On McEliece-Type Cryptosystems Using Self-Dual Codes With Large Minimum Weight

Mariot, Luca; Picek, Stjepan; Yorgova, Radinka

DOI
10.1109/ACCESS.2023.3271767

Publication date
2023

Document Version
Final published version

Published in
IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 10 February 2023, accepted 23 April 2023, date of publication 1 May 2023, date of current version 9 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3271767

RESEARCH ARTICLE

On McEliece-Type Cryptosystems Using Self-Dual Codes With Large Minimum Weight

LUCA MARIOT[1], STJEPAN PICEK[2,3], AND RADINKA YORGOVA[2]
[1]Semantics, Cybersecurity & Services Group, University of Twente, 7522 NB Enschede, The Netherlands
[2]Department of Intelligent Systems, Delft University of Technology, 2628 CD Delft, The Netherlands
[3]Digital Security Group, Radboud University, 6525 XZ Nijmegen, The Netherlands

Corresponding author: Stjepan Picek (s.picek@tudelft.nl)

ABSTRACT One of the Round 3 Finalists in the NIST post-quantum cryptography call is the Classic McEliece cryptosystem. Although it is one of the most secure cryptosystems, the large size of its public key remains a practical limitation. In this work, we propose a McEliece-type cryptosystem using large minimum distance error-correcting codes derived from self-dual codes. To the best of our knowledge, such codes have not been implemented in a code-based cryptosystem until now. Moreover, we modify the decryption step of the system by introducing a decryption algorithm based on two private keys. We determine the parameters of binary codes with large minimum distance, which, if implemented into a McEliece-type cryptosystem, would provide a security level respectively of 80, 128, and 256 bits. For the 80-bit security case, we construct a large minimum distance self-dual code of length 1 064, and use it to derive a random punctured code to be used in the corresponding McEliece-type cryptosystem. Compared to the original McEliece cryptosystem, the key size is reduced by about 38.5%, although an optimal decoding set is yet to be constructed to make the new system fully defined and usable.

INDEX TERMS Post-quantum cryptography, McEliece cryptosystem, self-dual codes.

I. INTRODUCTION

The process initiated by NIST to standardize one or more quantum-resistant public-key cryptographic algorithms is ongoing, and currently at the fourth round.[1] One of the candidate submissions for the public-key encryption and key-establishment algorithms is the Classic McEliece cryptosystem. This fact indicates that after a long time of research on the original encryption scheme [11], this public-key cryptosystem is still considered one of the most secure.

Still, there is a major drawback, namely the size of its public key. This is a practical limitation for broad use in the current communication systems. For comparison, for the 128 bits security level of the McEliece cryptosystem, the size of its public key is around 187.69 Kb [4], whereas the public key of RSA for the same bit security is 3 Kb (or equivalently, 3 072 bits) [13, Table 2].

A significant number of studies aim to minimize the key size of the McEliece cryptosystem by using different families of error-correcting codes, but most of these variants have been broken (see e.g. [6], [12], [14]).

This paper proposes a McEliece-type cryptosystem using codes with error-correction capability higher than the capability of the codes adopted until now. By increasing the minimum distance of the implemented codes, we aim to decrease the size of the public key of the cryptosystem. More specifically, we use high minimum distance punctured codes derived from self-dual codes. Such punctured codes have no specific structure and do not belong to any known family of error-correcting codes. Our choice to use binary self-dual codes as a source code is based on two reasons: first, self-dual codes with large minimum distance exist (e.g., the extended Golay Code), and second, there is an algorithm for contracting self-dual codes [7], [9], [22]. To the best of our knowledge, self-dual or punctured codes derived from them have not been implemented in a code-based cryptosystem until now. The reason is most likely twofold: first, binary self-dual codes with high minimum distance are known up to length 130,

The associate editor coordinating the review of this manuscript and approving it for publication was Oussama Habachi.

[1]As of April 2023.

which is too small for current security requirements. Second, there was no efficient decoding algorithm for such codes until recently [23], an exception being the extended Golay code [16].

The main contributions of this paper can be summarized follows:

- We determine the parameters of a putative optimal self-dual code, from which a punctured code would provide a classic security level of 80, 128, and 256 bits (respectively a quantum security level of 67, 101, and 183 bits) if implemented in a McEliece-type cryptosystem.
- For the 80-bit security case, we construct an optimal self-dual code of length 1 064. To the best of our knowledge, such a code is presented here for the first time.
- We derive a punctured code of this self-dual code to generate the public key of a McEliece-type cryptosystem. Further, we modify the decryption step of the system by introducing a decryption algorithm that uses two private keys, namely the punctured and the self-dual code.

Our theoretical analysis estimates that the security level of the so defined system is 80 and 67 bits against classical and quantum attacks, respectively. The size of the resulting public key is 276.39 Kb, whereas the best-known example of a binary Goppa code providing the same bit security level in the original McEliece cryptosystem is 449.85 Kb [4]. Therefore, in this case, we achieve a reduction of the key size around 38.5%. *The results on the 80-bit security case suggest that self-dual codes can be used in a McEliece-type cryptosystem to reduce the key size for the same security level.* However, a current limitation is that *to make this cryptosystem usable, one also needs to define an optimal decoding set*. The computational effort to search for an optimal decoding set is currently undergoing for the 80-bit security level case, and we leave the complete definition and analysis of our cryptosystem in this particular instance for future research.

In summary, the main innovation underlying this paper is the idea to investigate self-dual codes for McEliece-type cryptosystems. The motivation, in perspective, is to obtain more compact public keys in such cryptosystems, which is the main issue for their use. The approach proposed in this paper is still far from providing a practical solution, as the limitation outlined above on the optimal decoding set suggests. However, we deem this work to indicate a promising future research direction in code-based cryptosystems.

II. BACKGROUND

Let $\mathbb{F}_2^n$ be the n-dimensional vector space over the binary field $\mathbb{F}_2$, whose vector sum is the bitwise XOR between n-bit vectors, while the multiplication by a scalar corresponds to the logical AND between a single bit and a n-bit vector. The *Hamming distance* between two vectors in $\mathbb{F}_2^n$ is the number of coordinates where they differ, while the *Hamming weight* (or only *weight*) $wt(v)$ of a vector $v \in \mathbb{F}_2^n$ is the number of nonzero coordinates in v. A k-dimensional subspace $\mathcal{C}$ of $\mathbb{F}_2^n$ is called a $[n, k, d]$ *binary linear code* where d is the minimum Hamming distance between any pair of vectors (also called *codewords*) of $\mathcal{C}$. Equivalently, d is the minimum Hamming weight among all nonzero codewords of $\mathcal{C}$. Since a $[n, k, d]$ binary linear code $\mathcal{C}$ is a vector subspace, it can be spanned by a $k \times n$ *generator matrix* G of rank k. On the other hand, a *parity-check matrix* H for $\mathcal{C}$ is a $(n-k) \times n$ matrix such that $Hx^\top = \underline{0}$ if and only if $x \in \mathcal{C}$. The vector $s = Hx^\top$ is also called the *syndrome* of x. A *coset* of a vector $x \in \mathbb{F}_2^n$ is the set $x + \mathcal{C} = \{x + c : c \in \mathcal{C}\}$, and a *coset leader* is any element in $x + \mathcal{C}$ with minimum Hamming weight.

Two binary linear codes $\mathcal{C}_1$ and $\mathcal{C}_2$ of length n are called *equivalent* if one can be obtained from the other by a permutation of coordinates, that is, if there exists a permutation $\sigma \in S_n$, with S_n being the symmetric group of order n, such that $\sigma(\mathcal{C}_1) = \mathcal{C}_2$. In particular, if a permutation σ maps a code $\mathcal{C}$ to itself, then σ is called *an automorphism* of the code.

The inner product in $\mathbb{F}_2^n$ is given by

$$\langle u, v \rangle = u_1 v_1 + u_2 v_2 + \cdots + u_n v_n$$

for $u, v \in \mathbb{F}_2^n$, and u and v are *orthogonal* if such product is equal to 0. Then, $\mathcal{C}^\perp = \{v \in \mathbb{F}_2^n : \ \langle u, v \rangle = 0, \ \forall u \in \mathcal{C}\}$ is the orthogonal of the code $\mathcal{C}$.

The code $\mathcal{C}$ is called *self-orthogonal* if $\mathcal{C} \subset \mathcal{C}^\perp$, and *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$. It is known that the weight of any codeword of a binary self-dual code is even [17, p.9]. If an error-correcting code is a linear $[n, k, d]$ code then it can correct up to $t \leq (d-1)/2$ errors. Let $\mathcal{C}$ be a linear code and $\mathcal{C}_i$ the set of all words of $\mathcal{C}$ without the i-th coordinate. Then, $\mathcal{C}_i$ is the punctured code of $\mathcal{C}$ on the i-th position.

A. McEliece CRYPTOSYSTEM

The McEliece Cryptosystem is the first code-based cryptosystem, and it was proposed by Robert McEliece in 1978 [11]. The original cryptosystem uses a binary $[1\,024, 524]$ code with an error-correcting capability of 50 errors. The steps of the encryption scheme are as follows:

1) *Define the system parameters:*
   - k: the length of the message block.
   - n: the length of the ciphertext.
   - t: the number of the intentionally added errors (equal to the error-correcting capability of the implemented linear code).
2) *Key generation:* define:
   - G: a generating matrix of an $[n, k, 2t+1]$ code for which there is a fast decoding algorithm.
   - P: a random $n \times n$ permutation matrix.
   - S: a random dense $k \times k$ non-singular matrix and, compute $G' = SGP$, P^{-1}.
   - S^{-1}: the inverse of P and S.

   Note that G' generates a linear code with the same n, k and t. Then, one has:
   - (G', t): *Public key*.
   - (G, P, S) or (Dec_G, P, S): *Private key*, where Dec_G is the fast decoding algorithm.

3) *Encryption:* split the data for encryption into k-bit blocks. Then each block m is encrypted as $r = mG' + e$, where e is a random vector of length n and weight t.
4) *Decryption:* The received vector r is decrypted as follows:
   a) Compute $r' = rP^{-1}$, which is $mSG + eP^{-1}$.
   b) Decode r' into a codeword c' using the efficient decoding algorithm for the code with generator matrix G, $c' = mSG$.
   c) Compute c such that $cG = c'$ (If G is in a systematic form, then c is the first k bits of c').
   d) Compute $m = cS^{-1}$.

The scheme above can be applied with any linear code for which a fast decoding algorithm is known, and for which there is a significant number of different codes of this family for the chosen length, dimension, and error-correcting capability. The original system in [11] employs a binary [1 024, 524, 101] Goppa code.

B. CRYPTANALYSIS

As with any other public encryption scheme, the McEliece cryptosystem gives the following information to the attacker: the encryption parameters, the encryption and decryption algorithms, and the public key. Hence, the adversary can also select any plaintext and compute the corresponding ciphertext.

Concerning the adversary's goals (total break, partial break, or distinguishing break), there are three main categories of attacks:

- *Key-recovery attack*: the attacker deduces the private key.
- *Message-recovery attack*: the attacker obtains a part of or the complete plaintext corresponding to a ciphertext without knowing the private key.
- *Distinguishing attack*: the attacker can distinguish a ciphertext from a random message without knowledge about the private key, or the attacker can distinguish the public key from a random code.

Next, we review a few of the known attacks on the McEliece encryption scheme. For each attack, we evaluate *the probability of success* or the inverse problem of evaluating the average number of attempts of the attack until the attacker achieves its target.

For algorithmic attacks *the security level* of a system is defined as a minimum work factor. The *work factor* is the average number of elementary (binary) operations needed to perform a successful attack [1, p.72].

In the following sections, we describe the main attacks published in the relevant literature, assuming that a McEliece cryptosystem is defined by a private key (G, P, S), where G is a $k \times n$ generator matrix of a binary $[n, k, 2t+1]$ code, P is a random $n \times n$ permutation matrix, and S is a random dense $k \times k$ non-singular matrix. The public key is (G', t) where $G' = SGP$. Further, we assume that the attacker has access to a ciphertext c produced by the encryption scheme.

We start by first recalling the components over which brute-force attacks can be mounted. Then, we describe the basic Information Set Decoding (ISD) attack and its work factor, along with some of its improved versions, particularly Stern's ISD attack.

1) BRUTE-FORCE ATTACKS

A brute-force attack can be mounted towards different components of the encryption system:

- *Towards the message*: the attacker takes a random message m_1 of length k, encrypts it to $c_1 = m_1 \cdot G'$, and computes the difference $e_1 = c - c_1$. If the difference e_1 has weight $\leq t$, then the plaintext corresponding to the ciphertext c is exactly m_1 and the attack succeeds. Then the probability of success is $1/2^k$ since the number of all possible messages of length k is 2^k.
- *Towards the coset leaders of the code generated by G'*: the attacker computes the syndrome of all coset leaders. The coset leader with syndrome equal to the syndrome of the ciphertext c is the error vector. Knowing the error vector, one can compute the codeword and then the message. The number of the coset leaders is $|\mathbb{F}_2^n|/|\mathcal{C}'| = 2^{n-k}$. Therefore, the work factor of this attack is at least 2^{n-k}.
- *Towards the error-vector*: the attacker searches among the vectors e of length n and weight t such that the syndrome of e is equal to the syndrome of the received vector c (the ciphertext). Thus, it is a search on e such that $S(e) = e \cdot H^T$ equals $S(c)$, where H represents the parity-check matrix corresponding to G'. This problem is equivalent to finding a linear combination of t columns of H, which results in a column vector with weight $S(c)$. Since there are $\binom{n}{t}$ possible choices for the vector e, the work factor of the brute force attack towards the error vector is $\binom{n}{t}$.

2) INFORMATION SET DECODING ATTACKS

Prange introduced the ISD technique in 1962 [19] as an efficient decoding method for cyclic codes. Several works (e.g., [8], [10], [15]) considered increasingly improved versions of the ISD decoding algorithm to attack the original McEliece cryptosystem described in [11].

An *information set* for a $[n, k]$ code $\mathcal{C}$ is any subset $A = \{i_1, \cdots, i_k\}$ of k coordinates such that, for any given set of values $b_i \in \mathbb{F}_2$, with $i \in A$, there is a unique codeword $c \in \mathcal{C}$. The information set thus consists of any k indices such that the corresponding k columns of a generator matrix of $\mathcal{C}$ have rank k.

Let $v = mG' + e$, where G' is a generator matrix of an $[n, k, 2t+1]$ code $\mathcal{C}$ and e is an error vector of weight t. Let A be an information set of k coordinates such that all entries of the error vector indexed by A are 0. In summary, the algorithm for the ISD attack works as follows:

1) Choose k out of n indices for the information set. These k columns of G' are permuted to the first k positions,

which is $G'P = [A_k|A_{n-k}]$, where A_k are the chosen k columns and A_{n-k} is the rest of G'.

2) Transform the matrix $[A_k|A_{n-k}]$ in systematic form, which takes $\mathcal{O}(k^3)$ operations [11], since it entails solving k linear equations in k unknowns. This is equivalent to transforming $G'P$ into $[I_k|A'_{n-k}] = UG'P$, where U is the transformation matrix.
3) Compute m as $m = v_A U$, where v_A are the k coordinates of v in the positions of the information set A. Then $e = r - mG'$. If $wt(e) = t$, then m is the encrypted message. The possibilities for the error vector e to have 0 coordinates in the information set are k out of $n - t$ coordinates, i.e. $\binom{n-t}{k}$;
4) Estimate how many of the choices for k out of n columns have rank k of the generator matrices of the family of $[n, k, 2t + 1]$ binary codes. In the original code-based cryptosystem, Goppa codes were used, and for these codes, around 29% of the choices of k columns are invertible.

Therefore, the work factor for the ISD attack is

$$\frac{k^3\binom{n}{k}}{\beta\binom{n-t}{k}},$$

where β is the proportion of the invertible k columns out of n for the generator matrices of the family of $[n, k, 2t+1]$ codes. Note that β depends on the specific family.

3) STERN'S ISD ATTACK

Stern [21] proposed a refinement of the ISD attack, which is based on the use of the extended code generated by G'', defined as:

$$G'' = \begin{pmatrix} G' \\ x \end{pmatrix} = \begin{pmatrix} G' \\ u \cdot G' + e \end{pmatrix}. \tag{1}$$

It is known [1] that such code has only one minimum weight codeword, which coincides with e. Stern's attack consists in finding the unique codeword e of weight t in the code generated by G''. The algorithm is probabilistic, using two input parameters p and l with the parity check matrix of the extended code.

The work factor is $B = f_1 + f_2 + f_3$ for one iteration of the attack, where [21]:

$$f_1 = \frac{1}{2}(n-k)^3 + k(n-k)^2,$$
$$f_2 = 2pl\binom{k/2}{p},$$
$$f_3 = 2p(n-k)\frac{\binom{k/2}{p}^2}{2^l}.$$

The total work factor of the attack is $\frac{B}{P_t}$, where P_t is the probability of finding a codeword of weight t in one iteration. In particular, P_t is estimated in [21] as:

$$P_t = \frac{\binom{t}{2p}\binom{n-t}{k-2p}}{\binom{n}{k}} \cdot \frac{\binom{2p}{p}}{4^p} \cdot \frac{\binom{n-k-t+2p}{l}}{\binom{n-k}{l}}. \tag{2}$$

TABLE 1. Shorthand notation for the attacks considered in this paper.

Name	Attack
A_1	Brute force attack towards the message
A_2	Brute force attack towards the coset leaders of the private key
A_3	Brute force attack on the error-vector
A_4	Basic Information Set Decoding attack
A_5	Stern's attack
A_6	Basic Quantum Information Set Decoding attack

4) QUANTUM BASIC INFORMATION SET DECODING ATTACK

Let $v = mG + e$, G and e be defined as before. The Basic Quantum ISD attack first searches for an invertible submatrix G_S of G, by selecting k of its columns. Once it is found, the algorithm computes $(v_{i_1}, v_{i_2}, \ldots, v_{i_k}) \cdot {G_S}^{-1} = m$, with $m \in \mathbb{F}_2^k$, then determines $mG \in \mathbb{F}_2^n$, and finds the error vector $e = v - mG$, checking if its Hamming weight is t.

Regarding [3], randomly searching for a root can succeed in approximately $\binom{n}{k}/0.29\binom{n-t}{k}$ iterations, where one iteration of this function has around $O(n^3)$ bit operations. Grover's algorithm uses about square root of the number of iterations, i.e., $\sqrt{\binom{n}{k}/0.29\binom{n-t}{k}}$.

Then the work factor for the Basic Quantum ISD attack, which is the complete number of qubit operations for finding a solution, is $O(n^3)\sqrt{\binom{n}{k}/0.29\binom{n-t}{k}}$. Note that the meaning of 0.29 is that, on average, 29% of the selected matrices G_S are non-singular when G is a generator matrix of the Goppa code. A list of the described attacks with names used further in this work are reported in Table 1.

III. PARAMETERS ESTIMATION FOR SELF-DUAL CODES WITH BIT SECURITY 80, 128, AND 256

To estimate parameters for the self-dual codes, which would provide a security level of 80, 128, and 256 bits, we apply the upper bounds for the work factor of the attacks in the previous section to the known recently proposed Goppa codes with these security levels. Since our attacks are not the best known, we expect to obtain higher values for the upper bounds. These higher values we use further for the estimation of the parameters of the self-dual codes.

The private key of the original McEliece cryptosystem is a [1 024, 525] Goppa code with the error-correcting capability of 50 errors. Initially, it was estimated to provide a security of 64 bits. Later, via an improved version of Stern's attack in [4] the security of the system was reduced to 60.5 bits. In the same publication, the authors proposed parameters for the Goppa codes, where implementation in the McEliece cryptosystem would provide a security level of 80, 128, and 256 bits. The proposed codes are listed in Table 2. The latest proposed codes providing security levels of 128, 196, and 256 bits are in the NIST proposal [2].

From the results listed in Table 2, it follows that we have to search for codes providing a bit security level of 83, 148, and 302 to ensure that they would provide at least 80, 128, and 256 bit security concerning the latest attacks. In Table 3, we list the parameters of a few such codes.

TABLE 2. min(Log_2(*Workfactor*)) of the attacks $A_1, \ldots, A_6$ in Section II-B.

Goppa codes							
code	security	n	k	t	$k(n-k)$	$min_(A_1, ..., A_5)$	A_6
D_1	80 [4]	1 632	1 269	34	460 647	82.231	69.5887
D_2	128 [4]	2 960	2 288	57	1 537 536	129.8371	96.7078
D_3	128 [2]	3 488	2 720	64	2 088 960	147.4275	106.5127
D_4	256 [4]	6 624	5 129	117	7 667 855	259.2255	166.1179
D_5	256 [2]	6 688	5 024	128	8 359 936	265.2662	168.9545
D_6	256 [2]	6 960	5 413	119	8 373 911	266.0612	169.8205
D_7	256 [2]	8 192	6 528	128	10 862 592	302.1663	188.9797

Note that these are the parameters of the punctured $[n, k, 2t+1]$ codes. The corresponding self-dual codes must be of length $n+2$ and minimum weight $2t+3$, to ensure that the punctured codes are within the required parameters. The upper bounds for the minimum weight of a putative self-dual $[n_1, n_1/2, d_1]$ code are as follows [20]:

$$\begin{cases} d_1 \leq 4\lfloor \frac{n_1}{24} \rfloor + 4 & , \text{ if } n_1 \not\equiv 22 \pmod{24}, \\ d_1 \leq 4\lfloor \frac{n_1}{24} \rfloor + 6 & , \text{ if } n_1 \equiv 22 \pmod{24}. \end{cases} \tag{3}$$

Remark 1: In our estimation, we consider a minimum weight that is 15% smaller than the above bounds. In this way, we achieve the following:

- increasing the probability that such a code exist and can be constructed;
- if such a code exists, then a large number of codes with the same parameters, length, and minimum weight exist. This is a preliminary requirement for the security of the McEliece-type cryptosystem.

The size of the putative punctured codes B_1, B_9, and B_{31} is at least 38% smaller than the size of the proposed smallest Goppa codes D_1, D_2, and D_4 providing the security level of 80, 128, and 256 bits, correspondingly. In the next section, we will present a possible construction of a self-dual code where the punctured code has the parameters of B_1.

IV. A NEW EXAMPLE OF McEliece-TYPE CRYPTOSYSTEM WITH 80-BIT SECURITY

In this section, we first construct an example of a binary $[1\,064, 532, d \geq 162]$ self-dual code, to define a McEliece-type cryptosystem with 80 bit security. Then, we derive a punctured code from such code to generate the public key of the encryption scheme. Next, we discuss an efficient decoding algorithm suitable for the new self-dual code. The decoding is used in the decryption step of the cryptosystem. Further, we propose a modified decryption algorithm for the McEliece-type system with two private keys: the new binary $[1\,064, 532, d \geq 162]$ self-dual code and one of its punctured codes. The decryption integrates the decoding of the complete self-dual code. Finally, we discuss the bit security level of the McEliece-type cryptosystem thus defined.

A. A BINARY [1 064, 532, D ≥ 162] SELF-DUAL CODE

The upper bound for the minimum weight d of a binary $[1\,064, 532, d]$ self-dual code is 180 (Eq.(3)). Here we construct such a code where the aim is for d to be at least 162. Note that this value for d is much smaller than the upper bound (see Remark 1).

To construct a binary $[1\,064, 532, d \geq 162]$ self-dual code we use a known algorithm presented in [7] and [22].

Let us assume that a self-dual $[1\,064, 532, d \geq 162]$ code exists. Let B be such a code and let B have an automorphism σ of order 133 with 8 cycles of length 133 and no fixed points. Without loss of generality σ can be represented as:

$$\sigma = \Omega_1 \Omega_2 \ldots \Omega_8 \ ,$$

where Ω_i is a cycle of length 133 for $1 \leq i \leq 8$.

If $v \in B$, then v can be expressed as

$$v = (v|\Omega_1, v|\Omega_2, \ldots, v|\Omega_8) \ ,$$

where $v|\Omega_i = (v_0, v_1, \ldots, v_{132})$ denotes the coordinates of v in the i^{-th} cycle of σ. Let further $F_\sigma(B)$ and $E_\sigma(B)$ be respectively defined as $F_\sigma(B) = \{v \in B | \ v\sigma = v\}$ and $E_\sigma(B) = \{v \in B | \ wt(v|\Omega_i) \equiv 0 \pmod 2, \ i = 1, \ldots, 8\}$. According to [7], both $F_\sigma(B)$ and $E_\sigma(B)$ are linear subcodes of B. Moreover, $B = F_\sigma(B) \oplus E_\sigma(B)$, where $\oplus$ stands for the direct sum of linear subspaces. Then, a generator matrix of B can be decomposed as:

$$G = \begin{pmatrix} X \\ Y \end{pmatrix}, \tag{4}$$

where X is a generator matrix of $F_\sigma(B)$ and Y is a generator matrix of $E_\sigma(B)$.

The maps π and φ are defined as follows:

$$\pi : F_\sigma(B) \to \mathbb{F}_2^8 \ , \ \ \pi(v|\Omega_i) = v_j \ , \tag{5}$$

for some $j \in \Omega_i$, $i = 1, \ldots, 8$, and

$$\varphi : E_\sigma(B) \to \mathcal{P}^8 \ , \tag{6}$$

where $v|\Omega_i = (v_0, v_1, \ldots, v_{132})$ is identified with the polynomial $\varphi(v|\Omega_i)(x) = v_0 + v_1 x + \cdots + v_{132} x^{132}$ in $\mathcal{P}$ for $1 \leq i \leq 8$, and $\mathcal{P}$ is the set of even weight polynomials in the quotient ring $\mathcal{R}_1 = \mathbb{F}_2[x]/(x^{133} - 1)$.

An inner product in $\mathcal{P}^8$ is defined as:

$$\langle g, h \rangle = g_1(x)h_1(x^{-1}) + \cdots + g_8(x)h_8(x^{-1}) \tag{7}$$

for all $g, h \in \mathcal{P}^8$.

TABLE 3. min(Log_2(*Workfactor*)) of the attacks $A_1, \ldots, A_6$ in Section II-B.

Punctured codes							
code	n	k	t	$k(n-k)$	$min(A_1, ..., A_5)$	A_6	expected security
B_1	1 062	531	75	281 961	87.3248	67.5796	80
B_2	1 064	532	75	283 024	87.3264	67.5837	80
B_8	1 076	538	75	289 444	87.2886	67.6079	80
B_9	1 894	947	134	896 809	147.8721	101.2093	128
B_{10}	1 896	948	134	898 704	147.869	101.2097	128
B_{30}	1 940	970	136	940 900	149.8767	102.3316	128
B_{31}	4 006	2 003	284	4 012 009	303.9682	183.5916	256
B_{32}	4 008	2 004	284	4 016 016	303.9619	183.5895	256
B_{42}	4 028	2 014	284	4 056 196	303.8758	183.5694	256

Algorithm 1 Construction of a Self-Dual Code Having an Automorphism

1 Determine a generator matrix X' of $\pi(F_\sigma(B))$.
2 Find the generator matrix X of $F_\sigma(B)$ corresponding to X'.
3 Construct a generator matrix Y' of $\varphi(E_\sigma(B))$.
4 Find the generator matrix Y of $E_\sigma(B)$ corresponding to Y'.
5 **if** $G = \begin{pmatrix} X \\ Y \end{pmatrix}$ generates a code with a minimum weight d, **then**
6 return G (G generates B);
7 **else**
8 return to 1.

According to [22], for the images $\pi(F_\sigma(B))$ and $\varphi(E_\sigma(B))$ the following holds:

1) $\pi(F_\sigma(B))$ is a binary self-dual code of length 8;
2) $\varphi(E_\sigma(B))$ is a self-orthogonal code, i.e.,

$$u_1(x)v_1(x^{-1}) + \cdots + u_8(x)v_8(x^{-1}) = 0, \qquad (8)$$

for all $u, v \in \varphi(E_\sigma(B))$.

The code generation procedure, following [7] and [22] using the above-defined sets, images and properties, is summarized in Algorithm 1.

To construct the code B, we take the steps of Algorithm 1.

1) Determine a generator matrix of $\pi(F_\sigma(B))$.
Since the image $\pi(F_\sigma(B))$ is a binary self-dual code of length 8, a possible generator matrix of $\pi(F_\sigma(B))$ is:

$$X' = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

2) Find the corresponding generator matrix X of $F_\sigma(B)$.
The corresponding to X' generator matrix of $F_\sigma(B)$ is:

$$X = \begin{pmatrix} s & o & o & o & o & s & s & s \\ o & s & o & o & s & o & s & s \\ o & o & s & o & s & s & o & s \\ o & o & o & s & s & s & s & o \end{pmatrix}, \qquad (9)$$

where $s = (1, 1, \ldots, 1)$ is the all ones vector and o is the zero vector in $\mathbb{F}_2^{133}$.

3) Construct a generator matrix Y' of $\varphi(E_\sigma(B))$.
The image $\varphi(E_\sigma(B)) \subset \mathcal{P}^8$, where $\mathcal{P}$ is the set of even-weight polynomials in $\mathcal{R}_1$.
The factorization of the polynomial $x^{133} - 1$ over $\mathbb{F}_2$ is

$$x^{133} - 1 = h_0(x)h_1(x)\ldots h_9(x) \ ,$$

where $h_0 = x - 1$, $deg(h_1(x)) = deg(h_2(x)) = 3$ and $deg(h_j(x)) = 18$ for $j = 3, \ldots, 9$. Next, denote by:

- $g_j(x) = \frac{x^{133}-1}{h_j(x)}$.
- $I_j = \langle g_j(x) \rangle$: the ideal of $\mathcal{R}_1$ generated by $g_j(x)$.
- $e_j(x)$: the generator idempotent of I_j for $j = 0, \ldots, 9$.

Then, according to [18, p.56], we have:

- $\mathcal{R}_1 = I_0 \oplus I_1 \oplus \cdots \oplus I_9$.
- I_j is a field with $2^{deg(h_j(x))}$ elements, $j = 0, 1, \ldots, 9$.
- $e_i(x)e_j(x) = 0$, $i \neq j$.

After generating the idempotent $e_j(x)$ of the ideal I_j, for $j = 1, \ldots, 9$, we observe that $e_1(x^{-1}) = e_2(x)$, $e_3(x^{-1}) = e_4(x)$, $e_5(x^{-1}) = e_6(x)$, $e_7(x^{-1}) = e_8(x)$, and $e_9(x^{-1}) = e_9(x)$. The same relations also hold for the generator polynomials $g_i(x)$ for $1 \leq i \leq 9$, i.e., $g_1(x^{-1}) = g_2(x)$, $g_3(x^{-1}) = g_4(x)$, etc. Using these relations and the self-orthogonality of the image $\varphi(E_\sigma(B))$, we construct a generator matrix of $\varphi(E_\sigma(B))$ having the form:

$$Y' = \begin{pmatrix} Y_1 \\ \vdots \\ Y_9 \end{pmatrix}, \qquad (10)$$

where Y_j is 4×8 matrix with elements of I_j, for $j = 1, \ldots, 9$. The cells Y_1, Y_3, Y_5, and Y_7 are constructed under certain conditions, which we discuss at the end of this section. The cells Y_2, Y_4, Y_6, and Y_8 are obtained from the previous four cells using the orthogonality condition Eq. (8). Also there, we present a particular example of the complete generator matrix Y' of $\varphi(E_\sigma(B))$ in Eq. (10). We note that for each of Y_1, Y_3,

Y_5, Y_7, and Y_9 there are at least $16 \cdot 2^{18}$ choices. This leads to more than 2^{20} choices for Y'. Each of these choices can be mapped into 8! matrices by a column permutation. Therefore, we have at least 2^{35} choices for the matrix Y'.

4. Find the corresponding generator matrix Y of $E_\sigma(B)$. The matrix Y' defines the generator matrix of the subcode $E_\sigma(B)$ as

$$Y = \begin{pmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,8} \\ \vdots & \vdots & & \vdots \\ y_{36,1} & y_{36,2} & \cdots & y_{36,8} \end{pmatrix}, \qquad (11)$$

where each entry of the first 8 rows is a right circulant 3×133 matrix since I_j is a cyclic $[133, 3]$ code for $j = 1, 2$, and each entry of the rest of the rows is a right circulant 18×133 because I_j is a cyclic $[133, 18]$ code for $j = 3, \ldots, 9$.[2]

In step 3 we mentioned that the cells Y_1, Y_3, Y_5, and Y_7 are constructed under certain conditions. The matrix $Y', i = 1, \ldots, 9$, specifies the generator matrix Y of the subcode $E_\sigma(B)$. The minimum weight of the code B has to be greater than or equal to 162. Thus, the same has to hold for the minimum weight of Y. In this regard, we construct the Y_i cells according to the following requirements:

- each row of Y_i, $i = 1, \ldots, 9$, has at least four non zero elements, i.e., each row has weight greater than or equal to four;
- the weight of Y_1, $\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$, Y_3, $\begin{pmatrix} Y_3 \\ Y_4 \end{pmatrix}$, Y_5, $\begin{pmatrix} Y_5 \\ Y_6 \end{pmatrix}$, Y_7, $\begin{pmatrix} Y_7 \\ Y_8 \end{pmatrix}$ and Y_9 is at least 3.

In step 4, the matrix Y has to satisfy the following requirements:

- The first 24 rows, corresponding to $\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$ have a minimum weight of at least 162.
- Each next 18 rows corresponding to a row in Y_s, $s = 3, \ldots, 9$, have a minimum weight of at least 162.
- The linear combinations of up to 8 rows of the 144 rows of Y corresponding to $\begin{pmatrix} Y_3 \\ Y_4 \end{pmatrix}$, $\begin{pmatrix} Y_5 \\ Y_6 \end{pmatrix}$, $\begin{pmatrix} Y_7 \\ Y_8 \end{pmatrix}$ and the 72 rows corresponding to Y_9, have a weight of at least 162.

Once we have constructed the generator matrices for the subcodes $F_\sigma(B)$ and $E_\sigma(B)$, we can proceed to step 5. of Algorithm 1.

5. If $G = \begin{pmatrix} X \\ Y \end{pmatrix}$ generates a code with a minimum weight $d \geq 162$, where X and Y are given in Eq. (9) and Eq. (11) respectively, then G generates the code B. To obtain a generator matrix of B, we developed a software in C++ that performs the following operations:
   - Construct sub-matrices of Y corresponding to Y_s, with $s = 1, \ldots, 8$, which meet the conditions in Step 3.
   - Create the sub-matrix of Y corresponding to Y_9 defined in step 3.
   - Create the matrix G defined in Eq. (4) and the parity-check matrix H of G.
   - Compute the weight of all linear combinations up to 8 rows of G and of H. This calculation is performed by implementing the algorithm for efficiently computing the codewords of fixed weight in linear codes (for the binary case) presented in [5].

   Calculating the exact minimum weight has a work factor of 2^{87} (regarding Stern's attack, Section II-B), which is infeasible. Instead, the following computations are carried out:
   (1) All linear combinations of up to 8 vector rows of G and the corresponding parity check matrix are computed. The resulted codewords have a weight greater than or equal to 168.
   (2) Simulations of a random linear combination of a random number of rows of G on a single 16 RAM Intel7 PC for 30 days resulted in vectors with weight greater than or equal to 168.
   (3) From the requirements for steps 3 and 4, it follows that the first $24 \times 1\,064$ sub-matrix and every next $18 \times 1\,064$ sub-matrix of Y has a minimum weight of at least 168.+

   Based on this evidence, we expect the matrix G with the defined above sub-matrices X and Y to generate a self-dual $[1\,064, 532, d \geq 162]$ code.

Remark 2: Note that for the submatrix Y of $G = \begin{pmatrix} X \\ Y \end{pmatrix}$, there are at least 2^{35} possible choices (as explained at the end of Step 3, Algorithm 1). It is not known how many of the corresponding self-dual codes of length 1 064 generated by the matrices G have a minimum weight $d \geq 162$.

B. McEliece-TYPE CRYPTOSYSTEM USING THE NEW CODE EXAMPLE

Let B_p be a punctured $[1\,062, 531, d' \geq 160]$ code obtained from the self-dual code B with generator matrix G, from Section IV-A, by removing the first two columns and the first row of G. Denote this generator matrix of B_p by G_p.

A decoding scheme, recently introduced in [23], decodes binary self-dual codes having an automorphism ϕ of order pr including only cycles of length pr, for p and r being odd prime numbers. This decoding is a hard decision iterative decoding scheme using a set of cyclically different codewords with weights $d + o$, for $o = 0, 2, 4$ or any small number. Two codewords are called *cyclically different* if one cannot

[2]The first rows of the circulants corresponding to the polynomials in the matrix Y' and the corresponding binary generator matrix G can be found at the following repository: https://github.com/NoAuthorSubmission/McEliece_Data

be obtained from the other by applying ϕ^l, for some l, that is, $b \neq \phi^l(c)$ for $1 \leq l \leq pr-1, \forall\ b,\ c \in \mathcal{C}$. An optimal decoding set can be defined after experiments with sets of cyclically different codewords of minimum weight or mixed sets of codewords of different weights close to the minimum weight. The new self-dual code B possesses an automorphism σ of order 133 with eight cycles of length 133 and no fixed points. Therefore, the decoding scheme of [23] is a valid decoding scheme for B.

Next, we define the McEliece cryptosystem using the punctured code B_p. Recall that B_p is a $[1\,062, 531, d' \geq 160]$ punctured code obtained from the self-dual code B, while G and G_p are respectively generator matrices of B and B_p.

1) *System parameters*:
   - $k = 531$: the length of the message m.
   - $n = 1\,062$: length of the ciphertext r.
   - $t = 75$: number of the intentionally added errors.
2) *Key generation*:
   - G_p: a generator matrix of a $[1\,062, 531, 160]$ code, a punctured code of a self-dual $[1\,064, 532, d \geq 162]$ code.
   - P: a random $n \times n$ permutation matrix.
   - S: an invertible $k \times k$ matrix such that SG_pP is in a systematic form.
   - $G'_p = SG_pP$, P^{-1} and S^{-1}: the inverse of P and S.
   - *Public key*: $(G'_p,\ t)$.
   - *Private key*: (G_p, G, P, S).
3) *Encryption*:
   - e: a random error vector of length n and $wt(e) = t$.
   - $m \rightarrow r = m\,G'_p + e$
4) *Decryption:* For the decryption, we define two more elements. Let S_1 and P_1 be extended matrices of S and P defined as follows:

$$S_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & & & \\ \vdots & & S & \\ 0 & & & \end{pmatrix}, \tag{12}$$

$$P_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & & & \\ \vdots & & P & \\ 0 & & & \end{pmatrix}. \tag{13}$$

One can show that for the matrices defined above the following holds:

$$\begin{aligned} &(0|m) \cdot S_1 \cdot G \cdot P_1 \\ &= (0|m \cdot S) \cdot G \cdot P_1 \\ &= (m_1^*, m_2^*|m \cdot S \cdot G_p \cdot P) \\ &= (m_1^*, m_2^*|m \cdot G'_p). \end{aligned} \tag{14}$$

Thus, we can decode r' of length 1 062 via decoding a padded $(*, * \mid r')$ of length 1 064 by the initial self-dual code B. The decryption process, including this decoding strategy, is described in Algorithm 2.

Algorithm 2 Decryption Using Padded Ciphertext for McEliece Cryptosystem With Private Keys B_p and B

```
Denote s = [[0, 0], [0, 1], [1, 0], [1, 1]], i = 1, t = 75,
  n = 1062, k = n/2.
Compute r' = rP^{-1}, r-received vector of length k
while i < 5
    Pad r' into (s[i]||r')
    Decode (s[i]||r') into c_1, c_1 ∈ B, by Algorithm 1 [23]
  with B_mix.
    if 5) successful then
        Denote c_2 = c_1[3 : n + 2], m_2 = c_2[1 : k]
        Compute m_1 = m_2 · S^{-1}
        if (m_1 ∈ B_1 AND weight(m_1 · G'_p − r') == t) then
            return r as m_1
    i = i + 1
return 'Unsuccessful decryption'
```

Algorithm 2 includes the decoding Algorithm 1 [23] with decoding set B_{mix} containing cyclically different codewords of the self-dual code B of weight 168, 180, 184, and 188.

Remark 3: Note that first, the decoding set is much smaller (at least 133 times smaller) than the complete set of codewords with weight 168, 180, 184, and 188, and second, the decoding runs around t iterations (steps 2 till 8 of Algorithm 1 in [23], $t = 75$) for correcting the t errors.

An example of a self-dual [266, 133, 36] code, constructed via an automorphism of order 133 as the code B, is included in [23]. Using a set of only 2 614 codewords the mentioned decoding algorithm corrects up to $t-2$ errors in 100% of the cases, where $t = 17$. Since increasing the number of codewords with weight $d+o$, for $o = 0, 2, 4$, in the decoding set increases the decoding performance of the algorithm, there will be a set that decodes t errors in 100% of the cases.

Note that the minimum weight of the punctured code B_p is 160, which means B_p has an error-correcting capability of up to 79 errors. According to the estimation in Section III for a security level of 80 bits, code B_p is required to correct 75 errors, which is $t-4$. In such a setup, we expect that the decoding algorithm from [23] will decode B with the same or close to the efficiency of decoding of B_{266} when using a large enough decoding set.

The cryptanalysis of the system includes the attacks $A_1, \dots, A_5$ described in Section II-B. According to Table 2, the original McEliece cryptosystem with a security level of 80 bits has a security level of 82 bits against the attacks $A_1, \dots, A_5$. The new system with the public key G'_p, according to Table 2 (first row), has a security level of 87 bits against the same attacks. The gap of 7 bits is to ensure that the same system has a security level of 80 bits against improved versions of these attacks. The punctured code B_p and the public key G'_p created by it are not self-orthogonal and do not belong to any specific family of codes. Therefore, the problem of

decoding the public key or decoding B_1 is expected to be as difficult as the problem of decoding a random code.

V. CONCLUSION

This paper proposed a McEliece-type cryptosystem using high minimum distance self-dual codes and punctured codes derived from them. We determined the parameters of a putative optimal self-dual code, providing a classic (respectively, quantum) security level of 80, 128, and 256 (respectively, 67, 101, and 183) bits. For the 80-bit security case, we constructed an optimal self-dual code of length 1 064, reducing the key size by around 38.5% with respect to the original McEliece cryptosystem. The main limitation of our work is that a complete optimal decoding set is needed to make our cryptosystem practically usable. The computational search of such a decoding set is currently undergoing, and it is a direction for future research on the topic.

REFERENCES

[1] M. Baldi, *QC-LDPC Code-Based Cryptography (5.4 Cryptanalysis of the McEliece and Niederreiter Cryptosystems)*. Cham, Switzerland: Springer, 2014.

[2] D. Bernstein, C. Chou, T. Lange, I. von Maurich, R. Misoczki, R. Niederhagen, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, and W. Wang, "Classic McEliece: Conservative code-based cryptography," Tech. Rep., 2017.

[3] D. J. Bernstein, "Grover vs. McEliece," in *Proc. 3rd Int. Conf. Post-Quantum Cryptogr.*, 2010, pp. 73–80.

[4] D. J. Bernstein, T. Lange, and C. Peters, "Attacking and defending the McEliece cryptosystem," in *Post-Quantum Cryptography*, J. Buchmann and J. Ding, Eds. Berlin, Germany: Springer, 2008, pp. 31–46.

[5] I. Bouyukliev and V. Bakoev, "A method for efficiently computing the number of codewords of fixed weights in linear codes," *Discrete Appl. Math.*, vol. 156, no. 15, pp. 2986–3004, Aug. 2008.

[6] A. Couvreur, A. Otmani, and J.-P. Tillich, "Polynomial time attack on wild McEliece over quadratic extensions," in *Advances in Cryptology EUROCRYPT 2014*. Berlin, Germany: Springer, 2014, pp. 17–39.

[7] R. A. Dontcheva, A. J. vanZanten, and S. M. Dodunekov, "Binary self-dual codes with automorphisms of composite order," *IEEE Trans. Inf. Theory*, vol. 50, no. 2, pp. 311–318, Feb. 2004.

[8] Y. Hamdaoui and N. Sendrier, "A non asymptotic analysis of information set decoding," Cryptol. ePrint Arch., Tech. Rep. 2013/162, 2013. [Online]. Available: https://eprint.iacr.org/2013/162

[9] W. Huffman, "Automorphisms of codes with applications to extremal doubly even codes of length 48," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 3, pp. 511–521, May 1982.

[10] P. J. Lee and E. F. Brickell, "An observation on the security of McEliece's public-key cryptosystem," in *Advances in Cryptology EUROCRYPT 88*, D. Barstow, W. Brauer, P. B. Hansen, D. Gries, D. Luckham, C. Moler, A. Pnueli, G. Seegmüller, J. Stoer, N. Wirth, and C. G. Günther, Eds., Berlin, Germany: Springer, 1988, pp. 275–280.

[11] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *Deep Space Netw. Prog. Rep.*, vol. 44, pp. 114–116, Jan. 1978.

[12] L. Minder and A. Shokrollahi, "Cryptanalysis of the sidelnikov cryptosystem," in *Advances in Cryptology EUROCRYPT 2007*. Berlin, Germany: Springer, 2007, pp. 347–360.

[13] NIST. (2012). *Recommendation for Key Management. NIST*. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-57p1r3.pdf

[14] A. Otmani, J.-P. Tillich, and L. Dallot, "Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes," *Math. Comput. Sci.*, vol. 3, no. 2, pp. 129–140, Apr. 2010.

[15] C. Peters, "Information-set decoding for linear codes over F_q," in *Post-Quantum Cryptography*, N. Sendrier, Ed., Berlin, Germany: Springer, 2010, pp. 81–94.

[16] V. Pless, "Decoding the Golay codes," *IEEE Trans. Inf. Theory*, vol. IT-32, no. 4, pp. 561–567, Jul. 1986.

[17] V. Pless and W. Huffman, *Fundamentals Error-Correcting Codes*. Amsterdam, The Netherlands: Cambridge Univ. Press, 2003.

[18] V. S. Pless and W. C. E. Huffman," *Handbook Coding Theory*. Amsterdam, The Netherlands: Elsevier, 1998.

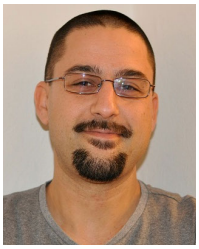
[19] E. Prange, "The use of information sets in decoding cyclic codes," *IEEE Trans. Inf. Theory*, vol. IT-8, no. 5, pp. 5–9, Sep. 1962.

[20] E. M. Rains, "Shadow bounds for self-dual codes," *IEEE Trans. Inf. Theory*, vol. 44, no. 1, pp. 134–139, Jan. 1998.

[21] J. Stern, "A method for finding codewords of small weight," in *Coding Theory and Applications* (Lecture Notes in Computer Science), vol. 388, G. Cohen and J. Wolfmann, Eds. Heidelberg, Germany: Springer, 1989, pp. 106–113.

[22] R. A. Yorgova, "On binary self-dual codes with automorphisms," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 3345–3351, Jul. 2008.

[23] R. Yorgova, "On decoding of a specific type of self-dual codes," 2021, *arXiv:2106.11146*.



LUCA MARIOT received the dual Ph.D. degree in computer science from the University of Milano-Bicocca, Italy, and Université Côte d'Azur, France, in 2018. From 2018 to 2020, he was a Postdoctoral Researcher with the University of Milano-Bicocca. He was also a Postdoctoral Researcher with the Delft University of Technology, The Netherlands, from 2020 to 2021, and Radboud University, The Netherlands, in 2022. He is currently an Assistant Professor with the University of Twente, The Netherlands. His research interests include cryptography, evolutionary computation, cellular automata, and machine learning.

STJEPAN PICEK received the Ph.D. degree, in 2015. From 2015 to 2017, he was a Postdoctoral Researcher with KU Leuven, Belgium, and MIT, USA, and from 2017 to 2021, he was an Assistant Professor with the Delft University of Technology, The Netherlands. He is currently an Associate Professor with Radboud University, The Netherlands. His research interests include security, machine learning, and evolutionary algorithms.

RADINKA YORGOVA received the M.Sc. degree from Shumen University, Bulgaria, in 1993, the Ph.D. degree in coding theory from the Delft University of Technology, The Netherlands, in 2002, and the master's degree in computer science, with a specialization in cybersecurity from the Delft University of Technology, in 2021. From 1993 to 1997, she was an Assistant Professor with Shumen University. From 2002 to 2010, she worked on postdoctoral and research projects with the Delft University of Technology, University of Liverpool, U.K., from 2004 to 2005, and University of Bergen, Norway, from 2005 to 2010. For the next eight years, she worked in industry developing mathematical models for an energy consultancy company in The Netherlands. Since November 2021, she has been a Lecturer in cybersecurity with the Amsterdam University of Applied Sciences. Her research interests include (code-based) cryptography, algebraic coding theory, combinatorics, privacy-enhancing technologies, algorithms, and applied computing.