Contributions to Discrete Mathematics

Volume 15, Number 1, Pages 52–66
ISSN 1715-0868

PELL CODING AND PELL DECODING METHODS WITH SOME APPLICATIONS

NİHAL TAŞ, SÜMEYRA UÇAR, AND NİHAL YILMAZ ÖZGÜR

ABSTRACT. We develop a new coding and decoding method using the generalized Pell (p, i)-numbers for $p = 1$. The relations among the code matrix elements, error detection, and correction have been established for coding theory when $p = i = 1$. We give two new blocking algorithms using Pell numbers and generalized Pell (p, i)-numbers for $p = 1$.

1. INTRODUCTION

Recently, in [3], the generalized Pell (p, i)-numbers have been defined by the following recurrence relations

(1.1)
$$P_p^{(i)}(n) = 2P_p^{(i)}(n-1) + P_p^{(i)}(n-p-1);\ n > p+1, 0 \leq i \leq p,\ p = 1, 2, 3, \ldots$$

with the initial terms

$$P_p^{(i)}(1) = \cdots = P_p^{(i)}(i) = 0,\ P_p^{(i)}(i+1) = \cdots = P_p^{(i)}(p+1) = 1.$$

For $i = p = 1$, the generalized Pell $(1, 1)$-number corresponds to the $(n+1)$th classical Pell number defined as

$$P_{n+1} = 2P_n + P_{n-1}, n \in \mathbb{Z}^+$$

with the initial terms

$$P_0 = 0, P_1 = 1.$$

For the basic properties of these numbers, one can see [3] and [4].

It is known that $\gamma = 1 + \sqrt{2}$ and $\delta = 1 - \sqrt{2}$ are the roots of the characteristic equation of the Pell recurrence relation $t^2 - 2t - 1 = 0$. Using these roots, the Binet formula for the Pell number is

$$P_n = \frac{\gamma^n - \delta^n}{\gamma - \delta} \text{ for all } n \geq 0.$$

In addition, $\lim_{n\to\infty} P_{n+1}/P_n = \gamma$.

Received by the editors April 25, 2018, and in revised form April 8, 2019 .

2010 *Mathematics Subject Classification.* 68P30, 14G50, 11T71, 11B39.

Key words and phrases. Coding/decoding method, Pell number, generalized Pell (p, i)-number, blocking algorithm.

Corresponding author: N. Taş.

The Pell P-matrix of order 2 has the following form:

$$P = \begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix}.$$

The nth power of the P-matrix and its determinant are given by

$$P^n = \begin{bmatrix} P_{n+1} & P_n \\ P_n & P_{n-1} \end{bmatrix} \tag{1.2}$$

and

$$\det(P^n) = P_{n+1}P_{n-1} - P_n^2 = (-1)^n.$$

In [3], the following matrix A was introduced:

$$A = \begin{bmatrix} 2 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \ddots & \vdots & 0 \\ \vdots & \cdots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}_{(p+1)\times(p+1)}. \tag{1.3}$$

Furthermore,

(1.4)

$$A^n = G_n = \begin{bmatrix} P_p^{(p)}(n+p+1) & P_p^{(p)}(n+1) & P_p^{(p)}(n+2) & \cdots & P_p^{(p)}(n+p) \\ P_p^{(p)}(n+p) & P_p^{(p)}(n) & P_p^{(p)}(n+1) & \cdots & P_p^{(p)}(n+p-1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ P_p^{(p)}(n+2) & P_p^{(p)}(n-p+2) & P_p^{(p)}(n-p+3) & \cdots & P_p^{(p)}(n+1) \\ P_p^{(p)}(n+1) & P_p^{(p)}(n-p+1) & P_p^{(p)}(n-p+2) & \cdots & P_p^{(p)}(n) \end{bmatrix}_{(p+1)\times(p+1)}.$$

Using the matrices given in the equations (1.3) and (1.4), we get

$$\det(A^n) = \det(G_n) = (-1)^{n(p+2)}, \tag{1.5}$$

for $n > 0$.

Recently, Fibonacci coding theory has been introduced and studied in many aspects (see [1], [2], [5], [7], [8], [9], and [13] for more details). For example, in [8], A. P. Stakhov presented a new coding theory using the generalization of the Cassini formula for Fibonacci p-numbers and Q_p-matrices. Later, B. Prasad developed a new coding and decoding method obtained by Lucas p matrix [6]. More recently, a new algorithm called the "Fibonacci Blocking Algorithm" using Fibonacci Q-matrices was presented in [10]. In [11], two new algorithms were presented called the Lucas blocking algorithm, which uses R-matrices and Lucas numbers, and the minesweeper model, which uses Fibonacci Q^n-matrices and R-matrices.

Motivated by the above studies, the main purpose of this paper is to develop a new coding and decoding method using the generalized Pell (p, i)-numbers for $p = 1$. The relations among the code matrix elements, error detection, and correction have been established for this coding theory with $p = i = 1$. As an application, we give two new blocking algorithms using Pell and generalized Pell (p,i)-numbers for $p=1$. For any $p \geq 2$, it is possible to use blocks of size $(p+1) \times (p+1)$.

2. Pell Coding and Decoding Method

In this section, we present a new coding and decoding method using the generalized Pell (p,i)-numbers. In our method, the nonsingular square matrix M with order $(p+1)$, where $p = 1,2,3,\ldots$ corresponds to our message matrix. We consider the matrix G_n of order $(p+1)$ as the coding matrix and its inverse matrix $(G_n)^{-1}$ as the decoding matrix. We introduce Pell coding and Pell decoding with transformations

$$M \times G_n = E$$

and

$$E \times (G_n)^{-1} = M,$$

respectively, where E is a code matrix.

Taking $p=i=1$, we give an example of the Pell coding and decoding method. Let M be a message matrix of the following form:

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

where m_1, m_2, m_3, m_4 are positive integers.

Let $n=3$. We construct the coding matrix G_n:

$$G_3 = \begin{bmatrix} P_1^{(1)}(n+2) & P_1^{(1)}(n+1) \\ P_1^{(1)}(n+1) & P_1^{(1)}(n) \end{bmatrix} = \begin{bmatrix} P_1^{(1)}(5) & P_1^{(1)}(4) \\ P_1^{(1)}(4) & P_1^{(1)}(3) \end{bmatrix} = \begin{bmatrix} 12 & 5 \\ 5 & 2 \end{bmatrix}.$$

Then we find inverse of G_3:

$$(G_3)^{-1} = \frac{1}{12 \cdot 2 - 5 \cdot 5} \begin{bmatrix} 2 & -5 \\ -5 & 12 \end{bmatrix} = \begin{bmatrix} -2 & 5 \\ 5 & -12 \end{bmatrix}.$$

Now, we calculate the code matrix E.

$$E = M \times G_3 = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} 12 & 5 \\ 5 & 2 \end{bmatrix} = \begin{bmatrix} 12m_1 + 5m_2 & 5m_1 + 2m_2 \\ 12m_3 + 5m_4 & 5m_3 + 2m_4 \end{bmatrix},$$

where $e_1 = 12m_1 + 5m_2$, $e_2 = 5m_1 + 2m_2$, $e_3 = 12m_3 + 5m_4$ and $e_4 = 5m_3 + 2m_4$.

Finally, the code matrix E is sent to a channel. The message matrix M can be obtained by decoding as the following way

$$\begin{aligned} M &= E \times (G_3)^{-1} = \begin{bmatrix} e_1 & e_2 \\ e_3 & e_4 \end{bmatrix} \begin{bmatrix} -2 & 5 \\ 5 & -12 \end{bmatrix} \\ &= \begin{bmatrix} -2e_1 + 5e_2 & 5e_1 - 12e_2 \\ -2e_3 + 5e_4 & 5e_3 - 12e_4 \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix}. \end{aligned}$$

Notice that the relation between the code matrix E and the message matrix M is

$$\det(E) = \det(M \times G_n) = \det(M) \times \det(G_n) = \det(M) \times (-1)^{n(p+2)} \tag{2.1}$$

to be used in the checking procedure and so, we have the following relation for the above example:

$$\det(E) = \det(M \times G_3) = \det(M) \times \det(G_3) = \det(M) \times (-1).$$

3. The Relationships between the Code Matrix Elements for $p = i = 1$

We write E and M as follows:

$$E = M \times G_n = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} P_1^{(1)}(n+2) & P_1^{(1)}(n+1) \\ P_1^{(1)}(n+1) & P_1^{(1)}(n) \end{bmatrix} = \begin{bmatrix} e_1 & e_2 \\ e_3 & e_4 \end{bmatrix}$$

and

$$M = E \times (G_n)^{-1} = \begin{bmatrix} e_1 & e_2 \\ e_3 & e_4 \end{bmatrix} \frac{1}{(-1)^n} \begin{bmatrix} P_1^{(1)}(n) & -P_1^{(1)}(n+1) \\ -P_1^{(1)}(n+1) & P_1^{(1)}(n+2) \end{bmatrix}$$
$$= \frac{1}{(-1)^n} \begin{bmatrix} e_1 P_1^{(1)}(n) - e_2 P_1^{(1)}(n+1) & -e_1 P_1^{(1)}(n+1) + e_2 P_1^{(1)}(n+2) \\ e_3 P_1^{(1)}(n) - e_4 P_1^{(1)}(n+1) & -e_3 P_1^{(1)}(n+1) + e_4 P_1^{(1)}(n+2) \end{bmatrix},$$

for $n = 2k+1$. Because m_1, m_2, m_3, m_4 are positive integers, we get

$$m_1 = -e_1 P_1^{(1)}(n) + e_2 P_1^{(1)}(n+1) > 0, \tag{3.1}$$

$$m_2 = e_1 P_1^{(1)}(n+1) - e_2 P_1^{(1)}(n+2) > 0, \tag{3.2}$$

$$m_3 = -e_3 P_1^{(1)}(n) + e_4 P_1^{(1)}(n+1) > 0, \tag{3.3}$$

$$m_4 = e_3 P_1^{(1)}(n+1) - e_4 P_1^{(1)}(n+2) > 0. \tag{3.4}$$

Using (3.1) and (3.2), we find

$$\frac{P_1^{(1)}(n+2)}{P_1^{(1)}(n+1)} < \frac{e_1}{e_2} < \frac{P_1^{(1)}(n+1)}{P_1^{(1)}(n)}. \tag{3.5}$$

Using (3.3) and (3.4), we get

$$\frac{P_1^{(1)}(n+2)}{P_1^{(1)}(n+1)} < \frac{e_3}{e_4} < \frac{P_1^{(1)}(n+1)}{P_1^{(1)}(n)}. \tag{3.6}$$

From the inequalities (3.5) and (3.6), we obtain

$$\frac{e_1}{e_2} \approx \gamma, \ \frac{e_3}{e_4} \approx \gamma. \tag{3.7}$$

For $n = 2k$, we obtain similar relations given in (3.7).

4. Error detection/correction for the Pell Coding and decoding method

The message may not reach the recipient clearly while transmitting the message text. Therefore some errors may occur in the code matrix E. We try to determine and correct these errors using the properties of the determinant in this process. Let $p = i = 1$ and the message matrix M be given by

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

We know that $\det(M) = m_1 m_4 - m_2 m_3$, the code matrix $E = M \times G_n$, and $\det(E) = \det(M) \times (-1)^n$. From the relationship between determinants of E and M, if n is an odd number, we have

$$\det(E) = -\det(M)$$

and if n is an even number, we have

$$\det(E) = \det(M).$$

The new method of error detection is an application of the matrix G_n. The basic idea of this method depends on calculating the determinants of M and E. After comparing the determinants obtained from the channel, the receiver can decide whether the code message E is damaged or not.

Actually, we cannot determine which element of the code message is damaged. In order to find the damaged element, we suppose different cases such as the case of a single error, two errors, etc.

Case 1: A single error in the code matrix E.

We can easily obtain that there are four places where single error appear in E :

$$\begin{bmatrix} t_1 & e_2 \\ e_3 & e_4 \end{bmatrix}, \begin{bmatrix} e_1 & t_2 \\ e_3 & e_4 \end{bmatrix}, \begin{bmatrix} e_1 & e_2 \\ t_3 & e_4 \end{bmatrix}, \begin{bmatrix} e_1 & e_2 \\ e_3 & t_4 \end{bmatrix},$$

where t_i $(i \in \{1, 2, 3, 4\})$ is the damaged element. To check the above four different cases, we can use the following relations:

$$t_1 e_4 - e_2 e_3 = (-1)^n \det(M), \tag{4.1}$$

$$e_1 e_4 - t_2 e_3 = (-1)^n \det(M), \tag{4.2}$$

$$e_1 e_4 - t_3 e_2 = (-1)^n \det(M), \tag{4.3}$$

$$e_1 t_4 - e_2 e_3 = (-1)^n \det(M), \tag{4.4}$$

where the single possible error is the element t_1 (resp. t_2, t_3, and t_4) given in relation (4.1) (resp. (4.2),(4.3), and (4.4)).

Using the above relations, we have

$$t_1 = \frac{(-1)^n \det(M) + e_2 e_3}{e_4}, \tag{4.5}$$

$$t_2 = \frac{(-1)^n \det(M) + e_1 e_4}{e_3}, \tag{4.6}$$

$$t_3 = \frac{(-1)^n \det(M) + e_1 e_4}{e_2}, \tag{4.7}$$

$$t_4 = \frac{(-1)^n \det(M) + e_2 e_3}{e_1}. \tag{4.8}$$

Since the elements of message matrix M are positive integers, we should find integer solutions of equations (4.5)–(4.8). If there are no integer solutions of these equations, we find that our cases related to a single error are incorrect or an error has occurred in the checking element, $\det(M)$. If $\det(M)$ is incorrect, we use the relations given in (3.7) to check the correctness of the code matrix E.

Case 2: Two errors in the code matrix E.

Let us consider the following case with a double error in E:

$$\begin{bmatrix} t_1 & t_2 \\ e_3 & e_4 \end{bmatrix}, \tag{4.9}$$

where t_1, t_2 are the damaged elements of E. Using the relation

$$\det(E) = (-1)^n \det(M),$$

we can write following equation for the matrix given in (4.9)

$$t_1 e_4 - t_2 e_3 = (-1)^n \det(M). \tag{4.10}$$

Also, we know the following relation between t_1 and t_2

$$t_1 \approx \gamma t_2. \tag{4.11}$$

It is clear that (4.10) is a Diophantine equation. Because there are many solutions to Diophantine equations, we should choose solutions t_1, t_2 satisfying relation (4.11).

Case 3: Three errors in the code matrix E.

We correct the triple errors in the code matrix E of the form

$$\begin{bmatrix} t_1 & t_2 \\ t_3 & e_4 \end{bmatrix},$$

where t_1, t_2 and t_3 are damaged elements of E. From (4.11), t_2 can be obtained. The remaining errors are reduced to the Case 2 solution.

Consequently, our method depends on checking possible damaged elements of E using the checking element $\det(M)$ and the checking relation $t_1 \approx \gamma t_2$. Our correctness method allows us to correct for 14 out of 15 possible cases because the method is inadequate in the case of four errors. We can say that the correction ability of our method is $14/15 = 0.9333$, or 93.33%.

Example 4.1. *Let us take $p = i = 1$ and consider the message matrix M as follows:*

$$M = \begin{bmatrix} 13 & 55 \\ 2 & 100 \end{bmatrix}$$

with $\det(M) = 1190$. *If we choose $n = 4$, we get*

$$G_4 = \begin{bmatrix} P_1^{(1)}(6) & -P_1^{(1)}(5) \\ -P_1^{(1)}(5) & P_1^{(1)}(4) \end{bmatrix} = \begin{bmatrix} 29 & 12 \\ 12 & 5 \end{bmatrix}$$

and

$$E = M \times G_4 = \begin{bmatrix} 13 & 55 \\ 2 & 100 \end{bmatrix} \begin{bmatrix} 29 & 12 \\ 12 & 5 \end{bmatrix} = \begin{bmatrix} 1037 & 431 \\ 1258 & 524 \end{bmatrix}.$$

Assume that the received elements are $1030, 431, 1258, 524$, *and* 1190. *Thus,*

$$E^* = \begin{bmatrix} 1030 & 431 \\ 1258 & 524 \end{bmatrix}.$$

Using (2.1), we have

$$\det(E^*) = -2,478 \neq (-1)^{12} \det(M) = 1190$$

and so we deduce that there are some errors in the received elements. Using equations (4.5)–(4.8), we obtain the following:

$$\begin{aligned} t_1 &= \frac{1190 + 542198}{524} = 1037, \\ t_2 &= \frac{1190 + 539720}{1258} = 429.9761526232, \\ t_3 &= \frac{1190 + 539720}{431} = 1255.0116009281, \\ t_4 &= \frac{1190 + 542198}{1030} = 527.5611650485. \end{aligned}$$

Notice that only t_1 satisfies the necessary conditions, using this value we obtain the corrected code matrix:

$$E = \begin{bmatrix} 1037 & 431 \\ 1258 & 524 \end{bmatrix}.$$

5. Blocking Methods As an Application of Pell Numbers and Generalized Pell (p, i)-Numbers

In this section we introduce new coding/decoding algorithms using Pell and generalized Pell (p, i)-numbers. We put our message in a matrix of even size adding a zero between two words and at the end of the message until the size of the message matrix is even. Dividing the $2m \times 2m$ message matrix $\mathcal{M}$ into 2×2 block matrices,B_i for $(1 \leq i \leq m^2)$, we can construct a new coding method.

Now we describe the notation used in our coding method. Assume that matrices B_i and E_i are of the following form:

$$B_i = \begin{bmatrix} b_1^i & b_2^i \\ b_3^i & b_4^i \end{bmatrix} \text{ and } E_i = \begin{bmatrix} e_1^i & e_2^i \\ e_3^i & e_4^i \end{bmatrix}.$$

We use the matrix P^n given in (1.2) and rewrite the elements of this matrix to

$$P^n = \begin{bmatrix} p_1 & p_2 \\ p_3 & p_4 \end{bmatrix}.$$

The number of the block matrices B_i is denoted by b. Using b, we choose a number n as follows:

$$n = \begin{cases} 3, & b \leq 3 \\ \left\lfloor \frac{b}{2} \right\rfloor, & b > 3 \end{cases}.$$

After choosing n, we write the following character table modulo 29 since we use 29 characters (this table can be extended according to the number of characters used in the message matrix). We represent each characters of the table with a number between n and $n+28$ beginning with n for the last character and continuing in reverse order.

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

Now we will describe the following new coding and decoding algorithms.

Algorithm 5.1 Pell Blocking Algorithm — Coding algorithm

Step 1. Divide the matrix $\mathcal{M}$ into blocks B_i $\left(1 \leq i \leq m^2\right)$.
Step 2. Choose n.
Step 3. Determine b_j^i $(1 \leq j \leq 4)$.
Step 4. Compute $\det(B_i) \to d_i$.
Step 5. Construct $K = \left[d_i, b_k^i\right]_{k \in \{1,3,4\}}$.

Algorithm 5.2 Pell Blocking Algorithm — Decoding algorithm

Step 1. Compute P^n.
Step 2. Determine p_j $(1 \leq j \leq 4)$.
Step 3. Compute $p_1 b_3^i + p_3 b_4^i \to e_3^i$ $\left(1 \leq i \leq m^2\right)$.
Step 4. Compute $p_2 b_3^i + p_4 b_4^i \to e_4^i$.
Step 5. Solve $(-1)^n \times d_i = e_4^i(p_1 b_1^i + p_3 x_i) - e_3^i(p_2 b_1^i + p_4 x_i)$.
Step 6. Substitute for $x_i = b_2^i$.

Step 7. Construct B_i.
Step 8. Construct $\mathcal{M}$.

In the following example we give an application of the above algorithm for $b > 3$.

Example 5.1. *Let us consider the message matrix for the following text:*

"MATH IS SWEET:)"

Using the text, we get the following message matrix:

$$\mathcal{M} = \begin{bmatrix} M & A & T & H \\ 0 & I & S & 0 \\ S & W & E & E \\ T & : &) & 0 \end{bmatrix}_{4\times 4}.$$

We begin by applying the coding algorithm to $\mathcal{M}$*:*

Step 1. *We divide the message matrix* $\mathcal{M}$ *into* 2×2 *matrices,* B_i $(1 \le i \le 4)$ *from left to right:*

$$B_1 = \begin{bmatrix} M & A \\ 0 & I \end{bmatrix},\ B_2 = \begin{bmatrix} T & H \\ S & 0 \end{bmatrix},\ B_3 = \begin{bmatrix} S & W \\ T & : \end{bmatrix},\ \text{and } B_4 = \begin{bmatrix} E & E \\ ) & 0 \end{bmatrix}.$$

Step 2. *Since* $b = 4 > 3$, *we calculate* $n = \lfloor b/2 \rfloor = 2$. *For* $n = 2$, *we use the following "letter table" for the message matrix* $\mathcal{M}$*:*

M	A	T	H	0	I	S	0
18	1	11	23	4	22	12	4
S	W	E	E	T	:	)	0
12	8	26	26	11	3	2	4

Step 3. *We have the elements of the blocks* B_i $(1 \le i \le 4)$ *as follows:*

$b_1^1 = 18$	$b_2^1 = 1$	$b_3^1 = 4$	$b_4^1 = 22$
$b_1^2 = 11$	$b_2^2 = 23$	$b_3^2 = 12$	$b_4^2 = 4$
$b_1^3 = 12$	$b_2^3 = 8$	$b_3^3 = 11$	$b_4^3 = 3$
$b_1^4 = 26$	$b_2^4 = 26$	$b_3^4 = 2$	$b_4^4 = 4$

Step 4. *Now we calculate the determinants* d_i *of the blocks* B_i :

$$d_1 = \det(B_1) = 392$$
$$d_2 = \det(B_2) = -232$$
$$d_3 = \det(B_3) = -52$$
$$d_4 = \det(B_4) = 52$$

Step 5. *Using Step 3 and Step 4 we obtain the following matrix* K*:*

$$K = \begin{bmatrix} 392 & 18 & 4 & 22 \\ -232 & 11 & 12 & 4 \\ -48 & 12 & 11 & 3 \\ 52 & 26 & 2 & 4 \end{bmatrix}.$$

Next, we apply the decoding algorithm to K:

Step 1. *By (1.2) we know that*

$$P^2 = \begin{bmatrix} P_3 & P_2 \\ P_2 & P_1 \end{bmatrix} = \begin{bmatrix} 5 & 2 \\ 2 & 1 \end{bmatrix}.$$

Step 2. *The elements of P^2 are denoted by*

$$p_1 = 5,\ p_2 = 2,\ p_3 = 2,\ \textit{and}\ p_4 = 1.$$

Step 3. *We compute the elements e_3^i to construct the matrix E_i:*

$$e_3^1 = 64,\ e_3^2 = 68,\ e_3^3 = 61,\ \textit{and}\ e_3^4 = 18.$$

Step 4. *We compute the elements e_4^i to construct the matrix E_i:*

$$e_4^1 = 30,\ e_4^2 = 28,\ e_4^3 = 25,\ \textit{and}\ e_4^4 = 8.$$

Step 5. *We calculate the elements x_i:*

$$\begin{aligned} (-1)^2(392) &= 30(90 + 2x_1) - 64(36 + x_1) \\ &\Rightarrow x_1 = 1, \\ (-1)^2(-232) &= 28(55 + 2x_2) - 68(22 + x_2) \\ &\Rightarrow x_2 = 23, \\ (-1)^2(-52) &= 25(60 + 2x_3) - 61(24 + x_3) \\ &\Rightarrow x_3 = 8, \\ (-1)^2(52) &= 8(130 + 2x_4) - 18(52 + x_4) \\ &\Rightarrow x_4 = 26. \end{aligned}$$

Step 6. *We rename the x_i as follows*:

$$x_1 = b_2^1 = 1,\ x_2 = b_2^2 = 23,\ x_3 = b_2^3 = 8,\ \textit{and}\ x_4 = b_2^4 = 26.$$

Step 7. *We construct the block matrices B_i :*

$$B_1 = \begin{bmatrix} 18 & 1 \\ 4 & 22 \end{bmatrix},\ B_2 = \begin{bmatrix} 11 & 13 \\ 12 & 4 \end{bmatrix},\ B_3 = \begin{bmatrix} 12 & 8 \\ 11 & 3 \end{bmatrix},\ \textit{and}\ B_4 = \begin{bmatrix} 26 & 26 \\ 2 & 4 \end{bmatrix}.$$

Step 8. *We recover the message matrix $\mathcal{M}$:*

$$\mathcal{M} = \begin{bmatrix} 18 & 1 & 11 & 23 \\ 4 & 22 & 12 & 4 \\ 12 & 8 & 26 & 26 \\ 11 & 3 & 2 & 4 \end{bmatrix} = \begin{bmatrix} M & A & T & H \\ 0 & I & S & 0 \\ S & W & E & E \\ T & : &) & 0 \end{bmatrix}.$$

Now we give an application of generalized Pell (p, i)-numbers for $p = 1$ using the blocking method. Assume that matrices B_k and E_k have the following forms:

$$B_k = \begin{bmatrix} b_1^k & b_2^k \\ b_3^k & b_4^k \end{bmatrix} \text{ and } E_k = \begin{bmatrix} e_1^k & e_2^k \\ e_3^k & e_4^k \end{bmatrix}.$$

We use the matrix G_n defined in (1.4) and relabel the elements of this matrix by

$$G_n = \begin{bmatrix} g_1 & g_2 \\ g_3 & g_4 \end{bmatrix}.$$

The number of the block matrices B_k are denoted by b. Fix the number n to

$$n = p + 2 = 3.$$

The generalized coding algorithm follows the same steps as the Pell Blocking coding algorithm. For the decoding algorithm, we have:

Algorithm 5.3 Generalized Pell Blocking Algorithm — Decoding algorithm

Step 1. Compute G_n.
Step 2. Determine g_j $(1 \leq j \leq 4)$.
Step 3. Compute $g_1 b_3^k + g_3 b_4^k \to e_3^k$ $\left(1 \leq k \leq m^2\right)$.
Step 4. Compute $g_2 b_3^k + g_4 b_4^k \to e_4^k$.
Step 5. Solve $(-1)^n \times d_k = e_4^k (g_1 b_1^k + g_3 x_k) - e_3^k (g_2 b_1^k + g_4 x_k)$.
Step 6. Substitute for $x_k = b_2^k$.
Step 7. Construct B_k.
Step 8. Construct $\mathcal{M}$.

We choose $p = i = 1$ in the following example.

Example 5.2. *Let us consider the message matrix for the following message text*:

"HAPPY BIRTHDAY TO YOU:)".

Using the text, we get the following message matrix

$$\mathcal{M} = \begin{bmatrix} H & A & P & P & Y & 0 \\ B & I & R & T & H & D \\ A & Y & 0 & T & O & 0 \\ Y & O & U & : &) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{6\times 6}.$$

We begin by applying the coding algorithm to $\mathcal{M}$.

Step 1. *We can divide the message text* $\mathcal{M}$ *of size* 6×6 *into the matrices, named* B_k $(1 \leq k \leq 9)$, *from left to right, each of size is* 2×2 :

$$B_1 = \begin{bmatrix} H & A \\ B & I \end{bmatrix}, B_2 = \begin{bmatrix} P & P \\ R & T \end{bmatrix}, B_3 = \begin{bmatrix} Y & 0 \\ H & D \end{bmatrix},$$
$$B_4 = \begin{bmatrix} A & Y \\ Y & O \end{bmatrix}, B_5 = \begin{bmatrix} 0 & T \\ U & : \end{bmatrix}, B_6 = \begin{bmatrix} O & 0 \\ ) & 0 \end{bmatrix},$$
$$B_7 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_8 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_9 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Step 2. *Since $p = 1$, we calculate $n = 3$. For $n = 3$, we use the following "letter table" for the message matrix $\mathcal{M}$:*

H	A	P	P	Y	0	B	I	R	T	H	D
24	2	16	16	7	5	1	23	14	12	24	28
A	Y	0	T	O	0	Y	O	U	:	)	0
2	7	5	12	17	5	7	17	11	4	3	5
0	0	0	0	0	0	0	0	0	0	0	0
5	5	5	5	5	5	5	5	5	5	5	5

Step 3. *We have the elements of the blocks B_k $(1 \leq k \leq 9)$ as follows:*

$b_1^1 = 24$	$b_2^1 = 2$	$b_3^1 = 1$	$b_4^1 = 23$
$b_1^2 = 16$	$b_2^2 = 16$	$b_3^2 = 14$	$b_4^2 = 12$
$b_1^3 = 7$	$b_2^3 = 5$	$b_3^3 = 24$	$b_4^3 = 28$
$b_1^4 = 2$	$b_2^4 = 7$	$b_3^4 = 7$	$b_4^4 = 17$
$b_1^5 = 5$	$b_2^5 = 12$	$b_3^5 = 11$	$b_4^5 = 4$
$b_1^6 = 17$	$b_2^6 = 5$	$b_3^6 = 3$	$b_4^6 = 5$
$b_1^7 = 5$	$b_2^7 = 5$	$b_3^7 = 5$	$b_4^7 = 5$
$b_1^8 = 5$	$b_2^8 = 5$	$b_3^8 = 5$	$b_4^8 = 5$
$b_1^9 = 5$	$b_2^9 = 5$	$b_3^9 = 5$	$b_4^9 = 5$

Step 4. *Now we calculate the determinants d_k of the blocks B_k :*

$$d_1 = \det(B_1) = 550, \; d_2 = \det(B_2) = -32, \; d_3 = \det(B_3) = 76,$$
$$d_4 = \det(B_4) = -15, \; d_5 = \det(B_5) = -112, \; d_6 = \det(B_6) = 70,$$
$$d_7 = \det(B_7) = 0, \quad d_8 = \det(B_8) = 0, \quad d_9 = \det(B_9) = 0.$$

Step 5. *Using Step 3 and Step 4 we obtain the following matrix K :*

$$K = \begin{bmatrix} 550 & 24 & 1 & 23 \\ -32 & 16 & 14 & 12 \\ 76 & 7 & 24 & 28 \\ -15 & 2 & 7 & 17 \\ -112 & 5 & 11 & 4 \\ 70 & 17 & 3 & 5 \\ 0 & 5 & 5 & 5 \\ 0 & 5 & 5 & 5 \\ 0 & 5 & 5 & 5 \end{bmatrix}.$$

Next, we apply the decoding algorithm to K:

Step 1. *Using the equation (1.4) we have*

$$G_n = \begin{bmatrix} P_p^{(p)}(n+p+1) & P_p^{(p)}(n+1) \\ P_p^{(p)}(n+p) & P_p^{(p)}(n) \end{bmatrix}$$

and

$$G_3 = \begin{bmatrix} P_1^{(1)}(5) & P_1^{(1)}(4) \\ P_1^{(1)}(4) & P_1^{(1)}(3) \end{bmatrix} = \begin{bmatrix} 12 & 5 \\ 5 & 2 \end{bmatrix}.$$

Step 2. *The elements of G_3 are denoted by*

$$g_1 = 12,\ g_2 = 5,\ g_3 = 5,\ \text{and}\ g_4 = 2.$$

Step 3. *We compute the elements e_3^k to construct the matrix E_k:*

$$\begin{aligned} e_3^1 &= 127, e_3^2 = 228, e_3^3 = 428, \\ e_3^4 &= 169, e_3^5 = 152, e_3^6 = 61, \\ e_3^7 &= 85,\ \ e_3^8 = 85,\ \ e_3^9 = 85. \end{aligned}$$

Step 4 . *We compute the elements e_4^k to construct the matrix E_k:*

$$\begin{aligned} e_4^1 &= 51,\ e_4^2 = 94,\ e_4^3 = 176, \\ e_4^4 &= 69,\ e_4^5 = 63,\ e_4^6 = 25, \\ e_4^7 &= 35,\ e_4^8 = 35,\ e_4^9 = 35. \end{aligned}$$

Step 5. *We calculate the elements x_k:*

$$\begin{aligned} (-1)(550) &= 51(288 + 5x_1) - 127(120 + 2x_1) \\ &\Rightarrow x_1 = 2, \\ (-1)(-32) &= 94(192 + 5x_2) - 228(80 + 2x_2) \\ &\Rightarrow x_2 = 16, \\ (-1)(76) &= 176(84 + 5x_3) - 428(35 + 2x_3) \\ &\Rightarrow x_3 = 5, \\ (-1)(-15) &= 69(24 + 5x_4) - 169(10 + 2x_4) \\ &\Rightarrow x_4 = 7, \\ (-1)(-112) &= 63(60 + 5x_5) - 152(25 + 2x_5) \\ &\Rightarrow x_5 = 12, \\ (-1)(70) &= 25(204 + 5x_6) - 61(85 + 2x_6) \\ &\Rightarrow x_6 = 5, \\ (-1)0 &= 35(60 + 5x_7) - 85(25 + 2x_7) \\ &\Rightarrow x_7 = 5, \\ (-1)0 &= 35(60 + 5x_8) - 85(25 + 2x_8) \\ &\Rightarrow x_8 = 5, \\ (-1)0 &= 35(60 + 5x_9) - 85(25 + 2x_9) \\ &\Rightarrow x_9 = 5. \end{aligned}$$

Step 6. *We rename x_k as follows*:

$$\begin{aligned} x_1 &= b_2^1 = 2,\ x_2 = b_2^2 = 16,\ x_3 = b_2^3 = 5, \\ x_4 &= b_2^4 = 7,\ x_5 = b_2^5 = 12,\ x_6 = b_2^6 = 5, \\ x_7 &= b_2^7 = 5,\ x_8 = b_2^8 = 5,\ \ x_9 = b_2^9 = 5. \end{aligned}$$

Step 7. *We construct the block matrices* B_k :

$$B_1 = \begin{bmatrix} 24 & 2 \\ 1 & 23 \end{bmatrix}, B_2 = \begin{bmatrix} 16 & 16 \\ 14 & 12 \end{bmatrix}, B_3 = \begin{bmatrix} 7 & 5 \\ 24 & 28 \end{bmatrix},$$
$$B_4 = \begin{bmatrix} 2 & 7 \\ 7 & 17 \end{bmatrix}, \quad B_5 = \begin{bmatrix} 5 & 12 \\ 11 & 4 \end{bmatrix}, B_6 = \begin{bmatrix} 17 & 5 \\ 3 & 5 \end{bmatrix},$$
$$B_7 = \begin{bmatrix} 5 & 5 \\ 5 & 5 \end{bmatrix}, \quad B_8 = \begin{bmatrix} 5 & 5 \\ 5 & 5 \end{bmatrix}, \quad B_9 = \begin{bmatrix} 5 & 5 \\ 5 & 5 \end{bmatrix}.$$

Step 8. *We recover the following message matrix* $\mathcal{M}$:

$$\mathcal{M} = \begin{bmatrix} 24 & 2 & 16 & 16 & 7 & 5 \\ 1 & 23 & 14 & 12 & 24 & 28 \\ 2 & 7 & 5 & 12 & 17 & 5 \\ 7 & 17 & 11 & 4 & 3 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 \end{bmatrix} = \begin{bmatrix} H & A & P & P & Y & 0 \\ B & I & R & T & H & D \\ A & Y & 0 & T & O & 0 \\ Y & O & U & : &) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

6. Conclusion

We have presented two new coding and decoding algorithms using the generalized Pell (p, i)-numbers for $p = 1$. Combining these algorithms together, we can generate a new mixed algorithm such as the Minesweeper model introduced in [11]. Furthermore it is possible to develop new mixed models using Fibonacci and Lucas blocking algorithms given in [10] and [11]. Taking $p \geq 2$, it possible to increase the size of blocks. For example, a new coding algorithm using 3×3 block matrices is given in [12].

References

[1] M. Basu and B. Prasad, *The generalized relations among the code elements for Fibonacci coding theory*, Chaos Solitons Fractals **41** (2009), no. 5, 2517–2525.

[2] M. Esmaeili and M. Esmaeili, *A Fibonacci-polynomial based coding method with error detection and correction*, Comput. Math. Appl. **60** (2010), no. 10, 2738–2752.

[3] E. Kılıç, *The generalized Pell (p, i)-numbers and their Binet formulas, combinatorial representations, sums*, Chaos Solitons Fractals **40** (2009), 2047–2063.

[4] T. Koshy, *Pell and Pell–Lucas numbers with applications*, Springer, Berlin (2014).

[5] S. Prajapat, A. Jain, and R. S. Thakur, *A novel approach for information security with automatic variable key using Fibonacci Q-matrix*, IJCCT **3** (2012), no. 3, 54–57.

[6] B. Prasad, *Coding Theory on Lucas p Numbers*, Discrete Math. Algorithms Appl. **8** (2016), no. 4, 17 pages.

[7] A. Stakhov, V. Massingue, and A. Sluchenkov, *Introduction into Fibonacci coding and cryptography*, Osnova, Kharkov (1999).

[8] A. P. Stakhov, *Fibonacci matrices, a generalization of the Cassini formula and a new coding theory*, Chaos Solitons Fractals **30** (2006), no. 1, 56–66.

[9] B. S. Tarle and G. L. Prajapati, *On the information security using Fibonacci series*, International Conference and Workshop on Emerging Trends in Technology (ICWET 2011)-TCET, Mumbai, India.

[10] N. Taş, S. Uçar, N. Y. Özgür and Ö. Ö. Kaymak, *A new coding/decoding algorithm using Fibonacci numbers*, Discrete Math. Algorithms Appl. **10** (2018), no. 2.

[11] S. Uçar, N. Taş, and N. Y. Özgür, *A new application to coding theory via Fibonacci and Lucas numbers*, Math. Sci. Appl. E-Notes **7** (2019), no. 1, 62–70.

[12] S. Uçar and N. Y. Özgür, *Right circulant matrices with generalized Fibonacci and Lucas polynomials and coding theory*, J. BAUN Inst. Sci. Technol. **21** (2019), no. 1, 306–322,

[13] F. Wang, J. Ding, Z. Dai, and Y. Peng, *An application of mobile phone encryption based on Fibonacci structure of chaos*, 2010 Second WRI World Congress on Software Engineering.

Balikesir University, Department of Mathematics, 10145 Balikesir, Turkey
E-mail address: nihaltas@balikesir.edu.tr

Balikesir University, Department of Mathematics, 10145 Balikesir, Turkey
E-mail address: sumeyraucar@balikesir.edu.tr

Balikesir University, Department of Mathematics, 10145 Balikesir, TURKEY
E-mail address: nihal@balikesir.edu.tr