Coding theory on Pell-Lucas $p$ numbers

P. Sundarayya $^1$, M.G. Vara Prasad $^2$

$^1$Department of Mathematics, GITAM (Deemed to be University), Visakhapatnam, India
psundarayya@gmail.com

$^2$Department of Mathematics, NSRIT, Visakhapatnam, India
prasadvaram11@gmail.com

Abstract:
In this paper, we developed a new coding and decoding method followed from Pell-Lucas $p$ numbers. We established the relations among the code matrix elements for $p = 1$ and $i=1$, error detection and correction for this coding theory. Correction ability of this method is 93.33% for $p = 1,i=1$ and for $p = 2,i=2$ the correction ability is 99.80%. In general, correction ability of this method increases as $p$ increases.

Keywords: Pell numbers; Lucas $p$ numbers; Pell-Lucas $p$ numbers; golden mean; code matrix.

Mathematics Subject Classification 2010: 11B39, 94B35, 94B25, 11T71

1. Introduction:

In [2], The generalized pell(p,i)-numbers have been defined the following recurrence relations.

$p_p(i)(n) = 2p_p(i)(n-1) + p_p(i)(n-p-1) for \ n > p + 1, 0 \leq i \leq p , p= 1,2,3 ... ...

With initial terms
$p_p(i)(1)=0, p_p(i)(i)=1 \ \text{for } i=1,2,3,\ldots, p$

For $i=p=1$, the generalized pell (1,1)-numbers is $(n+1)$th classical pell numbers defined as $P_n=2P_{n-1}+P_{n-2}$ for $n=1,2,3$ ...

With initial times $P_0=0, P_1=1$

The pell $P$ matrix of order 2 is $P=egin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix}$

The $n^{th}$ power of the $P$-matrix is $P^n = \begin{pmatrix} P_{n+1} & P_n \\ P_n & P_{n-1} \end{pmatrix}$

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

In [2], It was introduced the matrix $A$ in the form

$A = \begin{pmatrix} 2 & 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix}$ of order $(p+1)$

For $i=p$

$A^n = G_n = \begin{pmatrix} p_p(p)(n+p+1) & p_p(p)(n+1) & p_p(p)(n+2) & \cdots & p_p(p)(n+p) \\ p_p(p)(n+p) & p_p(p)(n) & p_p(p)(n+1) & \cdots & p_p(p)(n+p-1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_p(p)(n+2) & p_p(p)(n-p+2) & p_p(p)(n-p+3) & \cdots & p_p(p)(n+1) \\ p_p(p)(n+1) & p_p(p)(n-p+1) & p_p(p)(n-p+2) & \cdots & p_p(p)(n) \end{pmatrix}$

of order $(p+1)$

$\text{Det}(A^n) = \text{Det}(G_n) = (-1)^{n(p+2)}$

Generalized Lucas- pell (p,i) –numbers
\[ Q_p^{(i)}(n) = 2Q_p^{(i)}(n-1) + Q_p^{(i)}(n-p-1) \text{ for } n > p + 1, 0 \leq i \leq p, p = 1, 2, 3, \ldots. \tag{9} \]

With initial terms
\[ Q_p^{(i)}(1) = \ldots = Q_p^{(i)}(0) = 2, P_p^{(i)}(1) = \ldots = P_p^{(i)}(p+1) = p+2 \tag{10} \]

For \( i = p = 1 \), then the generalized Lucas Pell(1,1)-number is \((n+1)\text{th} \) classical Lucas-Pell numbers defined as
\[ Q_n = 2, P_n = 2 \text{ for } n = 1, 2, 3, \ldots. \]

2. **The Pell-Lucas p-matrix**

We define the \((p+1)\times(p+1)\) matrix defined as follows

\[
S_pA^n = \begin{pmatrix}
Q_p^{(p)}(n+p) & Q_p^{(p)}(n) & Q_p^{(p)}(n+1) & \cdots & Q_p^{(p)}(n+p-1) \\
Q_p^{(p)}(n+p-1) & Q_p^{(p)}(n-1) & Q_p^{(p)}(n) & \cdots & Q_p^{(p)}(n+p-2) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
Q_p^{(p)}(n+1) & Q_p^{(p)}(n-p+1) & Q_p^{(p)}(n-p+2) & \cdots & Q_p^{(p)}(n) \\
Q_p^{(p)}(n) & Q_p^{(p)}(n-p) & Q_p^{(p)}(n-p+1) & \cdots & Q_p^{(p)}(n-1)
\end{pmatrix} \tag{11}
\]

Theorem 1: Let \( n \) be a positive integer and the matrix \( S_pA^n \) then

\[
S_pA^n = \begin{pmatrix}
Q_p^{(p)}(n+p) & Q_p^{(p)}(n) & Q_p^{(p)}(n+1) & \cdots & Q_p^{(p)}(n+p-1) \\
Q_p^{(p)}(n+p-1) & Q_p^{(p)}(n-1) & Q_p^{(p)}(n) & \cdots & Q_p^{(p)}(n+p-2) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
Q_p^{(p)}(n+1) & Q_p^{(p)}(n-p+1) & Q_p^{(p)}(n-p+2) & \cdots & Q_p^{(p)}(n) \\
Q_p^{(p)}(n) & Q_p^{(p)}(n-p) & Q_p^{(p)}(n-p+1) & \cdots & Q_p^{(p)}(n-1)
\end{pmatrix} \tag{12}
\]

By using mathematical induction on \( n \), when \( p = 1 \), if \( n = 1 \) then

\[
S_1A = \begin{pmatrix}
Q_1^{(1)}(1) & Q_1^{(1)}(1) \\
Q_1^{(1)}(0) & Q_1^{(1)}(-1)
\end{pmatrix} \begin{pmatrix}
P_1^{(1)}(2) & P_1^{(1)}(1) \\
P_1^{(1)}(1) & P_1^{(1)}(0)
\end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 2 & -2 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 6 & 2 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} Q_1^{(1)}(1) & Q_1^{(1)}(1) \\ Q_1^{(1)}(0) & Q_1^{(1)}(0) \end{pmatrix}
\]

When \( p = 1 \), \( S_1A = \begin{pmatrix} Q_1^{(1)}(1) & Q_1^{(1)}(1) \\ Q_1^{(1)}(0) & Q_1^{(1)}(0) \end{pmatrix} \) is true for \( n = 1 \)

When \( p = 1 \), now assume that \( S_1A^{k-1} = \begin{pmatrix} Q_1^{(1)}(k) & Q_1^{(1)}(k-1) \\ Q_1^{(1)}(k-1) & Q_1^{(1)}(k-2) \end{pmatrix} \) is true for \( n = k-1 \)

\[
S_1A^k = \begin{pmatrix}
Q_1^{(1)}(1) & Q_1^{(1)}(1) \\
Q_1^{(1)}(0) & Q_1^{(1)}(-1)
\end{pmatrix} \begin{pmatrix}
P_1^{(1)}(k+1) & P_1^{(1)}(k) \\
P_1^{(1)}(k) & P_1^{(1)}(k-1)
\end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 2 & -2 \end{pmatrix} \begin{pmatrix} P_1^{(1)}(k+1) & P_1^{(1)}(k) \\ P_1^{(1)}(k) & P_1^{(1)}(k-1) \end{pmatrix} = \begin{pmatrix} 2P_1^{(1)}(k+1) + 2P_1^{(1)}(k) & 2P_1^{(1)}(k) + 2P_1^{(1)}(k-1) \\ 2P_1^{(1)}(k+1) - 2P_1^{(1)}(k) & 2P_1^{(1)}(k) - 2P_1^{(1)}(k-1) \end{pmatrix}
\]
\[
S_1 A^k = \begin{pmatrix}
Q_1^{(1)}(k+1) & Q_1^{(1)}(k) \\
Q_1^{(1)}(k) & Q_1^{(1)}(k-1)
\end{pmatrix}
\]

When \( p=1 \), \( S_1 A^k \) is true for \( k \)

When \( p=2, n=1 \)

\[
S_2 A = \begin{pmatrix}
Q_2^{(2)}(3) & Q_2^{(2)}(1) & Q_2^{(2)}(2) \\
Q_2^{(2)}(2) & Q_2^{(2)}(0) & Q_2^{(2)}(1) \\
Q_2^{(2)}(1) & Q_2^{(2)}(-1) & Q_2^{(2)}(0)
\end{pmatrix}
\begin{pmatrix}
P_2^{(2)}(4) & P_2^{(2)}(2) & P_2^{(2)}(3) \\
P_2^{(2)}(3) & P_2^{(2)}(1) & P_2^{(2)}(2) \\
P_2^{(2)}(2) & P_2^{(2)}(0) & P_2^{(2)}(1)
\end{pmatrix}
\]

\[
= \begin{pmatrix}
11 & 2 & 4 \\
4 & 3 & 2 \\
2 & 0 & 3
\end{pmatrix}
\begin{pmatrix}
2 & 0 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0
\end{pmatrix}
\]

\[
= \begin{pmatrix}
24 & 4 & 11 \\
11 & 2 & 4 \\
4 & 3 & 2
\end{pmatrix}
\begin{pmatrix}
Q_2^{(2)}(4) & Q_2^{(2)}(2) & Q_2^{(2)}(3) \\
Q_2^{(2)}(3) & Q_2^{(2)}(1) & Q_2^{(2)}(2) \\
Q_2^{(2)}(2) & Q_2^{(2)}(0) & Q_2^{(2)}(1)
\end{pmatrix}
\]

When \( p=2, n=1 \)

\[
S_2 A = \begin{pmatrix}
Q_2^{(2)}(3) & Q_2^{(2)}(1) & Q_2^{(2)}(2) \\
Q_2^{(2)}(2) & Q_2^{(2)}(0) & Q_2^{(2)}(1)
\end{pmatrix}
\]

When \( p=2, n=1 \)

\[
\text{Now assume that } S_2 A^k = \begin{pmatrix}
Q_2^{(2)}(k+3) & Q_2^{(2)}(k+1) & Q_2^{(2)}(k+2) \\
Q_2^{(2)}(k+2) & Q_2^{(2)}(k) & Q_2^{(2)}(k+1) \\
Q_2^{(2)}(k+1) & Q_2^{(2)}(k-1) & Q_2^{(2)}(k)
\end{pmatrix}
\]

is true for \( n=k \)

\[
S_2 A^{k+1} = S_2 A^k A = \begin{pmatrix}
Q_2^{(2)}(k+3) & Q_2^{(2)}(k+1) & Q_2^{(2)}(k+2) \\
Q_2^{(2)}(k+2) & Q_2^{(2)}(k) & Q_2^{(2)}(k+1) \\
Q_2^{(2)}(k+1) & Q_2^{(2)}(k-1) & Q_2^{(2)}(k)
\end{pmatrix}
\begin{pmatrix}
P_2^{(2)}(4) & P_2^{(2)}(2) & P_2^{(2)}(3) \\
P_2^{(2)}(3) & P_2^{(2)}(1) & P_2^{(2)}(2) \\
P_2^{(2)}(2) & P_2^{(2)}(0) & P_2^{(2)}(1)
\end{pmatrix}
\]

\[
= \begin{pmatrix}
Q_2^{(2)}(k+4) & Q_2^{(2)}(k+2) & Q_2^{(2)}(k+3) \\
Q_2^{(2)}(k+3) & Q_2^{(2)}(k+1) & Q_2^{(2)}(k+2) \\
Q_2^{(2)}(k+2) & Q_2^{(2)}(k) & Q_2^{(2)}(k+1)
\end{pmatrix}
\]

When \( p=2, n=1 \)

\[
S_2 A^{k+1} = \begin{pmatrix}
Q_2^{(2)}(k+3) & Q_2^{(2)}(k+1) & Q_2^{(2)}(k+2) \\
Q_2^{(2)}(k+2) & Q_2^{(2)}(k) & Q_2^{(2)}(k+1)
\end{pmatrix}
\]

Lemma: For fixed \( p \in \mathbb{N}, S_p A^n = \]

\[
\begin{vmatrix}
Q_p^{(p)}(n+p) & Q_p^{(p)}(n) & Q_p^{(p)}(n+1) & \cdots & Q_p^{(p)}(n+p-1) \\
Q_p^{(p)}(n+p-1) & Q_p^{(p)}(n-1) & Q_p^{(p)}(n) & \cdots & Q_p^{(p)}(n+p-2) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
Q_p^{(p)}(n) & Q_p^{(p)}(n-1) & Q_p^{(p)}(n-2) & \cdots & Q_p^{(p)}(n+p-1) \\
Q_p^{(p)}(n) & Q_p^{(p)}(n-1) & Q_p^{(p)}(n-2) & \cdots & Q_p^{(p)}(n+p-1)
\end{vmatrix}
\]

\[
\det(S_p)(-1)^{n(p+2)}
\]

3. Pell-Lucas coding and decoding method:

In this paper, we introduce a new coding theory which is known as Pell-Lucas coding and decoding method. In this method, we represent the message in the form non-singular matrix,
M of order (p+1) where p = 1, 2, 3, ..., and we represent the Pell-Lucas matrix $S_p A^n$ of order (p+1) as a coding matrix and its inverse matrix $(S_p A^n)^{-1}$ as a decoding matrix. We represent a transformation $M \times S_p A^n = E$ as Pell-Lucas coding and transformation $E \times (S_p A^n)^{-1} = M$ as Pell-Lucas decoding. We represent the matrix, E as code matrix.

3.1. Example of the Pell-Lucas coding and decoding method

Let us represent the initial message in the form of the non-singular $M$, of order 2 as follows

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

(14)

Consider $S_1 A^2 = \begin{pmatrix} Q_1^{(1)}(3) & Q_1^{(1)}(2) \\ Q_1^{(1)}(2) & Q_1^{(1)}(1) \end{pmatrix} = \begin{pmatrix} 14 & 6 \\ 6 & 2 \end{pmatrix}$

Then the inverse of inverse of $S_1 A^2$ is given by

$$(S_1 A^2)^{-1} = \begin{pmatrix} \frac{1}{8} & \frac{6}{6} \\ \frac{6}{6} & -\frac{1}{8} \end{pmatrix} = \begin{pmatrix} 2 & 6 \\ 6 & -14 \end{pmatrix}$$

Then the Pell-Lucas coding of the message (11) consists of the multiplication of the initial matrix (12) that is

$$M \times S_1 A^2 = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} \begin{pmatrix} 14 & 6 \\ 6 & 2 \end{pmatrix} = \begin{pmatrix} m_1 + 6m_4 & 6m_1 + 2m_2 \\ 14m_2 + 6m_4 & 6m_3 + 2m_4 \end{pmatrix} = \begin{pmatrix} e_1 & e_2 \\ e_3 & e_4 \end{pmatrix} = E$$

(15)

Where $e_1 = 14m_1 + 6m_2$, $e_2 = 6m_1 + 2m_2$, $e_3 = 14m_3 + 6m_4$, $e_4 = 6m_3 + 2m_4$

Then the code $E = e_1, e_2, e_3, e_4$ is sent to a channel.

The decoding of the code message, E is given with (13) is performed by the following way

$$(e_1, e_2, e_3, e_4) \times (\frac{1}{8}, \frac{6}{6}) = (\frac{1}{8}e_1 - 6e_2, \frac{6}{6}e_3 - 6e_4) = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} = M$$

(16)

4. Determinant of the code matrix $E$:

The code matrix, E is defined by the following way

$E = M \times S_p A^n$, According to matrix theory [4]

We have $\text{Det} E = \text{Det}(M \times S_p A^n) = \text{Det}(M) \times \text{Det}(S_p) \times \text{Det}(A^n)$

$$\text{Det} E = \text{Det}(M) \times \text{Det}(S_p) \times (-1)^n(p+2)$$

(17)

5. Relation between the code matrix element for $p = 1$ and $i = 1$

We consider the simplest Pell-Lucas coding and decoding method for $p = 1$ and $i = 1$

We can write the code matrix, E and the initial message M as the following

$$E = M \times S_p A^n = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} \begin{pmatrix} Q_{n+1} & Q_n \\ Q_n & Q_{n-1} \end{pmatrix} = \begin{pmatrix} e_1 & e_2 \\ e_3 & e_4 \end{pmatrix}$$

And $M = E \times (S_p A^n)^{-1} = \begin{pmatrix} e_1 & e_2 \\ e_3 & e_4 \end{pmatrix} \begin{pmatrix} 1 & -Q_{n+1} \\ -Q_n & 1 \end{pmatrix}$

$$M = \begin{pmatrix} -e_1 Q_{n-1} + e_2 Q_n & e_1 Q_n - e_2 Q_{n+1} \\ -e_3 Q_{n-1} + e_4 Q_n & e_3 Q_n - e_4 Q_{n+1} \end{pmatrix}$$

for $n = 2k+1$ (18)

Since $m_1, m_2, m_3, m_4$ are positive integers, we have

$$m_1 = \frac{-e_1 Q_{n-1} + e_2 Q_n}{8} > 0$$

(19)

$$m_2 = \frac{e_1 Q_{n-1} + e_2 Q_n}{8} > 0$$

(20)

$$m_3 = \frac{-e_3 Q_{n-1} + e_4 Q_n}{8} > 0$$

(21)

$$m_4 = \frac{e_3 Q_{n-1} + e_4 Q_n}{8} > 0$$

(22)

From (19) & (20) , we get

$$\frac{Q_{n+1}}{Q_n} < \frac{e_2}{e_1} < \frac{Q_n}{Q_{n-1}}$$

(23)

From (21) & (18) , we get

$$\frac{Q_{n+1}}{Q_{n-1}} < \frac{e_4}{e_3} < \frac{Q_{n}}{Q_n}$$

(24)
\[
\frac{q_{n+1}}{q_n} < \frac{e_3}{e_4} < \frac{q_n}{q_{n-1}} \tag{24}
\]

From the inequalities (19) and (20), we obtain

\[
\frac{e_1}{e_2} \approx \frac{\gamma}{\sqrt{2}} \tag{25}
\]

For \( n=2k \) we obtain the similar relations given (25)

6. Error Detection /Correction:

6.1. Error Detection:

Because of the many reasons arising in the channel, some errors may be occur in the code matrix \( E \). So we are going to try to determine and correct these errors using the properties of determinant in this process.

Case i: Let \( p = 1 \) and the message matrix \( M \) be as shown follows

\[
M = \begin{pmatrix}
m_1 & m_2 \\
m_3 & m_4
\end{pmatrix}
\text{ and } \text{Det}(M)=m_1m_4 - m_2m_3 \tag{26}
\]

\[
\text{code message } E = MXS_pA^n, \text{ According to matrix theory } [4] \tag{27}
\]

We have Det \( E = \text{Det}(MXS_pA^n) = \text{Det}(M) \times \text{Det}(S_p) \times \text{Det}(A^n) = \text{Det}(M) \times (\gamma)^n \times (-1)^{3n} = 8\text{Det}(M) \times (\gamma)^n \times (-1)^{3n+1} \tag{28}
\]

From the relationship between determinant of \( E \) and \( M \)

\[
\text{Det } E = \begin{cases} 8\text{Det}(M) & \text{n is even} \\ -8\text{Det}(M) & \text{n is odd} \end{cases} \tag{29}
\]

The above identity (28) is having some basis of the new method of the error detection based on the application of the Pell – Lucas \( S_pA^n \) matrix.

The essence of the method consists that the sender calculates the determinant of the initial message \( M \) represented in the matrix form (26) and sends it to the channel after the code message \( E \) (27). The receiver calculates the determinant of the code message, \( E \) (27) and compares the determinant of the initial message of \( M \) (26) received from the channel. If this comparison corresponds to (29), it means that the code message, \( E \) (27) is correct and the receiver can decode the code message, \( E \) (23) otherwise the code message, \( E \) (27) is not correct. Error detection is the first step in communication of messages.

6.2. Error Correction:

The possibility of restoration of the code message, \( E \) can be done by using the property of the Pell-Lucas \( S_pA^n \) matrix. For selecting \( p = 1, i = 1 \) and \( n = 4 \), Pell-Lucas \( S_pA^n \) matrix will be

\[
Pell-Lucas S_1A^4 = \begin{pmatrix} Q_5 & Q_4 \\ Q_4 & Q_3 \end{pmatrix} = \begin{pmatrix} 82 & 34 \\ 34 & 14 \end{pmatrix}
\]

Then the Lucas coding of the message (25) consists of the multiplication of the initial matrix (26) that is

\[
M \times S_pA^n = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} \begin{pmatrix} 82 & 34 \\ 34 & 14 \end{pmatrix} = \begin{pmatrix} 82m_1 + 34m_2 & 34m_1 + 14m_2 \\ 82m_3 + 34m_4 & 34m_3 + 14m_4 \end{pmatrix} = \begin{pmatrix} e_1 & e_2 \\ e_3 & e_4 \end{pmatrix} = E \tag{30}
\]

where \( e_1 = 82m_1 + 34m_2, e_3 = 82m_3 + 34m_4, e_2 = 34m_1 + 14m_2, e_4 = 34m_3 + 14m_4 \)

After constructing the code matrix, \( E \), we calculate the determinant of the initial matrix, \( M \) (22). The determinant is sent to the communication channel after the code message, \( E = e_1, e_2, e_3, e_4 \). Assume that the communication channel has the special means for the error detection in each of elements \( e_1, e_2, e_3, e_4 \) of the codemessage, \( E \).
Assume that the first element \( e_1 \) of \( E \) is received with the error. Then, we can represent the code message in the matrix form as given below

\[
E = \begin{pmatrix}
y_1 & e_2 \\
y_3 & e_4
\end{pmatrix}
\]  \hspace{1cm} \text{(31)}

where \( y_1 \) is the destroyed element of the code message, \( E \) but the rest matrix entries must be correct and equal to the following way:

\[
e_3 = 82m_3 + 34m_4, e_2 = 34m_1 + 14m_2, e_4 = 34m_3 + 14m_4
\]  \hspace{1cm} \text{(32)}

Then, according to the properties of the Lucas coding method, we can write the following equation for calculation of \( y_1 \):

\[
y_1e_4 - e_2e_3 = y_1 \left[ (34m_3 + 14m_4) - (34m_1 + 14m_2)(82m_3 + 34m_4) \right]
\]  \hspace{1cm} \text{(33)}

\[
y_1 = \frac{(m_1m_4 - m_2m_3)(-8)}{2m_1 + 34m_2} = e_1
\]  \hspace{1cm} \text{(34)}

Finally we conclude that \( y_1 = e_1 \). Thus, we have restored the code message \( E \) using the property of determinant of the Pell-Lucas matrix.

The main idea of this method depends on calculating the determinants of \( M \) and \( E \). Comparing the determinants obtained from the channel, the receiver can decide whether the code message \( E \) is true or not. Actually, we cannot determine which element of the code message is damaged. In order to find damaged element, we suppose different cases such as single error, two errors etc. Now, we consider the first case with a single error in the code matrix \( E \). We can easily obtain that there are four places where single error. Since the elements of message matrix \( M \) are positive integers, we should find integer solutions of the equations from If there are not integer solutions of these equations, we find that our cases related to a single error is incorrect or an error can be occurred in the checking element “\( \text{Det}(M) \)”. If \( \text{Det}(M) \) is incorrect, we use the relations given in (25) to check a correctness of the code matrix \( E \).

Similarly, we can check the cases with double errors in the code matrix \( E \). Let us consider the following case with double error in \( E \)

\[
(i) \begin{pmatrix} y_1 & e_2 \\ e_3 & e_4 \end{pmatrix} \quad (ii) \begin{pmatrix} y_2 & e_4 \\ e_3 & e_4 \end{pmatrix} \quad (iii) \begin{pmatrix} y_3 & e_2 \\ e_3 & e_4 \end{pmatrix} \quad (iv) \begin{pmatrix} y_4 & e_2 \\ e_3 & e_4 \end{pmatrix}
\]  \hspace{1cm} \text{(35)}

Finding determinants, we get following useful equations

\[
y_1e_4 - e_2e_3 = -8(-1)^{n(p+2)} \text{Det} M
\]  \hspace{1cm} \text{(36)}

\[
y_2e_4 - e_2e_3 = -8(-1)^{n(p+2)} \text{Det} M
\]  \hspace{1cm} \text{(37)}

\[
y_3e_4 - e_2e_3 = -8(-1)^{n(p+2)} \text{Det} M
\]  \hspace{1cm} \text{(38)}

\[
y_4e_4 - e_2e_3 = -8(-1)^{n(p+2)} \text{Det} M
\]  \hspace{1cm} \text{(39)}

By using above equations, we have to modify following way

\[
y_1 = \frac{-8(-1)^{n(p+2)} \text{Det} M + e_2e_3}{e_4}
\]  \hspace{1cm} \text{(40)}

\[
y_2 = \frac{-8(-1)^{n(p+2)} \text{Det} M + e_2e_3}{e_3}
\]  \hspace{1cm} \text{(41)}

\[
y_3 = \frac{-8(-1)^{n(p+2)} \text{Det} M + e_2e_3}{e_4}
\]  \hspace{1cm} \text{(42)}

\[
y_4 = \frac{-8(-1)^{n(p+2)} \text{Det} M + e_2e_3}{e_1}
\]  \hspace{1cm} \text{(43)}

Since the elements of message matrix \( M \) are positive integers, we should find integer solutions of the equations from (68) to (71). If there are not integer solutions of these equations, we find that our cases related to a single error is incorrect or an error can be occurred in the checking element “\( \text{Det}(M) \)”. If \( \text{Det}(M) \) is incorrect, we use the relations given in (25) to check a correctness of the code matrix \( E \). Similarly, we can check the cases with double errors in the code matrix \( E \).

Let us consider the following case with double error in \( E \)
where \(y_1, y_2\) are the damaged elements of \(E\). Using the relation

\[
\text{Det } E = -8(-1)^n \text{Det } M
\]  

(45)

we can write following equation for the matrix \(\text{Det } E = y_1 e_4 - y_2 e_3\)

\[
x_1 e_4 - x_2 e_3 = -8(-1)^n \text{Det } M
\]  

(46)

Also, we know the following relation between \(y_1\) and \(y_2\)

\[
\frac{y_1}{y_2} \approx y
\]  

(47)

It is clear that the equation (46) is a Diophantine equation. Because there are many solutions of Diophantine equations, we should choose the solutions \(y_1, y_2\) satisfying the above checking relation (46).

Using the similar approach, we can correct the triple errors in the code matrix \(E\) such that

\[
\begin{pmatrix}
  y_1 & y_2 \\
  y_3 & e_4
\end{pmatrix}
\]  

(48)

Where \(y_1, y_2\) and \(y_3\) are damaged elements of \(E\).

Consequently, our method depends on confirmation of various cases about damaged elements of \(E\) using the checking element \(\text{Det}(M)\) and checking relation \(y_1, y_2\). Our correctness method permits us to correct 14 cases among the 15 cases, because our method is inadequate for the case with four errors. So we can say that correction ability of our method is

\[
\frac{14}{15} = 0.9333 = 93.33\%
\]  

Case ii: Let \(p = 2\) and \(i=2\) the message matrix \(M\) be as shown follows

\[
M = \begin{pmatrix}
  m_1 & m_2 & m_3 \\
  m_4 & m_5 & m_6 \\
  m_7 & m_8 & m_9
\end{pmatrix}
\]  

(49)

\[
E = M \times (S_2 A^n) = \begin{pmatrix}
  m_1 & m_2 & m_3 \\
  m_4 & m_5 & m_6 \\
  m_7 & m_8 & m_9
\end{pmatrix} \begin{pmatrix}
  Q_2^{(2)}(n + 3) & Q_2^{(2)}(n + 1) & Q_2^{(2)}(n + 2) \\
  Q_2^{(2)}(n + 2) & Q_2^{(2)}(n) & Q_2^{(2)}(n + 1) \\
  Q_2^{(2)}(n + 1) & Q_2^{(2)}(n - 1) & Q_2^{(2)}(n)
\end{pmatrix}
\]  

\[
= \begin{pmatrix}
  e_1 & e_2 & e_3 \\
  e_4 & e_5 & e_6 \\
  e_7 & e_8 & e_9
\end{pmatrix}
\]  

\[
M = E \times (S_2 A^n)^{-1} = \begin{pmatrix}
  e_1 & e_2 & e_3 \\
  e_4 & e_5 & e_6 \\
  e_7 & e_8 & e_9
\end{pmatrix} \begin{pmatrix}
  Q_2^{(2)}(n + 3) & Q_2^{(2)}(n + 1) & Q_2^{(2)}(n + 2) \\
  Q_2^{(2)}(n + 2) & Q_2^{(2)}(n) & Q_2^{(2)}(n + 1) \\
  Q_2^{(2)}(n + 1) & Q_2^{(2)}(n - 1) & Q_2^{(2)}(n)
\end{pmatrix}^{-1}
\]  

\[
= \begin{pmatrix}
  m_1 & m_2 & m_3 \\
  m_4 & m_5 & m_6 \\
  m_7 & m_8 & m_9
\end{pmatrix}
\]  

The code matrix \(E\) may contain single, double, . . . , nine fold error similar to [1].

Thus, there are \(9c_1 + 9c_2 + 9c_3 + 9c_4 + 9c_5 + 9c_6 + 9c_7 + 9c_8 + 9c_9 = 2^4 - 1 = 511\)

(50)
codes of errors in the code matrix $E$. We can use hypotheses for single error, double error, . . . , nine fold error as [1]. The nine fold error of the code matrix is not correctable so the possibility to correct eight cases of the method is $\frac{510}{511} = 0.9980 \equiv 99.80\%$:

In general, for $p = s$ and $n > p + 1 = s + 1$ the correct possibility of the method is

$$\frac{2^{(s+1)^2-2}}{2^{(s+1)^2-1}} \approx 1 = 100\%$$

(51)

Therefore, for large value of $p = s$, the correct possibility of the method is

$$\frac{2^{(s+1)^2-2}}{2^{(s+1)^2-1}} \approx 1 = 100\%$$

(52)

7. Comparison of the Pell-Lucas Coding Method to the Classical Pell Coding Method:

The Pell-Lucas Lucas coding method is based on matrix approach which possesses many peculiarities and advantages in comparison to classical (algebraic) coding method. The use of matrix theory for designing new error-correction codes is the first peculiarity of the Pell-Lucas coding method. The large information units, in particular matrix elements, are objects of detection and correction of errors in the Pell-Lucas coding method. There is no theoretical restrictions for the value of the numbers that can be matrix elements whereas in algebraic coding theory there are very small information elements, bits and their combinations are the objects of detection and correction. Lucas coding method has very high correction ability in comparison to classical Pell coding method.

8. Conclusion:

The Pell-Lucas coding method is the main application of the $S_p A^n$ matrix. The Pell-Lucas coding method reduces to matrix multiplication, a well-known algebraic operation, which is realized very well in modern computers. The main practical peculiarity of this method is that large information units, in particular, matrix elements, are objects of detection and correction of errors. The elements of the initial matrix, $M$ and therefore the elements of the code matrix, $E$ can be the numbers of unlimited value. This means that theoretically the Pell-Lucas coding method allows correcting the numbers of unlimited value. The correction ability of this method for the simple case $p = 1$ and $i=1$ is equal 93.33% that exceeds essentially all well-known correcting codes. The correction ability of this method for the case $p = 2$ and $i=2$ is 99.80% which corrects up to eight-fold of errors among nine folds. Correction ability of this method increases as $p$ increases and for large value of $p$ it is approximately to 100%.

References

[1] A. P. Stakhov, Fibonacci matrices, a generalization of the Cassini formula and a new coding theory, Chaos Solitons and Fractals, 30 (2006), 56-66.
[2] E. Klic, The generalization pell(p,i)-numbers and their Binnet formula, combinatorial representations, sums Chaos Solitons and Fractals, 40 (2009), 2047-2063.
[3] T. Koshy, Pell and Pell-Lucas numbers with applications, Springer, Berlin (2014).
[4] B. Prasad, Coding theory on Lucas $p$ numbers, Discrete Mathematics, Algorithms and applications 8 (2016), 14-17.
[5] Nihl Tas, S. Ucar, N.Y. Ozgur, Pell coding and Pell decoding methods with some applications, arxiv, preprint arxiv-(2017)1706.04377.
[6] M. Basu, B. Prasad, The generalized relations among the code elements for Fibonacci coding theory, Chaos, Solitons and Fractals 41 (2009) 2517–2525.
[7] Kuhapatanakul, The Fibonacci $p$-numbers and Pascal’s triangle, Cogent Mathematics 2016, 3: 1264176.
[8] S. Prajapat, A. Jain, R. S. Takur, A Novel approach for Information security with automic variable key using Fibonacci Q-matrix, ijceb (2016), 2517–2525.
[9] A. P. Stakhov and B. Rozin theory of Binnet formulas for Fibonacci and Lucas $p$ numbers, Chaos, Solitons and Fractals 27 (2006) 1162–1177.
[10] S. Varda, Fibonacci and Lucas numbers and golden section, Ellis horwood limited, chichester, 1989.
[11] E. G. Kocer, N. Tuglu, A stakhov on m-extension of the fibonacci and Lucas $p$-numbers, Chaos, Solitons and Fractals 40 (2009) 1890–1904.
[12] E. G. Kocer, N. Tuglu, The Binnet formulas for the pell and PellS-Lucas $p$ numbers. ARS combinatorial (in press).