



Research article

Error detection and correction for coding theory on k-order Gaussian Fibonacci matrices

Suleyman Aydinyuz* and Mustafa Asci

Pamukkale University, Science Faculty, Department of Mathematics, Kinikli, Denizli, Turkey

* **Correspondence:** Email: aydinyuzsuleyman@gmail.com.

Abstract: In this study, the coding theory defined for k-order Gaussian Fibonacci polynomials is rearranged by taking $x = 1$. We call this coding theory the k-order Gaussian Fibonacci coding theory. This coding method is based on the Q_k, R_k and $E_n^{(k)}$ matrices. In this respect, it differs from the classical encryption method. Unlike classical algebraic coding methods, this method theoretically allows for the correction of matrix elements that can be infinite integers. Error detection criterion is examined for the case of $k = 2$ and this method is generalized to k and error correction method is given. In the simplest case, for $k = 2$, the correct capability of the method is essentially equal to 93.33%, exceeding all well-known correction codes. It appears that for a sufficiently large value of k , the probability of decoding error is almost zero.

Keywords: Fibonacci numbers; Gaussian Fibonacci numbers; k-order Gaussian Fibonacci numbers; k-order Gaussian Fibonacci matrices; k-order Gaussian Fibonacci coding/decoding; error detection and correction

1. Introduction

The Fibonacci sequence is defined $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$ with the initial conditions $F_0 = 0, F_1 = 1$. The Fibonacci sequence and the golden ratio are used in many fields such as cryptology, coding theory and quantum physics in [1–11]. Horadam defined the Gaussian Fibonacci numbers in [12,13] and gave some general identities about the Gaussian Fibonacci numbers. Jordan generalized the Gaussian Fibonacci numbers with a similar definition in [14]. The Gaussian Fibonacci sequence is defined as $GF_n = GF_{n-1} + GF_{n-2}$ with $n > 1$ where $GF_0 = i, GF_1 = 1$ in [14]. It can also be easily

seen F_n that $GF_n = F_n + iF_{n-1}$ where F_n is the n th Fibonacci number.

Asci and Gurel generalized these studies and defined the k -order Gaussian Fibonacci numbers in [15] by the following recurrence relation

$$GF_n^{(k)} = \sum_{j=1}^k GF_{n-j}^{(k)}, \text{ for } n > 0 \text{ and } k \geq 2$$

with boundary conditions for $1 - k \leq n \leq 0$,

$$GF_n^{(k)} = \begin{cases} 1-i & , \quad k=1-n \\ i & , \quad k=2-n \\ 0 & , \quad \text{otherwise.} \end{cases}$$

It can also be seen that $GF_n^{(k)} = F_n^{(k)} + iF_{n-1}^{(k)}$ where $F_n^{(k)}$ is the n th k -order Fibonacci number.

In [16], Asci and Aydinyuz defined the k -order Gaussian Fibonacci polynomials and gave some important results. The k -order Gaussian Fibonacci polynomials $\{GF_n^{(k)}\}_{n=0}^{\infty}$ are

$$GF_n^{(k)}(x) = \sum_{j=1}^k x^{k-j} GF_{n-j}^{(k)}(x)$$

for $n > 0$ and $k \geq 2$ with the initial conditions for $1 - k \leq n \leq 0$,

$$GF_n^{(k)}(x) = \begin{cases} 1-ix & , \quad k=1-n \\ i & , \quad k=2-n \\ 0 & , \quad \text{otherwise.} \end{cases}$$

It can be seen that

$$GF_n^{(k)}(x) = F_n^{(k)}(x) + iF_{n-1}^{(k)}(x)$$

where $F_n^{(k)}(x)$ is the n th k -order Fibonacci polynomial.

In order to ensure information security in terms of data transfer over communication channels, a lot of work has been done on this subject and continues to be done. Therefore, coding/decoding algorithms play an important role to ensure information security. Especially, Fibonacci coding theory is one of the most preferred in this field. We can see examples of these in many studies. For example, in [17], Stakhov gave a new approach to a coding theory using the generalization of the Cassini formula for Fibonacci p -numbers and Q_p -matrices in 2006. In 2009, Basu and Prasad in [18] presented the generalized relations among the code elements for the Fibonacci coding theory. Also, Basu and Das introduced a new coding theory for Tribonacci matrices in [19] and in 2014 they defined the coding theory for Fibonacci n -step numbers by generalizing the coding theory for Tribonacci numbers in [20]. Esmaeili in [21] also described a Fibonacci polynomial-based coding method with error detection and correction.

In this paper, we remember the coding theory for k -order Gaussian Fibonacci polynomials given in [16] by taking $x=1$ for k -order Gaussian Fibonacci numbers. We describe the coding theory for the k -order Gaussian Fibonacci numbers. We give illustrative examples. The interesting relation between the elements of the code-message matrix is analyzed and derived. The most important of this paper is that we deal with error detection and error correction methods and examples of error

probability is given.

2. k-Order Gaussian Fibonacci coding theory

When we take $x=1$ in the coding theory defined in [16], we obtain the k-order Gaussian Fibonacci coding method. First, we define the Q_k, R_k and $E_n^{(k)}$ matrices, which plays an important role in this coding theory. Q_k, R_k and $E_n^{(k)}$ are defined the $k \times k$ matrices as the following:

$$Q_k = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}_{k \times k}, \quad R_k = \begin{bmatrix} GF_{k-1}^{(k)} & GF_{k-2}^{(k)} & GF_{k-3}^{(k)} & \cdots & GF_2^{(k)} & GF_1^{(k)} & 0 \\ GF_{k-2}^{(k)} & GF_{k-3}^{(k)} & GF_{k-4}^{(k)} & \cdots & GF_1^{(k)} & 0 & 0 \\ GF_{k-3}^{(k)} & GF_{k-4}^{(k)} & GF_{k-5}^{(k)} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ GF_2^{(k)} & GF_1^{(k)} & 0 & \cdots & \vdots & 0 & 0 \\ GF_1^{(k)} & 0 & 0 & \cdots & 0 & 0 & i \\ 0 & 0 & 0 & \cdots & 0 & i & 1-i \end{bmatrix}_{k \times k}$$

and

$$E_n^{(k)} = \begin{bmatrix} GF_{n+k-1}^{(k)} & GF_{n+k-2}^{(k)} & \cdots & GF_{n+1}^{(k)} & GF_n^{(k)} \\ GF_{n+k-2}^{(k)} & GF_{n+k-3}^{(k)} & \cdots & GF_n^{(k)} & GF_{n-1}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ GF_{n+1}^{(k)} & GF_n^{(k)} & \cdots & GF_{n-k+3}^{(k)} & GF_{n-k+2}^{(k)} \\ GF_n^{(k)} & GF_{n-1}^{(k)} & \cdots & GF_{n-k+2}^{(k)} & GF_{n-k+1}^{(k)} \end{bmatrix}_{k \times k}$$

where $GF_n^{(k)}$ is the n th k-order Gaussian Fibonacci number.

Theorem 2.1. For $n \geq 1$, we get in [15] as follow:

$$Q_k^n R_k = E_n^{(k)}. \quad (2.1)$$

Corollary 2.1. For $k=2$, we get

$$\begin{aligned} Q^2 R &= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n \begin{bmatrix} 1 & i \\ i & 1-i \end{bmatrix} \\ &= \begin{bmatrix} GF_{n+1} & GF_n \\ GF_n & GF_{n-1} \end{bmatrix} \end{aligned}$$

where GF_n is the n th usual Gaussian Fibonacci number in [15].

Corollary 2.2. For $k=3$, we get

$$Q_3^n R_3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^n \begin{bmatrix} 1+i & 1 & 0 \\ 1 & 0 & i \\ 0 & i & 1-i \end{bmatrix}$$

$$= \begin{bmatrix} GT_{n+2} & GT_{n+1} & GT_n \\ GT_{n+1} & GT_n & GT_{n-1} \\ GT_n & GT_{n-1} & GT_{n-2} \end{bmatrix}$$

where GT_n is the n th Gaussian Tribonacci number in [23].

2.1. Applications of k -order Gaussian Fibonacci numbers to coding theory

In this section, we redefine k -order Gaussian Fibonacci coding theory using k -order Gaussian Fibonacci numbers and Q_k, R_k and $E_n^{(k)}$ play very important role in the construction of k -order Gaussian Fibonacci coding theory.

We now obtain the matrix $E_n^{(k)}$ using Q_k, R_k matrices for the $k=2, k=3$ values and examine the inverse.

- For $k=2$, introducing the square matrix Q_2 of order 2 as:

$$Q_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

and the square matrix R_2 of order 2 as:

$$R_2 = \begin{bmatrix} 1 & i \\ i & 1-i \end{bmatrix}$$

for $n=1$, we can use (2.1)

$$\begin{aligned} E_1^{(2)} &= Q_2 \cdot R_2 \\ &= \begin{bmatrix} 1+i & 1 \\ 1 & i \end{bmatrix} \\ &= \begin{bmatrix} GF_2^{(2)} & GF_1^{(2)} \\ GF_1^{(2)} & GF_0^{(2)} \end{bmatrix} \end{aligned}$$

such that

$$\begin{aligned} \det E_1^{(2)} &= \det(Q_2 \cdot R_2) \\ &= (-1) \cdot (2-i) \\ &= -2+i. \end{aligned}$$

The inverse of $E_1^{(2)}$ is as:

$$\left(E_1^{(2)}\right)^{-1} = \begin{bmatrix} \frac{1}{5} - \frac{2}{5}i & \frac{2}{5} + \frac{1}{5}i \\ \frac{2}{5} + \frac{1}{5}i & -\frac{1}{5} - \frac{3}{5}i \end{bmatrix}$$

$$\begin{aligned}
&= \frac{1}{\det E_1^{(2)}} \begin{bmatrix} GF_0 & -GF_1 \\ -GF_1 & GF_2 \end{bmatrix} \\
&= \frac{1}{-2+i} \begin{bmatrix} GF_0 & -GF_1 \\ -GF_1 & GF_2 \end{bmatrix}
\end{aligned}$$

such that

$$\begin{aligned}
\det(E_1^{(2)})^{-1} &= \frac{1}{-2+i} \\
&= -\frac{2}{5} - \frac{1}{5}i.
\end{aligned}$$

Also, by (2.1) for $n = 2$, we can get $E_2^{(2)}$ as follows:

$$\begin{aligned}
E_2^{(2)} &= Q_2^2 \cdot R_2 \\
&= \begin{bmatrix} 2+i & 1+i \\ 1+i & 1 \end{bmatrix} \\
&= \begin{bmatrix} GF_3^{(2)} & GF_2^{(2)} \\ GF_2^{(2)} & GF_1^{(2)} \end{bmatrix}
\end{aligned}$$

such that

$$\begin{aligned}
\det E_2^{(2)} &= \det(Q_2^2 \cdot R_2) \\
&= (-1)^2 \cdot (2-i) \\
&= 2-i.
\end{aligned}$$

The inverse of $E_2^{(2)}$ is as:

$$\begin{aligned}
(E_2^{(2)})^{-1} &= \begin{bmatrix} \frac{2}{5} + \frac{1}{5}i & -\frac{1}{5} - \frac{3}{5}i \\ -\frac{1}{5} - \frac{3}{5}i & \frac{3}{5} + \frac{4}{5}i \end{bmatrix} \\
&= \frac{1}{\det E_2^{(2)}} \begin{bmatrix} GF_1^{(2)} & -GF_2^{(2)} \\ -GF_2^{(2)} & GF_3^{(2)} \end{bmatrix} \\
&= \frac{1}{2-i} \begin{bmatrix} GF_1^{(2)} & -GF_2^{(2)} \\ -GF_2^{(2)} & GF_3^{(2)} \end{bmatrix}
\end{aligned}$$

such that

$$\det(E_2^{(2)})^{-1} = \frac{1}{2-i}$$

$$= \frac{2}{5} + \frac{1}{5}i.$$

Theorem 2.1.1. $E_n^{(2)} = \begin{bmatrix} GF_{n+1}^{(2)} & GF_n^{(2)} \\ GF_n^{(2)} & GF_{n-1}^{(2)} \end{bmatrix}$ where $E_1^{(2)} = \begin{bmatrix} 1+i & 1 \\ 1 & i \end{bmatrix}$.

Theorem 2.1.2. $(E_n^{(2)})^{-1} = \frac{1}{\det E_n^{(2)}} \begin{bmatrix} GF_{n-1}^{(2)} & -GF_n^{(2)} \\ -GF_n^{(2)} & GF_{n+1}^{(2)} \end{bmatrix}$
 $= \frac{1}{(\det Q_2)^n \cdot \det R_2} \begin{bmatrix} GF_{n-1}^{(2)} & -GF_n^{(2)} \\ -GF_n^{(2)} & GF_{n+1}^{(2)} \end{bmatrix}$.

- For $k=3$, introducing the square matrix Q_3 of order 3 as:

$$Q_3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

and the square matrix R_3 of order 3 as:

$$R_3 = \begin{bmatrix} 1+i & 1 & 0 \\ 1 & 0 & i \\ 0 & i & 1-i \end{bmatrix}$$

for $n=1$, we can use (2.1)

$$\begin{aligned} E_1^{(3)} &= Q_3 \cdot R_3 \\ &= \begin{bmatrix} 2+i & 1+i & 1 \\ 1+i & 1 & 0 \\ 1 & 0 & i \end{bmatrix} \\ &= \begin{bmatrix} GF_3^{(3)} & GF_2^{(3)} & GF_1^{(3)} \\ GF_2^{(3)} & GF_1^{(3)} & GF_0^{(3)} \\ GF_1^{(3)} & GF_0^{(3)} & GF_{-1}^{(3)} \end{bmatrix} \end{aligned}$$

such that

$$\begin{aligned} \det E_1^{(3)} &= \det(Q_3 \cdot R_3) \\ &= 1.2i \\ &= 2i. \end{aligned}$$

The inverse of $E_1^{(3)}$ is as:

$$\begin{aligned}
(E_1^{(3)})^{-1} &= \begin{bmatrix} \frac{1}{2} & -\frac{1}{2}-\frac{1}{2}i & \frac{1}{2}i \\ -\frac{1}{2}-\frac{1}{2}i & 1+i & \frac{1}{2}-\frac{1}{2}i \\ \frac{1}{2}i & \frac{1}{2}-\frac{1}{2}i & -\frac{1}{2}-i \end{bmatrix} \\
&= \frac{1}{\det E_1^{(3)}} \begin{bmatrix} GF_{-1}^{(3)}GF_1^{(3)} - (GF_0^{(3)})^2 & GF_0^{(3)}GF_1^{(3)} - GF_{-1}^{(3)}GF_2^{(3)} & GF_0^{(3)}GF_2^{(3)} - (GF_1^{(3)})^2 \\ GF_0^{(3)}GF_1^{(3)} - GF_{-1}^{(3)}GF_2^{(3)} & GF_3^{(3)}GF_{-1}^{(3)} - (GF_1^{(3)})^2 & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} \\ (GF_1^{(3)})^2 - GF_0^{(3)}GF_2^{(3)} & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 \end{bmatrix} \\
&= \frac{1}{\det R_3} \begin{bmatrix} GF_{-1}^{(3)}GF_1^{(3)} - (GF_0^{(3)})^2 & GF_0^{(3)}GF_1^{(3)} - GF_{-1}^{(3)}GF_2^{(3)} & GF_0^{(3)}GF_2^{(3)} - (GF_1^{(3)})^2 \\ GF_0^{(3)}GF_1^{(3)} - GF_{-1}^{(3)}GF_2^{(3)} & GF_3^{(3)}GF_{-1}^{(3)} - (GF_1^{(3)})^2 & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} \\ (GF_1^{(3)})^2 - GF_0^{(3)}GF_2^{(3)} & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 \end{bmatrix}
\end{aligned}$$

such that

$$\begin{aligned}
\det(E_1^{(3)})^{-1} &= \frac{1}{2i} \\
&= -\frac{i}{2}.
\end{aligned}$$

Also, by (2.1) for $n = 2$, we can get $E_2^{(3)}$ as follows:

$$\begin{aligned}
E_2^{(3)} &= Q_3^2 . R_3 \\
&= \begin{bmatrix} 4+2i & 2+i & 1+i \\ 2+i & 1+i & 1 \\ 1+i & 1 & 0 \end{bmatrix} \\
&= \begin{bmatrix} GF_4^{(3)} & GF_3^{(3)} & GF_2^{(3)} \\ GF_3^{(3)} & GF_2^{(3)} & GF_1^{(3)} \\ GF_2^{(3)} & GF_1^{(3)} & GF_0^{(3)} \end{bmatrix}
\end{aligned}$$

such that

$$\begin{aligned}
\det E_2^{(3)} &= \det(Q_3^2 . R_3) \\
&= 1^2 . 2i \\
&= 2i.
\end{aligned}$$

The inverse of $E_2^{(3)}$ is as:

$$\begin{aligned} (E_2^{(3)})^{-1} &= \begin{bmatrix} \frac{1}{2}i & \frac{1}{2}-\frac{1}{2}i & -\frac{1}{2}-i \\ \frac{1}{2}-\frac{1}{2}i & -1 & \frac{1}{2}+\frac{3}{2}i \\ -\frac{1}{2}-i & \frac{1}{2}+\frac{3}{2}i & 1+\frac{1}{2}i \end{bmatrix} \\ &= \frac{1}{\det E_2^{(3)}} \begin{bmatrix} GF_0^{(3)}GF_2^{(3)} - (GF_1^{(3)})^2 & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 \\ GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_0^{(3)}GF_4^{(3)} - (GF_2^{(3)})^2 & GF_2^{(3)}GF_3^{(3)} - GF_1^{(3)}GF_4^{(3)} \\ GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 & GF_2^{(3)}GF_3^{(3)} - GF_1^{(3)}GF_4^{(3)} & GF_2^{(3)}GF_4^{(3)} - (GF_3^{(3)})^2 \end{bmatrix} \\ &= \frac{1}{\det R_3} \begin{bmatrix} GF_0^{(3)}GF_2^{(3)} - (GF_1^{(3)})^2 & GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 \\ GF_1^{(3)}GF_2^{(3)} - GF_0^{(3)}GF_3^{(3)} & GF_0^{(3)}GF_4^{(3)} - (GF_2^{(3)})^2 & GF_2^{(3)}GF_3^{(3)} - GF_1^{(3)}GF_4^{(3)} \\ GF_1^{(3)}GF_3^{(3)} - (GF_2^{(3)})^2 & GF_2^{(3)}GF_3^{(3)} - GF_1^{(3)}GF_4^{(3)} & GF_2^{(3)}GF_4^{(3)} - (GF_3^{(3)})^2 \end{bmatrix}. \end{aligned}$$

Theorem 2.1.3. $E_n^{(3)} = \begin{bmatrix} GF_{n+2}^{(3)} & GF_{n+1}^{(3)} & GF_n^{(3)} \\ GF_{n+1}^{(3)} & GF_n^{(3)} & GF_{n-1}^{(3)} \\ GF_n^{(3)} & GF_{n-1}^{(3)} & GF_{n-2}^{(3)} \end{bmatrix}$ where $E_1^{(3)} = \begin{bmatrix} 2+i & 1+i & 1 \\ 1+i & 1 & 0 \\ 1 & 0 & i \end{bmatrix}$.

Theorem 2.1.4.

$$(E_n^{(3)})^{-1} = \frac{1}{\det R_3} \begin{bmatrix} GF_{n-2}^{(3)}GF_n^{(3)} - (GF_{n-1}^{(3)})^2 & GF_{n-1}^{(3)}GF_n^{(3)} - GF_{n-2}^{(3)}GF_{n+1}^{(3)} & GF_{n-1}^{(3)}GF_{n+1}^{(3)} - (GF_n^{(3)})^2 \\ GF_{n-1}^{(3)}GF_n^{(3)} - GF_{n-2}^{(3)}GF_{n+1}^{(3)} & GF_{n-2}^{(3)}GF_{n+2}^{(3)} - (GF_n^{(3)})^2 & GF_n^{(3)}GF_{n+1}^{(3)} - GF_{n-1}^{(3)}GF_{n+2}^{(3)} \\ GF_{n-1}^{(3)}GF_{n+1}^{(3)} - (GF_n^{(3)})^2 & GF_n^{(3)}GF_{n+1}^{(3)} - GF_{n-1}^{(3)}GF_{n+2}^{(3)} & GF_n^{(3)}GF_{n+2}^{(3)} - (GF_{n+1}^{(3)})^2 \end{bmatrix}.$$

For arbitrary k-positive integers, the square matrix $E_n^{(k)}$ of order k and inverses can be found similarly.

2.2. k-order Gaussian Fibonacci coding/decoding method

In this section, we describe a new k-order Gaussian Fibonacci coding theory. We put our message in a matrix of M and let us represent the initial message in the form of the square matrix M of order k. We take the k-order Gaussian Fibonacci matrix $E_n^{(k)}$ as a coding matrix and its inverse matrix $(E_n^{(k)})^{-1}$ as a decoding matrix for an arbitrary positive integer k. The transformation $M \times E_n^{(k)} = C$ is called k-order Gaussian Fibonacci coding and we name the transformation $C \times (E_n^{(k)})^{-1} = M$ as k-order Gaussian Fibonacci decoding. We define C as a code matrix.

2.3. Illustrative examples of k -order Gaussian Fibonacci coding/decoding method

The given example is solved using the alphabet table below.

Using the arbitrary value of $s \geq 0$, we write the following alphabet according to $\pmod{30}$. We can extend the characters in the table according to our wishes. We begin the “ s ” for the first character in Table 1.

Table 1. Alphabet table.

A	B	C	D	E	F	G	H	I	J
s	$s+1$	$s+2$	$s+3$	$s+4$	$s+5$	$s+6$	$s+7$	$s+8$	$s+9$
K	L	M	N	O	P	Q	R	S	T
$s+10$	$s+11$	$s+12$	$s+13$	$s+14$	$s+15$	$s+16$	$s+17$	$s+18$	$s+19$
U	V	W	X	Y	Z	0	!	?	.
$s+20$	$s+21$	$s+22$	$s+23$	$s+24$	$s+25$	$s+26$	$s+27$	$s+28$	$s+29$

Example 2.3.1. Let us consider the message matrix for the following message text:

$$\text{“CODE”} \quad (2.2)$$

- **Step 1:** Let’s create the message matrix using the message text:

$$M = \begin{bmatrix} C & O \\ D & E \end{bmatrix}_{2 \times 2}$$

- **Step 2:** Let’s write the message matrix M according to the alphabet table for the arbitrary value “ s ” we choose. For $s = 2$;

$$M = \begin{bmatrix} 4 & 16 \\ 5 & 6 \end{bmatrix}$$

- **Step 3:** For $k = 2$, $n = 2$, we use (2.1);

$$\begin{aligned} E_2^{(2)} &= Q_2^2 \cdot R_2 \\ &= \begin{bmatrix} 2+i & 1+i \\ 1+i & 1 \end{bmatrix} \end{aligned}$$

- **Step 4:** The code message is:

$$\begin{aligned} C &= M \times E_2^{(2)} \\ &= \begin{bmatrix} 4 & 16 \\ 5 & 6 \end{bmatrix} \begin{bmatrix} 2+i & 1+i \\ 1+i & 1 \end{bmatrix} \\ &= \begin{bmatrix} 24+20i & 20+4i \\ 16+11i & 11+5i \end{bmatrix} \end{aligned}$$

- **Step 5:** The decode message is:

$$\begin{aligned}
 M &= C \times (E_2^{(2)})^{-1} \\
 &= \begin{bmatrix} 24+20i & 20+4i \\ 16+11i & 11+5i \end{bmatrix} \begin{bmatrix} \frac{2}{5} + \frac{1}{5}i & -\frac{1}{5} - \frac{3}{5}i \\ -\frac{1}{5} - \frac{3}{5}i & \frac{3}{5} + \frac{4}{5}i \end{bmatrix} \\
 &= \begin{bmatrix} 4 & 16 \\ 5 & 6 \end{bmatrix} \\
 &= \begin{bmatrix} C & O \\ D & E \end{bmatrix}.
 \end{aligned}$$

Example 2.3.2. Let us consider the message matrix for the following message text:

“PUBLIC KEY”

- **Step 1:** Let's create the message matrix using the message text:

$$M = \begin{bmatrix} P & U & B \\ L & I & C \\ K & E & Y \end{bmatrix}_{3 \times 3}$$

- **Step 2:** Let's write the message matrix M according to the alphabet table for the arbitrary value “ s ” we choose. For $s = 5$:

$$M = \begin{bmatrix} 20 & 25 & 6 \\ 16 & 13 & 7 \\ 15 & 9 & 29 \end{bmatrix}$$

- **Step 3:** For $k = 3$, $n = 6$, we use (2.1):

$$\begin{aligned}
 E_6^{(3)} &= Q_3^6 \cdot R_3 \\
 &= \begin{bmatrix} 44+24i & 24+13i & 13+7i \\ 24+13i & 13+7i & 7+4i \\ 13+7i & 7+4i & 4+2i \end{bmatrix}
 \end{aligned}$$

- **Step 4:** The code message is:

$$\begin{aligned}
 C &= M \times E_6^{(3)} \\
 &= \begin{bmatrix} 1558+847i & 847+459i & 459+252i \\ 1107+602i & 602+327i & 327+178i \\ 1253+680i & 680+374i & 374+199i \end{bmatrix}
 \end{aligned}$$

- **Step 5:** The decode message is:

$$\begin{aligned}
 M &= C \times (E_3^{(6)})^{-1} \\
 &= \begin{bmatrix} P & U & B \\ L & I & C \\ K & E & Y \end{bmatrix} \\
 &= \begin{bmatrix} 20 & 25 & 6 \\ 16 & 13 & 7 \\ 15 & 9 & 29 \end{bmatrix}.
 \end{aligned}$$

3. Connections between the code matrix elements

In this section, we consider the k -order Gaussian Fibonacci coding/decoding method for $k = 2$. We have an interesting relation among the elements of a code matrix C that has a crucial role in the error-correction process, outlined as follows:

$$C = M \times E_n^{(2)} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \times \begin{bmatrix} GF_{n+1}^{(2)} & GF_n^{(2)} \\ GF_n^{(2)} & GF_{n-1}^{(2)} \end{bmatrix} \quad (3.1)$$

$$= \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} \quad (3.2)$$

and

$$M = C \times (E_n^{(2)})^{-1} \quad (3.3)$$

$$\begin{aligned}
 &= \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} \times \left(\frac{1}{(\det(Q_2))^n \det R_2} \begin{bmatrix} GF_{n-1}^{(2)} & -GF_n^{(2)} \\ -GF_n^{(2)} & GF_{n+1}^{(2)} \end{bmatrix} \right) \quad (3.4) \\
 &= \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}
 \end{aligned}$$

For the case for an even integer $n = 2m$, we obtain the following equation

$$\begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} \times \left(\frac{1}{2-i} \begin{bmatrix} GF_{n-1}^{(2)} & -GF_n^{(2)} \\ -GF_n^{(2)} & GF_{n+1}^{(2)} \end{bmatrix} \right) \quad (3.5)$$

since the $\det(Q_2) = 1$ and $\det(R_2) = 2 - i$.

It follows from (3.5) that the elements of the matrix M can be obtained according to the following formulas:

$$m_1 = \frac{1}{2-i} (GF_{n-1}^{(2)} c_1 - GF_n^{(2)} c_2) \quad (3.6)$$

$$m_2 = \frac{1}{2-i} \left(-GF_n^{(2)} c_1 + GF_{n+1}^{(2)} c_2 \right) \quad (3.7)$$

$$m_3 = \frac{1}{2-i} \left(GF_{n-1}^{(2)} c_3 - GF_n^{(2)} c_4 \right) \quad (3.8)$$

$$m_4 = \frac{1}{2-i} \left(-GF_n^{(2)} c_3 + GF_{n+1}^{(2)} c_4 \right) \quad (3.9)$$

Since the $s \geq 0$, the elements of the matrix M are

$$m_1 \geq 0, \quad m_2 \geq 0, \quad m_3 \geq 0 \quad \text{and} \quad m_4 \geq 0 \quad (3.10)$$

Because of the condition (3.10) we can write the equalities (3.6)–(3.9) as:

$$m_1 = \frac{1}{2-i} \left(GF_{n-1}^{(2)} c_1 - GF_n^{(2)} c_2 \right) \geq 0 \quad (3.11)$$

$$m_2 = \frac{1}{2-i} \left(-GF_n^{(2)} c_1 + GF_{n+1}^{(2)} c_2 \right) \geq 0 \quad (3.12)$$

$$m_3 = \frac{1}{2-i} \left(GF_{n-1}^{(2)} c_3 - GF_n^{(2)} c_4 \right) \geq 0 \quad (3.13)$$

$$m_4 = \frac{1}{2-i} \left(-GF_n^{(2)} c_3 + GF_{n+1}^{(2)} c_4 \right) \geq 0 \quad (3.14)$$

From the Eqs (3.11) and (3.12), we can get

$$\frac{GF_n^{(2)}}{GF_{n-1}^{(2)}} c_2 \leq c_1 \leq \frac{GF_{n+1}^{(2)}}{GF_n^{(2)}} c_2 \quad (3.15)$$

or

$$\frac{GF_n^{(2)}}{GF_{n-1}^{(2)}} \leq \frac{c_1}{c_2} \leq \frac{GF_{n+1}^{(2)}}{GF_n^{(2)}} \quad (3.16)$$

Similarly, from (3.13) and (3.14), we can obtain:

$$\frac{GF_n^{(2)}}{GF_{n-1}^{(2)}} c_4 \leq c_3 \leq \frac{GF_{n+1}^{(2)}}{GF_n^{(2)}} c_4 \quad (3.17)$$

or

$$\frac{GF_n^{(2)}}{GF_{n-1}^{(2)}} \leq \frac{c_3}{c_4} \leq \frac{GF_{n+1}^{(2)}}{GF_n^{(2)}} \quad (3.18)$$

Since two consecutive Gaussian Fibonacci numbers approach the golden ratio, we also obtain the following equations in (3.16) and (3.18) that connect the elements of the code matrix in (3.1):

$$c_1 \approx \tau c_2 \quad (3.19)$$

$$c_3 \approx \tau c_4 \quad (3.20)$$

where $\tau = \frac{1+\sqrt{5}}{2}$ is the “golden ratio”.

Similarly, for an odd integer $n = 2m + 1$, we can write the same approximate equalities (3.19) and (3.20) that connect the elements c_1, c_2, c_3 and c_4 of the code matrix in (3.1).

Thus, we have generated some important identities that connect the elements of the code matrix in (3.1) for the case $k = 2$.

If we consider the coding/decoding method for $k = 3$, we can obtain interesting identities as in $k = 2$. Since two consecutive Gaussian Tribonacci numbers approach Φ where $\Phi = 1.8393$. However, for general case of k we can find mathematical identities that connect the code matrix elements similar to the (3.19) and (3.20).

Example 3.1. Suppose $k = 2$. Thus, we have $\tau = \frac{1+\sqrt{5}}{2} \cong 1.618$. Assume that the following message-matrix is to be transmitted in (2.2):

For $s = 2$, we can get

$$M = \begin{bmatrix} C & O \\ D & E \end{bmatrix} = \begin{bmatrix} 4 & 16 \\ 5 & 6 \end{bmatrix}$$

- If $n = 2$ then

$$C = \begin{bmatrix} 24 + 20i & 20 + 4i \\ 16 + 11i & 11 + 5i \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix}$$

In this case, we have the following numbers rounded off to their first five digits:

$$\frac{c_1}{c_2} = 1.3462 + 0.7307i, \quad \frac{c_3}{c_4} = 1.5822 + 0.2808i$$

- For $n = 5$, we have

$$C = \begin{bmatrix} 112 + 68i & 68 + 44i \\ 70 + 43i & 43 + 27i \end{bmatrix}$$

In this case

$$\frac{c_1}{c_2} = 1.6171 - 0.0463i, \quad \frac{c_3}{c_4} = 1.6179 - 0.0159i$$

- For $n = 10$, we have

$$C = \begin{bmatrix} 1236 + 764i & 764 + 472i \\ 775 + 479i & 479 + 296i \end{bmatrix}$$

In this case

$$\frac{c_1}{c_2} = 1.618 + 0.0003i, \quad \frac{c_3}{c_4} = 1.618 + 0.0001i$$

- For $n = 15$, we have

$$C = \begin{bmatrix} 13708 + 8472i & 8472 + 5236i \\ 8595 + 5312i & 5312 + 3283i \end{bmatrix}$$

In this case

$$\frac{c_1}{c_2} = 1.618, \quad \frac{c_3}{c_4} = 1.618$$

These show that for $n = 15$ the relation (3.19) and (3.20) holds very well.

Therefore, we select the value of n large enough, as seen in Examples 2.3.1 we get a good approximation of τ . If we apply similar operation in Example 2.3.2, when n is chosen large enough we get a good approximation of Φ where $\Phi = 1.8393$ and Φ is the ratio between two Gaussian Tribonacci numbers.

4. Error detection and correction

The k -order Gaussian Fibonacci coding/decoding method considered above provides an interesting possibility to detect and correct “errors” in code message C . Error detection and correction for $k = 2$ depends on the states of the determinant of the matrix C and the connections with the code matrix elements at (3.19) and (3.20). These mathematical relations given by (3.19), (3.20) and the determinant of the code matrix C play a role in checking relations of the k -order Gaussian Fibonacci coding/decoding method. The error-correction algorithm for Fibonacci coding has been explained in [10] and [13]. This defined algorithm is valid and applicable to the k -order Gaussian Fibonacci numbers we have given in this study.

First, let’s calculate the determinant of our message matrix M and send it to the communication channel right after the code matrix elements. Here, the determinant of M will be the control element for the C code matrix received from the communication channel. After taking the determinant of the code matrix C and its control element M , we can calculate the determinant of the code matrix C and compare the determinant of M with the determinant of C with respect to the control relation. If the operations we have done after this check are exactly appropriate, we can conclude that the elements of the code matrix C are transmitted on the communication channel without errors. If the opposite happens, we can say that there are errors in the elements of the code matrix C or the determinant of the control element M .

Suppose the elements of the C code matrix errors. This matrix can have a one-fold error, double-fold error or k^2 – fold error. To explain how to correct these errors, we consider the 2×2 type matrix obtained for $k = 2$. These cases are considered.

Our first hypothesis is that we have the case of a “single” error in the code matrix C received from the communication channel. It is clearly seen that there are four types of single-fold errors in the C code matrix:

$$(a) \begin{bmatrix} x & c_2 \\ c_3 & c_4 \end{bmatrix} \quad (b) \begin{bmatrix} c_1 & y \\ c_3 & c_4 \end{bmatrix} \quad (c) \begin{bmatrix} c_1 & c_2 \\ z & c_4 \end{bmatrix} \quad (d) \begin{bmatrix} c_1 & c_2 \\ c_3 & t \end{bmatrix} \quad (4.1)$$

where x, y, z, t are possible “destroyed” elements.

Since it is $C = M \times E_n^{(2)}$, if we take the determinant of both sides, we get the following equations:

$$xc_4 - c_2c_3 = \det(M)(-1)^n(2 - i) \quad (\text{a possible “single error” is in the element } c_1) \quad (4.2)$$

$$c_1c_4 - yc_3 = \det(M)(-1)^n(2-i) \quad (\text{a possible "single error" is in the element } c_2) \quad (4.3)$$

$$c_1c_4 - c_2z = \det(M)(-1)^n(2-i) \quad (\text{a possible "single error" is in the element } c_3) \quad (4.4)$$

$$c_1t - c_2c_3 = \det(M)(-1)^n(2-i) \quad (\text{a possible "single error" is in the element } c_4) \quad (4.5)$$

or equivalently

$$x = \frac{\det(M)(-1)^n(2-i) + c_2c_3}{c_4} \quad (a) \quad (4.6)$$

$$y = \frac{-\det(M)(-1)^n(2-i) + c_1c_4}{c_3} \quad (b) \quad (4.7)$$

$$z = \frac{-\det(M)(-1)^n(2-i) + c_1c_4}{c_2} \quad (c) \quad (4.8)$$

$$t = \frac{\det(M)(-1)^n(2-i) + c_2c_3}{c_1} \quad (d) \quad (4.9)$$

The formulas (4.6)–(4.9) give four possible states of a “single error”. However, we need to choose the correct case from among x, y, z, t integer solutions. We have to choose such solutions, which satisfies to the additional “checking relations” (3.19) and (3.20). If calculations by formulas (4.6)–(4.9) do not give an integer result, we can conclude that our “single error” hypothesis is incorrect or that the determinant of control element M is “error”. In the second case, we can use approximate Eqs (3.19) and (3.20) to check the correctness of the C code matrix.

Similarly, we can easily check “double errors” of the C code matrix. Suppose that the code matrix C has elements x and y error as shown below:

$$\begin{bmatrix} x & y \\ c_3 & c_4 \end{bmatrix} \quad (4.10)$$

Then,

$$xc_4 - yc_3 = \det(M)(-1)^n(2-i) \quad (4.11)$$

However, according to the (3.19) there is the following relation between x and y

$$x \approx \tau y \quad (4.12)$$

Again, only integer solutions are acceptable. If no integer solution is obtained, then two-fold errors have not occurred.

Hence, we can show that with this type of approach, there is the possibility to correct all possible triple-fold errors in the C code matrix.

$$\begin{bmatrix} x & y \\ z & c_4 \end{bmatrix}.$$

Therefore, our method of error correction depends on the verification of different hypotheses

regarding errors in the code matrix using the determinant of the control relationship C matrix, (3.19) and (3.20), and the elements of the code matrix as integers. If all our solutions do not reach integer solutions, it means that the determinant of the control item M is wrong or there is a four-fold error in the C code matrix, and we should reject it. The code matrix C is defective and not correctable.

As a result, there are 15 error conditions in the C code matrix. According to the method given in [17], it means that the correctable probability of the method is equal since 14 cases between them can be corrected and

$$S_{cor} = \frac{14}{15} = 0.9333 = 93.33\%$$

If we generalize this equation as in [20], since only $k^2 - 1$ fold errors cannot be corrected with this method, the error correction capacity of the method is

$$\frac{2^{k^2} - 1}{2^{k^2}}$$

where k is the order of the message-matrix. So far sufficiently large values of k , the probability of decoding error is almost zero.

Consequently, for a sufficiently large value of k , in the case of $n = k$, the correct ability of this method is

$$\frac{2^{k^2} - 2}{2^{k^2} - 1}.$$

Therefore, for large value of k , the correct possibility of the method is

$$\frac{2^{k^2} - 2}{2^{k^2} - 1} \approx 1 = 100\%.$$

5. Conclusions

In this paper, we obtained the coding theory for k -order Gaussian Fibonacci polynomials given in [16] by taking $x=1$ for k -order Gaussian Fibonacci numbers and gave illustrative examples. This coding method differs from classical algebraic coding methods in this respect. We can show this difference with the following features:

1) Since the n, k and $s \geq 0$ values are arbitrarily chosen in the k -order Gaussian Fibonacci coding theory, it is difficult to estimate the information transmission between the two channels by a third channel, which increases the reliability of the information transmission. In addition, this method depends on matrix multiplication and can be performed quickly and easily by today's computers.

2) With k -order Gaussian Fibonacci coding theory, we can encrypt and send messages of the desired length by enlarging the k value sufficiently.

3) The main practical feature of this method is that matrix elements are error detection and correction objects.

4) The Gaussian Fibonacci coding method, which is the simplest coding method obtained for the $k=2$ value, has been redefined by handling the 2×2 type matrices with "single-fold, double-fold and triple-fold errors".

5) In the simplest case, for $k=2$, the correct capability of the method is essentially equal to

93.33%, exceeding all well-known correction codes.

6) This error correction method given for $k = 2$ is generalized and the correct ability of the errors of this method increases as k increases, and the correct ability approaches 100% for a sufficiently large value of k .

7) This article is just a brief outline of a new coding theory based on the k -order Gaussian Fibonacci matrices of the articles in [17,18,20,21] and [22].

Conflict of interest

The authors declare there is no conflict of interest.

References

1. T. Koshy, *Fibonacci and Lucas Numbers with Applications*, A Wiley-Interscience Publication, 2001. <https://doi.org/10.1002/9781118033067>
2. S. Vajda, *Fibonacci and Lucas Numbers and the Golden Section Theory and Applications*, Ellis Harwood Limited, 1989.
3. A. P. Stakhov, B. Rozin, The golden shofar, *Chaos Solitons Fractals*, **26** (2005), 677–684. <https://doi.org/10.1016/j.chaos.2005.01.057>
4. A. P. Stakhov, The golden section in the measurement theory, *Comput. Math. Appl.*, **17** (1989), 613–638. [http://dx.doi.org/10.1016/0898-1221\(89\)90252-6](http://dx.doi.org/10.1016/0898-1221(89)90252-6)
5. S. K. Sen, R. P. Agarwal, Golden ratio in science, as random sequence source, its computation, and beyond, *Comput. Math. Appl.*, **56** (2008), 469–498. <https://doi.org/10.1016/j.camwa.2007.06.030>
6. M. Ascı, D. Tascı, On Fibonacci, Lucas and special orthogonal polynomials, *J. Comput. Appl. Math.*, **2007** (2007). <https://doi.org/10.1016/j.cam.2007.01.026>
7. A. Stakhov, B. Rozin, Theory of Binet formulas for Fibonacci and Lucas p -numbers, *Chaos Solitons Fractals*, **27** (2006), 1162–1177. <https://doi.org/10.1016/j.chaos.2005.04.106>
8. M. El-Naschie, Modular groups in Cantorian $E(\infty)$ high-energy physics, *Chaos Solitons Fractals*, **16** (2003), 353–366. [https://doi.org/10.1016/S0960-0779\(02\)00440-X](https://doi.org/10.1016/S0960-0779(02)00440-X)
9. M. Esmaeili, On the weakly superincreasing distributions and the Fibonacci-Hessenberg matrices, *Ars Comb.*, **84** (2007), 217–224.
10. M. Esmaeili, More on the Fibonacci sequence and Hessenberg matrices, *Integers*, **6** (2006), A32.
11. M. Esmaeili, Polynomial Fibonacci-Hessenberg matrices, *Chaos Solitons Fractals*, **41** (2009), 2820–2827. <https://doi.org/10.1016/j.chaos.2008.10.012>
12. A. F. Horadam, A generalized Fibonacci sequence, *Am. Math. Mon.*, **68** (1961), 455–459. <http://www.jstor.org/stable/2311099>
13. A. F. Horadam, Complex Fibonacci numbers and Fibonacci quaternions, *Am. Math. Mon.*, **70** (1963), 289–291. <https://doi.org/10.2307/2313129>
14. J. H. Jordan, Gaussian Fibonacci and Lucas numbers, *Fibonacci Quart.*, **3** (1965), 31–318.
15. M. Ascı, E. Gurel, Some properties of k -order Gaussian Fibonacci and Lucas numbers, *Ars Combin.*, **135** (2017), 345–356.
16. M. Ascı, S. Aydinyuz, K -order Gaussian Fibonacci polynomials and applications to the coding/decoding theory, *J. Discrete Math. Sci. Cryptography*, **25** (2020), 1399–1416. <https://doi.org/10.1080/09720529.2020.1816917>

17. A. P. Stakhov, Fibonacci matrices, a generalization of the Cassini formula and a new coding theory, *Chaos Solitons Fractals*, **30** (2006), 56–66. <https://doi.org/10.1016/j.chaos.2005.12.054>
18. M. Basu, B. Prasad, The generalized relations among the code elements for Fibonacci coding theory, *Chaos Solitons Fractals*, **41** (2009), 2517–2525. <https://doi.org/10.1016/j.chaos.2008.09.030>
19. M. Basu, M. Das, Tribonacci matrices and a new coding theory, *Discrete Math. Algorithms Appl.*, **6** (2014), 1450008. <https://doi.org/10.1142/S1793830914500086>
20. M. Basu, M. Das, Coding theory on Fibonacci n-step numbers, *Discrete Math. Algorithms Appl.*, **6** (2014), 1450017. <https://doi.org/10.1142/S1793830914500177>
21. M. Esmaeili, M. Esmaeili, A Fibonacci-polynomial based coding method with error detection and correction, *Comput. Math. Appl.*, **60** (2010), 2738–2752. <https://doi.org/10.1016/j.camwa.2010.08.091>
22. A. P. Stakhov, V. Massingue, A. Sluchenkov, *Introduction into Fibonacci Coding and Cryptography*, Osnova, Kharkov, 1999.
23. M. Asci, G. Y. Lee, On the generalized gaussian fibonacci numbers, *Ars Combin.*, **132** (2017), 147–157.



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)