Cryptanalysis of MPPK/DS

Hao Guo[1], Ray A. Perlner[2], and Jintai Ding[3,*]

[1]Tsinghua University, Beijing, China
[2]National Institute of Standards and Technology, United States
[3]Yau Mathematical Sciences Center, Tsinghua University, Beijing, China
[*]jintai.ding@gmail.com

ABSTRACT

In 2022, Kuang and Perepechaenko introduced a new digital signature scheme called MPPK/DS, claiming it to be a quantum-safe multivariate scheme. In this paper, we conduct a cryptanalysis of MPPK/DS and present a method to forge signatures for arbitrary chosen messages. Our analysis reveals that the security of MPPK/DS can be compromised through the solution of a system of linear modular congruences. By using only the free online Magma Calculator, we successfully forge signatures in under 0.03 seconds for their level 1-5 parameters, indicating a significant vulnerability in the scheme's design.

1 Introduction

In the past 40 years, there have been significant advancements in the field of public-key cryptography, driven by the need for secure communication over untrusted networks. Unlike traditional symmetric cryptography, which relies on pre-established secret key exchange through trusted channels, public-key cryptography employs a key pair consisting of a public key and a secret key. The public key can be openly shared on the Internet, enabling encrypted messages to be sent by others without being vulnerable to potential adversaries.

Popular public-key cryptographic algorithms such as RSA, DSA, and Diffie-Hellman Key Exchange have formed the backbone of modern communication. However, in 1994, Peter Shor introduced Shor's Algorithm[1], which can efficiently factor large integers and solve discrete logarithm problems using quantum computers. With the advent of quantum computers and their potential threat to current cryptographic systems, there is an urgent need to transition to post-quantum public-key cryptosystems.

In response to this concern, NIST initiated a post-quantum cryptography standardization effort in 2016, aiming to replace vulnerable algorithms like RSA, DSA, and elliptic curve cryptosystems with quantum-resistant alternatives. After three rounds of submission, NIST announced the first four selected algorithms for standardization[2], three of which are lattice-based (Dilithium, KYBER, FALCON), and one is hash-based (SPHINCS+). Additionally, NIST called for proposals of additional digital signature schemes, with a preference for general-purpose signature schemes not based on structured lattices.

Another significant type of post-quantum cryptography is multivariate cryptosystems, alongside lattice-based and hash-based cryptosystems. Multivariate public-key cryptosystems are constructed through a bipolar approach[3], involving a system $\mathscr{F}: \mathbb{F}^n \to \mathbb{F}^m$ of m multivariate quadratic polynomials in n variables (the central map), which can be easily inverted. Two linear (affine) invertible maps $\mathscr{S}: \mathbb{F}^m \to \mathbb{F}^m$ and $\mathscr{T}: \mathbb{F}^n \to \mathbb{F}^n$ are used to mix the variables and polynomials, resulting in a quadratic public key $\mathscr{P} = \mathscr{S} \circ \mathscr{F} \circ \mathscr{T}: \mathbb{F}^n \to \mathbb{F}^m$ (see Figure 1). This public key is challenging to distinguish from a randomly generated quadratic map.

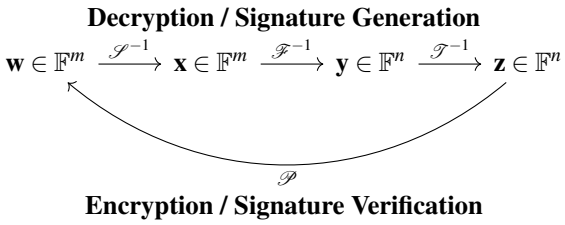

Figure 1. General workflow of bipolar schemes

The security of multivariate cryptosystems relies on the hardness of solving polynomial systems over finite fields. For instance, when the base field is GF(2), solving quadratic systems was proved by Fraenkel and Yesha[4] to be NP-complete. Since higher degree polynomials can be transformed into quadratic polynomials by adding more variables, it is sufficient to solve

quadratic systems over finite fields, known as the MQ (Multivariate Quadratic) problem. The MQ problem has even been utilized in the design of hash functions and stream ciphers, such as MQ-HASH[5] and QUAD[6].

One of the early multivariate public-key cryptosystems is the Matsumoto-Imai cryptosystem[7]. It constructs the central map as an easily invertible map with low Hamming weight terms over the extension field. By utilizing the isomorphism between the extension field and the base field, the central map is transformed into a quadratic map in a general-like form. However, Patarin[8] fully attacked the Matsumoto-Imai system in 1995, leading to the development of the Oil and Vinegar cryptosystem. These two systems represent the two main types of multivariate public-key cryptosystems to date: BigField type and UOV type.

BigField type cryptosystems encompass Hidden Fields Equation (HFE)[9] and its numerous variants (e.g., GeMSS[10], which was attacked by Tao et al.[11]). On the other hand, UOV type cryptosystems were originally represented by Rainbow[12], a Round 3 finalist that was later attacked by Beullens[13]. Currently, UOV type remains an active research topic with various UOV-like cryptosystems such as QR-UOV[14], MAYO[15], and NOVA[16].

In the present day, multivariate cryptosystems are often employed as digital signatures because the solution of a polynomial system is easy to verify and transfer. Despite the disadvantage of having a large public key size, the need for repetitive transmissions is usually not required, and the public key size does not significantly impact the signature's efficiency.

Regarding MPPK/DS, it was first introduced in[17] and later in[18]. The authors claimed to have proposed a new quantum-safe digital signature algorithm known as the Multivariate Polynomial Public Key Digital Signature. Their algorithm is based on Fermat's Little Theorem, which states that if p is a prime number and a is an integer not divisible by p, then $a^{p-1} \equiv 1 \pmod{p}$. As a result, the exponent can be viewed as an element in $\mathbb{Z}/\varphi(p)\mathbb{Z}$. They utilize this property by taking polynomial powers of nontrivial elements in $\mathrm{GF}(p)$, allowing for pure exponent addition/subtraction on $\mathbb{Z}/\varphi(p)\mathbb{Z}$ after fixing a generator g of $\mathrm{GF}(p)^\times$. The variables in their polynomials are divided into the message variable X_0 and the noise variables $X_1,\ldots,X_m$. For a chosen message $x_0 \in \mathbb{Z}/\varphi(p)\mathbb{Z}$, the signature must satisfy certain polynomial equations over $X_1,\ldots,X_m$, enabling arbitrary noise $x_1,\ldots,x_m$ to be plugged in for verification. As a result, forging a signature boils down to finding suitable coefficients for monomials in $X_1,\ldots,X_m$.

In this article, we present an analysis of the MPPK/DS digital signature scheme, with a focus on its vulnerabilities in the key generation process and the system of linear modular congruence equations. Moreover, we provide insights into the forging process, outlining the steps to forge a signature in the scheme. Through a toy example, we illustrate how these weaknesses can be exploited to forge signatures successfully. We believe that MPPK/DS is not a traditional Multivariate scheme, given the identified vulnerabilities and its departure from the conventional approach. Experimental results validate these vulnerabilities across different security levels. Finally, we emphasize the importance of refining the MPPK/DS scheme and fostering collaboration within the cryptography community to enhance cryptographic security.

2 Cryptanalysis

2.1 MPPK/DS

This subsection describes the MPPK/DS digital signature scheme, partially following the notation used in[18]. MPPK/DS utilizes an odd prime p of the form $p = 2^x \cdot q + 1$ where q is also an odd prime and 2^x and q have the same length in binary form. It is well-known that when p is prime, $\mathrm{GF}(p)^\times$, the collection of units of $\mathrm{GF}(p)$ (which is just $\mathrm{GF}(p) \setminus \{0\}$), forms a cyclic group of order $\varphi(p) = p-1 = 2^x \cdot q$, where $\varphi(p)$ is Euler's totient function for p. We denote $R = \mathbb{Z}/\varphi(p)\mathbb{Z}$.

The public key consists of four polynomials over $R[X_0;X_1,\ldots,X_m]$: $\mathrm{N_{const}}$, $\mathrm{N_{lead}}$, P, Q, where $\mathrm{N_{const}}$ has X_0-degree 0. The signature for a message $x_0 \in R$ will be denoted as $\sigma = (x_0,A,B,C,D)$ where $A,B,C,D \in \mathrm{GF}(p)$ are numbers to be determined later.

2.1.1 Key Generation

The signer performs the following steps to generate the public and private keys:

1. Choose two univariate polynomial f and h in $R[X_0]$ with degree λ, i.e.,

$$f(X_0) = \sum_{k=0}^{\lambda} f_k X_0^k \tag{1}$$

$$h(X_0) = \sum_{k=0}^{\lambda} h_k X_0^k \tag{2}$$

2. Select a multivariate polynomial $\mathrm{B} \in R[X_0;X_1,\ldots,X_m]$ which is linear in $X_1,\ldots,X_m$ and degree n in X_0, namely

$$\mathrm{B}(X_0;X_1,\ldots,X_m) = \sum_{k=0}^{n} \mathrm{B_k}(X_1,\ldots,X_m) X_0^k \tag{3}$$

3. Compute the polynomials $\Phi(X_0;X_1,\ldots,X_m)$ and $\Psi(X_0;X_1,\ldots,X_m)$ as follows:

$$\Phi(X_0;X_1,\ldots,X_m) = f(X_0)\mathrm{B}(X_0;X_1,\ldots,X_m) - f_0\mathrm{B}_0(X_1,\ldots,X_m) - f_\lambda \mathrm{B}_\mathrm{n}(X_1,\ldots,X_m) \tag{4}$$
$$\Psi(X_0;X_1,\ldots,X_m) = h(X_0)\mathrm{B}(X_0;X_1,\ldots,X_m) - h_0\mathrm{B}_0(X_1,\ldots,X_m) - h_\lambda \mathrm{B}_\mathrm{n}(X_1,\ldots,X_m) \tag{5}$$

4. Generate noise polynomials

$$\mathrm{N}_\mathrm{const}(X_1,\ldots,X_m) = R_{const}\mathrm{B}_0(X_1,\ldots,X_m) \tag{6}$$
$$\mathrm{N}_\mathrm{lead}(X_0;X_1,\ldots,X_m) = R_{lead}\mathrm{B}_\mathrm{n}(X_1,\ldots,X_m)X_0^{n+\lambda} \tag{7}$$

for some randomly chosen $R_{const}, R_{lead} \in R$.

5. Randomly choose invertible elements $\alpha, \beta \in R^\times$.

6. Compute the public key polynomials as:

$$\mathrm{P}(X_0;X_1,\ldots,X_m) = \alpha R_{const}\Phi(X_0;X_1,\ldots,X_m) \tag{8}$$
$$\mathrm{Q}(X_0;X_1,\ldots,X_m) = \beta R_{lead}\Psi(X_0;X_1,\ldots,X_m) \tag{9}$$

The public key is then composed of P, Q, $\mathrm{N}_\mathrm{const}$, and N_lead, while the private key consists of f, h, R_{const}, R_{lead}, α, and β. The polynomial B can be discarded after key generation.

2.1.2 Signing Process

To sign a message x_0 (or its hash), the signer performs the following steps using the private key:

1. Calculate the values $a(x_0)$, $b(x_0)$, $c(x_0)$, and $d(x_0)$ as follows:

$$a(x_0) = \beta^{-1}R_{const}f(x_0) \tag{10}$$
$$b(x_0) = \alpha^{-1}R_{lead}h(x_0) \tag{11}$$
$$c(x_0) = R_{lead}[h(x_0)f_0 - f(x_0)h_0] \tag{12}$$
$$d(x_0) = R_{const}[h(x_0)f_\lambda - f(x_0)h_\lambda] \tag{13}$$

2. Randomly choose a generator $g \in \mathrm{GF}(p)^\times$.

3. Calculate the signature components A, B, C, and D as:

$$A = g^{a(x_0)} \bmod p \tag{14}$$
$$B = g^{b(x_0)} \bmod p \tag{15}$$
$$C = g^{c(x_0)} \bmod p \tag{16}$$
$$D = g^{d(x_0)} \bmod p \tag{17}$$

4. The signature σ is then presented as $\sigma = (x_0, A, B, C, D)$.

2.1.3 Verification

To verify the signature σ for a message x_0, the verifier performs the following steps:

1. Randomly choose values $x_1,\ldots,x_m \in R$.

2. Use the public key to check if the equation (18) holds:

$$A^{\mathrm{Q}(x_0;x_1,\ldots,x_m)} \equiv B^{\mathrm{P}(x_0;x_1,\ldots,x_m)}C^{\mathrm{N}_\mathrm{const}(x_1,\ldots,x_m)}D^{\mathrm{N}_\mathrm{lead}(x_0;x_1,\ldots,x_m)} \pmod p \tag{18}$$

3. If equation (18) holds for several trials with different values of $x_1,\ldots,x_m$, the verifier considers the signature to be valid.

2.2 Conversion from Forging Signature to Solving System of Linear Modular Congruences

MPPK/DS has a notable vulnerability that allows forging signatures by solving a system of linear modular congruence equations. This vulnerability arises due to the multilinear property of B, which extends to P, Q, $\mathrm{N}_{\mathrm{const}}$, and $\mathrm{N}_{\mathrm{lead}}$.

Given a public key $(\mathrm{P},\mathrm{Q},\mathrm{N}_{\mathrm{const}},\mathrm{N}_{\mathrm{lead}})$ and a target message x_0, the attacker wants to forge a signature (x_0,A,B,C,D) for x_0.

Let $R=\mathbb{Z}/\varphi(p)\mathbb{Z}$. Since B is multilinear in $X_1,\ldots,X_m$, the public key polynomials can be represented as linear combinations of constant values for a fixed message $x_0 \in R$:

$$\mathrm{P}(x_0;X_1,\ldots,X_m)=\sum_{k=1}^{m}\mathrm{P_k}(x_0)X_k \tag{19}$$

$$\mathrm{Q}(x_0;X_1,\ldots,X_m)=\sum_{k=1}^{m}\mathrm{Q_k}(x_0)X_k \tag{20}$$

$$\mathrm{N}_{\mathrm{const}}(x_0;X_1,\ldots,X_m)=\sum_{k=1}^{m}\mathrm{N}_{\mathrm{const,k}}(x_0)X_k \tag{21}$$

$$\mathrm{N}_{\mathrm{lead}}(x_0;X_1,\ldots,X_m)=\sum_{k=1}^{m}\mathrm{N}_{\mathrm{lead,k}}(x_0)X_k \tag{22}$$

where $\mathrm{Q_k}(x_0)$, $\mathrm{P_k}(x_0)$, $\mathrm{N}_{\mathrm{const,k}}(x_0)$, and $\mathrm{N}_{\mathrm{lead,k}}(x_0)$ are constant values for a fixed message x_0.

To forge a signature for a given message x_0, the attacker needs to find a common solution in R of the following system of linear modular congruence equations:

$$V_1\mathrm{P_k}(x_0)-V_2\mathrm{Q_k}(x_0)-V_3\mathrm{N}_{\mathrm{const,k}}(x_0)-V_4\mathrm{N}_{\mathrm{lead,k}}(x_0)=0 \tag{23}$$

for all $1\le k\le m$.

If the attacker successfully finds a suitable set of $V_1,V_2,V_3,V_4 \in R$ that satisfy the system of equations, they can then forge a signature for the target message x_0. The forged signature will be (x_0,A,B,C,D), where:

$$A=g^{V_1} \bmod p \tag{24}$$

$$B=g^{V_2} \bmod p \tag{25}$$

$$C=g^{V_3} \bmod p \tag{26}$$

$$D=g^{V_4} \bmod p \tag{27}$$

where g is a randomly chosen generator of $\mathrm{GF}(p)^{\times}$.

3 Experiment Results and Toy CounterExamples

We used the free online Magma calculator[19] to simulate the key generation, signing process, forging process, and verification for various security levels (1-5) provided by the authors $((\log_2 q,x,\log_2 p,n,\lambda,m)=(32,32,64,6,3,2))$. The results were obtained in less than 0.5 seconds, which demonstrates the weakness of this digital signature scheme.

To illustrate the vulnerability in MPPK/DS, let's consider the following example based on the public key provided by[18]:

Public key:

$$\mathrm{P}(X_0;X_1,X_2)=(152X_0+318X_0^2+234X_0^3)X_1+(140X_0+344X_0^2+216X_0^3)X_2 \tag{28}$$

$$\mathrm{Q}(X_0;X_1,X_2)=(48X_0+240X_0^2+340X_0^3)X_1+(232X_0+248X_0^2+96X_0^3)X_2 \tag{29}$$

$$\mathrm{N}_{\mathrm{const}}(X_1,X_2)=248X_1+204X_2 \tag{30}$$

$$\mathrm{N}_{\mathrm{lead}}(X_0;X_1,X_2)=(140X_1+336X_2)X_0^4 \tag{31}$$

Let's suppose we want to forge a signature for $x_0=48$. The attacker aims to find suitable values $V_1,V_2,V_3,V_4 \in R$ that satisfy the following system of linear modular congruence equations:

$$\begin{cases} 96V_1-256V_2-248V_3-288V_4=0 \pmod{352} \\ 128V_1-0V_2-204V_3-128V_4=0 \pmod{352} \end{cases} \tag{32}$$

Magma reveals that $(V_1, V_2, V_3, V_4) = (51, 283, 200, 186)$ is a solution to the system of equations. With this solution, the attacker can forge a signature for $x_0 = 48$ as follows:

$$A = g^{51} \bmod 353 \tag{33}$$

$$B = g^{283} \bmod 353 \tag{34}$$

$$C = g^{200} \bmod 353 \tag{35}$$

$$D = g^{186} \bmod 353 \tag{36}$$

where $g = 3$ is a randomly chosen generator of $\mathrm{GF}(353)^{\times}$.

The forged signature is $(48, A, B, C, D)$ with $A = 112$, $B = 316$, $C = 88$, and $D = 255$.

Now, let's see how the verifier verifies the forged signature for $x_0 = 48$ twice. The verifier randomly chooses two different values $x_1, \ldots, x_m = (51, 121)$ and $x_1, \ldots, x_m = (259, 324)$ in R and uses the public key to check if the following equation holds:

$$A^{\mathrm{Q}(48;X_1,X_2)} \equiv B^{\mathrm{P}(48;X_1,X_2)} C^{\mathrm{N_{const}}(X_1,X_2)} D^{\mathrm{N_{lead}}(48;X_1,X_2)} \pmod{353} \tag{37}$$

For Trial 1, when $(x_1, \ldots, x_m) = (51, 121)$, $\mathrm{P} = 32$, $\mathrm{Q} = 320$, $\mathrm{N_{const}} = 20$, $\mathrm{N_{lead}} = 256$, and:

$$112^{320} = 337 = 256 \times 337 \times 131 = 316^{32} \times 88^{20} \times 255^{256}$$

For Trial 2, when $(x_1, \ldots, x_m) = (259, 324)$, $\mathrm{P} = 128$, $\mathrm{Q} = 160$, $\mathrm{N_{const}} = 88$, $\mathrm{N_{lead}} = 256$, and:

$$112^{160} = 185 = 58 \times 1 \times 131 = 316^{128} \times 88^{88} \times 255^{256}$$

Since the equation (37) holds for both trials, the verifier mistakenly believes that the signature $(48, 112, 316, 88, 255)$ is valid for $x_0 = 48$ in both cases, and the forged signature successfully deceives the verification process. The signature given by[18] is $(48, 262, 187, 22, 159)$.

4 Conclusion

In this study, we conducted a comprehensive analysis of the MPPK/DS digital signature scheme and revealed its susceptibility to forgery attacks. By carefully examining the key generation process and the underlying system of linear modular congruence equations, we identified critical vulnerabilities that allow an adversary to forge signatures for arbitrary messages.

The results from our illustrative toy example demonstrated how an attacker can exploit these weaknesses to forge a signature for a specific message $x_0 = 48$. Through the solution of the system of linear modular congruence equations, the attacker successfully obtained a forged signature that deceives the verification process. This example highlights the significance of robust security evaluations for cryptographic schemes to ensure their resilience against potential attacks.

Our experimental results, generated using only the free online Magma calculator, further emphasized the weaknesses in the MPPK/DS digital signature scheme across various security levels. These findings underscore the urgency for improvements and the need for more secure cryptographic designs.

We believe that the MPPK/DS scheme does not follow the traditional multivariate scheme approach, which further raises concerns about its overall security. The vulnerabilities identified in this study call for further research and refinement to enhance the robustness and reliability of the MPPK/DS scheme.

Response from the MPPK/DS team

We have contacted the authors of MPPK/DS, and they have acknowledged our attack and expressed their intention to make improvements to address it.

Disclaimer

Certain commercial products or company names are identified here to describe our study adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the products or names identified are necessarily the best available for the purpose.

Data availability

All data generated or analysed during this study are included in this published article and its supplementary information files.

Acknowledgement

The contribution of Hao Guo is supported by National Key R&D Program of China (No. 2021YFB3100100). The contribution of Jintai Ding is supported by Beijing Natural Science Foundation (No. M22001).

References

1. Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review* **41**, 303–332 (1999).
2. Alagic, G. *et al.* Status report on the third round of the nist post-quantum cryptography standardization process. *US Dep. Commer. NIST* (2022).
3. Ding, J., Petzoldt, A. & Schmidt, D. S. *Multivariate Public Key Cryptosystems*, vol. 80 (Springer Nature, 2020).
4. Fraenkel, A. S. & Yesha, Y. Complexity of problems in games, graphs and algebraic equations. *Discret. Appl. Math.* **1**, 15–30 (1979).
5. Billet, O., Robshaw, M. J. & Peyrin, T. On building hash functions from multivariate quadratic equations. In *Australasian Conference on Information Security and Privacy*, 82–95 (Springer, 2007).
6. Berbain, C., Gilbert, H. & Patarin, J. Quad: A multivariate stream cipher with provable security. *J. Symb. Comput.* **44**, 1703–1723 (2009).
7. Matsumoto, T. & Imai, H. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In *Advances in Cryptology—EUROCRYPT'88: Workshop on the Theory and Application of Cryptographic Techniques Davos, Switzerland, May 25–27, 1988 Proceedings 7*, 419–453 (Springer, 1988).
8. Patarin, J. Cryptanalysis of the matsumoto and imai public key scheme of eurocrypt'88. In *Advances in Cryptology—CRYPT0'95: 15th Annual International Cryptology Conference Santa Barbara, California, USA, August 27–31, 1995 Proceedings 15*, 248–261 (Springer, 1995).
9. Patarin, J. Hidden fields equations (hfe) and isomorphisms of polynomials (ip): Two new families of asymmetric algorithms. In *International Conference on the Theory and Applications of Cryptographic Techniques*, 33–48 (Springer, 1996).
10. Casanova, A. *et al. GeMSS: a great multivariate short signature*. Ph.D. thesis, UPMC-Paris 6 Sorbonne Universités; INRIA Paris Research Centre, MAMBA Team ... (2017).
11. Tao, C., Petzoldt, A. & Ding, J. Efficient key recovery for all hfe signature variants. In *Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part I 41*, 70–93 (Springer, 2021).
12. Ding, J. & Schmidt, D. Rainbow, a new multivariable polynomial signature scheme. In *International conference on applied cryptography and network security*, 164–175 (Springer, 2005).
13. Beullens, W. Breaking rainbow takes a weekend on a laptop. In *Annual International Cryptology Conference*, 464–479 (Springer, 2022).
14. Furue, H., Ikematsu, Y., Kiyomura, Y. & Takagi, T. A new variant of unbalanced oil and vinegar using quotient ring: Qr-uov. In *Advances in Cryptology–ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6–10, 2021, Proceedings, Part IV 27*, 187–217 (Springer, 2021).
15. Beullens, W. Mayo: practical post-quantum signatures from oil-and-vinegar maps. In *International Conference on Selected Areas in Cryptography*, 355–376 (Springer, 2021).
16. Wang, L.-C., Tseng, P.-E., Kuan, Y.-L. & Chou, C.-Y. Nova, a noncommutative-ring based unbalanced oil and vinegar signature scheme with key-randomness alignment. Cryptology ePrint Archive, Paper 2022/665 (2022). https://eprint.iacr.org/2022/665.
17. Kuang, R., Perepechaenko, M. & Barbeau, M. A new quantum-safe multivariate polynomial public key digital signature algorithm. *Sci. Reports* **12**, 13168 (2022).
18. Kuang, R. & Perepechaenko, M. Optimization of the multivariate polynomial public key for quantum safe digital signature. *Sci. Reports* **13**, 6363, DOI: 10.1038/s41598-023-32461-3 (2023).
19. Bosma, W., Cannon, J. & Playoust, C. The Magma algebra system. I. The user language. *J. Symb. Comput.* **24**, 235–265, DOI: 10.1006/jsco.1996.0125 (1997). Computational algebra and number theory (London, 1993).