarrow p \times q$
    [5]. $s \leftarrow r \times (x_1)^h \times (x_2)^k \ mod\,n$
    [6]. return $(r, s)$
```

3) Signature verification algorithm for message M

The signature verification algorithm of the proposed scheme is based on the following assumption:

$$(y_1)^{(s\,mod\,q)} \times (s\,mod\,p)^r \ mod\,p = (y_2)^{r\times h} \times (r)^{(s\,mod\,q)+r} \ mod\,p \qquad (13)$$

In other words, if the message M and the signature (r, s) satisfy Eq. (13), the signature is considered valid, and the message is verified as authentic and intact. Conversely, if the equation is not satisfied, the signature is deemed forged, and the message is rejected in terms of authenticity and integrity.

The signature verification algorithm of the proposed scheme is described as follows:

Algorithm DSS-5.2c:

```
Input: $p, q, y_1, y_2, M, (r, s)$
Output: True/False.
    [1]. $h \leftarrow H(M)$
    [2]. $a \leftarrow (y_1)^{(s \text{ mod } q)} \times (s \text{ mod } p)^r \text{mod } p$
    [3]. $b \leftarrow (y_2)^{r\times h} \times (r)^{(s \text{ mod } q)+r} \text{mod} p$
    [4]. if $(a = b)$ then return (True)
         else return (False)
```

where M denotes the message and (r, s) denotes the signature to be verified. If the result is True, the integrity and origin of M are confirmed; otherwise, if the result is False, the origin and integrity of M are rejected.

4) Proof of correctness of the DSS-5.2 scheme

What needs to be proved here is:

If $a = (y_1)^{(s \text{ mod } q)} \times (s \text{ mod } p)^r \text{mod } p$ (14) and $b = (y_2)^{r\times h} \times (r)^{(s \text{ mod } q)+r} \text{mod } p$ (15) then: $a = b$.

Indeed, if the signature and message to be verified are not forged, from (12) and (14) we will have:

$$\begin{aligned} a &= (y_1)^{(s\,mod\,q)} \times (s\,mod\,p)^r \ mod\,p \\ &= (x_1)^{x_2\times r\times(x_1)^h\times(x_2)^k} \times (r \times (x_1)^h \times (x_2)^k)^r \ mod\,p \\ &= (x_1)^{r\times(x_1)^h\times(x_2)^{k+1}} \times (x_1)^{h\times r} \times (x_2)^{k\times r} \times (r)^r \ mod\,p \qquad (16) \end{aligned}$$

and from Eqs. (11, 12, 15) we get:

$$\begin{aligned} b &= (y_2)^{r\times h} \times (r)^{(s\,mod\,q)+r} \ mod\,p \\ &= (y_2)^{r\times h} \times (r)^{(s\,mod\,q)} \times (r)^r \ mod\,p \\ &= (x_2)^{-x_1\times r\times h} \times \left((x_1)^{x_2+h\times(x_1)^{-h}\times(x_2)^{-k}} \times (x_2)^{(x_1\times h+k)\times(x_1)^{-h}\times(x_2)^{-k}}\right)^{(s\,mod\,q)} \times (r)^r \ mod\,p \\ &= (x_2)^{-x_1\times r\times h} \times (x_2)^{x_1\times h\times(x_1)^{-h}\times(x_2)^{-k}\times(s\,mod\,q)} \times (x_1)^{x_2\times(s\,mod\,q)} \\ &\quad \times (x_1)^{h\times(x_1)^{-h}\times(x_2)^{-k}\times(s\,mod\,q)} \times (x_2)^{k\times(x_1)^{-h}\times(x_2)^{-k}\times(s\,mod\,q)} \times (r)^r \ mod\,p \\ &= (x_2)^{-x_1\times r\times h} \times (x_2)^{x_1\times h\times(x_1)^{-h}\times(x_2)^{-k}\times r\times(x_1)^h\times(x_2)^k} \times (x_1)^{x_2\times r\times(x_1)^h\times(x_2)^k} \\ &\quad \times (x_1)^{h\times(x_1)^{-h}\times(x_2)^{-k}\times r\times(x_1)^h\times(x_2)^k} \times (x_2)^{k\times(x_1)^{-h}\times(x_2)^{-k}\times r\times(x_1)^h\times(x_2)^k} \times (r)^r \ mod\,p \\ &= (x_2)^{-x_1\times r\times h} \times (x_2)^{x_1\times r\times h} \times (x_1)^{r\times(k)^h\times(x_2)^{k+1}} \times (x_1)^{h\times r} \times (x_2)^{k\times r} \times (r)^r \ mod\,p \\ &= (x_1)^{r\times(x_1)^h\times(x_2)^{k+1}} \times (x_1)^{h\times r} \times (x_2)^{k\times r} \times (r)^r \ mod\,p \qquad (17) \end{aligned}$$

From Eqs. (16−17) we have: $a = b$.

Thus, the correctness of the scheme has been proved.

C. Security Level of the Constructed DS Schemes

1) Resistance to classical attacks of DSS-5.1 and DSS-5.2

Both DSS-5.1 and DSS-5.2 are designed based on nonlinear variants of the newly proposed hard problem over a prime finite field, with the goal of eliminating the applicability of traditional discrete logarithm solving algorithms. As a result, they offer strong resistance against two common types of attacks in current PKI environments: attacks aimed at recovering the private key, and signature forgery attacks.

a) For the DSS-5.1 scheme

Attacks on the private key: Attacks targeting the private key typically occur at the key generation algorithm (DSS-5.1a) and the signature generation algorithm (DSS-5.1b). The attacker's goal is to recover the private key x from the public key y, or from a generated signature on a message M.

- Key Generation Algorithm: In this phase, the public key is computed as $y = x^{-x^{-1}}\ mod\ p$. To recover x from y, an attacker would have to solve the nonlinear equation $x^x \equiv y\ mod\ p$, which is exactly Type 1 of the proposed hard problems. Currently, no known mathematical method can solve this problem in polynomial time, except by brute-forcing all possible values. Hence, the only feasible attack is exhaustive search with complexity about 2^n, where $n = |p|$.
- Signature Generation Algorithm: In steps such as [2] and [4], variables k and x are used, but k is randomly generated and not disclosed. Step [6] involves a complex expression combining k, x, and the hash value h, which is highly resistant to inversion. Therefore, attacking to recover x from the output of the signature generation algorithm also reduces to solving the equation $x^x \equiv y\ mod\ p$, which remains infeasible except through brute-force methods.

Forgery attacks: A forgery attack occurs when an attacker can generate a valid signature for a message without knowing the private key.

The signature verification algorithm (Algorithm DSS-5.1c) for this scheme specifies that a pair of values (r, s) is accepted as a valid signature for the message M if the following condition is satisfied:

$$(y)^{(s\ mod\ q)} \times (s\ mod\ p)^r\ mod\ p$$
$$= (r)^{(s\ mod\ q)\times h+r}\ mod\ p \qquad (*)$$

It can be seen that condition (*) essentially corresponds to Type 2 of the proposed hard problems, as discussed in Section IV. This is a nonlinear exponential-polynomial problem, with no clear group structure, making it impossible to apply the traditional DLP techniques. That means the attacker cannot solve this equation, and therefore cannot forge a valid signature.

b) For the DSS-5.2 scheme

Attack on the Private Key: Attacks on the private key typically target the key generation algorithm (DSS-5.2a) and the signature generation algorithm (DSS-5.2b). The attacker's goal is to recover the private key pair x_1, x_2 from the public key y_1, y_2 or from an existing signature generated for message M.

- However, the attacker cannot recover the private key from the public key, because in algorithm DSS-5.2a: Lines 2 and 4: Two random numbers $\alpha_1, \alpha_2 \in F_p$ are selected. Lines 3 and 5: Ensure that x_1 and x_2 are not equal to 1; Lines 6 and 7: The public keys are generated using nonlinear functions: $y_1 = f(x_1)\ mod\ p$ and $y_2 = g(x_2)\ mod\ p$, where f and g are nonlinear functions that cannot be reduced to forms solvable via discrete logarithms. Therefore, since no discrete logarithm computation can be applied, the attacker cannot recover the private key from the public key.
- The attacker also cannot derive the private key from the signature on message M, because in algorithm DSS-5.2b: Line 1: A random k is selected from the range $(1, q)$; Lines 3 and 4: The signature (r, s) is generated based on the newly proposed hard problem, not on standard multiplication. Thus, since the signature formula is based on the newly proposed hard problem, the attacker cannot solve the equations to recover the private key without also solving this hard problem, which is currently intractable.

Forgery Attack on Signatures: A forgery attack occurs when an attacker is able to generate a valid signature for a message without knowing the private key. However, in this scheme, an attacker cannot forge a signature because a forged signature must satisfy the condition:

$$(y_1)^{(s \bmod q)} \times (s \bmod p)^r \bmod p$$
$$= (y_2)^{r\times h} \times (r)^{(s \bmod q)+r} \bmod p$$

It can be observed that this condition essentially corresponds to the second type of the newly proposed hard problems. Therefore, in order to generate a valid signature, the attacker would have to solve this new hard problem. However, as of now, no algorithm other than brute-force can achieve this.

Thus, both schemes demonstrate strong resistance against classical attacks due to their novel nonlinear structures that cannot be reduced to the traditional DLP. DSS-5.1 employs a complex single-variable nonlinear function, while DSS-5.2 extends this to a nested two-variable system, thereby offering a higher level of security. Although both schemes are built on the same mathematical foundation, a prime finite field, they represent two distinct implementation approaches, similar in principle yet different in structure and cryptographic complexity.

2) Post-quantum resistance of the proposed digital signature schemes

As analyzed in Section IV.B, the newly proposed hard problems demonstrate resistance to the quantum algorithms Shor and Grover, thanks to their nonlinear structure and the fact that they cannot be reduced to the standard discrete logarithm form. However, the post-quantum security level of a digital signature scheme does not rely solely on the underlying hard problem, but also on how this problem is integrated into each step of the signing and verification process.

In this paper, both proposed schemes, DSS-5.1 and DSS-5.2, leverage different variants of the new hard problem, with a design that ensures the generator element is never publicly disclosed and the cyclic structure is not exposed, conditions required for Shor's algorithm to be applicable. At the same time, all parameters potentially affected by Grover's algorithm (e.g., symmetric keys, hash values) are chosen with sufficient bit-lengths to ensure post-quantum security.

a) Resistance to shor and grover of the dss-5.1 scheme

The DSS-5.1 scheme is constructed based on Type 1 of the newly proposed hard problem, a special nonlinear problem in which the unknown appears both as the base and the exponent. This form cannot be reduced to the traditional DLP, which is the primary target of Shor's quantum algorithm.

- **On resistance to Shor's algorithm:**

In the DSS-5.1a key generation algorithm, the private key x is not selected directly; instead, it is generated indirectly from a secret generator element α. This element α is never publicly disclosed, meaning that an attacker has no knowledge of the generator needed to apply Shor's algorithm, which relies on finding periodicity based on a known base.

Moreover, the public key y is computed through a nonlinear function, deliberately avoiding direct use of the expression $x^x \ mod \ p$, as that could expose the structure of the underlying hard problem. However, the nonlinear nature of the key generation function is still preserved, and there exists no clear periodic mapping that Shor's algorithm could exploit.

- **On resistance to Grover's algorithm:**

During the signature generation process (algorithm DSS-5.1b), a random number $k \in (1, q)$ is selected for each signing session. Additionally, the message is hashed using a cryptographic hash function $H(M)$, with the hash value hhh typically residing in a large space (e.g., 256 bits or more).

Grover's algorithm only reduces the number of brute-force steps over a key space from 2^n to $2^{n/2}$. Therefore, to maintain a security level equivalent to NIST standards (e.g., NIST Level 1, which corresponds to 128 bit security), the scheme must use parameters such as p, p, k, and h with a minimum length of 256 bits. The current design of DSS-5.1 satisfies this requirement.

In conclusion, the DSS-5.1 scheme resists Shor's algorithm by keeping the generator element secret and using a nonlinear key generation function. It also resists Grover's algorithm through sufficiently large parameter sizes and per-signature randomness. This ensures a high level of security in both classical and quantum environments.

b) Resistance to shor and grover of the dss-5.2 scheme

The DSS-5.2 scheme is constructed based on Type 3 of the newly proposed hard problems, a two-dimensional cross exponentiation system, in which the two unknownsx_1, x_2 appear simultaneously in both the base and the exponent. Therefore, Type 3 is considered a strongly nonlinear system of equations, lacking a clear group structure as in DLP, and also lacking the periodicity required by Shor's quantum algorithm.

- **Regarding resistance to Shor's algorithm:**

To solve the DLP or ECDLP, Shor's algorithm fundamentally requires the ability to construct a quantum oracle that performs a transformation such as: $| \, r\rangle \mapsto | \, g^r \ mod p\rangle$ in a finite field, or $| \, r\rangle \mapsto | \, r \times G\rangle$ on an elliptic curve.

However, in the proposed scheme over a finite field, the generator g is randomly generated and kept secret within the private key; the only public value is $y = x^x \ mod \ p$. This prevents the attacker from constructing the oracle $f(r) = g^r$, since g is unknown.

As a result, the scheme completely nullifies the applicability of Shor's algorithm, not only in theory, but also at the level of each concrete step in the quantum procedure.

- **Regarding resistance to Grover's algorithm:**

Grover's algorithm accelerates brute-force attacks by reducing the complexity from $O(2^n)$ to $O(2^{n/2})$. However, the proposed scheme requires a minimum private key length of$n \geq 256$, which means the attack cost remains $O(2^{128})$, still infeasible even for powerful quantum computers.

Moreover, in the signature generation process (algorithm DSS-5.2b), a valid signature depends not only on a single pair (r, s), but must simultaneously satisfy two independent verification equations, each involving distinct keys and generators. This forces an attacker to solve two separate hard problems concurrently, significantly expanding the search space and thus drastically reducing Grover's effectiveness.

The resistance of the scheme against Shor and Grover attacks stems not only from the hardness of the underlying problem but also from the algorithmic design itself.

The public key conceals the generator element; The signing and verification processes are decentralized;

No quantum oracle can be constructed for Shor's algorithm;

The complex search space significantly reduces the effectiveness of Grover's algorithm. As a result, the scheme offers strong post-quantum security while maintaining high performance and ease of integration into existing PKI infrastructures without major modifications.

VI. Discussion

The proposed scheme introduces a novel approach that does not rely on lattice-based, hash-based, or code-based constructions like Dilithium, Falcon, or SPHINCS+. By leveraging a non-standard hard problem, it provides resistance against both Shor's and Grover's algorithms while maintaining compatibility with existing PKI infrastructures. With key and signature sizes of reasonable magnitude, the scheme is easy to implement and facilitates a smoother transition to the post-quantum cryptographic environment. However, a more comprehensive evaluation is still required to fully assess its practical feasibility and security.

- Computational Cost Comparison between Proposed Schemes, DSA, and SPHINCS+: Table I provides a detailed comparison of computational costs among the traditional signature scheme (DSA), the proposed schemes (DSS-5.1 and DSS-5.2), and a representative post-quantum scheme from the hash-based family (SPHINCS+). The evaluated operations include: the number of large modular exponentiations (N.exp), modular multiplications (N.mul), modular inversions (N.inv), and hash function calls (N.h). This analysis offers a theoretical performance assessment of each scheme during the signing and signature verification phases.

TABLE I. Comparison of Computational Costs Between The Proposed Schemes, Dsa, and Sphincs+

Operation	DSA Sign	DSA Verify	DSS-5.1 Sign	DSS-5.1 Verify	DSS-5.2 Sign	DSS-5.2 Verify	SPHINCS+ Sign [27]	SPHINCS+ Verify [27]
N.exp	2	3	3	4	4	3	–	–
N.mul	3	2	5	3	6	4	–	–
N.inv	1	1	1	2	0	1	–	–
N.h	1	1	1	2	1	1	~65,000 – 200,000	~25,000 – 60,000

It can be seen that the two proposed schemes, DSS-5.1 and DSS-5.2, incur slightly higher computational costs than DSA/DCDSA, but remain feasible for practical deployment owing to the use of simple modular operations and the absence of complex structures such as those in hash-based or lattice-based schemes. In contrast, SPHINCS+ [27] achieves strong quantum-resistant security but incurs extremely high computational costs because it requires tens of thousands of hash operations per signature, rendering it less suitable for resource-constrained environments, including IoT and high-speed blockchain applications. Therefore, the proposed schemes strike a balanced trade-off between efficiency, security, and compatibility with current PKI infrastructures.

- Performance and Scalability: The proposed schemes achieve high performance by relying solely on modular multiplication and exponentiation, which are easily optimized in both software and hardware implementations. The public key is compatible with current PKI infrastructures, and the compact signature size is well-suited for blockchain, IoT, and smart cards. However, further experimental evaluation and comparison with other PQC schemes, especially on FPGA/ASIC platforms, are necessary.
- Resistance to Attacks: Both schemes are resistant to Shor's and Grover's algorithms as well as classical attacks, thanks to the use of a newly designed problem where the generator is kept secret, disrupting the underlying group structure crucial for solving discrete logarithms. Furthermore, brute-force attacks would require testing up to 2^{2n} cases (with $n \geq 256$), rendering them practically infeasible. These schemes are also immune to structure-specific attacks common in lattice-based cryptography (e.g., Dilithium). Nevertheless, further evaluation is needed regarding side-channel resistance, especially against timing and power analysis, and countermeasures such as masking or blinding should be considered to better protect the private key.
- Limitations and Future Work: Despite showing quantum resistance and PKI compatibility, the proposed schemes have yet to be experimentally verified across various platforms and remain unstandardized, unlike current PQC candidates such as Dilithium and Falcon. Future directions include algorithmic optimization, real-world implementation on PKI, blockchain, and IoT systems, and expansion to group signatures. Once fully validated, these schemes could offer a practical and flexible post-quantum signature solution.

VII. Conclusion

In this paper, we propose two post-quantum digital signature schemes, denoted as DSS-5.1 and DSS-5.2, based on two different forms of the same newly proposed hard problem over a prime finite field. Although they share the same mathematical foundation, these schemes adopt signature generation and verification algorithms which differ from those of previous schemes. This demonstrates that the newly proposed hard problem can serve as a foundation for various types of digital signature schemes with different structures, all aimed at the same goals: being secure against both classical and quantum attacks, and remaining compatible with current PKI infrastructures.

Theoretical analysis shows that both schemes are resistant to Shor's algorithm owing to their nonlinear structure and the concealment of the generator parameter. At the same time, the impact of Grover's algorithm is mitigated by appropriately choosing sufficiently large parameters. Compared to DSA and DCDSA, although the proposed schemes entail slightly higher computational costs, they avoid certain computationally expensive operations, such as the modular inverse in the signing phase (as in DSS-5.2), while achieving higher levels of security that are well suited for the quantum era.

These results pave the way for developing digital signature schemes that do not rely on existing post-quantum primitives but instead leverage newly proposed mathematical structures to achieve both efficiency and long-term security at the same time. In the future, practical implementations should be explored on specific platforms, such as blockchain and IoT devices, as well as further investigations into side-channel resistance, in order to provide a more comprehensive solution.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Tuan Nguyen Kim: Proposes a new hard problem as the foundation for constructing digital signature schemes and prepares the manuscript; Luu Hong Dung: Develops digital signature schemes based on the proposed new hard problem; Hoang Duc Tho: Proves the correctness of the proposed schemes; Ha Nguyen Hoang: Tests and evaluates the resistance of the proposed schemes against attacks; All authors had approved the final version.

References

[1] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120−126, Feb. 1978.

[2] T. E. Gamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” *IEEE Transactions on Information Theory*, vol. IT-31, no. 4, pp. 469−472, Jul. 1985.

[3] D. Johnson, A. Menezes, and S. Vanstone, “The elliptic curve digital signature algorithm (ECDSA),” *International Journal of Information Security*, vol. 1, no. 1, pp. 36−63, Aug. 2001.

[4] R. Housley, W. Polk, W. Ford, and D. Solo, “Internet X.509 public key infrastructure certificate and Certificate Revocation List (CRL) profile,” *IETF RFC 5280*, May 2008.

[5] J. P. D. Anvers, A. Karmakar, S. S. Roy, F. Vercauteren, and I. Verbauwhede, “Crystals-Dilithium: A lattice-based digital signature scheme,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, pp. 238−268, 2018.

[6] P. L. Schwabe, G. Cassiers, and F. Ducas, “Falcon: Fast-fourier lattice-based compact signatures over NTRU,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2020, no. 3, pp. 1−24, 2020.

[7] E.O. Kiktenko, A.A. Bulychev, P.A. Karagodin, N.O. Pozhar, M.N. Anufriev, and A.K. Fedorov, “SPHINCS+ post-quantum digital signature scheme with Streebog hash function,” in *Proc. AIP Conference Proceedings*, 2020, vol. 2241, no. 1, 020014. .

[8] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. Schanck, P. Schwabe, and D. Stehlé, “CRYSTALS-kyber: A CCA-secure module-lattice-Based KEM,” in *Proc. 2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, London, UK, 2018, pp. 353−367.

[9] A. Ali, M. A. H. Farquad, C. Atheeq, and C. Altaf, “A quantum encryption algorithm based on the rail fence mechanism to provide data integrity,” *Engineering, Technology and Applied Science Research*, vol. 14, iss. 6, pp. 18818−18823, 2024.

[10] D. Y. Guryanov, D. N. Moldovyan, and A. A. Moldovyan, “Post-quantum digital signature schemes: setting a hidden group with two-dimensional cyclicity,” *Informatization and Communication*, vol. 4, pp. 75−82, 2020.

[11] A. A. Moldovyan, N. A. Moldovyan, D. N. Moldovyan, A. A. Kostina, “A new approach to the development of digital signature algorithms based on the hidden discrete logarithm problem,” in *Proc the 7th International Conference on FICTA*, 2019, pp. 1−12.

[12] N. K. Tuan, N. H. Ha, H. N. Duy, “A new solution to enhance security in building digital signature schemes,” *Engineering, Technology and Applied Science Research*, vol. 15, no. 3, pp. 23613−23621, 2025.

[13] C. Battarbee *et al.*, “SPDH-sign: towards efficient, post-quantum group-based signatures,” in *Proc. International Conference on Post-Quantum Cryptography. Cham: Springer Nature Switzerland*, 2023, pp. 113−138.

[14] T. E. Gamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469−472, July 1985.

[15] P. W. Shor, “Algorithms for quantum computation: Discrete logarithms and factoring,” in *Proc. 35th Annual Symposium on Foundations of Computer Science (FOCS)*, Santa Fe, NM, USA, 1994, pp. 124−134.

[16] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *Proc. 28th Annual ACM Symposium on Theory of Computing (STOC)*, Philadelphia, PA, USA, 1996, pp. 212−219.

[17] Z. Gong *et al.*, “A survey on lattice-based digital signature,” *Cybersecurity*, vol. 7, Apr. 2024.

[18] C. Wolf, “Multivariate quadratic polynomials in public key cryptography,” Ph.D. thesis, Katholieke Universiteit Leuven, Nov. 2005.

[19] N. T. Courtois, M. Finiasz, and N. Sendrier, “How to achieve a McEliece-based digital signature scheme,” in *Proc. International Conference on the Theory and Application of Cryptology and Information Security*, 2001, pp. 157−174.

[20] A. Karakaya and A. Ulu, “A survey on post-quantum based approaches for edge computing security,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 16, no. 1, 2024.

[21] D. J. Bernstein *et al.*, “SPHINCS+ submission to the NIST post-quantum project,” *Submission to NIST*, 2017.

[22] A. Hülsing, J. Rijneveld, and J. Schwabe, “Qtesla: An efficient post-quantum signature scheme based on the ring learning with errors problem,” *Post-Quantum Cryptography*, pp. 1−17, 2017.

[23] N. Koblitz and A. Menezes, “A survey of public-key Cryptosystems,” *International Journal of Information Security*, vol. 1, no. 1, pp. 5−23, 2001.

[24] P. V. Tuan, L. T. Anh, and D. T. Linh, “A new digital signature scheme based on nonlinear modular inverse problem,” *Vietnam Journal of Computer Science*, vol. 6, no. 2, pp. 125−137, 2023.

[25] A. M. Shuaibu, S. Babuba, and V. J. Andrawus, “A survey of discrete logarithm algorithms in finite fields,” *Dutse Journal of Pure and Applied Sciences*, vol. 5, pp. 215−224, Jun. 2019.

[26] D. Hankerson and A. Menezes, “Elliptic curve discrete logarithm problem,” *Encyclopedia of Cryptography and Security*, pp. 186−189, 2025.

[27] T. G. Kang *et al.*, “On the performance analysis of SPHINCS+ verification,” *IEICE Trans. Inf. and Syst.*, no. 12, 2019.