IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 May 2025, accepted 9 June 2025, date of publication 16 June 2025, date of current version 23 June 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3579919

RESEARCH ARTICLE

An Integer Erasure Correction Coding and Its Application for Security Enhancement of Encryption

MIODRAG J. MIHALJEVIĆ[1,2,3], ALEKSANDAR RADONJIĆ[4], NEVENA MIJAJLOVIĆ[5], LIANHAI WANG[1,2], AND SHUJIANG XU[1,2]

[1]Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China
[2]Shandong Provincial Key Laboratory of Industrial Network and Information System Security, Shandong Fundamental Research Center for Computer Science, Jinan 250014, China
[3]Mathematical Institute, Serbian Academy of Sciences and Arts, 11001 Belgrade, Serbia
[4]Institute of Technical Sciences, Serbian Academy of Sciences and Arts, 11001 Belgrade, Serbia
[5]Faculty of Science and Mathematics, University of Montenegro, 81000 Podgorica, Montenegro

Corresponding author: Nevena Mijajlović (nevenami@ucg.ac.me)

This work was supported in part by the New 20 Project of Higher Education of Jinan, China, under Grant 202228017; in part by Shandong Natural Science Foundation of China under Grant ZR2024MF104 and Grant ZR2023QF129; in part by the Pilot Project for Integrated Innovation of Science, Education, and Industry of Qilu University of Technology (Shandong Academy of Sciences) under Grant 2024ZDZX08; and in part by the Talent Research Project of Qilu University of Technology under Grant 2023RCKY144. The work of Miodrag J. Mihaljević was supported by the Grant F-153 "Advanced Techniques for Information Security and Blockchain Technology" of Serbian Academy of Sciences and Arts. The work of Aleksandar Radonjić was supported by the Ministry of Science, Technological Development and Innovation of Republic of Serbia under Grant 451-03-136/2025-03/200175. The work of Nevena Mijajlović was supported by the Ministry of Education, Science and Innovation of Montenegro through the grant "Mathematical Analysis, Optimization and Machine Learning".

ABSTRACT This paper presents a new class of erasure-correcting codes (ECCs) aimed at enhancing cryptographic security of certain encryption schemes. The proposed ECCs employ integer arithmetic to encode and decode data bits and can correct all data bytes, each affected by exactly two erasures. In the enhanced encryption scheme, the ciphertext produced by the initial encryption undergoes further processing. The enhancement leverages specific fragmentation, the proposed ECCs and a simulated noisy channel. For legitimate users, the simulated noisy channel functions as a binary erasure channel, while for an attacker without the secret key, it acts as a channel with random deletions. Security notation and evaluation follow the traditional approach, assessing the attacker's advantage in distinguishing between two ciphertexts versus random guessing. This evaluation employs dedicated analysis based on information-theoretic findings on the capacity of certain deletion channels and is supported by illustrative numerical examples.

INDEX TERMS Integer codes, erasure correction, binary erasure channels, binary channel with random deletions, channel capacity, cryptographic security evaluation, lightweight encryption, security enhancement.

I. INTRODUCTION

In coding theory, there are many examples in which practical communication channels can be modeled as erasure channels. One example is modern communication networks, where messages are segmented into packets that are subsequently transmitted over the network. Some of these networks use erasure correcting codes (ECCs), thus enabling the receiver to reconstruct a certain number of packets that may be lost during transmission (see [1] and references therein). Similarly, in large-scale storage systems, where data is distributed across multiple storage units (such as disks or servers), ECCs are used to add redundant data segments. If some storage units fail, the constraints among the remaining units can be utilized to recover the original data (see [2] and references therein).

In the scenario described in this paper, the communication channel is modeled as an erasure channel from the

The associate editor coordinating the review of this manuscript and approving it for publication was Chien-Ming Chen.

receiver's perspective, whereas from the attacker's viewpoint it appears as a deletion channel. This is implemented by omitting two bits from each b-bit data byte ($b > 2$) at known erasure positions before transmitting the codeword. The positions of the omitted bits are known to both the transmitter and the receiver but remain concealed from the attacker.

In order for the receiver to be able to reconstruct the described codeword, the data must be encoded/decoded with ECCs having a very large minimum distance (MD). This requirement renders capacity-approaching (CA) codes, such as Polar and Low-Density Parity-Check (LDPC) codes, unsuitable for the potential application (CA codes do not have a clearly defined MD, while those for which it is possible to numerically determine the MD have high redundancy and/or enormous code lengths [3], [4], [5]). On the other hand, Reed-Solomon (RS) codes have a predefined MD and code rate. However, these codes would also be impractical in the present case, as correcting k erasure byte errors would require the use of k check-bytes [6]. An additional drawback of all these codes is that they are complex to encode/decode. In particular, in [7] and [8] it was shown that LDPC codes can be encoded in linear or quasi-linear time, whereas their decoding algorithms run in linear or log-linear time [9], [10]. Polar codes, on the other hand, can be encoded/decoded in log-linear time [11], [12], while the encoding/decoding procedures for RS codes have quasi-log-linear time complexity [13], [14].

Given all the above, in this paper, we will use dedicated integer ECCs that can correct k b-bit data bytes, each affected by exactly two erasures. The construction method of these codes is similar to those presented in [15], [16], and [17], which means that the codeword consists of k data bytes and only one check byte. Besides being rate-efficient, the proposed codes have the potential to run very fast in software, as they can be encoded/decoded in logarithmic time using one's complement arithmetic [18].

A. MOTIVATION FOR THE WORK

The growing need for frequent encryption and decryption operations in cyberspace underscores the importance of improving these techniques. This demand brings two key requirements: (i) minimizing implementation and operational overheads, and (ii) enhancing the security of encryption methods. Reducing overheads suggests the use of lightweight encryption techniques, which might only offer claimed computationally security. Recently, several methods have been proposed to construct encryption techniques, with a particular emphasis on coding-based methods to boost cryptographic security. Our goal is to provide additional advances within this direction proposing a suitable ECC and its application for enhancing cryptographic security. In addition, a targeted objective regarding ECC is the minimization of implementation and operational overhead, based on a design that employs integer arithmetic and exhibit exceptionally low redundancy.

B. SUMMARY OF THE RESULTS

The main contributions of this paper are summarized as follows.

1) A novel ECC is developed that belongs to the class of integer codes, which means that the codeword consists of k data bytes and one check byte. The proposed ECC employs integer arithmetic to encode and decode data bits and is capable of correcting k data bytes, each affected by exactly two erasures. The proposal provides a detailed explanation of the encoding and decoding algorithms, presents examples that illustrate the operation of the proposed codes, and explains the implementation advantages of the proposed codes over standard ones.
2) We employ the developed code as a dedicated one for the simulated noisy channel to enhance the cryptographic security of certain lightweight encryption schemes. In the security enhanced scheme, the noisy channels simulator degrades the encoded ciphertext by omitting certain bits. The simulator's control ensures that the party with the secret key knows the positions of the omitted bits, while an attacker without the key faces a difficult decoding challenge after a binary channel with deletion errors, thereby enhancing security.
3) Security evaluation is performed using traditional adversarial indistinguishability experiments and results on the capacity of communication channels with bit deletions. The considered indistinguishability experiment addresses the attacker's advantage in distinguishing between two ciphertexts versus random guessing.
4) The obtained security gain and implementation complexity are analyzed. The analytical results on the security gain are also illustrated numerically. The analysis provides guidelines for selecting the parameters of the proposed encryption scheme in order to achieve the desired cryptographic security and implementation complexity goals.

C. ORGANIZATION OF THE PAPER

Section II provides a summary of previous work relevant to this study. Section III proposes an error correction coding scheme for certain binary channels with erasures. Section IV introduces a security-enhanced encryption scheme based on the developed code. Section V evaluates the security of the proposed encryption scheme, while Section VI discusses the obtained security gain. The implementation complexity of the security enhancement components is addressed in Section VII. Finally, Section VIII concludes with several remarks. All proofs are given in the Appendix.

II. BACKGROUND

Enhancing the security and expanding the security margin of cryptographic primitives by integrating randomness has been explored in various designs, initially discussed in [19] and [20], as well as within the context of wiretap channel coding.

In symmetric key encryption techniques, two primary approaches can be identified. The first approach involves using a cryptographic key to govern error correction coding algorithms, as demonstrated in several studies, including [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32]. The second approach emphasizes using error correction coding and noisy channels to enhance the security of an encryption scheme, as reported in sources such as [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], and [44]. These security enhancements are based on the paradigms of additive noisy channels or channels with synchronization errors.

In the first approach, where encoding and decoding are controlled by a secret key, long secret keys are necessary because the error correction coding scheme must remain confidential. Conversely, the second approach, which uses error correction coding and noisy channels for security enhancement, allows for shorter secret keys, since the coding scheme itself does not need to be kept secret.

Several encryption techniques based on secret coding schemes have been proposed. The Rao-Nam (RN) cryptosystem [31] uses simple codes and employs a random non-singular invertible matrix, a generator matrix of a block code, and a permutation matrix for encryption. A variant of the RN cryptosystem using quasi-cyclic (QC) LDPC (QC-LDPC) codes was proposed in [32], and a nonlinear RN-like symmetric key encryption scheme was reported in [28], where QC-LDPC lattice codes are used. QC-LDPC codes have also been utilized in other encryption schemes, such as the one in [22], which eliminates permutation and scrambling matrices, and the scheme in [23], which randomly inserts and deletes bits in a QC-LDPC codeword.

Polar codes have been the foundation for several encryption designs. These codes have been employed in efficient secret key cryptosystems, where the generator matrix is kept secret from potential attackers, as seen in [21], [25], [26], [27], [29], and [30]. For example, some of these designs have utilized polar codes for physical layer encryption (PLE) [27], encryption based on chaotic sequences [29], and in conjunction with the Learning with Errors (LWE) problem [30]. Additionally, a scheme based on a linear block code and a simulator of a channel with synchronization errors was proposed in [24], where the simulator performs operations such as bit flipping, deletion, and insertion based on secret key-controlled Linear Feedback Shift Registers (LFSRs). However, it has been shown in [45] that this particular approach is vulnerable. A review of polar code-based encryption approaches is provided in [21].

The concept of introducing adjustable noise into encrypted data, known as the wiretap channel, has been explored for enhancing the security of the Data Encryption Standard (DES) and other block ciphers. For instance, the secrecy enhancement of the DES block cipher operating in cipher feedback mode (CFB) with adjustable noise is examined in [42]. The statistical properties of errors in block-ciphered cryptosystems are analyzed in [46].

Two paradigms for security enhancement have been discussed: encode → encrypt → degrade and encrypt → encode → degrade. These paradigms involve the use of additive noise, error correction coding, and simulations of various channels. Certain encryption techniques based on hard learning problems, such as Learning Parity in Noise (LPN) and LWE problems, have been used to achieve provable security guarantees. The security enhancements have been evaluated from both information-theoretic and computational complexity perspectives. The enhancement of security in stream ciphers is addressed using the "encode → encrypt → degrade" paradigm in [33], [34], [36], [41], and [43], while the "encrypt → encode → degrade" paradigm is discussed in [35], [37], [38], [39], and [44].

This paper focuses on encryption approaches that involve non-secret channel coding combined with noisy channels for security enhancement. The key components of these schemes include the initial encryption method, the employed channel coding, and the paradigm of the noisy channel. In the case of this paper, the noisy channel is modeled such that, from the receiver's perspective, it appears as an erasure channel, whereas from the attacker's viewpoint, it is effectively a deletion channel. In such a scenario, the use of integer codes is significantly more efficient than the use of traditional codes. The reason lies in the fact that integer codes are application-specific, i.e. that they are always constructed for particular (desired) types of channels in order to correct errors of a given type. Recently, in [47], integer codes that can correct up to two bit errors in a b-byte and can simultaneously correct some configurations of three or more erroneous bits, though not all possible ones, have been reported. The codes presented in this paper, however, are much more similar to the integer codes proposed in [48], [49], and [50].

III. INTEGER CODES FOR CORRECTION TWO ERASURES PER DATA BYTE

A. CONSTRUCTION

Before describing the encoding and decoding algorithms, we give two definitions that are necessary for understanding the concept of integer ECC.

Definition 1: An error is called a $2/k$–erasure if each of the k data bytes is affected by two erasures.

Definition 2: [39] Let $\mathcal{Z}_{2^b-1} = \{0, 1, \ldots, 2^b - 2\}$ be the ring of integers modulo 2^b–1 and let $B_j = \sum_{n=0}^{b-j} a_n \cdot 2^n$ be the integer representation of a b-bit byte, where $a_n \in \{0, 1\}$ and $1 \leq i \leq k$. Then, the code $C(b, k, c)$, defined as $\mathcal{C}(b, k, c) = \left\{(B_1, B_2, \ldots, B_k, B_{k+1}) \in \mathcal{Z}_{2^b-1}^{k+1} : \sum_{i=1}^{k} C_i \cdot B_i \equiv B_{k+1} (\mathrm{mod}\ 2^b - 1)\right\}$ is an $(kb + b, kb)$ integer ECC, where $c = (1, C_2, C_3, \ldots, C_k) \in \mathcal{Z}_{2^b-1}^{k}$ is the coefficient vector and $B_{k+1} \in \mathcal{Z}_{2^b-1}$ is an integer.

Before transmitting the codeword, the sender will omit two bits from each b-bit data byte at predetermined positions (the check-byte will be sent in the original form). Due to this, the sent codeword will have $(b-2)\cdot k+b$ bits instead of $(b+1)\cdot k$ bits (Fig. 1).

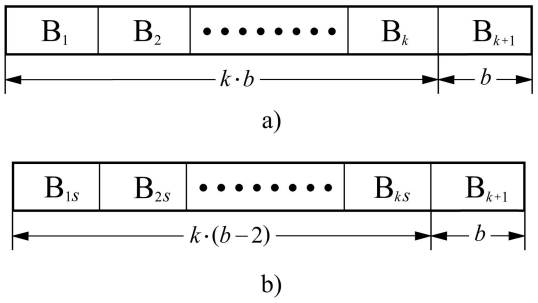

FIGURE 1. The codeword structure after: (a) the encoding process and (b) the bit omission process.

Upon receiving such a codeword, the decoder will insert $2k$ zeros into the known erasure positions. This means that the integer value of each "reconstructed" data byte will be reduced by $\mathcal{E} = 2^r \cdot x + 2^s \cdot y$, where $0 \le r < s \le b-1$ and $0 \le x, y \le 1$. Having this in mind, we can give the following definitions.

Definition 3: Let $V = \{0, 1\}$ and $P = \{0, 1, \ldots, b-1\}$. Then, the vectors representing the values and positions of the bits omitted at known erasure positions are respectively defined by

$$v = (v_1, v_2, \ldots, v_{2k}) \in \mathcal{V}^{2k}$$
$$p = (p_1, p_2, \ldots, p_{2k}) \in \mathcal{P}^{2k}.$$

Definition 4: Let $x = (B_1, B_2, \ldots, B_k, B_{k+1}) \in \mathcal{Z}_{2^b-1}^{k+1}$ be the original codeword and let $y = (\bar{B}_1, \bar{B}_2, \ldots, \bar{B}_k, \bar{B}_{k+1}) \in \mathcal{Z}_{2^b-1}^{k+1}$ be the received codeword in which two bits (of the known position) within each b-bit data byte are replaced by binary zeros. Then, the syndrome $\mathcal{S}$ of the received codeword is defined as

$$\begin{aligned}\mathcal{S} &= B_{k+1} - \sum_{i=1}^{k} \bar{B}_i \cdot C_i (\mathrm{mod} 2^b - 1) \\ &= \sum_{i=1}^{k} (B_i - \bar{B}_i) \cdot C_i (\mathrm{mod} 2^b - 1) \\ &= \sum_{i=1}^{k} e_i \cdot C_i (\mathrm{mod}\ 2^b - 1)\end{aligned}$$

where $e_i \in \{v_{2i-1} \cdot 2^{p_{2i-1}} + v_{2i} \cdot 2^{p_{2i}}\}$.

Definition 5: The set of syndromes corresponding to $2/k$–erasures is defined as

$$\varepsilon = \sum_{i=1}^{k} (v_{2i-1} \cdot 2^{p_{2i-1}} + v_{2i} \cdot 2^{p_{2i}}) \cdot C_i (\mathrm{mod}\ 2^b - 1).$$

From the above it is clear that the original codeword will be instantly reconstructed if $S = 0$. However, since the value of the vector v is not pre-known to the decoder, the question arises under which conditions it can reconstruct the original codeword when $S \neq 0$. The answer is given by the following theorem.

Theorem 1: Let $\xi \in \varepsilon \setminus \{0\}$ be the error set for $(kb+b, kb)$ integer codes correcting $2/k$-erasures. Then, $|\xi| = 4^k - 1$, where $|\xi|$ is the cardinality of ξ.

The proof is given in the Appendix A. Theorem 1 shows that the number of elements ξ is known in advance, but not their values. They will be known when the vector c is found with the help of a computer. The vector c is also needed to construct the syndrome table (ST) that enables the decoder to reconstruct the original codeword. Although it may seem complicated, the process of generating the ST is very simple and can be guessed from the proof of Theorem 1. Namely, since there is a 1:1 mapping between the syndrome S and the vector v, the ST can be automatically constructed by generating the elements of the set ξ (Fig. 2).

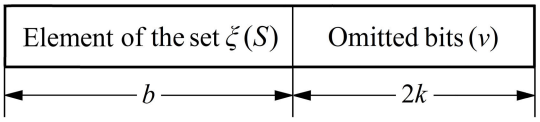

FIGURE 2. Bit-width of one ST entry.

Based on this fact, it is easy to conclude that the process of reconstructing the original codeword consists of two steps: finding the entry with the first b bits as that of the syndrome S and modifying the initially reconstructed codeword by XORing the vector v with the inserted binary zeros.

B. ENCODING & DECODING PROCEDURES AND ILLUSTRATIVE EXAMPLES

The pseudocodes of the encoding and decoding processes are shown in Fig. 3, while Fig. 4 illustrates the application of the proposed codes in a communication system.

To further clarify the operation of the proposed codes, we provide the following illustrative examples.

Example 1. Let $b = 7$, $k = 3$ and $p = (3, 4, 3, 5, 2, 5)$. By using a computer search, the sender and receiver will generate the coefficient vector $c = (1, 3, 22)$, while the receiver will additionally generate the ST having $|\xi| = 4^3-1=63$ entries (Table 1). Now, suppose that the sender wants to transmit 21 bits, say $m = (B_1, B_2, B_3) = (0101010_2, 1101111_2, 1000100_2) = (42, 111, 68)$. In that case, the value of the check-byte will be equal to

$$\begin{aligned}B_{k+1} = B_4 &= 1 \cdot 42 + 3 \cdot 111 + 22 \cdot 68\ (\mathrm{mod} 127) \\ &= 93 = 10111001_2\end{aligned}$$

and the codeword will have 28 bits, $x = (B_1, B_2, B_3, B_4) = (0101010_2, 1101111_2, 1000100_2, 1011101_2) = (42, 111, 68, 93)$. Given the value of the vector p, the sender will omit the 4th and 5th bits from the first byte, the 4th and 6th bits from the second byte and the 3rd and 6th bits from the third byte. As a result, the shortened codeword will have 22 bits, $x_s = (B_{1s}, B_{2s}, B_{3s}, B_4) = (01010_2, 11011_2, 10010_2, 1011101_2)$. When it receives such

ENCODER
//Encoding process//
Input: m = $(B_1, B_2, ..., B_k)$, $c = (1, C_2, ..., C_k)$, $p = (p_1, p_2, ..., p_{2k})$;
$B_{k+1} = 0$;
for $i = 1$ **to** k
  $B_{k+1} = C_i \cdot B_i + B_{k+1}$;
end
$x = (B_1, B_2, ..., B_k, B_{k+1})$; // the codeword with $b \cdot k + b$ bits
// Omitting two bits per data byte at agreed erasure positions //
for $i = 1$ **to** k
  omit two bits within the B_i at positions p_{2i-1} and p_{2i}
end
Output: $x_s = (B_{1s}, B_{2s}, ..., B_{ks}, B_{k+1})$; // the codeword with $(b-2) \cdot k + b$ bits

DECODER
// Inserting binary zeros at agreed erasure positions //
Input: $x_s = (B_{1s}, B_{2s}, ..., B_{ks}, B_{k+1})$, $c = (1, C_2, ..., C_k)$, $p = (p_1, p_2, ..., p_{2k})$, ST;
for $i = 1$ **to** k
  insert two binary zeros within the B_i at positions p_{2i-1} and p_{2i}
end
$y = (\overline{B}_1, \overline{B}_2, ..., \overline{B}_k, B_{k+1})$; // the initially reconstructed codeword (IRC)
// Reconstructing the original codeword //
$\overline{B}_{k+1} = 0$;
for $i = 1$ **to** k
  $\overline{B}_{k+1} = C_i \cdot \overline{B}_i + \overline{B}_{k+1}$;
end
$S = B_{k+1} - \overline{B}_{k+1}$;
Step 1. Use the value of S and lookup the ST to get the vector v
Step 2. Modify the IRC by XORing the vector v with the inserted binary zeros
Output: $y = x = (B_1, B_2, ..., B_k, B_{k+1})$;

FIGURE 3. Pseudocodes of encoding and decoding procedures.

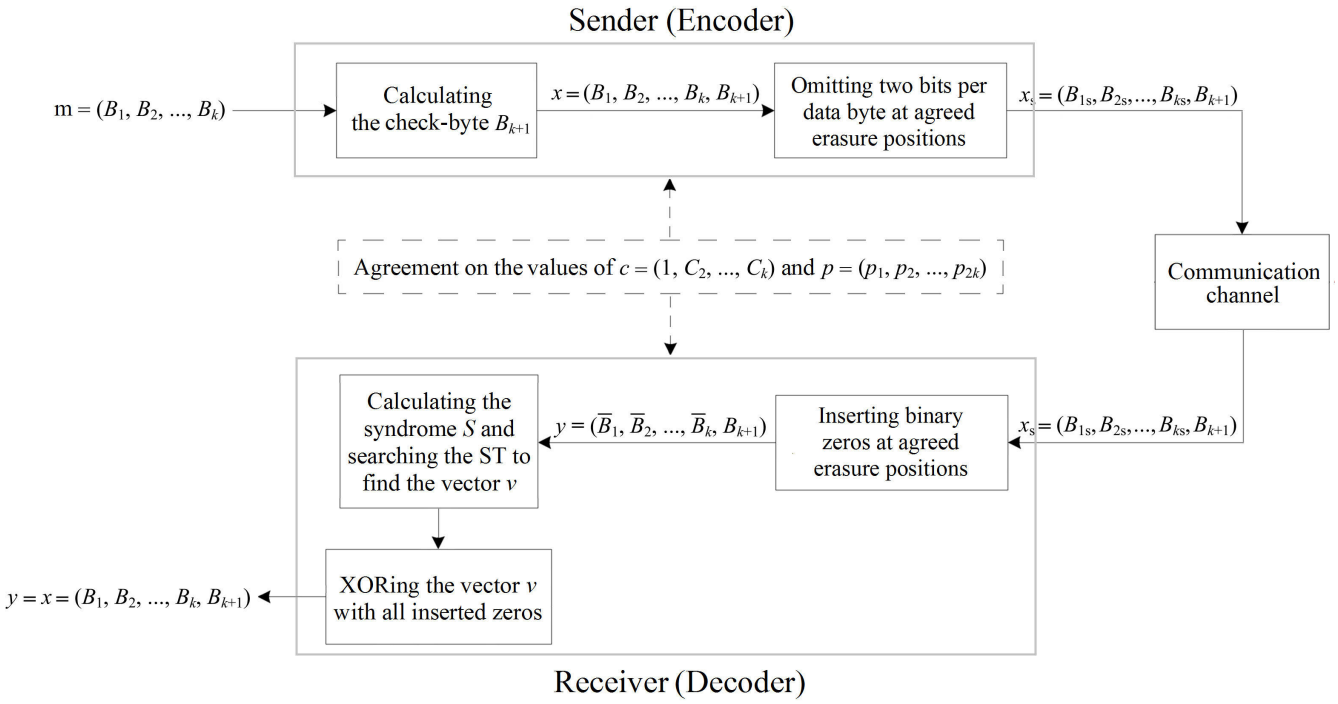

FIGURE 4. Framework of the proposed integer ECC employment.

a codeword, the receiver will first insert binary zeros at the known erasure positions,

$$y = (010\bar{0}\bar{0}10_2, 110\bar{0}1\bar{0}1_2, 10\bar{0}0\bar{1}0\bar{0}_2, 1011101_2) = (34, 101, 68, 93).$$

After that, it will calculate the value of the syndrome S

$$S = 93 - (1 \cdot 34 + 3 \cdot 101 + 22 \cdot 68) \ (mod 127) = 38$$

and lookup the ST to get the value of the vector v. When this procedure is completed, the receiver will modify the initially reconstructed codeword by XORing the vector $v = (1, 0, 1, 1, 0, 0)$ with the inserted binary zeros. As a result, the codeword will have the form $y = x = (B_1, B_2, B_3, B_4) = (0101010_2, 1101111_2, 1000100_2, 1011101_2)$.

Example 2. Let $b = 7$, $k = 3$ and $p = (1, 6, 2, 5, 3, 4)$. By using a computer search, the sender and receiver will generate the coefficient vector $c = (1, 3, 46)$, while the receiver will additionally generate the ST having $|\xi| = 4^3 - 1 = 63$ entries (Table 2). Suppose the sender wants to send the same data again, $m = (B_1, B_2, B_3) = (0101010_2, 1101111_2, 1000100_2) = (42, 111, 68)$. In that case, it will calculate the check-byte

$$B_{k+1} = B_4 = 1 \cdot 42 + 3 \cdot 111 + 46 \cdot 68 \ (mod 127) = 1001010_2$$

TABLE 1. The ST for the (28, 21) integer 2/7–erasure correcting code when $p = (3, 4, 3, 5, 2, 5)$.

	S	v		S	v		S	v		S	v		S	v
1	1	001110	**14**	18	110100	**27**	36	111000	**40**	53	101111	**53**	86	111101
2	3	101010	**15**	19	010011	**28**	38	101100	**41**	54	010101	**54**	98	000010
3	4	010000	**16**	21	000111	**29**	39	001011	**42**	56	110001	**55**	102	010010
4	5	011110	**17**	23	100011	**30**	42	111100	**43**	57	111111	**56**	104	000110
5	6	000100	**18**	24	001000	**31**	43	011011	**44**	58	100101	**57**	106	100010
6	7	111010	**19**	25	010111	**32**	44	000001	**45**	62	110101	**58**	108	010110
7	8	100000	**20**	27	110011	**33**	45	001111	**46**	68	001001	**59**	110	110010
8	9	101110	**21**	28	011000	**34**	47	101011	**47**	72	011001	**60**	112	100110
9	10	010100	**22**	29	100111	**35**	48	010001	**48**	74	001101	**61**	116	110110
10	12	110000	**23**	30	001100	**36**	49	011111	**49**	76	101001	**62**	122	001010
11	13	111110	**24**	32	101000	**37**	50	000101	**50**	78	011101	**63**	126	011010
12	14	100100	**25**	33	110111	**38**	51	111011	**51**	80	111001			
13	15	000011	**26**	34	011100	**39**	52	100001	**52**	82	101101			

TABLE 2. The ST for the (28, 21) integer 2/7–erasure correcting code when $p = (1, 6, 2, 5, 3, 4)$.

	S	v		S	v		S	v		S	v		S	v
1	1	010000	**14**	32	100000	**27**	51	010111	**40**	80	101000	**53**	99	011111
2	3	101111	**15**	33	110000	**28**	54	001100	**41**	81	111000	**54**	105	001001
3	4	111111	**16**	35	001010	**29**	55	011100	**42**	82	100111	**55**	106	011001
4	6	000100	**17**	36	011010	**30**	57	000001	**43**	83	110111	**56**	111	001101
5	7	010100	**18**	38	100100	**31**	58	010001	**44**	86	101100	**57**	112	011101
6	10	101001	**19**	39	110100	**32**	63	000101	**45**	87	111100	**58**	114	000010
7	11	111001	**20**	41	001110	**33**	64	010101	**46**	89	100001	**59**	115	010010
8	16	101101	**21**	42	011110	**34**	67	101010	**47**	90	110001	**60**	120	000110
9	17	111101	**22**	44	000011	**35**	68	111010	**48**	92	001011	**61**	121	010110
10	19	100010	**23**	45	010011	**36**	73	101110	**49**	93	011011	**62**	124	101011
11	20	110010	**24**	48	001000	**37**	74	111110	**50**	95	100101	**63**	125	111011
12	25	100110	**25**	49	011000	**38**	76	100011	**51**	96	110101			
13	26	110110	**26**	50	000111	**39**	77	110011	**52**	98	001111			

after which the codeword with 28 bits will be formed, $x = (B_1, B_2, B_3, B_4) = (0101010_2, 1101111_2, 1000100_2, 1001010_2) = (42, 111, 68, 74)$.

Given the value of the vector p, the sender will omit the 2nd and 7th bits from the first byte, the 3rd and 6th bits from the second byte and the 4th and 5th bits from the third byte. As a result, the shortened codeword will have 22 bits, $x_s = (B_{1s}, B_{2s}, B_{3s}, B_4) = (00101_2, 11111_2, 10000_2, 1001010_2)$. After receiving such a codeword, the receiver will first insert binary zeros at the known erasure positions, $y = (0\bar{0}101\bar{0}_2, 11\bar{0}11\bar{0}1_2, 100\bar{0}\bar{0}00_2, 1001010_2) = (10, 109, 64, 74)$, and then calculate the value of the syndrome S

$$S = 74 - (1 \cdot 10 + 3 \cdot 109 + 46 \cdot 64) \,(mod 127) = 95.$$

Since $S \neq 0$, the receiver will lookup the ST to get the value of the vector v. After that, in the next step, it will modify the initially reconstructed codeword by XORing the vector $v = (1, 0, 0, 1, 0, 1)$ with the inserted binary zeros. As a result, the codeword will have the form $y = x = (B_1, B_2, B_3, B_4) = (0101010_2, 1101111_2, 1000100_2, 1001010_2)$.

For the sake of completeness, we note that experiments have shown that the proposed codes can be constructed for values of $b \geq 6$, where the inequality $2k + 1 \leq b$ always holds.

IV. EMPLOYMENT OF THE PROPOSED INTEGER CODE FOR THE ENCRYPTION SECURITY ENHANCEMENT

Building on previously reported methods for enhancing the cryptographic security of lightweight encryption techniques (see for example [37], [38], [39]), this section summarizes a novel variant of the approach.

The considered security enhancement is based on post-processing, of the initially generated ciphertext, that includes dedicated error-correction coding and a simulated noisy channel. We employ the ECC presented in Section III, and the simulated noisy channel that appears as a binary erasure channel for legitimate receivers and a random deletion channel for potential adversaries, resulting in a substantial increase in the difficulty of cryptanalysis.

The proposed security-enhanced encryption scheme is displayed in Fig. 5.

The encryption operates as follows:

- The encryption algorithm generates ciphertext segments $c = Enc_k(m)$ and pseudorandom sequence segments s, where k and m are the secret key and message, respectively. The segments of s are arguments for certain functions that control the fragmenter and generate the vector $p = [p_1, p_2, \ldots, p_{2k}]$.

- The fragmeter, controlled by the sequence s, organizes c bits into q-bits segmants of k, b-bits bytes where k and b, $q = kb$, are the parameters and: (i) with the probability λ sends a q-bits fragment to the sub-channel 1, i.e. towards the encoder, or (ii) with the probability $1 - \lambda$ to the

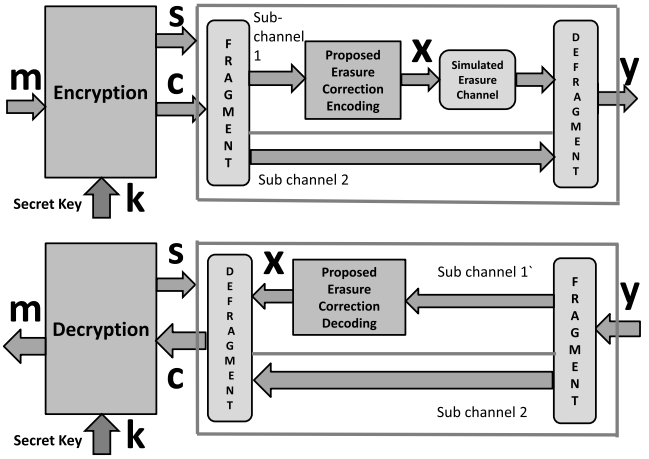

FIGURE 5. Security enhanced encryption with the error correction code proposed integer ECC.

error-free sub-channel 2 directly to the defragmenter, where λ, $0.5 < \lambda < 1$, is the parameter.

- The ciphertext c is encoded using erasure correction $Encode(c)$, producing a codeword x.

- The codeword x is then passed through a binary erasure channel controlled by the sequence s (the vector p), resulting in a degraded version y of the codeword x, where in each byte B_i, $i = 1, 2, \ldots, k$, two bits are erased on the positions controlled by the sequence s, and the parity byte B_{k+1} is kept error-free.

- The defragmenter, for each input segment of the sequence c, outputs the corresponding vector y.

The decryption process, corresponding to this encryption procedure, includes the following basic steps:

- The decryption algorithm generates pseudorandom sequence segments s identical to those produced during encryption.

- The received vector y is subjected to fragmentation and, with the probability λ, erasure correction decoding $Decode(y)$ if required, which recovers the error-free ciphertext c.

- The recovered ciphertext c is then used to retrieve the original message m as $m = Dec_k(c)$.

The security enhancement method is general and can be applied directly to block cipher encryption techniques involving block-by-block processing, as well as to certain stream ciphers where the encoded ciphertext segments are self-contained.

Note that an attacker does not have access to the secret key k and, therefore, cannot determine the vector s. As a result, while the legitimate receiver observes $y^{(n)}$ as the codeword $x^{(n)}$ after passing through the binary erasure channel, the attacker, lacking the sequence s, perceives $y^{(n)}$ as the codeword $x^{(n)}$ after passing through a binary deletion channel. To support this asymmetry in perception and enable secure decoding on the legitimate side, the system operates in a block-wise manner. While a traditional stream cipher processes the plaintext bit-by-bit, the proposed security enhancement relies on block-by-block processing of the initial ciphertext stream. This approach does not limit the total length of the ciphertext but, in order to maintain an efficient and implementable coding scheme, each block must be limited in size so that the memory requirements for storing the ST remain within practical bounds.

V. SECURITY EVALUATION OF THE ENHANCED ENCRYPTION

An attacker is confronted with the challenge of cryptanalysis in a known plaintext attack scenario, as illustrated in Fig. 6.

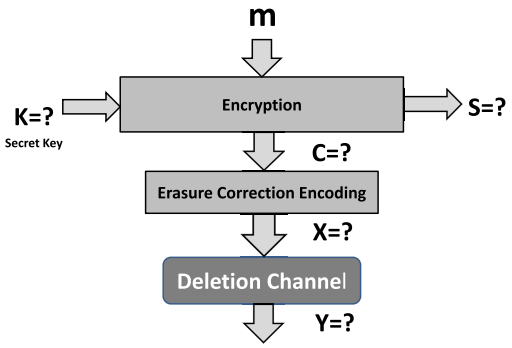

FIGURE 6. Model of encryption from the attacker's perspective under a known plaintext attack.

It's important to note that while the legitimate parties deal with the challenge of decoding after a binary erasure channel, the attacker faces a significantly more difficult task of decoding after a deletion channel. The attacker's knowledge is limited to the following: each byte of a codeword is independently transmitted through noisy sub-channel 1 with probability λ or through error-free sub-channel with probability $1 - \lambda$. If transmitted through sub-channel 1, in each codeword byte, except the last-parity byte, two bits are omitted and each one with the probability d. The attacker does not know the specific realization of the "individual channel selection events," meaning unawareness of from which sub-channel each output symbol was received.

A. PRELIMINARIES AND SECURITY NOTATION

Referring to Fig. 6 we consider a statistical model where m, c, x and y are realizations of the stochastic variables M, C, X and Y, respectively. and the main corresponding background statements are given in Table 3.

We use a traditional framework for analyzing cryptographic security that focuses on two key aspects: (i) defining what constitutes a "break" of the scheme, and (ii) outlining the assumed capabilities of the adversary. A cryptographic scheme is considered computationally secure if, for every probabilistic polynomial-time adversary $\mathcal{A}$ conducting a specified type of attack, and for any polynomial $p(n)$, there exists an integer N such that the probability of $\mathcal{A}$ succeeding

TABLE 3. Preliminaries.

$$\begin{aligned}
&\Pr(M=m_0|Y=y_0)=\frac{\Pr(M=m_0,Y=y_0)}{\Pr(Y=y_0)}\\
=\;&\frac{\sum_x\sum_c \Pr(M=m_0,Y=y_0,X=x,C=c)}{\Pr(Y=y_0)}\\
=\;&\frac{\sum_x\sum_c \Pr(Y=y_0|M=m_0,X=x,C=c)\Pr(M=m_0,X=x,C=c)}{\Pr(Y=y_0)}\\
=\;&\frac{\sum_x \Pr(Y=y_0|X=x)\sum_c \Pr(M=m_0,X=x,C=c)}{\Pr(Y=y_0)}\\
=\;&\frac{\sum_x \Pr(Y=y_0|X=x)\sum_c \Pr(M=m_0|X=x,C=c)\Pr(X=x,C=c)}{\Pr(Y=y_0)}\\
=\;&\frac{\sum_x \Pr(Y=y_0|X=x)\sum_c \Pr(M=m_0|C=c)\Pr(X=x|C=c)\Pr(C=c)}{\Pr(Y=y_0)}
\end{aligned}$$

Consequently, when C takes value c_0, we have:

$$\begin{aligned}
&\Pr(M=m_0|Y=y_0)=\\
=\;&\frac{\sum_x \Pr(Y=y_0|X=x)\Pr(M=m_0|C=c_0)\Pr(C=c_0)}{\Pr(Y=y_0)}\\
=\;&\Pr(M=m_0|C=c_0)\Pr(C=c_0)\frac{\sum_x \Pr(Y=y_0|X=x)}{\Pr(Y=y_0)}
\end{aligned}$$

Further on, we consider the security of encryption in the above statistical model.

(where the success of the attack is clearly defined) is less than $\frac{1}{p(n)}$ for all $n > N$. The following two definitions outline the security evaluation scenario and the corresponding security assertion.

Definition 6 ([51]): The Adversarial Indistinguishability Experiment consists of the following steps:

1) The adversary $\mathcal{A}$ chooses a pair of messages $(m_0; m_1)$ of the same length n, and passes them on to the encryption system for encrypting.
2) A bit $b \in \{0,1\}$ is chosen uniformly at random, and only one of the two messages $(m_0; m_1)$, precisely m_b, is encrypted into ciphertext $Enc(m_b)$ and returned to $\mathcal{A}$;
3) Upon observing $Enc(m_b)$, and without knowledge of b, the adversary $\mathcal{A}$ outputs a bit b_0;
4) The experiment output is defined to be 1 if $b_0 = b$, and 0 otherwise; if the experiment output is 1, denoted shortly as the event $(\mathcal{A} \to 1)$, we say that $\mathcal{A}$ has succeeded.

Definition 7 ([51]): An encryption scheme provides indistinguishable encryptions in the presence of an eavesdropper, if for all probabilistic polynomial-time adversaries $\mathcal{A}$

$$\Pr[\mathcal{A} \to 1|Enc(m_b)] \leq \frac{1}{2} + \varepsilon, \tag{1}$$

where $\varepsilon = negl(n)$ is a negligibly small function.

Definitions 6 and 7 are more precisely discussed in [51]. Please note that the encoding method used is a deterministic algorithm, ensuring it does not compromise the overall security of the encryption scheme. Assuming the decoding algorithm provides error-free decoding, it neither enhances nor diminishes the security of the encryption. The increase in the security margin arises from the use of a noisy channel, with the encoding simply correcting errors on the legitimate receiver's side.

B. ANALYSIS

We analyze the encryption system depicted in Fig. 5, considering that the legitimate parties share pseudo-random secret sequences instead of truly random ones. Our objective is to assess the advantage of the adversary $\mathcal{A}$ in the indistinguishability game defined in Definition 1, when $y \leftarrow Enc(m_b)$, where y is a specific realization of the random variable Y. This assessment assumes that the advantage of $\mathcal{A}$ is known when m_0 and m_1 are chosen realizations of the random variable M.

Proposition 1: Let the encryption mapping from M to C be such that the adversary $\mathcal{A}$'s advantage in the indistinguishability game (as defined in Definition 6) is $\frac{1}{2} + \varepsilon$ (as specified in Definition 7). Let $\Pr(X = x|Y = y)$ and $\Pr(Y = y|Z = z)$ represent the probability distributions associated with the noisy channels in the security-enhanced encryption scheme. Under these conditions, we have:

$$\Pr[\mathcal{A} \to 1|Y = y] = \frac{1}{2} + \varepsilon \cdot \Pr(X = x_b|Y = y).$$

Proposition 2: Let U and V be discrete random variables representing the input and output, respectively, of a communication channel. Suppose u and v are possible realizations of U and V, respectively, and the decision rule on U given V involves identifying a realization u given v. In the statistical model considered, the probability of correctly identifying v among 2^q categories, where $q > 1$, satisfies the following inequality:

$$\Pr(U = u|V = v) < \left(\frac{q(Cap^* - 1) + 1 + \log_2(2^q - 1)}{\log_2(2^q - 1)}\right)^q.$$

where Cap denotes the capacity of the noisy channel from the attacker's perspective.

Theorem 2:

$$Pr[\mathcal{A} \to 1|Y = y] \leq$$
$$\leq \left(\frac{-q(\log_2 \frac{8}{1+\sqrt{5}})(\sum_{p=1}^{2} \lambda_p d_p) + 1 + \log_2(2^q - 1)}{\log_2(2^q - 1)} \right)^q,$$

where $q = kb$, λ_p, $0 \leq d_p < 1$, $p = 1, 2$, are the parameters, $\sum_{p=1}^{2} \lambda_p = 1$.

VI. ANALYSIS OF A GAIN ON THE CRYPTOGRAPHIC SECURITY

It is important to note that implementing the security enhancement reduces the probability that an attacker can make a correct decision in the security game outlined in Definition 6. As stated in Proposition 1 and Theorem 2, the security enhancement diminishes the bias in favor of the attacker winning the game, and the reciprocal of this reduction can be interpreted as the security gain. Based on this, we define the security gain as follows.

Definition 8: The security gain γ is a measure of the lower bound on the parameter ε reduction, (in Definition 7) specified as follows:

$$\gamma(q, d_p, \lambda_p, p = 1, 2)$$
$$= \left(\frac{-q(\log_2 \frac{8}{1+\sqrt{5}})(\sum_{p=1}^{2} \lambda_p d_p) + 1 + \log_2(2^q - 1)}{\log_2(2^q - 1)} \right)^{-q}$$

Certain numerical illustrations of $\gamma(q, \lambda_p, d_p, p = 1, 2)$ are given in the following Table 4.

The following Figures 7 and 8 illustrate $\gamma(q, d_p, \lambda_p, p = 1, 2)$ and $\frac{1}{\gamma(\cdot)}$ as the function of q and λ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

TABLE 4. Numerical illustration of the cryptographic security gain $\gamma(q, \lambda_p, d_p, p = 1, 2)$ when the erasure-correction codes with the parameters k and $b = 7$ are employed, and $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

k, $(q = 7k)$, λ	lower bound on the security gain γ
$k = 3$ $(q = 21)$, $\lambda = 0.6$	58.617
$k = 3$ $(q = 21)$, $\lambda = 0.7$	155.13
$k = 3$,$(q = 21)$, $\lambda = 0.8$	430.46
$k = 4$ $(q = 28)$, $\lambda = 0.6$	342.28
$k = 4$ $(q = 28)$, $\lambda = 0.7$	1277.7
$k = 4$ $(q = 28)$, $\lambda = 0.8$	5089.8
$k = 5$ $(q = 35)$, $\lambda = 0.6$	2005.8
$k = 5$ $(q = 35)$, $\lambda = 0.7$	10564
$k = 5$ $(q = 35)$, $\lambda = 0.8$	60445
$k = 6$ $(q = 42)$, $\lambda = 0.6$	11776
$k = 6$ $(q = 42)$, $\lambda = 0.7$	87528
$k = 6$ $(q = 42)$,, $\lambda = 0.8$	719430
$k = 7$ $(q = 49)$, $\lambda = 0.6$	69213
$k = 7$ $(q = 49)$, $\lambda = 0.7$	726040
$k = 7$ $(q = 49)$, $\lambda = 0.8$	8573800

Since the security gain function exhibits exponential growth, which, when plotted on a linear scale, tends to obscure certain characteristics of the graphics

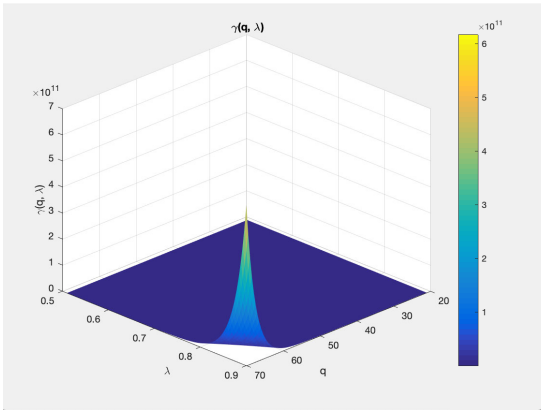

FIGURE 7. Illustration of $\gamma(q, d_p, \lambda_p, p = 1, 2, 3)$ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

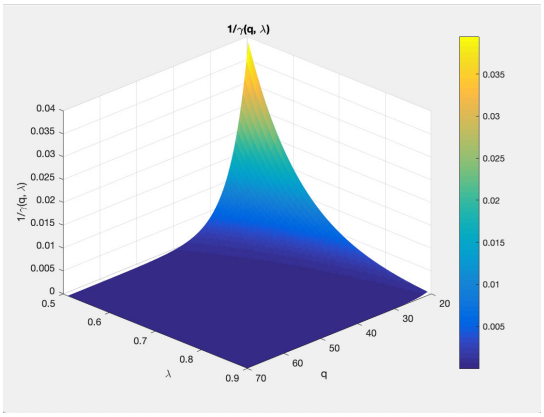

FIGURE 8. Illustration of $\frac{1}{\gamma(q, d_p, \lambda_p, p=1,2,3)}$ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

in Figures 7 and 8, it appears suitable to consider its logarithmic form, as well. Therefore, for simplicity in both presentation and analysis, instead of directly examining $\gamma(q, d_p, \lambda_p, p = 1, 2)$, we use the following security gain measure $\alpha(q, d_p, \lambda_p, p = 1, 2)$:

$$\alpha(q, d_p, \lambda_p, p = 1, 2)$$
$$= \log_2 \frac{1}{\gamma(q, d_p, \lambda_p, p = 1, 2)}$$
$$= \log_2 \left(\frac{-q(\log_2 \frac{8}{1+\sqrt{5}})(\sum_{p=1}^{2} \lambda_p d_p) + 1 + \log_2(2^q - 1)}{\log_2(2^q - 1)} \right)^q$$
$$= q\Bigg(\log_2 \Bigg(- q(\log_2 \frac{8}{1+\sqrt{5}})(\sum_{p=1}^{2} \lambda_p d_p) + 1$$
$$+ \log_2(2^q - 1)) - \log_2(\log_2(2^q - 1)) \Bigg) \Bigg)$$

Figures 9 and 10 illustrate $\alpha(q, d_p, \lambda_p, p = 1, 2)$ as the function of q and λ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

The examples shown in Fig. 9 and Fig. 10 suggest the potential for optimizing certain parameters of the security gain.

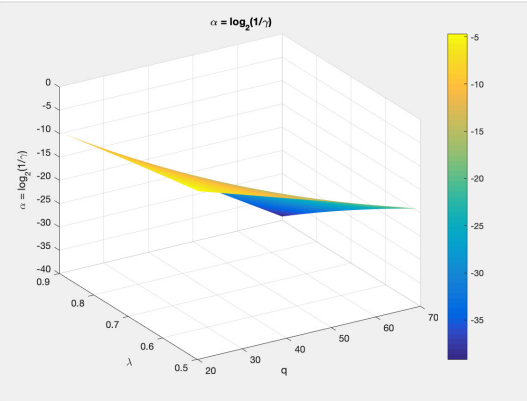

FIGURE 9. Illustration of $\alpha(q, d_p, \lambda_p, p = 1, 2, 3)$ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

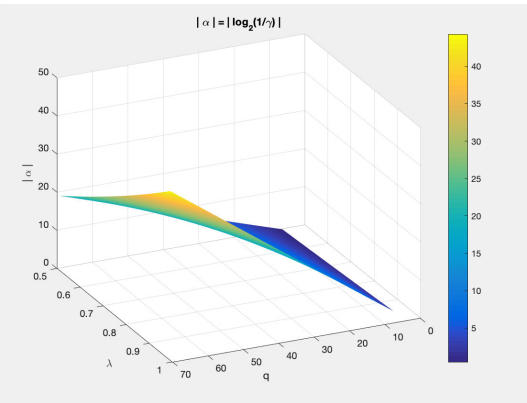

FIGURE 10. Numerical illustration of $|\alpha(q, d_p, \lambda_p, p = 1, 2, 3)|$ when $q = kb$, $b = 7$, $d_1 = \frac{2}{7}$, $d_2 = 0$, and $\lambda_1 = \lambda$, $\lambda_2 = 1 - \lambda$, $0.5 < \lambda < 1$.

VII. IMPLEMENTATION COMPLEXITY OF THE COMPONENTS FOR THE SECURITY ENHANCEMENT

A. A SUMMARY ON TIME COMPLEXITY OF THE ENCODING AND DECODING PROCEDURES

In the analyzed scenario, the use of traditional linear codes (LCs), such as Polar, LDPC, and RS codes, would be problematic not only in terms of redundancy and the complexity of the encoding/decoding procedures, but also from the perspective of practical implementation. Specifically, if LCs were used, a hardware-based receiver would be extremely complex, while a software-based receiver would significantly reduce data transmission rates. This is because traditional LCs rely on finite field (FF) arithmetic, which general-purpose processors (GPPs) do not natively support. As a result, GPPs must emulate FF arithmetic, and thus to perform tens of operations to process one bit [52], [53], [54]. On the other hand, [18] demonstrated that all integer codes are well-suited for implementation on GPPs. Furthermore, it was shown in [18] that, if a binary tree structure is used: (i) any integer encoder requires $\lceil \log_2 k \rceil + 1$ operations to compute the check byte B_{k+1}, (ii) any integer decoder requires $\lceil log_2(k + 1) \rceil + 1$ operations to generate the syndrome S, and in addition $\lfloor log_2 |\xi| \rfloor + 2$ operations for the codeword recovery.

TABLE 5. The coding average computational overheads per a bit of ciphertext and memory overhead in the proposed security enhancement encryption scheme.

	proposed encryption scheme
computational encoding overhead per a bit of ciphertext	$\sim \frac{\lambda}{kb} (\lceil \log_2 k \rceil + 1)$
computational decoding overhead per a bit of ciphertext	$\sim \frac{\lambda}{kb} (\lceil \log_2 (k+1) \rceil + \lfloor log_2 (4^k - 1) \rfloor + 3)$
memory overhead for decoding	$(4^k - 1) \times (2k + b)$

Essentially, this means that the only difference among various classes of integer codes lies in the amount of memory required by the decoder to store the ST. In the case of the proposed codes, the size of the ST grows exponentially with the number of data bytes [the ST has $4^k - 1$ entries, where each entry is $(2k + b)$-bits wide]. Because of this, the direct application of the proposed codes to long ciphertexts becomes memory-demanding. However, this limitation can be effectively addressed by partitioning the ciphertext into smaller blocks, typically a few hundred bits each, which are then individually encoded. This strategy allows the proposed codes to be applied even to long ciphertexts, while keeping the memory requirements within reasonable bounds (on the order of a few megabytes)

B. IMPLEMENTATION COMPLEXITY OF SIMULATED NOISY CHANNEL AND A SUMMARY OF THE OVERHEAD

The implementation of the simulated noisy channel includes: (i) the implementation of the output function that provides the sequence s from the encryption scheme; (ii) the implementation of the functions that map segments s; and (iii) the byte-by-byte implementation of the simulated noisy channel.

The output function that provides the control sequence for the simulation of the noisy channel could be a simple look-up table that implements a substitution box for example, and accordingly, it can be efficiently implemented.

The functions that map segments s perform certain hashing operations over the successive segments of the sequence s of length a, where a is a parameter. Taking into account that a is small, one option is to evaluate these functions employing two column look-up tables with 2^a rows. Another option is to employ, as the mapping functions those with a low-complexity algebraic evaluation. Final implementation of the noisy channel is simple because it requires just erasure of two bits from each data byte. So, the noisy channel simulator has a low complexity of the implementation. According to the above discussion, the implementation complexity of the coding dominates over the implementation of the noisy channel simulation.

Table 5 summarizes the implementation complexity overhead of the proposed security enhanced encryption scheme.

VIII. CONCLUSION

This paper presents a novel integer ECC and its application for enhancing the security of certain encryption schemes. The codeword of the proposed code consists of k data bytes and a single check byte. In addition to being rate-efficient, the proposed codes are well-suited for software implementation, as both encoding and decoding can be performed in logarithmic time using one's complement arithmetic.

The security enhancement arises from the asymmetry in how the ciphertext is perceived: while the legitimate receiver obtains the encoded ciphertext degraded by a binary erasure channel, an attacker, lacking access to the secret key, observes a sequence resembling transmission through a random deletion channel. It is shown that this construction provides a provable enhancement of cryptographic security in the information-theoretic sense. This statement results from the analysis based on combination of the traditional notation of the encryption security and results on capacity bounds of the deletion channels. The derived analytical results on the security gain are illustrated with numerical examples.

From a practical standpoint, the main constraint of the proposed security-enhanced scheme lies in the memory required to store the ST. This limitation, however, can be effectively addressed by partitioning long ciphertexts into smaller blocks, which are then individually encoded. Under this approach, the overall memory overhead remains modest, making the scheme suitable for a wide range of real-world applications Also, certain implementation issues, including diverse and implementation-related attacking scenarios, as well as evaluation of ECC implementation, could be subject of further work for enhancing the overall robustness and applicability of the proposed scheme.

APPENDIX A
THE PROOFS

Proof of Theorem 1: From Definition 5 we indirectly know that the error set can be expressed as

$$\xi = \cup_{i=1}^{4^k-1} s_i$$

where

$$\begin{aligned}
s_1 &= \{0+0+\ldots+0+(0\cdot 2^{p_{2k-1}}+1\cdot 2^{p_{2k}})\\
&\quad \cdot C_k \ (\mathrm{mod}\ 2^b-1)\},\\
s_2 &= \{0+0+\ldots+0+(1\cdot 2^{p_{2k-1}}+0\cdot 2^{p_{2k}})\\
&\quad \cdot C_k \ (\mathrm{mod}\ 2^b-1)\},\\
s_3 &= \{0+0+\ldots+0+(1\cdot 2^{p_{2k-1}}+1\cdot 2^{p_{2k}})\\
&\quad \cdot C_k \ (\mathrm{mod}\ 2^b-1)\},\\
s_4 &= \{0+0+\ldots+0+(0\cdot 2^{p_{2k-3}}+1\cdot 2^{p_{2k-2}})\cdot C_{k-1}\\
&\quad +(0\cdot 2^{p_{2k-1}}+0\cdot 2^{p_{2k}})\cdot C_k \ (\mathrm{mod}\ 2^b-1)\},\\
s_5 &= \{0+0+\ldots+0+(0\cdot 2^{p_{2k-3}}+1\cdot 2^{p_{2k-2}})\cdot C_{k-1}\\
&\quad +(0\cdot 2^{p_{2k-1}}+1\cdot 2^{p_{2k}})\cdot C_k \ (\mathrm{mod}\ 2^b-1)\},\\
s_6 &= \{0+0+\ldots+0+(0\cdot 2^{p_{2k-3}}+1\cdot 2^{p_{2k-2}})\cdot C_{k-1}\\
&\quad +(1\cdot 2^{p_{2k-1}}+0\cdot 2^{p_{2k}})\cdot C_k(\mathrm{mod}\ 2^b-1)\},\\
s_7 &= \{0+0+\ldots+0+(0\cdot 2^{p_{2k-3}}+1\cdot 2^{p_{2k-2}})\cdot C_{k-1}\\
&\quad +(1\cdot 2^{p_{2k-1}}+1\cdot 2^{p_{2k}})\cdot C_k(\mathrm{mod}\ 2^b-1)\},\\
s_{4^k-1} &= \{(1\cdot 2^{p_1}+1\cdot 2^{p_2})\cdot 1+(1\cdot 2^{p_3}+1\cdot 2^{p_4})\\
&\quad \cdot C_2+\cdots+(1\cdot 2^{p_{2k-1}}+1\cdot 2^{p_{2k}})\\
&\quad \cdot C_k \ (\mathrm{mod}\ 2^b-1)\},
\end{aligned}$$

Since the value of the vector $p=(p_1,p_2,\ldots,p_{2k})\in P^{2k}$ is pre-known to the decoder, it is clear that the above subsets will be nonzero and mutually different only if there exists the coefficient vector $c=(1,C_2,C_3,\ldots,C_k)\in \mathcal{Z}^k_{2^b-1}$ such that $s_1\cap s_2\cap s_3\cdots\cap s_{4^k-1}=\oslash$.

If we add to this the fact that each subset has only one element, we obtain that

$$|\xi|=\sum_{i=1}^{4^k-1}|s_i|=(4^k-1)\cdot 1=4^k-1.$$

Q.E.D.

Proof of Proposition 1: For simplicity, Proposition 1 addresses a restricted case where the adversary $\mathcal{A}$'s advantage in the indistinguishability game is $\frac{1}{2}+\varepsilon$. Let the index b of the selected message be a realization of the random variable B, reflecting the output distribution of the adversary $\mathcal{A}$. The probability $\Pr(B=b|Y=y)$ that $\mathcal{A}$ wins the game is determined as follows:

$$\begin{aligned}
\Pr(B=b|Y=y)&=\frac{\Pr(B=b,Y=y)}{\Pr(Y=y)}\\
&=\frac{\sum_x \Pr(B=b,Y=y,X=x)}{\Pr(Y=y)}\\
&=\frac{\sum_x \Pr(B=b|Y=y,X=x)\Pr(Y=y,X=x)}{\Pr(Y=y)}\\
&=\frac{\sum_x \Pr(B=b|X=x)\Pr(Y=y,X=x)}{\Pr(Y=y)}.
\end{aligned}$$

Considering that $c\rightarrow x$ is a one-to-one error-correction encoding mapping, and under the assumption of the proposition, we have:

$$\Pr(B=b|X=x_b)=\frac{1}{2}+\varepsilon,$$

where x_b corresponds to the selected m_b, and

$$\Pr(B=b|X=x)=\frac{1}{2}\ \ \textit{for any}\ x\neq x_b.$$

Consequently,

$$\begin{aligned}
&\Pr(B=b|Y=y)\\
&=\frac{\Pr(B=b|X=x_b)\Pr(Y=y,X=x_b)}{\Pr(Y=y)}\\
&\quad+\frac{\sum_{x:x\neq x_b}\Pr(B=b|X=x)\Pr(Y=y,X=x)}{\Pr(Y=y)},
\end{aligned}\tag{2}$$

and

$$\begin{aligned}
&\Pr(B=b|Y=y)\\
&=\frac{(\frac{1}{2}+\varepsilon)Pr(Y=y,X=x_b)-\frac{1}{2}Pr(Y=y,X=x_b)}{Pr(Y=y)}\\
&\quad+\frac{\frac{1}{2}\sum_x Pr(Y=y,X=x)}{Pr(Y=y)}\\
&=\frac{1}{2}+\varepsilon\cdot\frac{Pr(X=x_b,Y=y)}{Pr(Y=y)}\\
&=\frac{1}{2}+\varepsilon\cdot\frac{Pr(X=x_b|Y=y)Pr(Y=y)}{Pr(Y=y)}\\
&=\frac{1}{2}+\varepsilon\cdot Pr(X=x_b|Y=y).
\end{aligned}$$

Q.E.D.

Proof of Proposition 2: The equivocation, or the conditional entropy $H(U|V)$, represents the average amount of information lost about U when V is given. According to [57] and [58], we have the following upper bound on the equivocation:

$$H(U|V)\le h(P_{err})+P_{err}log_2(2^q-1) \tag{3}$$

where $h(\cdot)\le 1$ is the binary entropy function, P_{err} is the bit error probability, and the conditional entropy is defined as

$$H(U|V)=\sum_{y\in supp(V)} Pr(V=v)H(U|V=v)$$

where

$$H(U|V=v)=\sum_{x\in supp(X)} Pr(U=u|V=v)\log_2\frac{1}{Pr(U=u|V=v)},$$

and $Pr(\cdot)$ denotes the probability of the considered event.

Recall that

$$H(U|V)=H(U)-I(U,V)$$

where

$$H(U)=\sum_{x\in supp(U)} Pr(U=u)\log_2\frac{1}{Pr(U=u)},$$

and the mutual information $I(U,V)$ is upper-bounded by the capacity Cap of the channel as follows:

$$I(U,V)\le qCap.$$

In the evaluation scenario considered in (3), the inequality above can be rewritten as:

$$q(1-Cap)\le 1+P_{err}log_2(2^q-1)$$

yielding

$$P_{err}\ge\frac{q(1-Cap)-1}{log_2(2^q-1)}$$

where Cap is the capacity of the employed channel. Accordingly, from

$$\Pr(U=u|V=v)=(1-P_{err})^q$$

we obtain

$$\Pr(U=u|V=v)<\left(\frac{q(Cap^*-1)+1+log_2(2^q-1)}{log_2(2^q-1)}\right)^q.$$

where Cap^* is the channel capacity from the attacker's perspective. Q.E.D.

Proof of Theorem 2: Propositions 1 and 2 imply that

$$\Pr[\mathcal{A}\to 1|Y=y]\le\mathcal{F}(Cap^*,q,\lambda_p,d_p,p=1,2)$$

where $\mathcal{F}(\cdot)$ is certain function and Cap^* is the channel capacity from the attacker's perspective, modeled as a structure of parallel deletion channels. According to [55] and [56], this structure is equivalent to a single deletion channel with a deletion probability d_e given by:

$$d_e=\sum_{p=1}^{2}\lambda_p d_p.$$

Considering that the capacity $Cap(d_p)\le 1-d_p$ for each $p=1,2$, is less than the corresponding erasure channel capacity, we have

$$Cap(d_p)\le 1-d_p,\quad p=1,2.$$

On the other hand, according to the reported in [59] we have

$$Cap^*<1-\left(log_2\frac{8}{1+\sqrt{5}}\right)\left(\sum_{p=1}^{2}\lambda_p d_p\right)$$

Combining the above results with the statements of Propositions 1 and 2, we arrive at the theorem's conclusion. Q.E.D.

REFERENCES

[1] D. Leong and T. Ho, "Erasure coding for real-time streaming," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2012, pp. 289–293.

[2] Z. Shen, Y. Cai, K. Cheng, P. P. C. Lee, X. Li, Y. Hu, and J. Shu, "A survey of the past, present, and future of erasure coding for storage systems," *ACM Trans. Storage*, vol. 21, no. 1, pp. 1–39, Jan. 2025.

[3] L. Liu, J. Huang, W. Zhou, and S. Zhou, "Computing the minimum distance of nonbinary LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 7, pp. 1753–1758, Jul. 2012.

[4] B. K. Butler and P. H. Siegel, "Bounds on the minimum distance of punctured quasi-cyclic LDPC codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4584–4597, Jul. 2013.

[5] J. Guo, A. G. I. Fàbregas, and J. Sayir, "Fixed-threshold polar codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2013, pp. 947–951.

[6] H. Dau, I. M. Duursma, H. M. Kiah, and O. Milenkovic, "Repairing Reed–Solomon codes with multiple erasures," *IEEE Trans. Inf. Theory*, vol. 64, no. 10, pp. 6567–6582, Oct. 2018.

[7] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 638–656, Feb. 2001.

[8] J. Lu and J. M. F. Moura, "Linear time encoding of LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 233–249, Jan. 2010.

[9] A. Frolov and V. Zyablov, "On the multiple threshold decoding of LDPC codes over GF(q)," *Adv. Math. Commun.*, vol. 11, no. 1, pp. 123–137, Jan. 2017.

[10] P. Rybin, K. Andreev, and V. Zyablov, "Error exponents of LDPC codes under low-complexity decoding," *Entropy*, vol. 23, no. 2, p. 253, Feb. 2021.

[11] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[12] B. Li, H. Shen, and D. Tse, "An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check," *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, Dec. 2012.
[13] N. Tang and Y. Lin, "Fast encoding and decoding algorithms for arbitrary (n, k) Reed–Solomon codes over $\mathbb{F}_{2^m}$," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 716–719, Apr. 2020.
[14] N. Tang, C. Chen, and Y. S. Han, "Fast error and erasure decoding algorithm for Reed–Solomon codes," *IEEE Commun. Lett.*, vol. 28, no. 4, pp. 759–762, Apr. 2024.
[15] A. Radonjic and V. Vujicic, "Integer codes correcting high-density byte asymmetric errors," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 694–697, Apr. 2017.
[16] A. Radonjic and V. Vujicic, "Integer codes correcting sparse byte errors," *Cryptography Commun.*, vol. 11, no. 5, pp. 1069–1077, Sep. 2019.
[17] A. Radonjic and V. Vujicic, "Integer codes correcting burst asymmetric errors within a byte and double asymmetric errors," *Cryptogr. Commun.*, vol. 12, no. 2, pp. 221–230, Mar. 2020.
[18] A. Radonjic and V. Vujicic, "Logarithmic time encoding and decoding of integer error control codes," *Eng. Rep.*, vol. 5, no. 11, p. e12675, Nov. 2023.
[19] R. L. Rivest and A. T. Sherman, "Randomized encryption techniques," in *Proc. Adv. Cryptol.*, Jan. 1983, pp. 145–163.
[20] M. Willett, "Deliberate noise in a modern cryptographic system (Corresp.)," *IEEE Trans. Inf. Theory*, vol. IT-26, no. 1, pp. 102–104, Jan. 1980.
[21] C. An, Y. Liu, and X. Lu, "Evolution of the polar code-based encryption schemes," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Madrid, Spain, Dec. 2021, pp. 1–6, doi: 10.1109/GCWkshps52748.2021.9681980.
[22] M. Esmaeili, M. Dakhilalian, and T. A. Gulliver, "New secure channel coding scheme based on randomly punctured quasi-cyclic-low density parity check codes," *IET Commun.*, vol. 8, no. 14, pp. 2556–2562, Sep. 2014.
[23] M. Esmaeili and T. A. Gulliver, "Joint channel coding-cryptography based on random insertions and deletions in quasi-cyclic-low-density parity check codes," *IET Commun.*, vol. 9, no. 12, pp. 1555–1560, Aug. 2015.
[24] M. Esmaeili and T. A. Gulliver, "A secure code based cryptosystem via random insertions, deletions, and errors," *IEEE Commun. Lett.*, vol. 20, no. 5, pp. 870–873, May 2016.
[25] R. Hooshmand, M. K. Shooshtari, and M. R. Aref, "Secret key cryptosystem based on polar codes over binary erasure channel," in *Proc. 10th Int. ISC Conf. Inf. Secur. Cryptol. (ISCISC)*, Aug. 2013, pp. 1–6, doi: 10.1109/ISCISC.2013.6767351.
[26] R. Hooshmand, M. R. Aref, and T. Eghlidos, "Physical layer encryption scheme using finite-length polar codes," *IET Commun.*, vol. 9, no. 15, pp. 1857–1866, Oct. 2015.
[27] R. Hooshmand and M. R. Aref, "Efficient polar code-based physical layer encryption scheme," *IEEE Wireless Commun. Lett.*, vol. 6, no. 6, pp. 710–713, Dec. 2017.
[28] K. Bagheri, T. Eghlidos, M.-R. Sadeghi, D. Panario, and H. Khodaiemehr, "A joint encryption, channel coding and modulation scheme using QC-LDPC lattice-codes," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 4673–4693, Aug. 2020.
[29] X. Lu, J. Lei, W. Li, K. Lai, and Z. Pan, "Physical layer encryption algorithm based on polar codes and chaotic sequences," *IEEE Access*, vol. 7, pp. 4380–4390, 2019.
[30] A. Rajagopalan, A. Thangaraj, and S. Agrawal, "Wiretap polar codes in encryption schemes based on learning with errors problem," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Vail, CO, USA, Jun. 2018, pp. 1146–1150, doi: 10.1109/ISIT.2018.8437896.
[31] T. R. N. Rao and K.-H. Nam, "Private-key algebraic-code encryptions," *IEEE Trans. Inf. Theory*, vol. 35, no. 4, pp. 829–833, Jul. 1989.
[32] C. M. Stuart, S. K., D. K. J., and D. P. Pattathil, "Design and implementation of hardware-efficient modified Rao–Nam scheme with high security for wireless sensor networks," *J. Inf. Secur. Appl.*, vol. 29, pp. 65–79, Aug. 2016.
[33] B. Applebaum, D. M. Cash, C. Peikert, and A. Sahai, "Fast cryptographic primitives and circular-secure encryption based on hard learning problems," in *Proc. Annu. Int. Cryptol. Conf.*, Aug. 2009, pp. 595–618.
[34] M. J. Mihaljević and H. Imai, "An approach for stream ciphers design based on joint computing over random and secret data," *Computing*, vol. 85, nos. 1–2, pp. 153–168, Jun. 2009.
[35] M. J. Mihaljević, A. Kavcic, and K. Matsuura, "An encryption technique for provably secure transmission from a high performance computing entity to a tiny one," *Math. Problems Eng.*, vol. 2016, pp. 1–10, Jan. 2016, doi: 10.1155/2016/7920495.
[36] M. J. Mihaljević and F. Oggier, "Security evaluation and design elements for a class of randomised encryptions," *IET Inf. Secur.*, vol. 13, no. 1, pp. 36–47, Jan. 2019.
[37] M. J. Mihaljević, "A security enhanced encryption scheme and evaluation of its cryptographic security," *Entropy*, vol. 21, no. 7, p. 701, Jul. 2019, doi: 10.3390/e21070701.
[38] M. J. Mihaljević, L. Wang, and S. Xu, "An approach for security enhancement of certain encryption schemes employing error correction coding and simulated synchronization errors," *Entropy*, vol. 24, no. 3, p. 406, Mar. 2022, doi: 10.3390/e24030406.
[39] M. J. Mihaljević, A. Radonjić, L. Wang, and S. Xu, "Security enhanced symmetric key encryption employing an integer code for the erasure channel," *Symmetry*, vol. 14, no. 8, p. 1709, Aug. 2022.
[40] M. Devipriya and M. Brindha, "Image encryption using modified perfect shuffle-based bit level permutation and learning with errors based diffusion for IoT devices," *Comput. Electr. Eng.*, vol. 100, May 2022, Art. no. 107954.
[41] H. Gilbert, M. J. B. Robshaw, and Y. Seurin, "How to encrypt with the LPN problem," in *Proc. ICALP*, Aug. 2008, pp. 679–690.
[42] Y. S. Khiabani, S. Wei, J. Yuan, and J. Wang, "Enhancement of secrecy of block ciphered systems by deliberate noise," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 5, pp. 1604–1613, Oct. 2012.
[43] F. Oggier and M. J. Mihaljevic, "An information-theoretic security evaluation of a class of randomized encryption schemes," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 2, pp. 158–168, Feb. 2014.
[44] S. Wei, J. Wang, R. Yin, and J. Yuan, "Trade-off between security and performance in block ciphered systems with erroneous ciphertexts," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 4, pp. 636–645, Apr. 2013.
[45] Y. Lee, Y.-S. Kim, and J.-S. No, "Ciphertext-only attack on linear feedback shift register-based esmaeili-gulliver cryptosystem," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 971–974, May 2017.
[46] J. Wang, J. Mu, S. Wei, C. Jiang, and N. C. Beaulieu, "Statistical characterization of decryption errors in block-ciphered systems," *IEEE Trans. Commun.*, vol. 63, no. 11, pp. 4363–4376, Nov. 2015.
[47] H. Kostadinov and N. Manev, "A general construction of integer codes correcting specific errors in binary communication channels," *Mathematics*, vol. 11, no. 11, p. 2521, May 2023.
[48] A. Radonjic, "(Perfect) integer codes correcting single errors," *IEEE Commun. Lett.*, vol. 22, no. 1, pp. 17–20, Jan. 2018.
[49] A. Radonjic and V. Vujicic, "Integer codes correcting burst and random asymmetric errors within a byte," *J. Franklin Inst.*, vol. 355, no. 2, pp. 981–996, Jan. 2018.
[50] A. Radonjic, "Integer codes correcting double errors and triple-adjacent errors within a byte," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 8, pp. 1901–1908, Aug. 2020.
[51] J. Katz and Y. Lindell, *Ntroduction To Modern Cryptography*. Boca Raton, FL, USA: CRC Press, 2007.
[52] Z. Wu, C. Gong, and D. Liu, "Computational complexity analysis of FEC decoding on SDR platforms," *J. Signal Process. Syst.*, vol. 89, no. 2, pp. 209–224, Nov. 2017.
[53] Y. Chen, X. Qiao, K. Deng, S. Song, and Z. Wang, "3.8-gbps polar belief propagation decoder on GPU," *IEEE Commun. Lett.*, vol. 27, no. 5, pp. 1247–1251, May 2023.
[54] J. Dai, H. Yin, Y. Lv, W. Xu, and Z. Yang, "Multi-gbps LDPC decoder on GPU devices," *Electronics*, vol. 11, no. 21, p. 3447, Oct. 2022.
[55] M. Rahmati and T. M. Duman, "Upper bounds on the capacity of deletion channels using channel fragmentation," *IEEE Trans. Inf. Theory*, vol. 61, no. 1, pp. 146–156, Jan. 2015.
[56] M. Cheraghchi and J. Ribeiro, "An overview of capacity results for synchronization channels," *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 3207–3232, Jun. 2021.
[57] D. L. Tebbe and S. J. Dwyer, "Uncertainty and the probability of error," *IEEE Trans. Inf. Theory*, vol. IT-24, pp. 516–518, 1968.
[58] M. Feder and N. Merhav, "Relations between entropy and error probability," *IEEE Trans. Inf. Theory*, vol. 40, no. 1, pp. 259–266, Jan. 1994.
[59] M. Cheraghchi, "Capacity upper bounds for deletion-type channels," *J. ACM*, vol. 66, no. 2, pp. 1–79, Apr. 2019.

MIODRAG J. MIHALJEVIĆ has been holding long-term visiting positions with various universities and research institutes in Japan, including The University of Tokyo, Sony Research Laboratories, and the National Institute AIST, Tokyo, since 1997. He is currently a Distinguished Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan), China, as well as a Research Professor and the Deputy Director with the Mathematical Institute, Serbian Academy of Sciences and Arts, Belgrade. His main research interests include cryptology, information security, and blockchain technology. Since 2014, he has been an Elected Member of the Academia Europaea. He has been an Elected Member of Serbian Academy of Sciences and Arts, since 2021. In 2013, he received the National Award of Serbian Academy of Sciences and Arts, for ten years achievements. In the five consecutive years 2020–2024, he was included in the ranked list colloquially known as ''World's Top 2% Scientists'' (by Elsevier and Stanford University), regarding his career achievements. For more information, please visit http://www.mi.sanu.ac.rs/cv/cvmihaljevic.htm.

ALEKSANDAR RADONJIĆ received the Dipl.Ing.El. degree in telecommunications, the Dipl.Ing.El. degree in instrumentation and measurement, and the Ph.D. degree in electrical and computer engineering from the University of Novi Sad, Serbia. He is currently a Principal Research Fellow with the Institute of Technical Sciences, Serbian Academy of Sciences and Arts. In addition, he is a Full Professor with the University Union Nikola Tesla, where he teaches several courses in the field of computing and informatics. His research interests include error control coding, fault-tolerant computing, VLSI design, and computer networking.

NEVENA MIJAJLOVIĆ was born in 1985. She received the Bachelor of Science, Master of Science, and Ph.D. degrees in mathematics from the University of Montenegro, Podgorica, Montenegro, in 2007, 2009, and 2015, respectively.

Since 2008, she has been with the University of Montenegro, initially as a Teaching Assistant, later as an Assistant Professor, and currently as an Associate Professor with the Faculty of Natural Sciences and Mathematics, Podgorica. Her research interests include optimization methods, variational inequalities, equilibrium problems, and blockchain technology.

Prof. Mijajlović is a member of the Committee for Mathematics and Physics of Montenegrin Academy of Sciences and Arts (MASA) and the Center for Young Scientists and Artists of MASA. She is also a member of the University of Montenegro's Board of Directors, representing an Academic Staff.

LIANHAI WANG received the M.E. degree in computer software and theory and the Ph.D. degree in computer science and technology from Shandong University, Jinan, China, in 2003 and 2014, respectively. He is currently a Research Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan) and the School of Cyber Security, Qilu University of Technology (Shandong Academy of Science), Jinan. His current research interests include the areas of information security, digital forensics, and blockchain.

SHUJIANG XU received the Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently a Research Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan) and the School of Cyber Security, Qilu University of Technology (Shandong Academy of Science), Jinan, China. His research interests include blockchain, data security, and multimedia information security.