

On the k -order Jacobsthal polynomials and a new approach to coding theory

Fatih Yılmaz *

Samet Arpacı †

Aybüke Ertaş §

Department of Mathematics

Ankara Hacı Bayram Veli University

Ankara 06900

Turkey

Mustapha Rachidi ‡

Intitut of Mathematics

Universidade Federal de Mato Grosso do Sul Campo Grande

Mato Grosso do Sul

MS- Brazil

Abstract

In this paper, we introduce a novel coding and decoding framework based on a generalized form of the k -order Jacobsthal polynomials. To this end, we first investigate the structure of these polynomials and derive a general expression for the powers of their associated companion matrices. Specifically, we develop a coding/decoding scheme utilizing integer powers of the companion matrix $Q_k(x)$ whose entries are determined by the coefficients of the k -order Jacobsthal polynomials. Furthermore, we define a suitable alphabet structured by generalized Jacobsthal numbers, integrating this alphabet into the proposed coding methodology. This construction enables a new perspective on algebraic coding theory by embedding the properties of k -order Jacobsthal polynomials into the design of the coding system. To demonstrate the practicality of the method, several illustrative

* ORCID-ID: 0000-0001-7873-1979

† ORCID-ID: 0000-0002-6646-8197

§ ORCID-ID: 0000-0002-6133-5520

‡ ORCID-ID: 0000-0002-8210-7383

* E-mail: fatih.yilmaz@hbu.edu.tr (Corresponding Author)

† E-mail: samet.arpaci@hbu.edu.tr

§ E-mail: aybkerts99@gmail.com

‡ E-mail: mustapha.rachidi@ufms.br

examples and applications are presented. Finally, potential directions for future research and implications for coding theory are discussed.

Subject Classification: 11B83, 11B37, 11T71, 94B35.

Keywords: *Jacobsthal polynomials, k-order Jacobsthal polynomials, Coding/decoding method.*

1. Introduction

Over the past decades, coding theory has garnered significant attention and has become a central field in information and communication technologies. At its core, coding theory addresses the reliable and secure transmission of information over potentially noisy channels. The foundations of the field trace back to the seminal work of Claude Shannon whose theorems laid the groundwork for modern information theory. Since then, the field has evolved considerably, drawing heavily on the algebraic structures of polynomials over finite fields and the theory of finite fields itself (for further details, see [1, 2, 3, 4, 5, 6, 7, 8, 9] and the references therein).

One of the principal goals of coding theory is to develop efficient and cost-effective methods for error-free message transmission. To achieve this, researchers have proposed a variety of algebraic codes such as block codes, affine ciphers, and shift ciphers that provide systematic frameworks for encoding and decoding. These methods are not only mathematically elegant but also practical, offering algorithmic simplicity and ease of implementation.

The general structure of these methods is typically illustrated by a simplified communication model (see Figure 1). In this model, messages originate from a sender and are transmitted through a channel. It is important to note that this study focuses on systems with a finite set of discrete signals (e.g., telegraph systems), in contrast to those involving continuous signal sources such as radio transmissions.

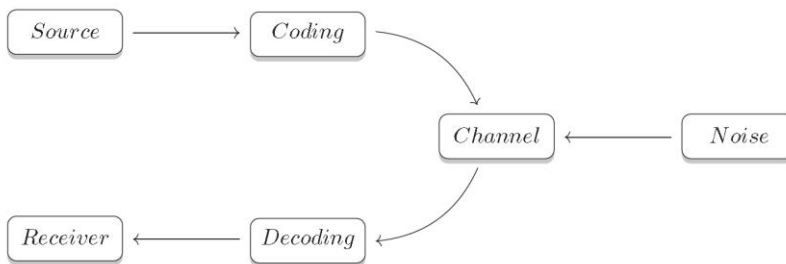


Figure 1
A simple scheme of a communication transmission system

Various techniques have been proposed in the literature to analyze the aforementioned coding processes. In this study, we focus on a new approach that employs Jacobsthal polynomials to construct an alternative coding and decoding

procedure. It is worth recalling that Horadam, in [10], introduced the Jacobsthal numbers via the recurrence relation

$$J_n = J_{n-1} + 2J_{n-2}, \text{ for every } n \geq 2$$

with initial terms $J_0 = 0$ and $J_1 = 1$. Over the years, the Jacobsthal number sequence and its generalizations have attracted significant attention, leading to a rich body of literature exploring their algebraic and combinatorial properties. Building upon these developments, our goal is to extend their application to the field of coding theory. Moreover, in [10, 11], Horadam provides the concept of Jacobsthal polynomials by considering the following recurrence relation

$$J_n(x) = J_{n-1}(x) + 2xJ_{n-2}(x), \text{ for every } n \geq 2, \quad (1)$$

where the initial conditions are $J_0(x) = 0$ and $J_1(x) = 1$. In the Table 1, we have the first few terms of the Jacobsthal polynomials.

Table 1
Some values of Jacobsthal polynomials

n	0	1	2	3	4	5	6
$J_n(x)$	0	1	1	$1 + 2x$	$1 + 4x$	$1 + 6x + 4x^2$	$1 + 8x + 12x^2$

In addition, Horadam investigated several notable properties of the Jacobsthal polynomials, including summation identities, Binet-type analytical expressions, Simpson formulas, and generating functions. The remarkable interesting results are obtained in [12] where Horadam derives convolution identities related to Jacobsthal-type polynomials. For further details and related results, the reader is referred to [13, 14, 15, 16, 17, 18] and the references therein. The analytic Binet's formula for the Jacobsthal polynomial is given as below

$$J_n(x) = \frac{\lambda^n(x) - \lambda^{-n}(x)}{\lambda(x) - \lambda^{-1}(x)}, \quad (2)$$

where $\lambda(x) = \frac{1}{2}(1 + \sqrt{1 + 8x})$ and $\lambda^{-1}(x) = \frac{1}{2}(1 - \sqrt{1 + 8x})$ denote the roots of the characteristic equation $t^2 - t - 2x = 0$ (see [19, 20]). Additionally, the Cassini identity of Jacobsthal polynomials is defined as follows

$$J_{n+1}(x)J_{n-1}(x) - J_n^2(x) = (-1)^n(2x)^{n-1}, \quad \text{for every } n \geq 2.$$

(see, for instance, [19]).

The primary objective of this study is to investigate the unexplored relationship between k -order Jacobsthal polynomials and coding theory. Although Jacobsthal polynomials have been extensively studied and applied in various mathematical disciplines, their potential applications within the framework of coding theory have, to the best of our knowledge, not been addressed in the existing literature. This work seeks to fill this gap by analyzing how the structural properties of k -order Jacobsthal polynomials can be leveraged to construct efficient encoding and decoding algorithms. Special emphasis is placed on the case $k = 2$, where we propose novel methods for both error detection and correction. Given the fundamental importance of coding theory in

modern communication systems, and the rich combinatorial structure of Jacobsthal sequences, this study aims to establish a meaningful connection between the two and open avenues for further research in this direction.

The organization of this paper is as follows. Section 2 provides a detailed examination of the novel Jacobsthal polynomial matrix $Q_k(x)$ outlining its fundamental properties and theoretical underpinnings. In Section 3, we develop the coding and decoding algorithms that utilize k -order Jacobsthal polynomials, including a comprehensive study of the determinant of the code matrix and the intrinsic relationships between its components. Section 4 is dedicated to the discussion of error detection and correction mechanisms within this framework. This section further explores a comparative analysis between the proposed Jacobsthal coding method and classical coding approaches, highlighting the advantages and limitations of each. Additionally, the critical role of polynomials in ensuring cryptographic security is examined in depth. Finally, Section 5 presents concluding remarks, summarizing the main contributions of the paper and suggesting potential directions for future research in this area.

2. Main Results on the k -order Jacobsthal polynomials

A wide range of studies in the literature has examined the mathematical structure and properties of Jacobsthal polynomials. In this section, we introduce the theoretical background of the k -order Jacobsthal polynomial sequence, which forms the foundation for the coding framework developed in this work. We also define the notation and terminology used throughout the paper to ensure clarity and consistency.

2.1 k -order Jacobsthal polynomials and their related matrices

For every integer $k \geq 2$, the family of k -th order Jacobsthal polynomials $\{J_n^{(k)}(x)\}_{n \geq 0}$ is defined by the linear recurrence relation

$$J_n^{(k)}(x) = J_{n-1}^{(k)}(x) + 2xJ_{n-2}^{(k)}(x) + \sum_{j=3}^k x^{j-1}J_{n-j}^{(k)}(x), n \geq 1, \quad (3)$$

subject to the initial conditions

$$J_i^{(k)}(x) = \begin{cases} 1, & \text{if } k + i = 1, \\ 0, & \text{otherwise,} \end{cases} \quad (1 - k \leq i \leq 0). \quad (4)$$

This definition was first introduced in [21]. Explicit representations of these polynomials for various orders k are provided in [21], where their structural properties are also illustrated. Furthermore, when evaluating the sequence at $x = 1$, the resulting integer values coincide with those obtained independently by Yilmaz and Bozkurt [22], thereby confirming the consistency of the formulation.

Indeed, in [21], Arpacı and Yilmaz developed a method for computing powers of the matrix $Q_k^n(x)$ using k -order Jacobsthal polynomials. In their approach, they derived explicit formulas for the matrix entries based on the Fibonacci–Jacobsthal fundamental system, contributing to the structural understanding of these polynomials.

The following property can be formulated for 2 -order Jacobsthal polynomials

Theorem 1: *The ratio two consecutive Jacobsthal polynomials is*

$$\lim_{n \rightarrow \infty} \frac{J_{n+1}^{(2)}(x)}{J_n^{(2)}(x)} = \frac{1}{2}(1 + \sqrt{1 + 8x}).$$

Proof: Let us suppose that $\lim_{n \rightarrow \infty} \frac{J_{n+1}^{(2)}(x)}{J_n^{(2)}(x)} = \lambda(x)$. Then, using Expression (2), we get

$$\lambda(x) = \lim_{n \rightarrow \infty} \left[\frac{J_n^{(2)}(x) + 2xJ_{n-1}^{(2)}(x)}{J_n^{(2)}(x)} \right] = 1 + \frac{2x}{\lambda(x)}.$$

Therefore, since $\lambda^2(x) - \lambda(x) - 2x = 0$ we show that the roots of this equation are $\lambda(x) = \frac{1}{2}(1 + \sqrt{1 + 8x})$ and $\lambda^{-1}(x) = \frac{1}{2}(1 - \sqrt{1 + 8x})$.

In the aim to prepare the ground of the coding method, let us consider the following important matrices $Q_k(x)$, $E_{k,n}(x)$ and $R_k(x)$. First, the matrix $Q_k(x)$ is the $k \times k$ companion matrix defined as follows

$$Q_k(x) = \begin{pmatrix} 1 & 2x & x^2 & \dots & x^{k-2} & x^{k-1} \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 1 & 0 \end{pmatrix}_{k \times k} \quad (5)$$

Second, for every $n \geq k$, we consider the $k \times k$ matrix $E_{k,n}(x)$ defined by

$$E_{k,n}(x) = \begin{pmatrix} J_{n-1}^{(k)}(x) & J_{n-2}^{(k)}(x) & J_{n-3}^{(k)}(x) & \dots & J_{n-k-1}^{(k)}(x) & J_{n-k}^{(k)}(x) \\ J_{n-2}^{(k)}(x) & J_{n-3}^{(k)}(x) & J_{n-4}^{(k)}(x) & \dots & J_{n-k}^{(k)}(x) & J_{n-k-1}^{(k)}(x) \\ J_{n-3}^{(k)}(x) & J_{n-4}^{(k)}(x) & J_{n-5}^{(k)}(x) & \dots & J_{n-k-1}^{(k)}(x) & J_{n-k-2}^{(k)}(x) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ J_{n-k+1}^{(k)}(x) & J_{n-k}^{(k)}(x) & J_{n-k-1}^{(k)}(x) & \dots & J_{n-2k+2}^{(k)}(x) & J_{n-2k+1}^{(k)}(x) \\ J_{n-k}^{(k)}(x) & J_{n-k-1}^{(k)}(x) & J_{n-k-2}^{(k)}(x) & \dots & J_{n-2k+1}^{(k)}(x) & J_{n-2k}^{(k)}(x) \end{pmatrix}_{k \times k} \quad (6)$$

where we set $J_{-n}^{(k)}(x) = 0$, for every $1 \leq n \leq k$. Finally, let $R_k(x)$ be the matrix given by

$$R_k(x) = E_{k,k}(x) = \begin{pmatrix} J_{k-1}^{(k)}(x) & J_{k-2}^{(k)}(x) & J_{k-3}^{(k)}(x) & \dots & J_1^{(k)}(x) & J_0^{(k)}(x) \\ J_{k-2}^{(k)}(x) & J_{k-3}^{(k)}(x) & J_{k-4}^{(k)}(x) & & J_0^{(k)}(x) & 0 \\ J_{k-3}^{(k)}(x) & J_{k-4}^{(k)}(x) & J_{k-5}^{(k)}(x) & & 0 & 0 \\ \vdots & & & & & \vdots \\ J_1^{(k)}(x) & J_0^{(k)}(x) & 0 & & 0 & 0 \\ J_0^{(k)}(x) & 0 & 0 & \dots & 0 & 1 \end{pmatrix}_{k \times k}.$$

A straightforward computations allows us to get

$$E_{k,n+1}(x) = Q_k(x)E_{k,n}(x), \text{ for every } n \geq k.$$

Moreover, an iterative using the preceding matrix equation allows us to establish the following result.

Theorem 2: For every $n \geq 0$,

$$E_{k,n+k}(x) = Q_k^n(x)R_k(x). \quad (7)$$

Proof: The proof is obtained by using the principle of mathematical induction based on an iterative process.

Theorem 2 shows the importance of calculating the matrix powers Q_k^n of the matrix Q_k . For illustrative purpose of Theorem 2, we consider the special case $k = 2$. In this case, we have the following corollary.

Corollary 1: For every $n \geq 1$,

$$Q_2^n(x) = \begin{bmatrix} J_{n+1}(x) & 2xJ_n(x) \\ J_n(x) & 2xJ_{n-1}(x) \end{bmatrix}. \quad (8)$$

Similarly, the powers $E_{2,2}^n(x)$ of the matrix $E_{2,2}(x)$ are given by

$$E_{2,2}^n(x) = \begin{bmatrix} J_{n-1}(x) & J_{n-2}(x) \\ J_{n-2}(x) & J_{n-3}(x) \end{bmatrix}. \quad (9)$$

Proof: Let us prove the first assertion by using induction. For $n = 1$ the equation

$$(8) \text{ is satisfied. That is, we have } Q_2(x) = \begin{bmatrix} J_2(x) & 2xJ_1(x) \\ J_1(x) & 2xJ_0(x) \end{bmatrix} = \begin{bmatrix} 1 & 2x \\ 1 & 0 \end{bmatrix}.$$

Suppose that Expression (8) holds for $n = p$, namely, $Q_2^p(x) = \begin{bmatrix} J_{p+1}(x) & 2xJ_p(x) \\ J_p(x) & 2xJ_{p-1}(x) \end{bmatrix}$. For $n = p + 1$ we have

$$Q_2^{p+1}(x) = Q_2^p(x)Q_2(x) = \begin{bmatrix} J_{p+1}(x) & 2xJ_p(x) \\ J_p(x) & 2xJ_{p-1}(x) \end{bmatrix} \begin{bmatrix} 1 & 2x \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} J_{p+2}(x) & 2xJ_{p+1}(x) \\ J_{p+1}(x) & 2xJ_p(x) \end{bmatrix}.$$

Therefore, the proof of the first assertion is completed.

For the second assertion, we can proceed in the same way. That is, for $n = 1$ the equation is clearly satisfied. That is, $E_{2,2}(x) = \begin{bmatrix} J_0(x) & J_{-1}(x) \\ J_{-1}(x) & J_{-2}(x) \end{bmatrix} =$

$\begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix}$. Suppose that Expression (9) holds for $n = p$, namely, $E_{2,2}^p(x) = \begin{bmatrix} J_{p-1}(x) & J_{p-2}(x) \\ J_{p-2}(x) & J_{p-3}(x) \end{bmatrix}$. For $n = p + 1$ we have

$$E_{2,2}^{p+1}(x) = E_{2,2}^p(x)E_{2,2}(x) = \begin{bmatrix} J_{p-1}(x) & J_{p-2}(x) \\ J_{p-2}(x) & J_{p-3}(x) \end{bmatrix} \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2xJ_{p-1}(x) & J_{p-2}(x) \\ 2xJ_{p-2}(x) & J_{p-3}(x) \end{bmatrix}.$$

Therefore, the proof of assertion (9) is completed.

From the general Expression (7) and the special cases given by Expressions (8)-(9), we can derive some identities and matrix equations. For example, for every x , we have the two following identities

$$\det E_{2,2}^2(x) = 2x \text{ and } \det E_{2,2}^n(x) = (-1)^{n-2} x^{n-3}.$$

Indeed, the first identity is derived from a simple direct computation. For the second identity, it is clear that $\det E_{2,2}^n(x) = J_{n-1}(x)J_{n-3}(x) - J_{n-2}^2(x)$. Hence, we get the result by exploiting the Cassini property for the Jacobsthal polynomials (see [19]). Additionally, a straightforward computation shows that $Q_2^n(x).R_2(x) = \begin{bmatrix} 1 & 2x \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} J_{n-1}(x) & J_{n-2}(x) \\ J_{n-2}(x) & J_{n-3}(x) \end{bmatrix}$, which allows us to obtain the following matrices identity

$$Q_2^n(x).R_2(x) = E_{2,n}(x).$$

Additionally, for $x = 1$ we obtain the following matrix representation

$$Q_2^n(1).R_2(1) = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} J_{n-1} & J_{n-2} \\ J_{n-2} & J_{n-3} \end{bmatrix} = E_{2,n}.$$

Finally, for $k = 3$ a direct computation shows that

$$Q_3^n(x).R_3(x) = \begin{bmatrix} 1 & 2x & x^2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^n \begin{bmatrix} x^2 & x^2 & 0 \\ x^2 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} J_{n-1}^{(3)}(x) & J_{n-2}^{(3)}(x) & J_{n-3}^{(3)}(x) \\ J_{n-2}^{(3)}(x) & J_{n-3}^{(3)}(x) & J_{n-4}^{(3)}(x) \\ J_{n-4}^{(3)}(x) & J_{n-5}^{(3)}(x) & J_{n-6}^{(3)}(x) \end{bmatrix}.$$

Therefore, the powers $Q_3^n(x)$ of the matrix $Q_3(x)$ are given under the form.

$$Q_3^n(x).R_3(x) = E_{3,n}(x).$$

3. The Coding/Decoding Method for the k -Order Jacobsthal Polynomials

In this section, we define a new coding/decoding method via the k -order Jacobsthal polynomials. The main steps of our method are described here below.

- **The Coding Algorithm** The coding steps can be summarized under the following form.
 - **Step 1:** Consider the text is a structure which occurs from m character and choose the arbitrary values r and i .

- **Step 2:** Substitute a general Jacobsthal number for each character from the Latin alphabet given in Table 2 and obtain the message matrix M of order k .
- **Step 3:** Constitute the matrix $E_{k,n}(x)$, given by (6). We call the matrix $E_{k,n}(x)$ as "encoding matrix".
- **Step 4:** Multiply the message matrix M with $E_{k,n}(x)$ and then, obtain the encrypted matrix C .
- **Step 5:** Send the encrypted matrix C to the receiver.
- **The Decoding Algorithm** The decoding steps can be summarized as follows.
 - **Step 1:** Consider the encrypted matrix C and constitute the inverse matrix $(E_{k,n}(x))^{-1}$. Here n is received from the source.
 - **Step 2:** Compute $C \times (E_{k,n}(x))^{-1}$ and constitute the message matrix M whose entries are general Jacobsthal numbers.
 - **Step 3:** Taking into account the arbitrary values r and i which are received from the source, substitute the character from the Latin alphabet given in Table 2 for each general Jacobsthal number and obtain the message matrix M of order k .
 - **Step 4:** Convert the decoded matrix to the text message by using the Latin alphabet.

This alphabet is written according to $\text{mod } 40$ for the arbitrary value of J_r^i . In addition to all these, we emphasize the importance of the method by pointing out that with this method, we can encrypt the message matrix by dividing it into arbitrary square matrix blocks, or we can encrypt it directly without dividing it into blocks.

Table 2
The representation of characters with general Jacobsthal numbers.

A	B	C	D	E	F	G	H
$1 + J_r^i$	$2 + J_{r+1}^i$	$3 + J_{r+2}^i$	$4 + J_{r+3}^i$	$5 + J_{r+4}^i$	$6 + J_{r+5}^i$	$7 + J_{r+6}^i$	$8 + J_{r+7}^i$
I	J	K	L	M	N	O	P
$9 + J_{r+8}^i$	$10 + J_{r+9}^i$	$11 + J_{r+10}^i$	$12 + J_{r+11}^i$	$13 + J_{r+12}^i$	$14 + J_{r+13}^i$	$15 + J_{r+14}^i$	$16 + J_{r+15}^i$
Q	R	S	T	U	V	W	X
$17 + J_{r+16}^i$	$18 + J_{r+17}^i$	$19 + J_{r+18}^i$	$20 + J_{r+19}^i$	$21 + J_{r+20}^i$	$22 + J_{r+21}^i$	$23 + J_{r+22}^i$	$24 + J_{r+23}^i$
Y	Z	0	1	2	3	4	5
$25 + J_{r+24}^i$	$26 + J_{r+25}^i$	$27 + J_{r+26}^i$	$28 + J_{r+27}^i$	$29 + J_{r+28}^i$	$30 + J_{r+29}^i$	$31 + J_{r+32}^i$	$32 + J_{r+33}^i$
6	7	8	9	$!$	$?$	$.$	$,$
$33 + J_{r+32}^i$	$34 + J_{r+33}^i$	$35 + J_{r+34}^i$	$36 + J_{r+35}^i$	$37 + J_{r+34}^i$	$38 + J_{r+35}^i$	$39 + J_{r+38}^i$	$40 + J_{r+39}^i$

For more clarity and conciseness, let us illustrate that the coding/decoding method works efficiently with two examples. The first example concerns the encoding/decoding of a single word and the second example involves the encoding/decoding of a sentence. We point out that, for these examples the steps of our method also can be checked by utilizing some famous mathematical software such as Maple or Mathematica.

Example 1: Let us code/decode the following word utilizing our method

"MATHEMATICS".

Initially, we encrypt the text taking into account the steps of the coding algorithm given above.

- **Step 1:** Firstly, let us create the message matrix M as a square matrix with the help of the given word "MATHEMATICS". Such matrix is as follows

$$M = \begin{bmatrix} M & A & T & 0 \\ H & E & M & 0 \\ A & T & I & 0 \\ C & S & ! & ! \end{bmatrix}.$$

- **Step 2:** Let us substitute a general Jacobsthal number for each character from the Latin alphabet given in Table 2 and obtain the message matrix M . For $r = 1$ and $i = 2$ we have

$$M = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} = \begin{bmatrix} 2744 & 2 & 349545 & 44739270 \\ 93 & 16 & 2744 & 44739270 \\ 2 & 349545 & 180 & 44739270 \\ 6 & 174782 & 45812984528 & 45812984528 \end{bmatrix}.$$

- **Step 3:** For $k = 2$ and $n = 1$ we have

$$E_{2,1}(x) = Q_2(x)R_2(x) = \begin{bmatrix} 1 & 2x \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2x & 2x \\ 2x & 0 \end{bmatrix}.$$

- **Step 4:** The encrypted submatrices are given by

$$c_{11} = m_{11} \times E_{2,1}(x) = \begin{bmatrix} 2744 & 2 \\ 93 & 16 \end{bmatrix} \begin{bmatrix} 2x & 2x \\ 2x & 0 \end{bmatrix} = \begin{bmatrix} 5492x & 5488x \\ 218x & 186x \end{bmatrix},$$

$$\begin{aligned} c_{12} &= m_{12} \times E_{2,1}(x) = \begin{bmatrix} 349545 & 44739270 \\ 2744 & 44739270 \end{bmatrix} \begin{bmatrix} 2x & 2x \\ 2x & 0 \end{bmatrix} \\ &= \begin{bmatrix} 90177630x & 699090x \\ 89484028x & 5488x \end{bmatrix}, \end{aligned}$$

$$c_{21} = m_{21} \times E_{2,1}(x) = \begin{bmatrix} 2 & 349545 \\ 6 & 174782 \end{bmatrix} \begin{bmatrix} 2x & 2x \\ 2x & 0 \end{bmatrix} = \begin{bmatrix} 699094x & 4x \\ 349576x & 12x \end{bmatrix},$$

$$\begin{aligned} c_{22} &= m_{22} \times E_{2,1}(x) = \begin{bmatrix} 180 & 44739270 \\ 45812984528 & 45812984528 \end{bmatrix} \begin{bmatrix} 2x & 2x \\ 2x & 0 \end{bmatrix} \\ &= \begin{bmatrix} 89478900x & 360x \\ 183251938112x & 91625969056x \end{bmatrix}. \end{aligned}$$

Hence, we obtain the encrypted matrix C as follows

$$C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} 5492x & 5488x & 90177630x & 699090x \\ 218x & 186x & 89484028x & 5488x \\ 699094x & 4x & 89478900x & 360x \\ 349576x & 12x & 183251938112x & 91625969056x \end{bmatrix}.$$

- **Step 5:** Send the encrypted matrix C to the receiver.

We decrypt the coded matrix taking into account the steps of the decoding algorithm given above.

- **Step 1:** Let us consider the encrypted matrix C and create the inverse $(E_{2,1}(x))^{-1}$ of the coding matrix, which is easily given by $(E_{2,1}(x))^{-1} = \begin{bmatrix} 0 & 1/2x \\ 1/2x & -1/2x \end{bmatrix}$.
- **Step 2:** To decode the encrypted message, we multiply the submatrices with their inverse, namely,

$$\begin{aligned}
 m_{11} &= c_{11} \times (E_{2,1}(x))^{-1} = \begin{bmatrix} 5492x & 5488x \\ 218x & 186x \end{bmatrix} \begin{bmatrix} 0 & 1/2x \\ 1/2x & -1/2x \end{bmatrix} \\
 &= \begin{bmatrix} 2744 & 2 \\ 93 & 16 \end{bmatrix}, \\
 m_{12} &= c_{12} \times (E_{2,1}(x))^{-1} = \begin{bmatrix} 90177630x & 699090x \\ 89484028x & 5488x \end{bmatrix} \begin{bmatrix} 0 & 1/2x \\ 1/2x & -1/2x \end{bmatrix} \\
 &= \begin{bmatrix} 349545 & 44739270 \\ 2744 & 44739270 \end{bmatrix}, \\
 m_{21} &= c_{21} \times (E_{2,1}(x))^{-1} = \begin{bmatrix} 699094x & 4x \\ 349576x & 12x \end{bmatrix} \begin{bmatrix} 0 & 1/2x \\ 1/2x & -1/2x \end{bmatrix} \\
 &= \begin{bmatrix} 2 & 349545 \\ 6 & 174782 \end{bmatrix}, \\
 m_{22} &= c_{22} \times (E_{2,1}(x))^{-1} = \\
 &\begin{bmatrix} 89478900x & 360x \\ 183251938112x & 91625969056x \end{bmatrix} \begin{bmatrix} 0 & 1/2x \\ 1/2x & -1/2x \end{bmatrix} \\
 &= \begin{bmatrix} 180 & 44739270 \\ 45812984528 & 45812984528 \end{bmatrix}.
 \end{aligned}$$

Hence, we obtain matrix of the decode message under the following form

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} 2744 & 2 & 349545 & 44739270 \\ 93 & 16 & 2744 & 44739270 \\ 2 & 349545 & 180 & 44739270 \\ 6 & 174782 & 45812984528 & 45812984528 \end{bmatrix}.$$

- **Step 3:** Taking into account the character from the Latin alphabet given in Table 2, we obtain the message matrix M of order k . Such matrix is given under the form

$$\begin{aligned}
 M &= \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} 2744 & 2 & 349545 & 44739270 \\ 93 & 16 & 2744 & 44739270 \\ 2 & 349545 & 180 & 44739270 \\ 6 & 174782 & 45812984528 & 45812984528 \end{bmatrix} \\
 &= \begin{bmatrix} M & A & T & 0 \\ H & E & M & 0 \\ A & T & I & 0 \\ C & S & ! & ! \end{bmatrix}_{4 \times 4}.
 \end{aligned}$$

- **Step 4:** The decoded message is obtained as "*MATHEMATICS*".

Example 2: By utilizing the same method, let us code/decode the following sentence

"MATHISEVERYWHERE"

Initially, we encrypt the text taking into account the steps of the coding algorithm given above.

- **Step 1:** Let us create the code matrix using the message text. Such matrix is given under the following form

$$M = \begin{bmatrix} M & A & T & H \\ I & S & E & V \\ E & R & Y & W \\ H & E & R & E \end{bmatrix}_{4 \times 4}$$

- **Step 2:** Let us write the code matrix C according to the alphabet table for arbitrary value s . For $r = 1$ and $i = 2$ we have

$$M = \begin{bmatrix} 2744 & 2 & 349545 & 93 \\ 180 & 174782 & 16 & 1398123 \\ 16 & 87399 & 11184836 & 2796226 \\ 93 & 16 & 87399 & 16 \end{bmatrix}.$$

- **Step 3:** For $k = 4$ and $n = 1$ we get the following matrix

$$E_{4,5}(x) = Q_4(x)R_4(x) = \begin{bmatrix} x^3 + 4x^4 + x^5 & x^3 + 2x^4 & x^3 & x^3 \\ x^3 + 2x^4 & x^3 & x^3 & 0 \\ x^3 & x^3 & 0 & 0 \\ x^3 & 0 & 0 & 0 \end{bmatrix}.$$

- **Step 4:** The matrix related to the code message is given by

$$C = M \times E_{4,5}(x) = \begin{bmatrix} 2744 & 2 & 349545 & 93 \\ 180 & 174782 & 16 & 1398123 \\ 16 & 87399 & 11184836 & 2796226 \\ 93 & 16 & 87399 & 16 \end{bmatrix} \begin{bmatrix} x^3 + 4x^4 + x^5 & x^3 + 2x^4 & x^3 & x^3 \\ x^3 + 2x^4 & x^3 & x^3 & 0 \\ x^3 & x^3 & 0 & 0 \\ x^3 & 0 & 0 & 0 \end{bmatrix} \\ = \begin{bmatrix} 349638x^3 + 2(x^3 + 2x^4) + 2744(x^3 + 4x^4 + x^5) & 349547x^3 + 2744(x^3 + 2x^4) & 2746x^3 & 2744x^3 \\ 1398139x^3 + 174782(x^3 + 2x^4) + 180(x^3 + 4x^4 + x^5) & 174798x^3 + 180(x^3 + 2x^4) & 174962x^3 & 180x^3 \\ 13981062x^3 + 87399(x^3 + 2x^4) + 16(x^3 + 4x^4 + x^5) & 11272235x^3 + 16(x^3 + 2x^4) & 87415x^3 & 16x^3 \\ 87415x^3 + 16(x^3 + 2x^4) + 93(x^3 + 4x^4 + x^5) & 87415x^3 + 93(x^3 + 2x^4) & 109x^3 & 93x^3 \end{bmatrix}.$$

- **Step 5:** Send the encrypted matrix C to the receiver.

We decrypt the coded matrix taking into account the steps of the decoding algorithm given above.

- **Step 1:** Let us consider the encrypted matrix C and create the inverse $(E_{4,5}(x))^{-1}$ of the coding matrix. This inverse is given by

$$(E_{4,5}(x))^{-1} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{x^3} \\ 0 & 0 & \frac{1}{x^3} & \frac{-1}{x^3} \\ 0 & \frac{1}{x^3} & \frac{-1}{x^3} & \frac{-2}{x^2} \\ \frac{1}{x^3} & \frac{-1}{x^3} & \frac{-2}{x^2} & \frac{-1}{x} \end{bmatrix}.$$

- **Step 2:** To decode the encrypted message, we multiply the submatrices with their inverse

$$\begin{aligned} M &= C \times (E_{4,5}(x))^{-1} \\ &= \begin{bmatrix} 349638x^3 + 2(x^3 + 2x^4) + 2744(x^3 + 4x^4 + x^5) & 349547x^3 + 2744(x^3 + 2x^4) & 2746x^3 & 2744x^3 \\ 1398139x^3 + 174782(x^3 + 2x^4) + 180(x^3 + 4x^4 + x^5) & 174798x^3 + 180(x^3 + 2x^4) & 174962x^3 & 180x^3 \\ 13981062x^3 + 87399(x^3 + 2x^4) + 16(x^3 + 4x^4 + x^5) & 11272235x^3 + 16(x^3 + 2x^4) & 87415x^3 & 16x^3 \\ 87415x^3 + 16(x^3 + 2x^4) + 93(x^3 + 4x^4 + x^5) & 87415x^3 + 93(x^3 + 2x^4) & 109x^3 & 93x^3 \end{bmatrix} \\ &\quad \times \begin{bmatrix} 0 & 0 & 0 & \frac{1}{x^3} \\ 0 & 0 & \frac{1}{x^3} & \frac{-1}{x^3} \\ 0 & \frac{1}{x^3} & \frac{-1}{x^3} & \frac{-2}{x^2} \\ \frac{1}{x^3} & \frac{-1}{x^3} & \frac{-2}{x^2} & \frac{-1}{x} \end{bmatrix} = \begin{bmatrix} 2744 & 2 & 349545 & 93 \\ 180 & 174782 & 16 & 1398123 \\ 16 & 87399 & 11184836 & 2796226 \\ 93 & 16 & 87399 & 16 \end{bmatrix}. \end{aligned}$$

Therefore, the matrix of the decoded message is obtained under the form

$$M = \begin{bmatrix} 2744 & 2 & 349545 & 93 \\ 180 & 174782 & 16 & 1398123 \\ 16 & 87399 & 11184836 & 2796226 \\ 93 & 16 & 87399 & 16 \end{bmatrix}.$$

- **Step 3:** Taking into account the character from the Latin alphabet given in Table 2, the message matrix M of order $k = 4$ is obtained under the form

$$M = \begin{bmatrix} M & A & T & H \\ I & S & E & V \\ E & R & Y & W \\ H & E & R & E \end{bmatrix}_{4 \times 4}.$$

- **Step 4:** The decoded message is obtained as "*MATH IS EVERYWHERE*".

4. Error Detection for $k = 2$

In this section, we describe a method for the construction of error-detecting and error-correcting codes as a part of algebraic coding theory. For this purpose and for greater clarity we will base ourselves on the previous examples.

Let us describe the first message as a nonsingular square matrix M of order 2, written under the form

$$M = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}. \quad (10)$$

Assume that all matrix entries are positive integers, i.e., $m_1, m_2, m_3, m_4 > 0$. Let us now choose any value of n for a $E_{2,2}(x)$ matrix in (6), namely, $E_{2,2}(x) = \begin{bmatrix} J_1(x) & J_0(x) \\ J_0(x) & J_{-1}(x) \end{bmatrix} = \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix}$. Then, the inverse is $(E_{2,2}(x))^{-1} = \frac{1}{2x} \begin{bmatrix} J_{-1}(x) & -J_0(x) \\ -J_0(x) & J_1(x) \end{bmatrix} = \begin{bmatrix} \frac{1}{2x} & 0 \\ 0 & 1 \end{bmatrix}$. Given the foregoing, we have the following statement

$$M \times E_{2,2}(x) = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} = C,$$

where

$$c_1 = 2xm_1, \quad c_2 = m_2, \quad c_3 = 2xm_3 \quad \text{and} \quad c_4 = m_4.$$

Then, the code message $C = (c_1, c_2, c_3, c_4)$ is sent to a channel. In the following statement, it is explained how to decode the code message E .

$$\begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} \begin{bmatrix} \frac{1}{2x} & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1/2x & c_2 \\ c_3/2x & c_4 \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} = M.$$

As a general form, the coding matrix C and the message matrix M can be written as

$$C = M \times E_{2,2}^n(x) = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} J_{n-1}(x) & J_{n-2}(x) \\ J_{n-2}(x) & J_{n-3}(x) \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix},$$

and

$$\begin{aligned} M &= C \times (E_{2,2}^n(x))^{-1} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} \begin{bmatrix} \frac{J_{n-3}(x)}{(-1)^{n-2}x^{n-3}} & -\frac{J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} \\ -\frac{J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} & \frac{J_{n-1}(x)}{(-1)^{n-2}x^{n-3}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{J_{n-3}(x)c_1 - c_2J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} & \frac{-J_{n-2}(x)c_1 + c_2J_{n-1}(x)}{(-1)^{n-2}x^{n-3}} \\ \frac{J_{n-3}(x)c_3 - c_4J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} & \frac{-J_{n-2}(x)c_3 + c_4J_{n-1}(x)}{(-1)^{n-2}x^{n-3}} \end{bmatrix}. \end{aligned}$$

Since m_1, m_2, m_3, m_4 are positive integers, we obtain

$$m_1 = \frac{J_{n-3}(x)c_1 - c_2J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} > 0, \quad m_2 = \frac{-J_{n-2}(x)c_1 + c_2J_{n-1}(x)}{(-1)^{n-2}x^{n-3}} > 0, \quad (11)$$

$$m_3 = \frac{J_{n-3}(x)c_3 - c_4J_{n-2}(x)}{(-1)^{n-2}x^{n-3}} > 0, \quad m_4 = \frac{-J_{n-2}(x)c_3 + c_4J_{n-1}(x)}{(-1)^{n-2}x^{n-3}} > 0. \quad (12)$$

We show that from (12) we get $\frac{J_{n-2}(x)}{J_{n-3}(x)} < -\frac{c_1}{c_2} < \frac{J_{n-1}(x)}{J_{n-2}(x)}$. And Expression (12) implies that we have $\frac{J_{n-2}(x)}{J_{n-3}(x)} < -\frac{c_3}{c_4} < \frac{J_{n-1}(x)}{J_{n-2}(x)}$. Consequently, when n converges to infinity we obtain

$$-\frac{c_1}{c_2} \approx \lambda(x), \quad -\frac{c_3}{c_4} \approx \lambda(x) \quad \text{where} \quad \lambda(x) = \frac{1}{2}(1 + \sqrt{1 + 8x}). \quad (13)$$

4.1 Error detection

Detecting and correcting errors in the code message C , caused by some external factors such as noise in the communication channel, is one of the main

goals of coding theory. The main idea is to use the matrix's determinant property as the message's verification criterion. Let's say the first message M gets transmitted by

$$\begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}$$

where all elements m_1, m_2, m_3, m_4 of the matrix M are positive integers. The determinant of M is $\det M = m_1 m_4 - m_2 m_3$ and the matrix of the code message C is given under the form

$$C = M \times E_{2,2}^n(x). \quad (14)$$

Hence,

$$\det C = \det(M \times E_{2,2}^n(x)) = \det M \times \det E_{2,2}^n(x) = \det M \times (-1)^{n-2} x^{n-3}. \quad (15)$$

This result establishes a direct connection between the determinant of the original message matrix M and that of the encoded matrix C . The relation expressed in equation (15) offers a novel approach for error detection utilizing the matrix $E_{2,2}^n(x)$. In this method, the sender calculates the determinant of the original message M (as defined in equation (10)) and transmits it together with the encoded message C through the communication channel. The receiver then computes the determinant of the received matrix C and compares it to the originally sent determinant M . If the equality described in (14) holds, the received code matrix C is confirmed to be free of errors and can be accurately decoded. Conversely, a discrepancy indicates the presence of errors in the transmitted message. Detecting such errors is a crucial initial step in ensuring reliable communication.

4.2 Error correction

The $E_{2,2}^n(x)$ matrix can be used to realize the possibility of restoring the code message, C . For $n = 2$ we get

$$E_{2,2}^2(x) = \begin{bmatrix} J_1(x) & J_0(x) \\ J_0(x) & J_{-1}(x) \end{bmatrix} = \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix}.$$

Subsequently, the multiplication of the message matrix, namely,

$$M \times E_{2,2}^2(x) = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} 2x & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2xm_1 & m_2 \\ 2xm_3 & m_4 \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} = C \quad (16)$$

where

$$c_1 = 2xm_1, \quad c_2 = m_2, \quad c_3 = 2xm_3 \quad \text{and} \quad c_4 = m_4.$$

We determine the determinant of the message matrix M after creating the code matrix E . The determinant $C = (c_1, c_2, c_3, c_4)$ is transmitted to the contact channel after the coding message. Assume that the code message C contains a unique way for the communication channel to identify the mistake in each of its constituents c_1, c_2, c_3 and c_4 . Assume for the moment that an error is encountered with c_1 , the first element in C . When u is the corrupted element of code message C , we can then describe the code message in matrix form as

follows $C' = \begin{bmatrix} u & c_2 \\ c_3 & c_4 \end{bmatrix}$. However, the remaining matrix entries need to be accurate and equivalent to the following

$$c_2 = m_2, \quad c_3 = 2xm_3 \quad \text{and} \quad c_4 = m_4.$$

Then, using the coding method's principles, we may create the following equation to calculate u

$$uc_4 - c_2c_3 = um_4 - 2xm_2m_3 = 2x(m_1m_4 - m_2m_3). \quad (17)$$

From (17) we get

$$u = 2xm_1. \quad (18)$$

By comparing the computed value in (18) with the matrix entry c_1 of the code matrix C given in (16), we identify that $u = c_1$. This comparison utilizes the determinant property of the matrix $E_{2,2}^n(x)$ to reconstruct the encoded message C . However, in practical scenarios, it is typically unknown which element of the received code has been corrupted. To address this uncertainty, we consider multiple hypotheses regarding the potentially erroneous elements and evaluate each case accordingly. An additional constraint in our model is that all entries in the code matrix C are required to be integers.

We begin with the assumption that a single error has occurred in the matrix C during transmission. Under this assumption, we distinguish four possible types of single-entry errors that may affect the integrity of the matrix.

$$(a) \begin{bmatrix} u & c_2 \\ c_3 & c_4 \end{bmatrix}, (b) \begin{bmatrix} c_1 & v \\ c_3 & c_4 \end{bmatrix}, (c) \begin{bmatrix} c_1 & c_2 \\ w & c_4 \end{bmatrix}, (d) \begin{bmatrix} c_1 & c_2 \\ c_3 & z \end{bmatrix} \quad (19)$$

where u, v, w, z are the corrupted elements. Here, we may test many theories (19). Based on the checking relation (15), we can build the following algebraic equations to test the hypotheses (a), (b), (c), and (d)

$$uc_4 - c_2c_3 = \det M \det(E_{2,2}^n(x)) \quad (a \text{ single-error scenario in which the corrupted entry is } c_1), \quad (20)$$

$$c_1c_4 - vc_3 = \det M \det(E_{2,2}^n(x)) \quad (a \text{ single-error scenario in which the corrupted entry is } c_2), \quad (21)$$

$$c_1c_4 - c_2w = \det M \det(E_{2,2}^n(x)) \quad (a \text{ single-error scenario in which the corrupted entry is } c_3), \quad (22)$$

$$c_1z - c_2c_3 = \det M \det(E_{2,2}^n(x)) \quad (a \text{ single-error scenario in which the corrupted entry is } c_4), \quad (23)$$

According to (20)-(23), there are four ways of calculating the probable single mistakes.

$$u = \frac{\det M \det(E_{2,2}^n(x)) + c_2c_3}{c_4}, \quad (24)$$

$$v = \frac{-\det M \det(E_{2,2}^n(x)) + c_1c_4}{c_3}, \quad (25)$$

$$w = \frac{-\det M \det(E_{2,2}^n(x)) + c_1 c_4}{c_2}, \quad (26)$$

$$z = \frac{\det M \det(E_{2,2}^n(x)) + c_2 c_3}{c_1}. \quad (27)$$

The formulas (24)–(27) offer four distinct variations corresponding to a single-error case. However, the correct version must be identified by selecting integer-valued solutions for the variables u, v, w and z , while also ensuring that these values satisfy the additional consistency conditions specified in (13). If the evaluations of (24) or (27) do not yield integers, it may indicate that the assumed error position is incorrect, or alternatively, that the check element $\det(M)\det(E_{2,2}^n(x))$ contains an error. In such a case, the approximations provided by (13) can be used to assess the validity of the code matrix C . Similarly, all possible double-error configurations in the code matrix can be systematically tested using this approach. For instance, consider the double-error case represented by the matrix

$$\begin{bmatrix} u & v \\ c_3 & c_4 \end{bmatrix} \quad (28)$$

where u and v are the corrupted elements. Applying the first consistency condition (15), we derive the following algebraic equation corresponding to matrix (28):

$$uc_4 - vc_3 = \det M \det(E_{2,2}^n(x)). \quad (29)$$

On the other hand, per the second checking relation (13), the relationship between u and v is as follows

$$u \approx -\lambda(x)v. \quad (30)$$

It is important to emphasize that equation (29) is a Diophantine equation that admits multiple solutions. Among these, the values of u and v must be chosen in such a way that the consistency condition given by (30) is satisfied. By analogy, one can show that all possible double-error patterns in the code matrix can be corrected by jointly applying the checking relations (13) and (15) together with a suitable solution to a Diophantine equation similar to (29). Furthermore, this approach can be extended to correct triple-error cases in the code matrix C , such as

$$\begin{bmatrix} u & v \\ w & c_4 \end{bmatrix}$$

where the corrupted elements are u, v and w .

Our approach to repair error relies on exploiting the relations (13), (15), and the fact that the code matrix's elements are integers to verify various hypotheses regarding defects in the code matrix. We must reject the code matrix C as defective and irrecoverable if none of our solutions result in integer outcomes that satisfy (13) and (15). This indicates that the checking element $\det M$ is incorrect or that there is a case of a fourfold error in C . Among the $({}^4C_1 + {}^4C_2 + {}^4C_3 + {}^4C_4) = 2^4 - 1 = 15$ examples, our approach permits the rectification of 14 cases. This also indicates that the method's corrective ability is $\frac{14}{15} = 0.9333 = 93.33\%$.

5. Concluding Remarks

In this study, we propose a novel coding and decoding framework that utilizes a generalized form of Jacobsthal polynomials. Unlike existing methods, our approach introduces a new alphabet constructed from generalized Jacobsthal numbers and incorporates k -order Jacobsthal polynomials into the coding process. Several illustrative examples are provided to demonstrate the practical application of the method. To the best of our knowledge, the results presented here have not yet been addressed in the existing literature.

Polynomials are known to play a fundamental role in mathematics and have become an essential component of modern cryptographic systems. In particular, they are central to the processes of encryption and decryption in secure communication. In this context, the study opens the door to new perspectives in coding theory. Specifically, the matrix-based structure underlying the Jacobsthal coding technique offers distinctive advantages over classical algebraic methods. One of its key features is the integration of matrix theory in constructing novel error-correcting codes, where errors can be detected and corrected at the level of matrix elements, enabling efficient handling of larger information units.

6. Acknowledgement

This work was partially presented at the ICMASE 2024 (V International Conference on Mathematics and its Applications in Science and Engineering) conference. The author Mustapha Rachidi, Permanent Professor in PPGEdumat, of the INMA of the Federal University of Mato Grosso do Sul (MS, Brazil), expresses his sincere thanks to the INMA and the UFMS for their precious support.

References

- [1] M. Asci, S. Aydinyuz, "k-order Gaussian Fibonacci polynomials and applications to the coding/decoding theory," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 26, no. 5, pp. 1399-1416, (2022).
- [2] A. Kumar Yadav, M. Pruthi, "Generalized k-step Jacobsthal numbers coding and decoding method," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 26, no. 6, pp. 1773-1799, (2023).
- [3] B. Kuloğlu, E. Özkan, "Applications of Jacobsthal and Jacobsthal-Lucas numbers in coding theory," *Mathematica Montisnigri*, vol. 57, pp. 54-64, (2023).
- [4] M. Basu, B. Prasad, "The generalized relations among the code elements for Fibonacci coding theory," *Chaos Solitons Fractals*, vol. 41, no. 5, pp. 2517-2525, (2009).

- [5] S. Uçar, N. Taş, and N. Y. Özgür, "A new application to coding theory via Fibonacci and Lucas numbers," *Mathematical Sciences and Applications E-Notes*, vol. 7, no. 1, pp. 62–70 (2019).
- [6] N. Taş, S. Uçar, N. Y. Özgür, and Ö. Ö. Kaymak, "A new coding/decoding algorithm using Fibonacci numbers," *Discrete Mathematics, Algorithms and Applications*, vol. 10, no. 2, pp. 1850028 (2018).
- [7] P. Sundarayya and M. G. Vara Prasad, "Coding theory on Pell–Lucas p-numbers," *Journal of Physics: Conference Series*, vol. 1344, no. 1, pp. 012017 (2019).
- [8] M. Basu and M. Das, "Coding theory on generalized Fibonacci n-step polynomials," *Journal of Information and Optimization Sciences*, vol. 38, no. 1, pp. 83–131 (2017).
- [9] B. Prasad, "Coding theory based on balancing polynomials," *Control and Cybernetics*, vol. 50, no. 2, pp. 335–346 (2021).
- [10] A. F. Horadam, "Jacobsthal representation numbers," *The Fibonacci Quarterly*, vol. 34, pp. 40–54 (1996).
- [11] A. F. Horadam, "Jacobsthal representation polynomials," *The Fibonacci Quarterly*, vol. 35, no. 2, pp. 137–148 (1997).
- [12] A. F. Horadam, "Convolutions for Jacobsthal-type polynomials," *The Fibonacci Quarterly*, vol. 40, no. 3, pp. 212–222 (2002).
- [13] P. Catarino and M. L. Morgado, "On generalized Jacobsthal and Jacobsthal–Lucas polynomials," *Analele Ştiinţifice ale Universităţii Ovidius din Constanţa*, vol. 24, no. 3, pp. 61–78 (2016).
- [14] G. B. Djordjević, "Generalized Jacobsthal polynomials," *The Fibonacci Quarterly*, vol. 38, no. 3, pp. 239–243 (2000).
- [15] V. E. Hoggatt Jr. and M. Bicknell-Johnson, "Convolution arrays for Jacobsthal and Fibonacci polynomials," *The Fibonacci Quarterly*, vol. 16, no. 5, pp. 385–402 (1978).
- [16] N. Kiliç, "On split k-Jacobsthal and k-Jacobsthal–Lucas quaternions," *Ars Combinatoria*, vol. 142, pp. 129–139 (2019).
- [17] N. Kiliç, "On k-Jacobsthal and k-Jacobsthal–Lucas octonions," *JP Journal of Algebra, Number Theory and Applications*, vol. 41, no. 1, pp. 1–17 (2019).
- [18] A. Tereszkievicz and I. Wawreniuk, "Generalized Jacobsthal polynomials and special points for them," *Applied Mathematics and Computation*, vol. 268, pp. 806–814 (2015).

- [19] T. Koshy and M. Salmassi, "Cassini-like formulas for Jacobsthal and Koshy polynomials," *The Mathematical Gazette*, vol. 90, no. 517, pp. 68–70 (2006).
- [20] T. Koshy, *Fibonacci and Lucas Numbers with Applications*, New York, NY, USA: Wiley (2001).
- [21] S. Arpacı and F. Yilmaz, "On k-order Jacobsthal polynomials and their properties," in *Mathematical Methods for Engineering Applications*, Cham, Switzerland: Springer, pp. 305–313 (2024).
- [22] F. Yilmaz and D. Bozkurt, "The generalized order-k Jacobsthal numbers," *International Journal of Contemporary Mathematical Sciences*, vol. 4, no. 34, pp. 1685–1694 (2009).

Received December, 2024