Anatolian Journal of Computer Sciences

Volume:5 No:1 2020 © Anatolian Science

pp:31-41 ISSN:2548-1304

Designing a New Hybrid Cryptographic Model using Coordinate Axes

Burhan SELÇUK[1], Ayşe Nur A. TANKÜL[1], Ayşegül DÜNDAR[2], Zehra AKKUŞ [2,3], Mehmet ARSLAN[3]*
[1]Karabuk University, Computer Engineering Department, Karabuk, Turkey (bselcuk@karabuk.edu.tr aysenuraltintas@karabuk.edu.tr)
[2]Malatya Science High School, Malatya, Turkey (adundar1313@gmail.com , zehraakkis@gmail.com)
[3]Malatya Science and Art Center, Malatya, Turkey (marslanmat@gmail.com)
[*] Corresponding author

Received Date : Dec. 14, 2019.
Acceptance Date : Feb. 6, 2020.
Published Date : Jun. 1, 2020.

Abstract: With the rapid development of technology, one of the most important requirements in today's systems is the reliable transfer of information and confidentiality. Thus, military, electronics, banking systems and many other places have become the fields of use of cryptography science. Cryptology methods are used to solve these problems. In this study, a new poly-alphabetic substitution cipher is designed using the coordinate axes. This hybrid method is a mix of the Polybius square cipher and the Vigenère cipher, reinforced with the RSA cryptography algorithm. There are multiple points for a letter in the coordinate axis and there is randomness in the calculation of these points, so the proposed method is a strong encryption method that is difficult to decode.

Keywords : Poly-alphabetic substitution cipher, Vigenère cipher, Polybius square cipher, RSA, Coordinate axis.

1. Introduction

Cryptology is a set of techniques and applications based on mathematically challenging problems that enable communicating parties to exchange information securely. The science of cryptology has two main sub-branches called cryptography and cryptanalysis. Cryptography investigates the methods used to encrypt and decrypt documents; cryptoanalysis examines the mechanisms established by crypto logical systems and tries to find unknown keys ([8],[9],[12]).

In terms of key encryption, there are two categories for cryptography, symmetric and asymmetric cryptography. The main difference between these two encryption methods is the use of a key in the encryption algorithm. While symmetric encryption algorithms use the same key for both encryption and decryption, an asymmetric encryption algorithm, by contrast, uses another key to decrypt a key to encrypt data. In asymmetric systems, the key used for encryption is known as the public key and can be easily shared with other people. On the other hand, the key used for decryption is a private key and must be kept confidential [10]. Although such a difference is seemingly simple, it determines the functional differences between the two cryptographic techniques and how they are used.

Advanced Encryption Standards (AES) and Data Encryption Standards (DES) are examples of symmetric key cryptography. DES was accepted as a standard by the National Institute of Standards and Technology (NIST) in 1977 and published as the Federal Information Processing Standards (FIPS) [5]. DES is an example of block encryption. In other words, by simply splitting a plaintext into pieces (blocks), it encrypts each part independently and performs the same operation on blocks to open a

ciphertext. The length of these blocks is 64 bits. DES also receives a 64-bit key. However, the valid length of this key is 56 bits because it is spent on an 8-bit parity. DES algorithm has been used as a standard for many years, but in 2001 AES is announced by the NIST as a new standard. The basis of the emergence of AES is that DES encryption algorithm is vulnerable to attacks. AES algorithm is a block cipher algorithm that encrypts 128-bit data blocks with 128, 192 or 256-bit key options [4]. The difference between the key length bit numbers changes the number of AES tour cycles. This algorithm is more efficient both in software and hardware implementations.

RSA and Elliptical Curve Cryptography (ECC) are an asymmetric block coder used to encrypt and decrypt information. In 1977, Ron Rivest, Adi Shamir and Leonard Aldeman [16] developed the RSA algorithm and used the initials of the surnames as the name of the algorithm. The RSA logic is based on the fact that it is more difficult to factor an integer than to find it by multiplying new integers. The base value is obtained by multiplying the two sufficiently large prime numbers. Other key parameters are derived from the same two prime numbers [2]. For ECC, to find the discrete logarithm of a randomly chosen element of an elliptic curve by using a known point is not feasible [11]. The problem difficulty is determined by the elliptic curve size.

In cryptography, a hash function is a branch like encryption. The difference of a hash function from encryption is that a hash function does not use a key to do the encoding as it does for encryption. A hash function maps plaintext with different sizes into a sequence of bits of a certain length, a fixed-length hash value also called a hash code. For hashing, any change in the data changes the hash value ([15]). There are many types of hash algorithms. Message Digest 5 (MD5) and Secure Hash Algorithm (SHA) algorithms are the most commonly used cryptographic hash algorithms. MD5 was proposed by Ron Rivest in 1991 [17]. This algorithm takes an input and gives a 128 bits output in a digest form. Input processed in 512 bits block size which is divided into 16 subblocks with the size of 32 bits. SHA was first announced by the NIST in 1993, and the algorithm was modified to improve security in 1995 is called SHA1. In 2001, SHA2 was proposed which is the revised version of SHA1. SHA2 is more powerful against attacks and can be used with larger data entries [3]. The SHA-2 hash function is used in some common security applications and protocols, such as TLS and SSL, PGP, SSH, S/MIME, and IPsec. It is also used in digital signature and crypto money such as bitcoin.

Section 2 provides information about poly-alphabetic substitution cipher. Section 3 introduces star coordinates that form coordinates on axis and their properties are examined. In section 4, a new encryption model is designed using star coordinates. In section 5, the developed application is explained, and examples to better understand the proposed encryption model. Finally, the results obtained in this study are examined.

2. Poly-Alphabetic Substitution Cipher

In Caesar and a substitution cipher, a key is first selected and then each character is mapped to a single character. These are good examples of a monoalphabetic cipher. The best example for substitution cipher is Polybius square cipher [6]. Let's give a simple example;

	1	2	3	4	5
1	a	b	c	d	e
2	f	g	h	i,j	k
3	l	m	n	o	p
4	q	r	s	t	u
5	v	w	x	y	z

In the above table, every number pairs denote one letter. Using the above matrix, if we use encryption operation on plaintext which is "thischannelissafe", then we get ciphertext is {44,23,24,43,13,23,11,33,33,15,31,24,43,43,11,21,15}.

The disadvantage of Cesar cipher is that it has a small key space. As for substitution cipher, even if the number of permutations is high, it is weak against statistical attack as in the case of a dictionary attack. To overcome the disadvantage of these two ciphers, the polyalphabetic substitution cipher approach has emerged as a powerful method of statistical attack with large key space. Vigenère cipher is a good

example. Every letter in English alphabet is denoted one number as 0,1,…,25, respectively. Let's use Vigenère encryption operation for "thisexampleisaboutpolyalphabeticcipher" using the above table and key set is {2,8,15,7,4,17}

19	7	8	18	4	23	0	12	15	11	4	8	18	0	1	14	20	19	15	14	11	24	0	11
2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17
21	**15**	**23**	**25**	**8**	**14**	**2**	**20**	**4**	**18**	**8**	**25**	**20**	**8**	**16**	**21**	**24**	**10**	**17**	**22**	**0**	**5**	**4**	**2**
15	7	0	1	4	19	8	2	2	8	15	7	4	17										
2	8	15	7	4	17	2	8	15	7	4	17	2	8										
17	**15**	**15**	**8**	**8**	**10**	**10**	**10**	**17**	**15**	**19**	**24**	**6**	**25**										

The ciphertext is obtained as "vpxziocuesizuiqvykrwafecrppiikkkrptygz". In this cipher, the same letter is replaced by another letter in different places. When the key sequence is finished, the encryption starts again. The length of the key is called the period of encryption.

In the matrix array symmetric key method, a two-dimensional array is initialized with plaintext character codes and manipulated using a 128-bit secret key to obtain the mapping process [13]. This method is fast in terms of working time and also reliable. With a 128-bit key, it is necessary to try 2^{128} different combinations to decrypt without a secret key in encryption, which takes centuries to solve. The Infinite Number of Alphabetical Tables method is another Polyalphabetic Substitution method [7]. This method is similar to the Vigenère cipher method. Differently, the key sequence is generated randomly and a new key sequence for encryption is generated when each key sequence is used. This method is more reliable and complex than the Vigenère method. The drawback of the method is the need to generate a large number of key sequences for a long text. In addition to monoalphabetic and polyalphabetic methods, the methods obtained by combining these two different methods are used for encryption. Vigenère cipher with Affine cipher method is a new method that the polyalphabetic Vigenère method and monoalphabetic Affine method have been combined to provide for better results [1]. Similarly, Vigenère and Caesar cipher methods have been used together for more secure encryption [18].

The proposed method is a polyalphabetic method and the motivation of this method is Polybius square cipher, Vigenère cipher and the RSA encryption algorithm. This hybrid method is a combination of Polybius square cipher and Vigenère cipher algorithm, and it is enhanced with the RSA encryption algorithm that strengthens the method against attacks.

3. Star Coordinates

In this section, we focus on a new sequence of numbers called star coordinates in XY coordinate axes. We descript the following algorithm to introduce star coordinates. To increase randomness, we select the first three steps as the RSA algorithm.

Algorithm 1. (Construct of Star Coordinates)

Step 1. Select p and q randomly large two prime numbers,

Step 2. Compute $n = p * q$ and $\phi(n) = (p-1)(q-1)$,

Step 3. Select at random e, $\gcd(e, \phi(n)) = 1$ where $1 < e < \phi(n)$,

Step 4. Divide an array with n elements into four parts

if $mod(n,4) = 1$, then

$m = (n-1)/4$

$n = [1, \dots, m+1, -1, \dots, -m, m+2, \dots, 2m+1, -m-1, \dots, -2m]$

elseif $mod(n,4) = 3$, then

$m = (n-3)/4$

$n = [1, \dots, m+1, -1, \dots, -m-1, m+2, \dots, 2m+2, -m-2, \dots, -2m-1]$

Step 5. Construct 2 dimensions array using n such that $\{n/e\} = \begin{bmatrix} n \\ n \end{bmatrix}$

Step 6. In the {n/e} sequences, e step goes back from the end in the second column. We match the first element of the new line of arrays with first element in the first column, and then all points are matched.

Remark 1. (i) In table 1, we divide the array into four parts with n elements as step 4 in Algorithm 1. If we also reconstruct the Table 1 with different version of their parts, then we get Table 2;

Table 1. Division of the array into four parts with *n* elements

Cases	n value	Part I	Part II	Part III	Part IV
Case I	$n = 4m + 1$	$m + 1$	m	m	m
Case II	$n = 4m + 3$	$m + 1$	$m + 1$	$m + 1$	m

Table 2. Different version of Table 1

Cases	n value	Part I	Part II	Part III	Part IV
Case I	$n = 4m + 1$	m	m	m	$m + 1$
Case II	$n = 4m + 3$	m	$m + 1$	$m + 1$	$m + 1$

A total number of different versions of Table 1 is $C(4,1).C(4,3)$ i.e. 16. Indeed, Case I has four alternative separations, $C(4,1)$, via new extra one element of an array with $n = 4m + 1$, and Case II has four alternative separations, $C(4,3)$, via new extra three elements of an array with $n = 4m + 3$. If we divide the array into five parts with n elements as step 4 in Algorithm 1, then we get a division simple in table 3;

Table 3. Division of the array into five parts with *n* elements

Cases	n value	Part I	Part II	Part III	Part IV	Part V
Case I	$n = 5m$	m	m	m	m	m
Case II	$n = 5m + 1$	m	$m + 1$	m	m	m
Case III	$n = 5m + 2$	$m + 1$	$m + 1$	m	m	m
Case IV	$n = 5m + 3$	$m + 1$	$m + 1$	m	$m + 1$	m
Case V	$n = 5m + 4$	$m + 1$	m	$m + 1$	$m + 1$	$m + 1$

A total number of different versions of Table 3 is $C(5,0).C(5,1).C(5,2).C(5,3).C(5,4)$ i.e. 2500. In fact, case 1 in table 3 does not occur. Because, p and q were chosen as randomly large prime numbers in step 1 in Algorithm 1. Also, the result does not change from $C(5,0) = 1$.

If we generalize and divide an array with n elements into l parts, then we get a total number of different versions is

$$\text{if } l \text{ is even, } C(l,1).C(l,3)\dots C(l,l-1),$$
$$\text{if } l \text{ is odd, } C(l,0).C(l,1)\dots C(l,l-1).$$

(ii) In fact, the most important step in Algorithm 1 is the construction of step 4. If Algorithm 1 is desired to be used in an encryption protocol, the choice of step 4 for the sender and receiver is a secret key. Two functions that make this choice are given in the appendix.

We give these functions written in Matlab, *divide_array* and *vperms*. The function *divide_array* divides an array with n elements into l parts and return one of the divisions that is chosen from all possible divisions. This function takes n, l and ch values as an input and calculates the array of partition for ch the index of chosen version from the number of possible partitions. The partition vector is a matrix with the sizes of each part. Output of the function is the partition array. The function $vperms$ is used by *divide_array* to calculate all permutations of the first partition vector. This function takes vector v and returns all permutation of it as a matrix. Step 4 in the Algorithm 1 can be arranged by new function.

Example 1. For ease of operation, we select $p = 7, q = 3$. Compute $n = pq = 21$ and $\phi(n) = 12$. We may select $e = 7$ since $\gcd(e, \phi(n)) = \gcd(7,12) = 1$.

Let us find the star coordinates $\{21/7\}$ using the Algorithm 1 as follows;

$$\begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & -1 & -2 & -3 & -4 & -5 & 7 & 8 & 9 & 10 & 11 & -6 & -7 & -8 & -9 & -10 \\ 10 & 11 & -6 & -7 & -8 & -9 & -10 & 1 & 2 & 3 & 4 & 5 & 6 & -1 & -2 & -3 & -4 & -5 & 7 & 8 & 9 \end{matrix}$$

$$\{21/7\} = \{(1,10), (2,11), (3,-6), (4,-7), (5,-8), (6,-9), (-1,-10), (-2,1), (-3,2), (-4,3), (-5,4)$$
$$(7,5), (8,6), (9,-1), (10,-2), (11,-3), (-6,-4), (-7,-5), (-8,7), (-9,8), (-10,9)\}$$

where $n = 21$ satisfies the Case I in Table 1.

4. A New Hybrid Model

In this section, new poly-alphabet cipher method has been developed by using star coordinates obtained from the Algorithm 1. Since the same letter is represented by a different coordinate in different places, a powerful method against statistical attacks is presented.

Example 2. We let the alphabet Ω = {A, C, D, E, K, L, P, R, T, Y} and follow alphabetical order for simplicity. Using the alphabet Ω, the star coordinates of the $\{21/7\}$ and $(6,-9)$ initial coordinate, encrypt the following words;

(a) "CYPERATTACK"

(b) "ACYPERATTACKDETECTED"

In Example 1, if we place the letters of the alphabet in the star coordinates of the $\{21/7\}$, we obtained;

1	2	3	4	5	6	−1	−2	−3	−4	−5	7	8	9	10	11	−6	−7	−8	−9	−10
10	11	−6	−7	−8	−9	−10	1	2	3	4	5	6	−1	−2	−3	−4	−5	7	8	9
P	R	T	Y	A	A	C	D	E	K	L	P	R	T	Y	A	C	D	E	K	L

Let z be the quotient of $n/(s(\Omega))$ and $r = n - s(\Omega)z$. Number of repetitions of the alphabet and number of elements remaining is found as $z = 2$ and $r = 2$, respectively, where $n = 21$ and $s(\Omega) = 10$. The alphabet repeats exactly 2 times and the rest of the 1 element, A, continue to repeat operations. Thus, the first element in the alphabet repeats 3 times, the remaining 9 elements repeats 2 times.

(a) If we encrypt the word "CYPERATTACK" using the alphabet Ω, we get;

"$C_1Y_1P_1E_1R_1A_1T_1T_2A_2C_2K_1$"

$(-1,-10),(10,-2),(7,5),(-3,2),(8,6),(6,-9),(9,-1),(3,-6),(11,-3),(6,-4),(-4,3)$

(b) If we encrypt the word "ACYPERATTACKDETECTED" using the alphabet Ω, we get;

"$A_1C_1Y_1P_1E_1R_1A_2T_1T_2A_3C_2K_1D_1E_2T_1E_1C_1T_2E_2D_2$"

$(6,-9),(-1,-10),(10,-2),(7,5),(-3,2),(8,6),(11,-3),(9,-1),(3,-6),(5,-8),(6,-4),(-4,3),$

$(-2,1),(-8,7),(9,-1),(-3,2),(-1,-10),(3,-6),(-8,7),(-7,-5).$

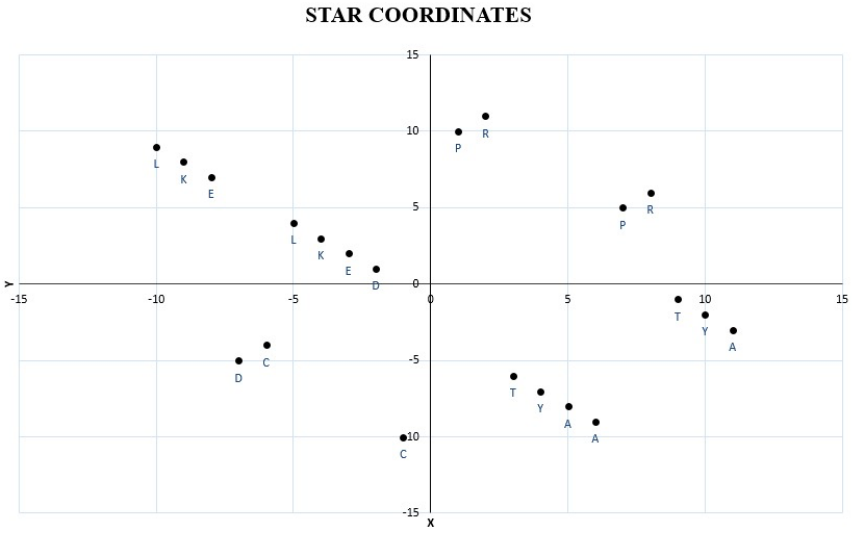

Fig. 1. Star coordinates in example 2

Remark 2. We let the alphabet Ω = {A, C, D, E, K, L, P, R, T, Y} and initial letter E. Using the alphabet Ω, the star coordinates of the $\{21/7\}$ and $(6,-9)$ initial coordinate, encrypt the same word in Example 2-(b); "ACYPERATTACKDETECTED".

In Example 1, if we place the letters of the alphabet in the star coordinates of the $\{21/7\}$, we obtained;

1	2	3	4	5	6	−1	−2	−3	−4	−5	7	8	9	10	11	−6	−7	−8	−9	−10
10	11	−6	−7	−8	−9	−10	1	2	3	4	5	6	−1	−2	−3	−4	−5	7	8	9
Y	A	C	D	E	E	K	L	P	R	T	Y	A	C	D	E	K	L	P	R	T

If we encrypt the word "ACYPERATTACKDETECTED" using the alphabet Ω, we get;

"$A_1C_1Y_1P_1E_1R_1A_2T_1T_2A_1C_2K_1D_1E_2T_1E_3C_1T_2E_1D_2$"

$(8,6),(9,-1),(7,5),(-3,2),(6,-9),(-4,3),(2,11),(-5,4),(-10,9),(8,6),(3,-6),(-1,-10),$

$(10,-2),(11,-3),(-5,4),(5,-8),(9,-1),(-10,9),(6,-9),(4,-7).$

Thus, we have obtained a different sequence from Example 2-(b) by selecting the initial letter of the alphabet differently.

4.1. Proposed Method.

The star coordinates are found as mentioned in Algorithm 1, and any initial point is selected from these points. All letters in the alphabet are placed in sequence from the initial point.

I. Key Generation.

Sender and receiver agree with Ω, p and q. Ω is the alphabet to be used for cipher. p and q are randomly large prime numbers.

II. Encryption

Step 1. Sender to calculate star coordinates puts p and q in Algorithm 1. Sender selects a random e, it satisfies $\gcd(e,\phi(n))=1$ where $1<e<\phi(n)$. Sender calculates star coordinates by following the other steps of the Algorithm 1.

Step 2. Sender select S and A are randomly parameters, which store the position of the initial coordinate (point) in the star coordinates and the position of the initial letter in the alphabet, respectively.

Step 3. Sender to assign alphabet letters to star coordinates. After the alphabet is finished, assignment of alphabet is started again. This process is continued until n points are used (in the last round, the alphabet does not have to end completely). Let z be the quotient of $n/(s(\Omega))$. The alphabet repeats exactly z times, the rest of the r elements continue to repeat operations. Thus, an initial letter and the first $r-1$ element in the alphabet after the initial letter repeats $z+1$ times, the remaining $s(\Omega)-r$ element repeats z times (see Example 2).

Step 4. The star coordinates of the letters of the text we will encrypt are written.

Step 5. Sender encrypts S and A using the RSA algorithm as $CS=S^e \bmod n$ and $CA=A^e \bmod n$. Sender transmits receiver $\{n,e,CS,CA\}$ and the ciphertext. Sender's public keys are $\{n,e,CS,CA\}$.

III. Decryption

A receiver to get the position of the initial point and initial letter calculates the secret key d such that $ed\equiv 1 \bmod \phi(n)$ and decrypts CS and CA using the RSA algorithm as $S=CS^d \bmod n$ and $A=CA^d \bmod n$. Decryption is performed in reverse order through the above steps over the encrypted text.

Remark 3. (i) Time complexity of the encryption process is $O(nt)$ where t is the plain text size and n is the number of points in the star coordinates for this method.

(ii) Frequency of cipher text characters will be one for large values of n. It is important to select the minimum value of n to get the frequency as 1 for each character, because runtime depends on the value of n. The minimum n limit can be calculated as alphabet size multiplied by the maximum character frequency in the given plain text.

(iii) If plain text that is encrypted is too long, the running time is too long accordingly. Thus, for some high frequency characters, multiple repetitions of the assigned points can be allowed for shorter run time and the method remains safe.

(iv) The method proposed for encryption and decryption is a fast method since symmetric encryption is used. Key generation and key exchange are conducted with asymmetric encryption ensuring a high level of safety. In the proposed method the strengths of the two encryption methods have been used.

(v) As explained in Remark 1 (ii), the *ch* parameter of *divide_array* function used in the construction of step 4 of the Algorithm 1 can be encrypted with RSA, such as S and A parameters in step 5 in the encryption method proposed above, and transmitted to the other party. This allows us to take security to the next level.

5. Application

We have developed an application that performs the encryption and decryption operations with the alphabet given using the star coordinates. This application was developed by using the MATLAB R2017_a program on a computer with 1.6 GHz Intel Core i5 processor.

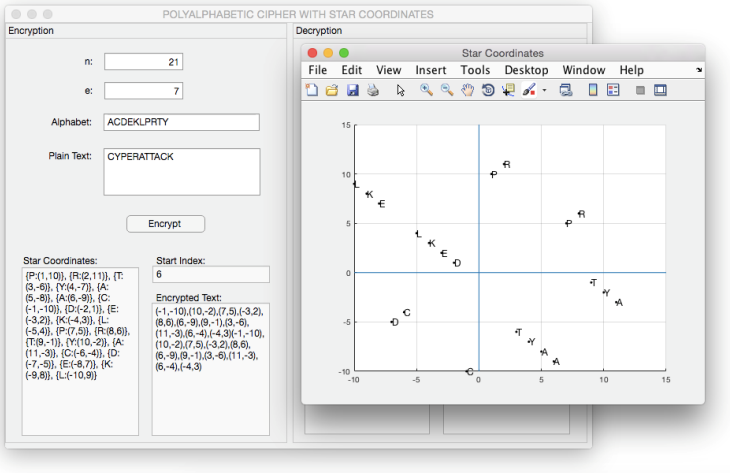

Fig. 2. The encryption solution of the example 2(a) on the application

In the "Polyalphabetic Cipher with Star Coordinates" application, the number of star coordinates $'n'$, the number of steps to go back $'e'$ in the second column from the end to and the alphabet that will be used for encryption are taken as the input in the encryption section. After the encryption is done in the application, the output is obtained as star coordinates, the index where the alphabet stars on the coordinates array and the encrypted text. In addition, the star coordinates are shown on the XY coordinate plane. The resulting encrypted text corresponds to one star coordinate for each character. If there is a character that is not in the given alphabet in the text to be encrypted, this character is not included in the encrypted text. Figure 2 shows an example of encryption.

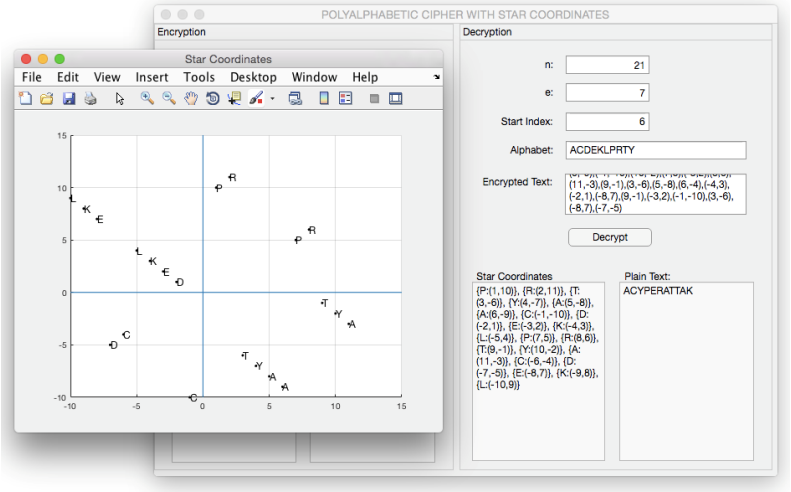

Fig. 3. The decryption solution of the example 2(b) on the application

In the developed application, as in the encryption section in the decryption section, the number of star coordinates $'n'$, the number of steps to go back $'e'$ in the second column from the end, the alphabet to be used for encryption are taken and the text to be encrypted is taken instead of the encrypted text. Additionally, the initial point of the alphabet on the coordinates array is taken as an input. The outputs are star coordinates values and decoded text. The star coordinates are shown in a new window on the XY axis as in the encryption section. Decryption example can be examined in Figure 3.

If the given alphabet has more character than the number of star coordinates, the alphabet encryption is continued from an array with having the same 'n' value and different 'e' value after assigning letters on the first array. For the example shown in the encryption section in Figure 4, only the value of the first array is taken and the initial indexes are determined for each array starting from the first. In Figure 4, 24-pointed star coordinates and a 26 character alphabet are used.

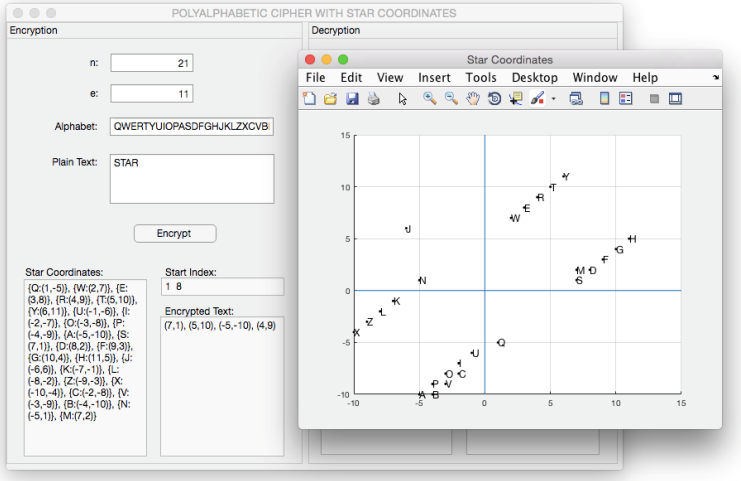

Fig. 4. Polyalphabetic cipher with star coordinates application main screen

In practice, encryption and decryption operate independently of each other. As shown in Figure 4, encryption and decryption can be performed using different alphabets.

If $p = 1051$ and $q = 8647$ are put in the RSA, then $n = 9087997$ and $\phi(n) = 9078300$ are calculated and $e = 701$ is selected randomly. Also, the alphabet Ω is selected as

{1234567890qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM .,?!:;'()-}

The plaintext [14] to be used for encryption is given below;

PERSONS OF THE DIALOGUE. Socrates, who is the narrator. Glaucon. Adeimantus. Polemarchus. Cephalus. Thrasymachus. Cleitophon. And others who are mute auditors. The scene is laid in the house of Cephalus at the Piraeus; and the whole dialogue is narrated by Socrates the day after it actually took place to Timaeus, Hermocrates, Critias, and a nameless person, who are introduced in the Timaeus. I went down yesterday to the Piraeus with Glaucon the son of Ariston, that I might offer up my prayers to the goddess (Bendis, the Thracian Artemis.); and also because I wanted to see in what manner they would celebrate the festival, which was a new thing. I was delighted with the procession of the inhabitants; but that of the Thracians was equally, if not more, beautiful. When we had finished our prayers and viewed the spectacle, we turned in the direction of the city; and at that instant Polemarchus the son of Cephalus chanced to catch sight of us from a distance as we were starting on our way home, and told his servant to run and bid us wait for him. The servant took hold of me by the cloak behind, and said: Polemarchus desires you to wait. I turned round, and asked him where his master was. There he is, said the youth, coming after you, if you will only wait. Certainly we will, said Glaucon; and in a few minutes Polemarchus appeared, and with him Adeimantus, Glaucon's brother, Niceratus the son of Nicias, and several others who had been at the procession. Polemarchus said to me: I perceive, Socrates, that you and your companion are already on your way to the city. You are not far wrong, I said. But do you see, he rejoined, how many we are? Of course. And are you stronger than all these? for if not, you will have to remain where you are. May there not be the alternative, I said, that we may persuade you to let us go? But can you persuade us, if we refuse to listen to you? he said. Certainly not, replied Glaucon. Then we are not going to listen; of that you may be assured. Adeimantus added: Has no one told you of the torch race on horseback in honour of the goddess which will take place in the evening? With horses! I replied: That is a novelty. Will horsemen carry torches and pass them one to another during the race? Yes, said Polemarchus, and not only so, but a festival will be celebrated at night, which you certainly ought to see. Let us rise soon after supper and see this festival; there will be a gathering of young men, and we will have a good talk. Stay then, and do not be perverse. Glaucon said: I suppose, since you insist, that we must. Very good, I replied. Accordingly we went with Polemarchus to his house; and there we found his brothers Lysias and Euthydemus, and with them Thrasymachus the Chalcedonian, Charmantides the Paeanian, and Cleitophon the son of Aristonymus. There too was Cephalus the father of Polemarchus, whom I had not seen for a long time, and I thought him very much aged. He was seated on a cushioned chair, and had a garland on his head, for he had been sacrificing in the court; and there were some other chairs in the room arranged in a semicircle, upon which we sat down by him. He saluted me eagerly, and then he said: You don't come to see me, Socrates, as often as you ought: If I were still able to go and see you I would not ask you to come to me. But at my age I can hardly get to the city, and therefore you should come oftener to the Piraeus. For let me tell you, that the more the pleasures of the body fade away, the greater to me is the pleasure and charm of conversation. Do not then deny my request, but make our house your resort and keep company with these young men; we are old friends, and you will be quite at home with us.

If selecting index of initial coordinate is 40 and using (9087997,701) star coordinates, then obtained ciphertext;

(12,-4543309),	(5,-4543302),	(6,-4543303),	(14,-4543311),	(11,-4543308),	(27,-4543324),	(87,-4543384),
(29,-4543326),	(84,-4543381),	(16,-4543313),	(102,-4543399),	(7,-4543304),	(18,-4543315),	(78,-4543375),
(175,-4543472),	(15,-4543312),	(10,-4543307),	(13,-4543310),	(21,-4543318),	(157,-4543454),	(17,-4543314),
(9,-4543306),	(151,-4543448),	(30,-4543327),	(248,-4543545),	(160,-4543457),	(58,-4543355),	(71,-4543368),
(53,-4543350),	(60,-4543357),	(54,-4543351),	(52,-4543349),	(61,-4543358),	(31,-4543328),	(321,-4543618),
(51,-4543348),	(65,-4543362),	(131,-4543428),	(394,-4543691),	(57,-4543354),	(134,-4543431),	(467,-4543764),
(127,-4543424),	(138,-4543435),	(125,-4543422),	(540,-4543837),	(1,-4543298),	(133,-4543430),	(126,-4543423),
(199,-4543496),	(206,-4543503),	(200,-4543497),	(204,-4543501),	(272,-4543569),	(103,-4543400),	...

6. Conclusion and future work

In this paper, text decryption and encryption algorithm based on Polybius square cipher, Vigenère cipher and the RSA encryption algorithm is designed and implemented. The proposed hybrid method is a poly-alphabet cipher method that uses the given alphabet multiple times and maps different star coordinates for the same letter. If the value n is selected large, then the frequency of every coordinate in ciphertext will be one. Also, if the initial coordinate (point) in star coordinates and the initial letter in the alphabet is randomly selected, then the probability of selecting of each element in the definition space is almost the same. That makes this method powerful against statistical attacks. By using the RSA, security is increased by one time and a highly secure method is obtained. New hybrid methods may be studied by using elliptical curve encryption instead of RSA in our proposed method, as a future work.

Acknowledgement: The authors are very grateful to Professor Ali Aydın Selçuk (The University of Tobb Etu, Turkey) and Dr. Sedat Akleylek (The University of Ondokuz Mayıs, Turkey) for his valuable suggestions, which helped to improve the paper significantly. The authors also thank for the Edior in Chief and the referee for improving and correcting the paper with their rigorous reviews.

References:

[1] T. M. Aung, H. H. Naing, and N. N. Hla, A Complex Transformation of Monoalphabetic Cipher to Polyalphabetic Cipher: (Vigenère-Affine Cipher). International Journal of Machine Learning and Computing, (2019) 9 (3) (June).

[2] P. Chaudhury, S. Dhang, M. Roy, S. Deb, J. Saha, A. Mallik, S. Bal, S. Roy, M. K. Sarkar, R. Das, ACAFP: Asymmetric Key based Cryptographic Algorithm using Four Prime Numbers to Secure Message Communication. A Review on RSA Algorithm, 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON), Bangkok, (2017) 332–337.

[3] R. Chaves, L. Sousa, N. Sklavos, A. P. Fournaris, G. Kalogeridou, P. Kitsos, F. Sheikh, Secure Hashing: SHA-1, SHA-2, and SHA-3, Taylor & Francis Group, LLC, 2016.

[4] FIPS 197, Announcing the ADVANCED ENCRYPTION STANDARD (AES). National Institute of Standarts and Technology (NIST) 2001.

[5] FIPS 46-3, Data Encryption Standard. National Institute of Standarts and Technology (NIST), 1999.

[6] H. J. Highland, Data encryption: A non-mathematical approach, Computer & Security Volume 16, Issue 5, 1997, Pages 369-386.

[7] R. S. Kartha, V. Paul, An Efficient Algorithm for Polyalphabetic Substitution Using Infinite Number of Alphabetical Tables. International Journal of Applied Engineering Research, (2018) 13 (4), 14-20.

[8] J. Katz, Y. Lindell, Introduction to Modern Cryptography: Principles and Protocols, Chapman & Hall, 2008.

[9] Ç. Koç, Cryptographic Engineering, Springer-Verlag, 2009.

[10] Y. Kumar, R. Munjal, H. Sharma, Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures. IJCSMS International Journal of Computer Science and Management Studies, (2011) 11 (03), 60–63.

[11] V. Miller, Use of elliptic curves in cryptography. Conference on the theory and application of cryptographic techniques, Springer, (1985) 417–426.

[12] C. Paar, J. Pelzl, Understanding Cryptography: A Textbook for Student and Practitioners, Springer, 2010.

[13] A.J. Paul, P. Mythili, Poly-alphabetic Substitution Mapping for Cryptographic Transformations. Conference: National Conference on Recent Innovations in Technology, (2009) (March).

[14] Plato (2002) The Republic. IDPH. Retrieved from, http://www.idph.net/conteudos/ebooks/republic.pdf

[15] A. A. P. Ratna, P. D. Purnamasari, A. Shaugi, M. Salman, Analysis and Comparison of MD5 and SHA-1 Algorithm Implementation in Simple-O Authentication based Security System. 2013 International Conference on QiR, 2013 99–104.

[16] R. L. Rivest, A. Shamir, and L.M. Adleman, A method for obtaining digital signatures and public-key cryptosystems, CACM **21**,2 (1978) pp. 120--126.

[17] R. Rivest, The MD5 Message-Digest Algorithm. Network Working Group, MIT Laboratory for Computer Science and RSA Data Security, Inc., April 1992.

[18] A. Saraswat, C. Khatri, Sudhakar, P. Thakral, P. Biswas, An Extended Hybridization of Vigenère and Caesar Cipher Techniques for Secure Communication. Procedia Computer Science 92 (2016) 355 – 360.

APPENDIX

```
function [arrayC arrayA] = divide_array(n, L, ch)
% function takes array size n, number of divide l and choise ch as an input
% return  vector arrayC chosen division and arrayA all divisions of array
    r = mod(n,L);
    m = floor(n/L);
    arrayN = ones(L,1)*m;
    i=L;
    for j=1:r
        arrayN(i) = arrayN(i)+1;
    end
    arrayA = unique(vperms(arrayN),'rows');
    arrayC = arrayA(ch,:);
end
```

```
function Perm = vperms(v)
    %  recursive method to generate permutations of given vector v
    s = numel(v);
    if s <= 1
        Perm = v;
    else
        fact = factorial(s-1);
        A = 1:fact;
        Perm = zeros(factorial(s),s);
        for i = s:-1:1
            Perm(A,1) = v(i);
            Perm(A,2:s) = vperms(v(setdiff(1:s,i)));
            A = A + fact;
        end
    end
end
```