New Method of Cryptography Using Laplace and Sumudu Transforms With Python Code

Raut Priti P., *Member, IAENG,* Hiwarekar Anil P., *Member, IAENG*

***Abstract*—Due to the increase in computer networks and the vast use of the internet, individuals and organizations need to secure their data. Cryptography ensures confidence through encryption-decryption techniques via mathematics. This paper presents a new method using the successive application of the Laplace-Sumudu transform combining two functions for encoding and the corresponding inverse transform for decoding. Starting with the standard results on Laplace and Sumudu transforms, we introduce our encryption technique and derive it as new theorems. Further, it is generalized and we used a step-by-step iterative technique to increase the level of security of the algorithm. We used Python code to implement this method with examples. We also illustrate our method for Marathi language words and successfully applied it for encrypting paragraphs. Finally, we compared our method with existing cryptosystems for the advancements.**

***Index Terms*—Cryptography, Sumudu Transform, Laplace Transform, Encryption, Decryption.**

I. Introduction

INFORMATION protection is the most essential part of today's digital world, which can be done through cryptography. For developing a new cryptosystem various mathematical techniques are required. G. Naga Lakshmi, B. Ravi Kumar, and A. Chandra Shekhar [7] introduced a new cryptographic scheme using Laplace transforms. Hiwarekar A. P. [2], [3] introduced a new cryptosystem using the Laplace transform of functions $sinh\ at$, $cosh\ at$, e^{at}. E. Adeyefai, L. Akinolai, and O. Agbolade [6] used a combination of two linear functions and Laplace transforms for encoding decoding. Pranajaya A. A. and Iwan Sugiarto [1] used the Maclaurian series and Laplace transform for encryption and decryption. Shaikh J. S. and Mundhe G. A. [9], used Elzaki transform for encoding and the corresponding inverse Elzaki transform for decoding. Raut P. P. and Hiwarekar A. P. [14], used Elzaki transform and Linear Combination of functions for encoding and decoding. Bodakhe D. S. and Panchal S. K. [5], introduced the encryption-decryption method using Sumudu transform. A new method of identity (ID) based Elgamal type encryption-decryption is described by B. S. Sahana Raj, Venugopal Achar Sridhar [4].

There are several ways in cryptography in which other transforms are used along with the Laplace Transform. Jadhav S. and Hiwarekar A. P. [16], introduced a new technique for encryption and decryption by using the Laplace-Elzaki transform. Mampi Saha introduced a new cryptosystem using the Laplace-Mellin transform [12]. Hemant K. Undegaonkar and R. N. Ingle used the Laplace and Sumudu transform [8]. The existing cryptographic method that uses the Sumudu transform can be broken by general attacks [13], Tuncay M. It is crucial to develop new cryptosystems using advanced complex mathematical techniques. Therefore, in this paper, we introduced a new cryptosystem using a linear combination of two functions and using Laplace and Sumudu transforms which will be resilient to the various attacks.
We need the following definitions and notations.

Manuscript received May 28, 2024; revised January 30, 2025.

Raut Priti P. is an Assistant Professor in the Department of Humanities and Applied Sciences, Shree L. R. Tiwari College of Engineering, Thane, Maharashtra, India. (Corresponding Author e-mail: rautpriti2020@gmail.com)

Hiwarekar Anil P. is a Professor of Mathematics at Vidya Pratishthan's Kamalnayan Bajaj, Institute of Engineering and Technology, Baramati, Pune, Maharashtra, India. (e-mail: hiwarekaranil@gmail.com).

II. Definitions and Notations

We use the definition of Plain text, Cipher text, and Encryption-Decryption from [14] and other definitions and standard results from [11], and [15].

Laplace Transform: If $f(t)$ is a function defined for all positive values of t, then the Laplace Transform of $f(t)$ is defined as,
$L(f(t)) = \int_0^\infty e^{-st} f(t)dt, t \geq 0$
provided that the integral exists. Here the parameter s is a real or complex number [14].
The corresponding inverse Laplace transform is $L^{-1}F(s) = f(t)$.

Sumudu Transform: Sumudu Transform of function $f(t)$ for all real numbers, $t \geq 0$ is defined as,
$T(u) = \int_0^\infty \frac{1}{u} e^{\frac{-t}{u}} f(t)dt,\ t \geq 0$
provided that the integral exists [5].
The corresponding Inverse Sumudu Transform is $S^{-1}T(u) = f(t)$.

$$S(t^n) = n!u^n \qquad S^{-1}(n!u^n) = t^n. \tag{1}$$

We also required the following expansions.

$$e^{2t} = \frac{(2t)^0}{0!} + \frac{(2t)^1}{1!} + \frac{(2t)^2}{2!} + ... + \frac{(2t)^i}{i!} \tag{2}$$

$$= \sum_{i=0}^{\infty} \frac{(2t)^i}{i!}. \tag{3}$$

$$\cosh\ 2t = \frac{(2t)^0}{0!} + \frac{(2t)^2}{2!} + \frac{(2t)^4}{4!} + ... + \frac{(2t)^{2i}}{(2i)!} \tag{4}$$

$$= \sum_{i=0}^{\infty} \frac{(2t)^{2i}}{(2i)!}. \tag{5}$$

Here we use the following Notations:

N - Set of Natural Numbers,
n - Length of Plain Text,
q - Length of Cipher Text,
P - Plain Text.

Laplace transform and Sumudu transform have many applications in various fields such as electric circuits, solving differential equations, and in engineering [10], but in the next section, we used them for cryptography.

III. Encryption-Decryption Using Laplace-Sumudu Transform

In this section, we have discussed a new cryptosystem using a linear combination of e^{2t} and $cosh\ 2t$ and the Laplace-Sumudu transform. The proposed methodology is as follows.

A. Method of Encryption:

Here we consider

$$f(t) = a\ E\ (e^{rt} + cosh\ rt), \quad a,\ r \in N. \tag{6}$$

Here we take $a = 1$ and $r = 2$.
The following steps are involved in the process of encryption.

Step 1: Choose the plain text E, then change each letter into its numerical form so that,
A = 0, B = 1,..., Y = 24, Z = 25.

Step 2: Chosen plain text based on Step 1, E is transformed to numerals and is represented as E_i^k, where $k = 0, 1, 2, ...$ denotes the number of iterations and $i = 0, 1, 2...$ indicates the position of each letter. Let us consider the given plain text to be COMPUTER.
Here $n = 8$, given message becomes
C = 2, O = 14, M = 12, P = 15, U = 20, T = 19, E = 4, R = 17, so let us assume that,

$$\begin{aligned} &E_0^0 = 2,\ E_1^0 = 14,\ E_2^0 = 12,\ E_3^0 = 15,\ E_4^0 = 20,\\ &E_5^0 = 19,\ E_6^0 = 4, E_7^0 = 17,\ E_n^0 = 0,\ \forall\ n \geq 8. \end{aligned} \tag{7}$$

Step 3: Write numbers as the coefficient of $[e^{2t} + cosh\ 2t]$, we consider,

$$f(t) = E\ (e^{2t} + cosh\ 2t). \tag{8}$$

$$\text{Thus,} \quad f(t) = \sum_{i=0}^{\infty} \frac{(2t)^i}{i!} E_i^0 + \sum_{i=0}^{\infty} \frac{(2t)^{2i}}{(2i)!} E_i^0, \tag{9}$$

using (3) and (5)

$$\begin{aligned} f(t) = \ & \frac{(2t)^0}{0!}E_0^0 + \frac{(2t)^1}{1!}E_1^0 + \frac{(2t)^2}{2!}E_2^0 \\ &+ \frac{(2t)^3}{3!}E_3^0 + \frac{(2t)^4}{4!}E_4^0 + \frac{(2t)^5}{5!}E_5^0 \\ &+ \frac{(2t)^6}{6!}E_6^0 + \frac{(2t)^7}{7!}E_7^0 + \frac{(2t)^0}{0!}E_0^0 \\ &+ \frac{(2t)^2}{2!}E_1^0 + \frac{(2t)^4}{4!}E_2^0 + \frac{(2t)^6}{6!}E_3^0 \\ &+ \frac{(2t)^8}{8!}E_4^0 + \frac{(2t)^{10}}{10!}E_5^0 + \frac{(2t)^{12}}{12!}E_6^0 \\ &+ \frac{(2t)^{14}}{14!}E_7^0. \end{aligned} \tag{10}$$

Step 4: Using (7) and taking the Sumudu transform of the function $f(t)$, represented by equation (10), we get

$$\begin{aligned} T(u) &= S\ [f(t)] \\ &= S\ [E\ (e^{2t} + cosh\ 2t)] \\ &= 4u^0 + 28u^1 + 104u^2 + 120u^3 + 512u^4 \\ &+ 608u^5 + 1216u^6 + 2176u^7 + 5120u^8 \\ &+ 19456u^{10} + 16384u^{12} + 278528u^{14}. \end{aligned} \tag{11}$$

Step 5: Divide coefficient of u^n by $(n-1)!,\ n \geq 1$ in (11), we get

$$\begin{aligned} Z(u) = 4u^0 &+ \frac{28u^1}{0!} + \frac{104u^2}{1!} + \frac{120u^3}{2!} + \frac{512u^4}{3!} + \frac{608u^5}{4!} \\ &+ \frac{1216u^6}{5!} + \frac{2176u^7}{6!} + \frac{5120u^8}{7!} + \frac{19456u^{10}}{9!} \\ &+ \frac{16384u^{12}}{11!} + \frac{278528u^{14}}{13!}. \end{aligned} \tag{12}$$

Step 6: Taking Laplace transform of the function $Z(u)$, represented by (12), we get

$$\begin{aligned} F(s) &= L\ [Z(u)] \\ &= \frac{4}{s^1} + \frac{28(1)}{s^2} + \frac{104(2)}{s^3} + \frac{120(3)}{s^4} + \frac{512(4)}{s^5} \\ &+ \frac{608(5)}{s^6} + \frac{1216(6)}{s^7} + \frac{2176(7)}{s^8} + \frac{5120(8)}{s^9} \\ &+ \frac{19456(10)}{s^{11}} + \frac{16384(12)}{s^{13}} + \frac{278528(14)}{s^{15}}. \end{aligned} \tag{13}$$

The coefficient of $\frac{1}{s^1}, \frac{1}{s^2}, \frac{1}{s^3}, ...$ are denoted by M_i^1 for $i = 0, 1, 2, ...$

Step 7: Add p to M_i^1 to increase the security to gives

$$E_i^1 = (M_i^1 + p)\ mod\ 26 \quad \text{and Key} \quad R_i^1 = \frac{M_i^1 + p - E_i^1}{26}, \tag{14}$$

where $0 \leq p \leq 25$. In this case, we choose $p = 7$.

TABLE I: An Encryption Value Table

i	$\mathbf{M_i^1}$	$\mathbf{M_i^1 + p}$	$\mathbf{E_i^1}$	$\mathbf{R_i^1}$
0	4	11	11	0
1	28	35	9	1
2	208	215	7	8
3	360	367	3	14
4	2048	2055	1	79
5	3040	3047	5	117
6	7296	7303	23	280
7	15232	15239	3	586
8	40960	40967	17	1575
9	194560	194567	9	7483
10	196608	196615	3	7562
11	3899392	3899399	23	149976

From TABLE I, the values of
$E_0^1 = 11,\ E_1^1 = 9,\ E_2^1 = 7,\ E_3^1 = 3,\ E_4^1 = 1,$
$E_5^1 = 5,\ E_6^1 = 23,\ E_7^1 = 3,\ E_8^1 = 17,\ E_9^1 = 9,$
$E_{10}^1 = 3,\ E_{11}^1 = 23,$
be the encoded message and key is obtained as
$R_0^1 = 0,\ R_1^1 = 1,\ R_2^1 = 8,\ R_3^1 = 14,\ R_4^1 = 79,$

$R_5^1 = 117,\ R_6^1 = 280,\ R_7^1 = 586,\ R_8^1 = 1575,$
$R_9^1 = 7483,\ R_{10}^1 = 7562,\ R_{11}^1 = 149976.$

Therefore, the plain text **COMPUTER** gets converted to ciphertext **LJHDBFXDRJDX** and the corresponding key as 0, 1, 8, 14, 79, 117, 280, 586, 1575, 7483, 7562, 149976. As a result, the above-mentioned encryption method is expressed in the following theorem.

Theorem 1: The n-long plain text in terms of $E_i^0, i = 0, 1, 2, ...$ can be converted to cipher text E_i^1 under Sumudu-Laplace transform of $E_i^0\ [e^{2t} + cosh\ 2t]$ (i.e., E_i^0 as a coefficient of $[e^{2t} + cosh\ 2t]$ and then taking its Sumudu transform followed by dividing coefficient of u^n by $(n-1)!,\ n \geq 1$ and then by taking its Laplace transform to get M_i^1),
where $E_i^1 = (M_i^1 + p)\ mod\ 26, \quad p \in N, \quad 0 \leq p \leq 25$ and $E_i^0 = 0, \forall\ i \geq n$. where

$$M_i^1 = \begin{cases} 2^i(2E_i^0), \\ i < n \text{ and } i = 0; \\ \\ 2^i(i)(E_i^0 + E_{\frac{i}{2}}^0), \\ i < n \text{ and } i \text{ is even}; \\ \\ 2^i(i)E_i^0, \\ i < n \text{ and } i \text{ is odd}; \\ \\ 2^{(2i-n)}(2i-n)E_{(i-\frac{n}{2})}^0, \\ i \geq n \text{ and } n \text{ is even}; \\ \\ 2^{(2i-n+1)}(2i-n+1)E_{(i-\frac{n+1}{2}+1)}^0, \\ i \geq n \text{ and } n \text{ is odd}, \end{cases} \tag{15}$$

and Key $R_i^1 = \frac{(M_i^1 + p - E_i^1)}{26}$.

Theorem 1 is illustrated with the following examples.

Example based on Theorem 1:

Example 1: INTERNET becomes
WGCYWGKAIGCE with $(r, p) = (2, 6)$.

Example 2: INTERNET becomes
NXTPNXBRZXTV with $(r, p) = (2, 23)$.

Example 3: MATHEMATICS becomes
GIEUYESCOUSIGCYY with $(r, p) = (2, 8)$.

Example 4: MATHEMATICS becomes
PRNDHNBLXDBRPLHH with $(r, p) = (2, 17)$.

Example 5: SECURITY becomes
XVJZHTNPPHHF with $(r, p) = (2, 13)$.

Example 6: SECURITY becomes
CAOEMYSUUMMK with $(r, p) = (2, 18)$.

To increase the security of this new cryptosystem we used shift cipher. Now we extend Theorem 1 for more generalized functions, which are included as;

Theorem 2: The n-long plain text in terms of $E_i^0,\ i = 0, 1, 2, ...$ can be converted to cipher text E_i^1, Sumudu-Laplace transform of $E_i^0\ a\ [e^{rt} + cosh\ rt]$ (i.e., E_i^0 as a coefficient of $a\ [e^{rt} + cosh\ rt]$ and then taking its Sumudu transform followed by dividing coefficient of u^n by $(n-1)!, n \geq 1$ and then by taking its Laplace transform to get M_i^1),
where $E_i^1 = (M_i^1 + p)\ mod\ 26, \quad a,\ r,\ p \in N,$
$0 \leq p \leq 25$ and $E_i^0 = 0, \forall\ i \geq n$.
Here

$$M_i^1 = \begin{cases} ar^i(2E_i^0), \\ i < n \text{ and } i = 0; \\ \\ ar^i(i)(E_i^0 + E_{\frac{i}{2}}^0), \\ i < n \text{ and } i \text{ is even}; \\ \\ ar^i(i)E_i^0, \\ i < n \text{ and } i \text{ is odd}; \\ \\ ar^{(2i-n)}(2i-n)E_{(i-\frac{n}{2})}^0, \\ i \geq n \text{ and } n \text{ is even}; \\ \\ ar^{(2i-n+1)}(2i-n+1)E_{(i-\frac{n+1}{2}+1)}^0, \\ i \geq n \text{ and } n \text{ is odd}, \end{cases} \tag{16}$$

and Key $R_i^1 = \frac{(M_i^1 + p - E_i^1)}{26}$.

We illustrate it with the following examples.

Example based on Theorem 2:

Example 7: SECURE becomes
GQCWEYKSK with $(a, r, p) = (5, 3, 8)$.

Example 8: SECURE becomes
LPDFHBNBV with $(a, r, p) = (9, 12, 25)$.

Example 9: FLOWERS becomes
YEQQAYOGYA with $(a, r, p) = (12, 11, 8)$.

Example 10: FLOWERS becomes
PSLBHIHNDB with $(a, r, p) = (7, 5, 23)$.

Example 11: NETWORK becomes
MUQSIIEYKU with $(a, r, p) = (7, 4, 12)$.

Example 12: NETWORK becomes
DFZBPHXRJJ with $(a, r, p) = (15, 10, 3)$.

Now, for the more secure form of our method, we use an iterative approach based on [3] Hiwarekar A. P. In this section, we apply Theorem 2 consecutively on each output so that cipher text in the previous step becomes input (Plain text) for the next step and so on. Thus, to obtain cipher text, we can use this procedure k times on plain text. This procedure is formalized in the following new Theorem.

Theorem 3: The n-long plain text in terms of $E_i^0, i = 0, 1, 2, ...$ can be converted to cipher text E_i^k,

Sumudu-Laplace transform of $E_i^0\ a\ [e^{rt} + cosh\ rt]$ successively k times (i.e., E_i^0 as a coefficient of $a\ [e^{rt} + cosh\ rt]$ and then taking successively k times its Sumudu transform followed by dividing coefficient of u^n by $(n-1)!, n \geq 1$ and then by taking its Laplace transform to get M_i^k) where,
$E_i^k = (M_i^k + p)\ mod\ 26,\ \ a,\ r,\ p \in N,\quad 0 \leq p \leq 25$ and $E_i^{k-1} = 0, \forall i\ \geq n$.
Here

$$M_i^k = \begin{cases} ar^i(2E_i^0), \\ i < n \text{ and } i = 0; \\ \\ ar^i(i)(E_i^{k-1} + E_{\frac{i}{2}}^{k-1}), \\ i < n \text{ and } i \text{ is even}; \\ \\ ar^i(i)E_i^{k-1}, \\ i < n \text{ and } i \text{ is odd}; \\ \\ ar^{(2i-n)}(2i-n)E_{(i-\frac{n}{2})}^{k-1}, \\ i \geq n \text{ and } n \text{ is even}; \\ \\ ar^{(2i-n+1)}(2i-n+1)E_{(i-\frac{n+1}{2}+1)}^{k-1}, \\ i \geq n \text{ and } n \text{ is odd}, \end{cases} \tag{17}$$

and key $R_i^k = \frac{(M_i^k + p - E_i^k)}{26}$.

Theorem 3 can be applied with the following examples.

Example based on Theorem 3:

Example 13: TEXT becomes
XXVBVVHPDFVPJ with $(a, r, p, k) = (2, 4, 15, 3)$.

Example 14: TEXT becomes
DBTZFJFXJPJNJPZNTVXDRZNJNZPVNFXBPXX
VXJNPBPTLRTPBRNBPPLTRFZVPXXHNDPNFD
BBBRNZZLDPDZRLNNLLHXPZPVZNNPXRJFXZ
HPXXNXTXBVTFVDPRRBLDVPXTTRHPZHTXP
FZXNFVZZZVJJPXTLVBTJBFBPZPNVBBRBTBD
TJDPZNBTVTRZRJLDPZDNJBDPTPVDFPHVVX
PJHBZX with $(a, r, p, k) = (2, 4, 15, 10)$.

Example 15: MATHS becomes
IASHUUCSEQAKSMCSUMCYMQ with
$(a, r, p, k) = (7, 3, 12, 4)$.

Example 16: MATHS becomes
EAWHMUSMSOASUMSOYMCUMUWAAWMMMO
OOQIGEMSMMEIEEGOIOOUQSMQMEKGEAQCC
AWMCQUIAEWWGMQSQQKQKWCISYMGMQSA
IECEGOSMIKYEYYY with $(a, r, p, k) = (7, 3, 12, 8)$.

Example 17: TIME becomes
VTLNXRPFDPJHLZXLPFBBXXZNJHHD with
$(a, r, p, k) = (10, 5, 25, 5)$.

Example 18: TIME becomes
HZHTDTNDLPNLXZNPTPNBJBBFFZZJXTTTHDNPB
VJZTJTPRXZZDDRDHXNFVRLRZFV
with $(a, r, p, k) = (10, 5, 25, 7)$.

Remark 1:
From Theorem 2 with $r = 2,\ a = 1$ we get Theorem 1 as a special case.
Remark 2:
Similarly from Theorem 3 with $k = 1,\ r = 2,\ a = 1$ we get Theorem 1.
Remark 3:
With $k = 1$ in Theorem 3 we get Theorem 2.
Remark 4:
For different combinations of a, r, p, k we get different cipher text.
For decryption, we use the following.

B. Method of Decryption

For decoding, we proceed in the reverse direction to find the original text using the known cipher text and key. We obtain the following results.

Theorem 4: The given cipher text in terms of $E_i^1, i = 0, 1, 2, ...$ with a given value of p and key R_i^1 can be converted to plain text E_i^0 under the inverse Laplace-Sumudu transform of $E_i^0\ [e^{2t} + cosh\ 2t]$, where

$$E_i^0 = \begin{cases} \frac{(26R_i^1 + E_i^1 - p) - (2^i E_{\frac{i}{2}}^0)}{2^i}, \\ i < n \text{ and } i = 0; \\ \\ \frac{(26R_i^1 + E_i^1 - p) - (2^i E_{\frac{i}{2}}^0)}{i2^i}, \\ i < n \text{ and } i \text{ is even}; \\ \\ \frac{26R_i^1 + E_i^1 - p}{i2^i}, \\ i < n \text{ and } i \text{ is odd}; \\ \\ \frac{(26R_i^1 + E_i^1 - p) - (2^{(2i-n)} E_{i-(\frac{n}{2})}^0)}{(2i-n)2^i}, \\ i \geq n \text{ and } n \text{ is even}; \\ \\ \frac{(26R_i^1 + E_i^1 - p) - 2^{(2i-n+1)} E_{(i-\frac{(n+1)}{2}+1)}^0}{(2i-n+1)2^i}, \\ i \geq n \text{ and } n \text{ is odd}. \end{cases} \tag{18}$$

Here, n is represented by equation (21).

We illustrate Theorem 4 with the following examples.

Example Based on Theorem 4:

Example 19: WGCYWGKAIGCE becomes INTERNET with $(r, p) = (2, 6)$ and key 0, 1, 10, 3, 88, 80, 118, 655, 1339, 5120, 7562, 167621.

Example 20: NXTPNXBRZXTV becomes INTERNET with $(r, p) = (2, 23)$ and key 1, 1, 10, 4, 89, 80, 119, 655, 1339, 5120, 7562, 167621.

Example 21: GIEUYESCOUSIGCYY becomes MATHEMATICS with $(r, p) = (2, 8)$ and key 1, 0, 6, 6, 56, 74, 103, 655, 945, 354, 11815, 0, 167621, 322639, 362968, 14518744.

Example 22: PRNDHNBLXDBRPLHH becomes MATHEMATICS with $(r,p) = (2,17)$ and key 1, 0, 6, 7, 57, 74, 104, 655, 945, 355, 11816, 0, 167621, 322639, 362969, 14518745.

Example 23: XVJZHTNPPHHF becomes SECURITY with $(r,p) = (2,13)$ and key 1, 0, 2, 18, 47, 49, 576, 827, 1339, 3151, 35919, 211732.

Example 24: CAOEMYSUUMMK becomes SECURITY with $(r,p) = (2,18)$ and key 2, 1, 2, 19, 47, 49, 576, 827, 1339, 3151, 35919, 211732.

Its generalized form is included in the next theorem.

Theorem 5: The given cipher text in terms of $E_i^1, i = 0,1,2,...$ with a given value of a,p,r and key R_i^1 can be converted to plain text E_i^0 under the inverse Laplace-Sumudu transform of $E_i^0\ a\ [e^{rt} + cosh\ rt]$, where

$$E_i^0 = \begin{cases} \frac{(26R_i^1+E_i^1-p)-(ar^i E^0_{\frac{i}{2}})}{ar^i}, \\ i < n \text{ and i = 0;} \\ \\ \frac{(26R_i^1+E_i^1-p)-(ar^i E^0_{\frac{i}{2}})}{iar^i}, \\ i < n \text{ and i is even;} \\ \\ \frac{26R_i^1+E_i^1-p}{iar^i}, \\ i < n \text{ and i is odd;} \\ \\ \frac{(26R_i^1+E_i^1-p)-(ar^{(2i-n)} E^0_{i-(\frac{n}{2})})}{(2i-n)ar^i}, \\ i \geq n \text{ and n is even;} \\ \\ \frac{(26R_i^1+E_i^1-p)-(ar^{(2i-n+1)} E^0_{(i-\frac{(n+1)}{2}+1)})}{(2i-n+1)ar^i}, \\ i \geq n \text{ and n is odd.} \end{cases} \quad (19)$$

Here, n is represented by equation (21).
The above theorem, Theorem 5 can be illustrated with the following examples.

Example based on Theorem 5:

Example 25: GQCWEYKSK becomes SECURE with $(a,r,p) = (5,3,8)$ and key 7, 2, 21, 311, 1184, 934, 16823, 171595, 454223.

Example 26: LPDFHBNBV becomes SECURE with $(a,r,p) = (9,12,25)$ and key 13, 17, 599, 35890, 545517, 1722684, 124033182, 20242215228, 857317350794.

Example 27: YEQQAYOGYA becomes FLOWERS with $(a,r,p) = (12,11,8)$ and key 4, 56, 2792, 40544, 486532, 6318154, 196234449, 3165915781, 2035090237924, 312877167402340.

Example 28: PSLBHIHNDB becomes FLOWERS with $(a,r,p) = (7,5,23)$ and key 3, 15, 337, 2222, 12116, 71515, 1009616, 3365385, 446965145, 14197716347.

Example 29: MUQSIIEYKU becomes NETWORK with $(a,r,p) = (7,4,12)$ and key 7, 4, 198, 1137, 9098, 23434, 211732, 1976162, 47992517, 542033132.

Example 30: DFZBPHXRJJ becomes NETWORK with $(a,r,p) = (15,10,3)$ and key 15, 23, 2653, 38077, 761538, 4903846, 110769230, 6461538461, 980769230769, 69230769230769.

Using Theorem 5, we apply iterative technique k times successively to get plain text. It is presented in the form of the following Theorem.

Theorem 6: The given cipher text in terms of $E_i^k, i = 0,1,2,...$ with a given value of a,p,r,k and key R_i^k can be converted to plain text $E_i^{(k-1)}$ under the successively inverse Laplace-Sumudu transform of $E_i^{(k-1)}\ a\ [e^{rt} + cosh\ rt]$, where

$$E_i^{k-1} = \begin{cases} \frac{(26R_i^k+E_i^k-p)-(ar^i E^{k-1}_{\frac{i}{2}})}{ar^i}, \\ i < n \text{ and i = 0 ;} \\ \\ \frac{(26R_i^k+E_i^k-p)-(ar^i E^{k-1}_{\frac{i}{2}})}{iar^i}, \\ i < n \text{ and i is even;} \\ \\ \frac{26R_i^k+E_i^k-p}{iar^i}, \\ i < n \text{ and i is odd;} \\ \\ \frac{(26R_i^k+E_i^k-p)-(ar^{(2i-n)} E^{k-1}_{i-(\frac{n}{2})})}{(2i-n)ar^i}, \\ i \geq n \text{ and n is even;} \\ \\ \frac{(26R_i^k+E_i^k-p)-(ar^{(2i-n+1)} E^{k-1}_{(i-\frac{(n+1)}{2}+1)})}{(2i-n+1)ar^i}, \\ i \geq n \text{ and n is odd.} \end{cases} \quad (20)$$

Here,

$$n = \begin{cases} \frac{2q}{3}, & \forall q \in 3N \\ \\ \frac{2q+1}{3}, & \forall q \notin 3N. \end{cases} \quad (21)$$

We illustrate above theorem with few examples as,

Example based on Theorem 6:

Example 31: XXVBVVHPDFVPJ becomes TEXT with $(a,r,p,k) = (2,4,15,3)$.

Example 32: DBTZFJFXJPJNJPZNTVXDRZNJNZP VNFXBPXXVXJNPBPTLRTPBRNBPPLTRFZVPXXHND PNFDBBBRNZZLDPDZRLNNLLHXPZPVZNNPXRJFXZ HPXXNXTXBVTFVDPRRBLDVPXTTRHPZHTXPFZXN FVZZZVJJPXTLVBTJBFBPZPNVBBRBTBDTJDPZNBTV TRZRJLDPZDNJBDPTPVDFPHVVXPJHBZX becomes TEXT with $(a,r,p,k) = (2,4,15,10)$.

Example 33: IASHUUCSEQAKSMCSUMCYMQ becomes MATHS with $(a,r,p,k) = (7,3,12,4)$.

Example 34: EAWHMUSMSOASUMSOYMCUMUWA

AWMMMOOOQIGEMSMMEIEEGOIOOUQSMQMEKGE AQCCAWMCQUIAEWWGMQSQQKQKWCISYMGMQS AIECEGOSMIKYEYYY becomes MATHS with $(a, r, p, k) = (7, 3, 12, 8)$.

Example 35: VTLNXRPFDPJHLZXLPFBBXXZNJHHD becomes TIME with $(a, r, p, k) = (10, 5, 25, 5)$.

Example 36: HZHTDTNDLPNLXZNPTPNBJBBFFZ-ZJXTTTHDNPBVJZTJTPRXZZDDRDHXNFVRLRZFV becomes TIME with $(a, r, p, k) = (10, 5, 25, 7)$.

IV. Programmatic Solution

Based on encryption-decryption Theorems, we developed corresponding Python code useful for implementation.

Python Program:
The code is as follows. The code for Encryption given in Fig. 1.

```
import string
r1 = int(input("Input a value r* = "))
a1 = int(input("Input a value a*= "))
p1 = int(input("Input a value p*= "))
givenText = input("Enter giventext= ")
q = len(givenText)
m=0
if q%2==0:   m = int(((3*q)/2))
elif q%2!=0:   m = int((((3*q)-1)/2))
def pvalue(alphabet): return
string.ascii_lowercase.index(alphabet.lower())
encoText = {}
print("EncodedText of giventext",givenText,"is ",end='')
for i in range(0,m):    if i==0:
encoText[i]=str(chr(65+int((a1*(pow(r1,i))*(pvalue(giv-
enText[i])+pvalue(givenText[int(i/2)]))+p1)%26)))
print(encoText[i],end='')
elif i<q and i%2==0:
enco-
Text[i]=str(chr(65+int(((a1*(pow(r1,i))*i*(pvalue(giv-
enText[i])+pvalue(givenText[int(i/2)])))+p1)%26)))
 print(encoText[i],end='')
 elif i<q and i%2!=0:
enco-
Text[i]=str(chr(65+int((a1*(pow(r1,i))*i*(pvalue(giv-
enText[i]))+p1)%26))) print(encoText[i],end='')
elif i>=q and q%2==0:
encoText[i]=str(chr(65+int(((a1*(pow(r1,(2*i)-
q))*((2*i)-q)*(pvalue(givenText[i-int(q/2)]))+p1)%26)))
print(encoText[i],end='')
elif i>=q and q%2!=0:
encoText[i]=str(chr(65+int(((a1*(pow(r1,(2*i)-
(q)+1))*((2*i)-(q)+1)*(pvalue(givenText[i-
int((q+1)/2)+1])))+p1)%26)))
print(encoText[i],end='')
```

Fig. 1: Python Code For Encryption

The code given in Fig. 2 is helpful to get the key quickly. The code given in Fig. 3 is helpful to get the original text quickly.

```
keys1 = {}
print("\nEncryption key of giventext",givenText,"is
",end='')
for i in range(0,m):
 if i==0:
keys1[i]=
int((a1*(pow(r1,i))*(pvalue(givenText[i])+pvalue(giv-
enText[int(i/2)])))+p1-
(a1*(pow(r1,i))*(pvalue(givenText[i])+pvalue(giv-
enText[int(i/2)]))+p1)%26)/26
print(keys1[i],end='')
elif i<q and i%2==0:
keys1[i]=
int((a1*(pow(r1,i))*(i)*(pvalue(giv-
enText[i])+pvalue(givenText[int(i/2)])))+p1-
(a1*(pow(r1,i))*(i)*(pvalue(givenText[i])+pvalue(giv-
enText[int(i/2)]))+p1)%26)/26
print(keys1[i],end='')
elif i<q and i%2!=0:
keys1[i]=int((a1*(pow(r1,i))*(i)*(pvalue(giv-
enText[i])))+p1-
(a1*(pow(r1,i))*(i)*(pvalue(givenText[i]))+p1)%26)/26
print(keys1[i],end='')
elif i>=q and q%2==0:
keys1[i]=int((a1*(pow(r1,(2*i)-q))*((2*i)-
q)*(pvalue(givenText[i-int(q/2)])))+p1-
(a1*(pow(r1,(2*i)-q))*((2*i)-q)*(pvalue(givenText[i-
int(q/2)]))+p1)%26)/26
print(keys1[i],end='')
elif i>=q and q%2!=0:
keys1[i]=int((a1*(pow(r1,(2*i)-(q)+1))*((2*i)-
(q)+1)*(pvalue(givenText[i-int((q+1)/2)+1])))+p1-
(a1*(pow(r1,(2*i)-(q)+1))*((2*i)-
(q)+1)*(pvalue(givenText[i-
int((q+1)/2)+1]))+p1)%26)/26
print(keys1[i],end='')
```

Fig. 2: Python Code For Key

```
oriText = ''
print("\nDecodedText of text",encoText,"is ",end='')
for i in range(0,q):
if i==0: print(str(chr(65+int(((26*keys1[i]+pvalue(encoText[i])-p1)-
((a1*(pow(r1,i)))*pvalue(giv-
enText[int(i/2)]))/(a1*(pow(r1,i))))))),end='')
elif i<m and i%2==0:
print(str(chr(65+int(((26*keys1[i]+pvalue(encoText[i])-p1)-
((a1*(pow(r1,i)))*i*pvalue(giv-
enText[int(i/2)]))/(a1*i*(pow(r1,i))))))),end='
elif i<m and i%2!=0:
print(str(chr(65+int((26*keys1[i]+pvalue(encoText[i])-
p1)/(a1*i*(pow(r1,i))))))),end='')
elif i>=m and n%2==0:
print(str(chr(65+int(((26*keys1[i]+pvalue(encoText[i])-p1)-
(a1*(pow(r1,(2*i)-n))*(pvalue(givenText[i-int(n/2)])))))/(a1*((2*i)-
n)*(pow(r1,i))))))),end='')
elif i>=m and n%2!=0:
print(str(chr(65+int(((26*keys1[i]+pvalue(encoText[i])-p1)-
(a1*(pow(r1,(2*i)-n+1))*(pvalue(givenText[i-
int((n+1)/2)+1])))))/(a1*((2*i)-n+1)*(pow(r1,i))))))),end='')
```

Fig. 3: Python Code For Decryption

V. Extended Theorem For ASCII code and For Marathi Language

A. For ASCII code:

Our results of section III are applicable for the word which includes alphabets only. Now we extend it for different data sets. By using ASCII code we obtained a new compresssive encryption-decryption technique, which can applied it to this theorem not only for single word but also for paragraphs including alphabet, numbers and symbols.

Theorem 7: The n-long plain text in terms of $E_i^0, i = 0, 1, 2, ...$ can be converted to cipher text E_i^k, Sumudu-Laplace transform of E_i^0 $a\ [e^t + cosh\ t]$ successively k times (i.e., E_i^0 as a coefficient of $a\ [e^t + cosh\ t]$ and then taking successively k times its Sumudu transform followed by dividing coefficient of u^n by $(n-1)!, n \geq 1$ and then by taking its Laplace transform to get M_i^k) where,
$E_i^k = (M_i^k + p)\ mod\ 97,\ \ a,\ p \in N,\ \ 0 \leq p \leq 96$ and $E_i^{k-1} = 0, \forall i \geq n$.
Here

$$M_i^k = \begin{cases} a(2E_i^0), \\ i < n \text{ and } i = 0; \\ \\ a(i)(E_i^{k-1} + E_{\frac{i}{2}}^{k-1}), \\ i < n \text{ and } i \text{ is even}; \\ \\ a(i)E_i^{k-1}, \\ i < n \text{ and } i \text{ is odd}; \\ \\ a(2i-n)E_{(i-\frac{n}{2})}^{k-1}, \\ i \geq n \text{ and } n \text{ is even}; \\ \\ a(2i-n+1)E_{(i-\frac{n+1}{2}+1)}^{k-1}, \\ i \geq n \text{ and } n \text{ is odd}, \end{cases} \tag{22}$$

and key $R_i^k = \frac{(M_i^k + p - E_i^k)}{97}$.

Theorem 8: The given cipher text in terms of $E_i^k, i = 0, 1, 2, ...$ with a given value of a, p, k and key R_i^k can be converted to plain text $E_i^{(k-1)}$ under the successively inverse Laplace-Sumudu transform of $E_i^{(k-1)}$ $a\ [e^t + cosh\ t]$, where

$$E_i^{k-1} = \begin{cases} \frac{(97R_i^k + E_i^k - 32 - p) - (aE_{\frac{i}{2}}^{k-1})}{a}, \\ i < n \text{ and i = 0 ;} \\ \\ \frac{(97R_i^k + E_i^k - 32 - p) - (aE_{\frac{i}{2}}^{k-1})}{ia}, \\ i < n \text{ and i is even;} \\ \\ \frac{97R_i^k + E_i^k - 32 - p}{ia}, \\ i < n \text{ and i is odd;} \\ \\ \frac{(97R_i^k + E_i^k - 32 - p) - (aE_{i-(\frac{n}{2})}^{k-1})}{(2i-n)a}, \\ i \geq n \text{ and n is even;} \\ \\ \frac{(97R_i^k + E_i^k - 32 - p) - (aE_{(i-\frac{(n+1)}{2}+1)}^{k-1})}{(2i-n+1)a}, \\ i \geq n \text{ and n is odd.} \end{cases} \tag{23}$$

Here

$$n = \begin{cases} \frac{2q}{3}, & \forall q \in 3N \\ \\ \frac{2q+1}{3}, & \forall q \notin 3N. \end{cases}$$

Example Based on Theorem 7 and 8:

Example 37: Here we consider plain text paragraph as "Cryptography is the practice of securing communication and protecting sensitive data, and understanding the mathematical concepts behind these algorithms are crucial for working with them effectively."
Using Theorem 7 this paragraph is encrypted as

```
nFg~YO^P1pDOK2.{>P)X+Gz;0G:xHzPLjBN].=tngvbzZ:
mp$uL38,$ W%<9_77JQD 9WvwX-LI|E,0k$U$P,?C9j:JF
eP=[$>0e+O n33^"$-&= ))etT#;#N!gBGmp/_rt uFZe!
Q}aK$= DWtHLGS fDk]nkybj$Ox/8AU?G@eE|dp6Wl0xg =
GIKS.ml_$Nhf:l#2;}SBS7O$6(R}$.>c$"F"N :` U%:G_
.yuhQEu~}=I$Cy@)tlAjt{$dk5'Q ?O$Jk:Lo%:j\EKsY[
S)'<4I/V(n%!5Uu9$xDG S
```

with (a, p) = (2, 4).
Example 38: Similarly, we consider plain text paragraph as "If B is a set and x is an element of B, this is written in shorthand as x ∈ B, which can also be read as "x belongs to B", or "x is in B". The statement "y is not an element of B" is written as y ∉ B, which can also be read as "y is not in B"."
Using Theorem 7 we can encrypt it with (a, p) = (3, 7) as

```
Y6C3>` +Jn{z*Vwcu/jW'po"9TqrTrgKO>ZFaN|r!04bY#qz$&]/Qs
 yOOZ)T'z<o'tAU?z=]sMpG+w$@|%9'}g,+!pG'f')#!H+OY~qVC-o$
Dv$gU7g^@Ml[HM KfnU,^b _2?R.b,'C9[Qqnk}|N&N|  bO'/DUv,m\
p.<D-KD86rqRYOrL}.&\'  ('&:$7uWqf=cL]uz*g' _SsJym5-'mk-@
bk7e[a!ie04eX[APbVYMP&iJxoDF">4)v6=c}.<>'`Kz{cT{rcu:uoz
]<g'|a.d^R-X5Qi3KaIE.Z?*V\arY%/'r)'%~!Gb|ZTS>qpb'-h'?6F
^JjXCQ':N'YHNUBv:<Je94BL.ty
```

Example 39: Plain text "The natural numbers are the numbers 0, 1, 2, 3, etc., possibly excluding 0." By using Theorem 7 it will be converted to

```
,  (Vb_M4-\])8-a'eJgXuNA0DU2zPS<nz2(
PP*Fu]VAJ.4>/ 4H5l+``6r)ZV$1<Wi+/AI
k?B^/!~s|+Rd,1L\q4`5G /e<."N|!2@~Q&
pR`EfT%.!k
```

with (a, p) = (12, 16).
Example 40: We consider the plain text paragraph as "Cryptography is a technique of securing information and communications through the use of codes so that only those persons for whom the information is intended can understand and process it. Thus preventing unauthorized access to information. The prefix "crypt" means "hidden" and the suffix "graphy" means "writing"." By Theorem 7 it will be converted to

```
,  2y<)Pz;O6=nzdlVIbF[G`=s]cD~ZJf:QNOM5T(RBJ
.9b6f9]l*&X,?XZOn;nym;"$qCMq}~MfW)H+)4'gK7(
U[+T}uM<lGg$4=mxj*lO4_rO5!8a[MUJy& WgB~nqZm
D/WRvt}zazt|*{3cJ6-S P_)P041 < =P(7}hm}Ffr`8
4tNpB5 1\@^ox}H?f\_kIwN>6/;N(Nst\g/ $3Ite]+
m.M:])B vTkx5lf^jJ\0/i,,1.p{ddh)XpgLM$1,<<o
`+wjSg2<ptT%2xtKZCYhufW1m]B`#i@5>Z}?Wh]+c7(
2I'ax~W^ xO{PhzW,dqyIt}!eU!08z<-p}qoL8mV\nJ
]M?N}(}:W8W?*)gs?J7c=MU73|!>((p5X<pu\ACRD9F
=3vPHU3l`$yf)g9&Y@znG&vE&`t(,`p/wdg2P>LYCk?
eV R)K>4~?.s:oFM1)s<Bo
```

with (a, p) = (11, 96).

Example 41: Plain text "Laplace transform is named in honour of the great French mathematician, Pierre Simon De Laplace (1749-1827). Like all transforms, the Laplace transform changes one signal into another according to some fixed set of rules or equations. The best way to convert differential equations into algebraic equations is the use of Laplace transformation." By Theorem 7 it converted to

v□N,~.MNv*>{-%G2CiG"T(wg<58h_vs{1
hmg,30]+Vu-Z?□K2b9.v@U&maqn~□CVZ"H
1HsmHB!}`fT[HlX`?8d4□Oc_8E,4□oPM|I
fo73W|I3<vp.r6Th|qS%,}#?vmauQKEr□]
7D0pnq; 'm*fCg(5rA5C{nQ$*-5.insKF.
!ZnBo-O;=f7k□(%jdgAT&%□!_a-~5NXWsQ
9$(~>i>Ue:_i_G-o_v{C6□qZp+VD4YF5hf
o-RK~□L/kb$`L#Xnj,h3kQ^g6MI9e□~Ip]
v□;g|4Yd$MFm{□qE=□V□Iek1 \C+7B3T]c6
b[<KFOQ%*t<lY6w.□[/P<&□]_UAZOxI>Sr
q.5hf□< XUM_W□2%UoO,)l,□$cC6}D~=/W#
W*)}[S)+)VPSUp6_(m{h)□c 548.j43;L]M
_<tx:□B@6EwAW(U]B:hHZ#3O7|9T>□}t';
j□XO1J8In|F[d□cl8□A□DK9DN9□8(f/)oG
NE<ZBn/#;6_Zm3□o/□<@^Z%□j?'d5 □te'%
c

with $(a, p) = (21, 95)$.

B. For Marathi Language:

Our results of section III and section V are applicable for the English language only. Now we extend it for different languages. We apply our encryption-decryption technique for Marathi language. So, we convert plain text of Marathi letters into numbers using TABLE II given below.

TABLE II: Numerical Conversion For Marathi Language

Letter	अ	आ	इ	ई	उ	ऊ	ऋ	ए
Number	1	2	3	4	5	6	7	8
Letter	ऐ	ओ	औ	अं	अः	क	ख	ग
Number	9	10	11	12	13	14	15	16
Letter	घ	ङ	च	छ	ज	झ	ञ	ट
Number	17	18	19	20	21	22	23	24
Letter	ठ	ड	ढ	ण	त	थ	द	ध
Number	25	26	27	28	29	30	31	32
Letter	न	प	फ	ब	भ	म	य	र
Number	33	34	35	36	37	38	39	40
Letter	ल	व	श	ष	स	ह	ळ	क्ष
Number	41	42	43	44	45	46	47	48
Letter	ज्ञ	ा	ि	ी	ु	ू	े	ै
Number	49	50	51	52	53	54	55	56
Letter	ो	ौ	ं	ः	ॅ	c	ृ	ॐ

We illustrate our encryption method for Marathi Language in TABLE III.

TABLE III: Illustrations on Marathi Language

Sr. No.	*a*	*r*	*p*	Plain Text	Cipher Text
1.	3	2	11	सकाळ	ठदऔइऔऔ
2.	5	2	45	गणित	अ:उउउसस
3.	12	5	7	नेटवर्क	दशळखयळत्रऋत्र
4.	15	4	49	इंटरनेट	औउघज्ञज्ञज्ञज्ञज्ञज्ञ
5.	7	10	19	ऊर्जा	यघढइचच

Similarly, we can extend this cryptosystem for different languages.

VI. Cryptanalysis

In this method, we used two integral transforms and a linear combination of two functions. In addition, we used an iterative method which may be helpful to protect against various attacks, which are mentioned below.

A. Cipher text Only Attack:

In a Cipher text-only attack, the attacker has access to only a specific set of cipher texts. Suppose the attacker knows the cipher text ZHBRNDXTX. The length of cipher text is 9 but the length of plain text SECURE is 6, as we used a linear combination of two functions that increase the length of cipher text. Therefore, this algorithm may prevent Cipher text-only attacks.

B. Known-Plain text Attack:

In a Know-Plain text Attack, the attacker has access to both the plain text and corresponding Cipher text. Suppose the attacker knows plain text TEXT and the corresponding cipher text NXVDTJXTJFJPLPTNPJNDRPPVFXPDZD-BVVFLDPFDPXJPPFNBPNVJFLVJXJZNZPNZ. The length of plain text is 4 and the length of cipher text is 63 letters. The length of cipher text is 15 times the length of plain text. Therefore, this algorithm may prevent a Known-Plain text attack.

C. Chosen Plain text and Chosen Cipher text attack:

In both chosen plain text and chosen cipher text attacks, the attacker tries to solve the matrix equation (19), which is not possible as the inverse of the matrix does not exist. Therefore, this algorithm may prevent Chosen Plain text and Chosen Cipher text attacks.

VII. Comparison with Other Cryptosystems

Based on literature survey, we compare our method of encrytpion-decryption with existing methods and which have following advantages over [1], [2], [3], [8], [12] and [16] .

A. Length of Cipher Text:

In many existing algorithms as mentioned above, the length of plain text and cipher text are same. But, in our cryptosystem length of cipher text is greater than the plain text, which may prevents Cipher text only attacks, Known-plain text attacks. subsectionUse of Python Code: For our method we have developed Python code that will be useful for implementation and getting output quickly.

B. Use of Iterative Method:

In this cryptosystem we used a step-by-step iterative technique which increases the security of the algorithm.

C. Use of Combination of Different Integrals:

In our method we used combination of two functions and two different integral transforms which increases the security level.

D. Use of Caesar Cipher:

To increase the security of our cryptosystem we used Caesar cipher by shifting output by p to prevent different attacks.

E. Polyalphabetic cipher:

In [8] new cryptosystem is obtained by using Laplace and Sumudu transform which is monoalbhabetc cipher. But, our method is Polyalphabetic cipher which may prevent frequency attack.

VIII. Concluding Remarks and Future Scope

In this work, we developed a new cryptosystem using a combination of Laplace and Sumudu transforms of two functions with Python code. The main advantage of this algorithm is that we can get different output for same input by changing any one value of a or r or k or p or all the values at a time. Moreover, in this method the length of cipher text is greater than that of plain text which may protect various attacks. We extend this work for ASCII code and demonstrate its successful application to paragraph encryption. Additionally, we illustrate our technique for Marathi words also. Extension, of this work is possible by using other suitable functions and transforms and data set.

Acknowledgement

Raut Priti P. is thankful to the Principal Dr. Deven Shah of Shree L. R. Tiwari College of Engineering, Mira Road, Thane and S.P. College Pune (Research center of Mathematics) for their support to this work. Hiwarekar Anil P. is thankful to the Principal, VPKBIET, Baramati, and to the management of Vidya Pratishthan Baramati for the entire support to this work.

References

[1] A. A. Pranajaya and Iwan Sugiarto, “Simulation and Analysis on Cryptography by Maclaurin Series and Laplace Transform”, IAENG International Journal of Applied Mathematics, vol. 52, no. 2, pp 441–449, 2022.

[2] A. P. Hiwarekar, “Application of Laplace transform for Cryptographic Scheme”, Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2013, WCE 2013, 3-5 July, 2013, London, U.K., pp 95–100.

[3] A. P. Hiwarekar, “Encryption-Decryption using Laplace Transforms”, Asian Journal of Mathematics and Computers, vol. 12, pp 201–209, 2016.

[4] B. S. Sahana Raj and Venugopalachar Sridhar, “Identity-Based Cryptography Using Matrices”, Wireless Personal Communication Springer, vol. 120, pp 1637–1657, 2021.

[5] D. S. Bodakhe and S. K. Panchal, “Use of Sumudu Transform in Cryptography”, Bulletin of Marathwada Mathematical Society, vol. 16, no. 2, pp 1–6, 2015.

[6] E. Adeyefa, L. Akinolai, O. Agbolade, “Application of Laplace Transform to Cryptography Using Linear Combination of Functions”, TWMS Journal of Applied and Engineering Mathematics, vol. 11, pp 1050–1060, 2021.

[7] G. Naga Lakshmi, B. Ravi Kumar, A. Chandra Shekhar, “A cryptographic Scheme of Laplace Transforms”, International Journal of Mathematical Archieve, vol. 2, pp 2515–2519, 2011.

[8] H. K. Undegaonkar and R. N. Ingle, “Role of some integral transforms in cryptography”, International Journal of Engineering and advanced technology, vol. 9, no. 3, pp 376–380, 2020.

[9] J. S. Shaikh and G. A. Mundhe, “Application of Elzaki Transform in Cryptography”, IJMSET, vol. 3, no. 3, pp 46–48, 2016.

[10] J. Vashi and M. G. Timol, “Laplace and Sumudu Transforms and Their Application”, International Journal of Innovative Science, Engineering and Technology, vol. 3, no. 8, pp 538–542, 2016.

[11] Lokenath Debnath and Dambaru Bhatta, *Integral Transforms and their applications*, 2nd ed., Chapman and Hall/CRC 2007, pp 143–150.

[12] Mampi Saha, “Application of Laplace-Mellin Transform for Cryptography”, Rai Journal of Technology Research ad Innovation, vol. 5, no. 1, pp 12–17, 2017.

[13] M. Tuncay, “Cryptanalysis use of Sumudu Transform in Cryptography”, ITM Web conferences, CME 2017, ICAAM, vol. 13, pp 1–5, 2017.

[14] P. P. Raut and A. P. Hiwarekar, “New Method of Cryptography with Python Code Using Elzaki Transform and Linear Combination of Function”, Communications in Mathematics and Applications, vol. 14, no. 3, pp 1245–1254, 2023.

[15] R. S. Douglas, *Cryptography Theory and Practice*, 3rd ed., Chapman and Hall/CRC 2011.

[16] S. Jadhav and A. P. Hiwarekar, “New Method for Cryptography using Laplace-Elzaki Transform”, Psychology and Education, vol. 58, no. 5, pp 1–6, 2021.