http://www.newtheory.org ISSN: 2149-1402

Journal of New Theory

Received: 14.01.2016 *Year*: **2016**, *Number*: **14**, *Pages*: **58-72**
Published: 24.07.2016 *Original Article*[**]

FRAGMENTED POLYALPHABETIC CIPHER

Yunus Aydoğan <yunus.programmer@gmail.com>
Naim Çağman <naim.cagman@gop.edu.tr>
Irfan Şimşek[*] <irfan.simsek@gop.edu.tr>

Department of Mathematics, Faculty of Arts and Sciences, Gaziosmanpaşa University
60240 Tokat, Turkey

Abstract − In this study, we define a polyalphabetic cipher method that is called fragmented polyalphabetic (FP) cipher that is based on the fragmented Caesar (FC) cipher. In the FP-cipher, plaintext is encrypted by multiple encryption alphabets which are obtained by using the FC-cipher. We then construct a mathematical modeling and make a computer program of the method.

Keywords − *Encryption, Decryption, Cipher, Polyalphabetic Cipher, Fragmented Polyalphabetic Cipher.*

1 Introduction

One of the earliest well known cryptographic systems was used by Julius Caesar [5]. In Caesar cipher that is a simple substitution cipher and an example of monoalphabetic cipher, each letter in the plaintext is shifted by a letter a certain number of positions down the alphabet. The Caesar cipher can be decrypted in an easy way with the brute-force attack [4]. One of the first polyalphabetic ciphers called Vigenere cipher dates back to the 16th century. This cipher was named after Vigenere (1523-1596). The Vigenere cipher works by using different shift ciphers to encrypt different letters [3].

Aydoğan et al. [1] defined the fragmented Caesar (FC) cipher which is based on the basic logic of Caesar cipher. The FC-cipher has more possibility then the classical Caesar cipher because of the fragmented alphabet is used to cipher. They also construct a mathematical modeling and make a computer program of the FC-cipher.

In this study, we define a polyalphabetic cipher that is called fragmented polyalphabetic (FP) cipher. The FP-cipher is based on the FC-cipher. The FP-cipher is

[**] *Edited by* Oktay Muhtaroğlu *(Area Editor)*
[*] *Corresponding Author.*

also generalized of the Vigenere cipher. In the FC-cipher, the alphabet is broken into small fragments and each letter is replaced by a letter some arbitrary number of positions down in each fragment. The FP-cipher uses different alphabets that are obtained by using the FC-cipher to encrypt different letters. We then construct a mathematical modeling and make a computer program of this cipher method.

The present paper is a condensation of part of the dissertation [2].

2 Preliminary

In this section, we give definitions and properties of the FC-cipher which are taken directly from [1].

2.1 Mathematical Model of FC-cipher

Throughout this paper, ASCII (**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange) is used, $I_n = \{1, 2, ..., n\}$ for all $n \in \mathbb{N}$ is an index set and U is a set of using characters which is ordered according to the ASCII.

Definition 1. Let $|U| = n$ and $X = \{x_i : i \in I_n\}$ be an ordered set according to the index set I_n. Then,

$$\alpha : U \to X, \quad \alpha(i\text{-th element}) = x_i, i \in I_n,$$

is called **indexing function** of U. Here, x_i is called **indexed element** of ith element of U and the set X is called **indexed character set** of U.

Example 2. Let 1, 9, b, M, $<$ and $+$ be using characters. Then, the character set U is written as $U = \{+, 1, 9, <, M, b\}$ since

x	$+$	1	9	$<$	M	b
$ASCII$	43	49	57	60	77	98

Therefore the indexed character set of U is obtained as $X = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ since

x	$+$	1	9	$<$	M	b
$\alpha(x)$	x_1	x_2	x_3	x_4	x_5	x_6

Definition 3. Let X be an indexed character set and $|X| = n$. Then, for $a_p \in \mathbb{N}$, $p \in I_n$, a fragmentation algorithm is set up as follows:

Algorithm of Fragmentation:

Step 1: Choose a_1 such that $2 \leq a_1 \leq n_1 = n - 2$

Step 2: Let $n_2 = n_1 - a_1$. If $n_2 > 4$, choose a_2 such that $2 \leq a_2 \leq n_2 - 2$, if not $a_2 = n_2$ which means the process is terminated.

$\vdots$

Step p: Let $n_p = n_{p-1} - a_{p-1}$. If $n_p > 4$, choose a_p such that $2 \leq a_p \leq n_p - 2$, if not $a_p = n_p$ which means the process is terminated.

Here, p is called a **fragment number**, a_p is number of characters in a fragment and $P = (a_1, a_2, ..., a_p)$ is called **fragment key** of X.

We can briefly choose values a_p as follow, for $p \in I_n$ and $i \in I_p$,

$$\begin{cases} 2 \leq a_1 \leq n_1, & \text{if } p = 1,\ n_1 = n - 2 \\ 2 \leq a_p \leq n_i - 2, & \text{if } p > 1,\ 4 < n_p,\ n_p = n_{p-1} - a_{p-1} \\ n_p = a_p, & \text{if } p > 1,\ n_p < 4,\ n_p = n_{p-1} - a_{p-1} \end{cases}$$

Example 4. Let X be an indexed character set and $|X| = 13$. If the fragmentation algorithm is working as follows,

Step 1: Choose $a_1 = 5$ such that $2 \leq a_1 \leq n_1 = 13 - 2 = 11$,

Step 2: Choose $a_2 = 6$ such that $2 \leq a_2 \leq 8 - 2$, because of $n_2 = 13 - 5 = 8$ and $8 > 4$,

Step 3: Choose $a_3 = 2$ because of $n_3 = 8 - 6 = 2$ and $2 < 4$.

Then, we obtain that $p = 3$ and $P = (5, 6, 2)$.

Definition 5. Let $X = \{x_1, x_2, ..., x_n\}$ be an indexed character set. For all $i \in I_n$ and $k \in I_{n-i}$, the set $W = \{x_i, x_{i+1}, ..., x_{i+k}\}$ is called as a **block subset** of X and denoted by $W \sqsubseteq X$.

Definition 6. Let X be an indexed character set, p be a number of fragment of X. If $X_i \sqsubseteq X$ has the following conditions, then family of set $\{X_i : i \in I_p\}$ is called an **ordered fragmentation** of X.

1. $|X_i| = a_i$,
2. $X_i \cap X_j = \emptyset$ for $i, j \in I_p$, $i \neq j$,
3. $X = \bigcup_{i \in I_p} X_i$,
4. $x_{max(X_i)+1} = x_{min(X_{i+1})}$ for $i \in I_p$, where $x_{min(X_i)}$ and $x_{max(X_i)}$ be the first and the last element of X_i, respectively.

Here, the X_i is called a **fragment** of X for $i \in I_p$.

Example 7. Let us consider Example 4 where $X = \{x_1, x_2, ..., x_{13}\}$ and $P = (5, 6, 2)$. Then, for $a_1 = 5$, $a_2 = 6$ and $a_3 = 2$ the ordered fragmentation of X are respectively as follow,

$$\begin{aligned} X_1 &= \{x_1, x_2, x_3, x_4, x_5\} \\ X_2 &= \{x_6, x_7, x_8, x_9, x_{10}, x_{11}\} \\ X_3 &= \{x_{12}, x_{13}\} \end{aligned}$$

Therefor, the ordered fragmentation of X is obtained as $\{X_1, X_2, X_3\}$.

Definition 8. Let X_i be a fragment of X and a_i be the number of X_i for all $i \in I_p$. If $0 < r_i < a_i$, then $R = (r_1, r_2, ..., r_p)$ is called **rotation key** of X.

Here, the r_i is called a **number of rotation** of X_i for all $i \in I_p$.

Note that the key of this method has two part, one of them is a fragment key P and the other is a rotation key R.

Example 9. Let us consider Example 4, if we choose number of rotations $r_1 = 3$, $r_2 = 4$ and $r_3 = 1$ for X_1, X_2, X_3, respectively. Then, the rotation key of X would be $R = (3, 4, 1)$.

Definition 10. Let X be an indexed character set, $\{X_1, X_2, ..., X_p\}$ be an ordered fragmentation of X and $P = (a_1, a_2, ..., a_p)$ be a fragment key of X. Then, m_i is defined by

$$m_i = \begin{cases} 0, & i = 0 \\ m_{i-1} + a_i, & i \in I_p \end{cases}$$

and called a **module** of X_i for all $i \in I_p$.

It is clear to see that $x_{m_i} = x_{max(X_i)}$ for $i \in I_p$.

Definition 11. Let X be an indexed character set and $\{X_1, X_2, ..., X_p\}$ be an ordered fragmentation of X. If m_i is a module of X_i for all $i \in I_p$ and $R = (r_1, r_2, ..., r_p)$ is a rotation key of X, then X_i**-rotation function**, denoted by β_i, is defined by

$$\beta_i : X_i \to X_i,\ \beta_i(x_t) = \begin{cases} x_{t+r_i}, & t + r_i \leq m_i \\ x_{(t+r_i)(mod\, m_i)+m_{i-1}}, & t + r_i > m_i \end{cases}$$

where $t \in I_{a_i}$.

Definition 12. Let X be an indexed character set and $\{X_1, X_2, ..., X_p\}$ be an ordered fragmentation of X. If β_i is an X_i-rotation function for all $i \in I_p$, then the following function

$$\beta : X \to X,\ \beta(x) = \begin{cases} \beta_1(x), & x \in X_1 \\ \beta_2(x), & x \in X_2 \\ \quad\vdots & \\ \beta_p(x), & x \in X_p \end{cases}$$

is called a **rotation function** of X.

Definition 13. Let $\alpha : U \to X$ be an indexing function. Then for all $t \in I_n$, inverse of α is called a **characterization function** and defined by

$$\alpha^{-1} : X \to U, \quad \alpha^{-1}(x_t) = \text{"}t\text{-th element of U "}$$

Definition 14. If $\alpha : U \to X$, $\alpha^{-1} : X \to U$ and $\beta : X \to X$ be indexing, characterization and rotation functions, respectively, then an **encryption function** on U is defined by

$$\gamma : U \to U, \quad \gamma(x) = \alpha^{-1}(\beta(\alpha(x)))$$

Definition 15. Let X be an indexed character set and $\{X_1, X_2, ..., X_p\}$ be an ordered fragmentation of X. If $R = (r_1, r_2, ..., r_p)$ is a rotation key of X, $\beta_i : X_i \to X_i$ is an X_i-rotation function and m_i is a module of X_i for all $i \in I_p$, then **inverse of rotation function** of X_i, denoted by β_i^{-1}, is defined by

$$\beta_i^{-1} : X_i \to X_i,$$
$$\beta_i^{-1}(x_t) = \begin{cases} x_{t+m_i-(r_i+m_{i-1})}, & t + m_i - (r_i + m_{i-1}) \leq m_i \\ x_{(t+m_i-(r_i+m_{i-1}))(mod\, m_i)+m_{i-1}}, & t + m_i - (r_i + m_{i-1}) > m_i \end{cases}$$

where $t \in I_{a_i}$.

Definition 16. Let X be an indexed character set and $\{X_1, X_2, ..., X_p\}$ be an ordered fragmentation of X. If β_i^{-1} is an inverse of rotation function of X_i for all $i \in I_p$, then the following function

$$\beta^{-1} : X \to X,\ \beta^{-1}(x) = \begin{cases} \beta_1^{-1}(x), & x \in X_1 \\ \beta_2^{-1}(x), & x \in X_2 \\ \vdots & \\ \beta_p^{-1}(x), & x \in X_p \end{cases}$$

is called a **inverse of rotation function** of X.

Definition 17. If $\alpha : U \to X$, $\alpha^{-1} : X \to U$, $\beta^{-1} : X \to X$ and $\gamma : U \to U$ be indexing, characterization, inverse of rotation and encryption functions, respectively, then a **decryption function** on U is defined by

$$\gamma^{-1} : U \to U, \quad \gamma^{-1}(x) = \alpha^{-1}(\beta^{-1}(\alpha(x)))$$

It is clear to see that $\gamma^{-1}(x) = \alpha^{-1}(\beta^{-1}((\alpha^{-1})^{-1}(x))) = \alpha^{-1}(\beta^{-1}(\alpha(x)))$.

Definition 18. Let U be a character set, P be a fragment key, R be a rotation key and γ be an encryption function. The four tuple (U, P, R, γ) is called an **FC-cipher encryption** on U. The four tuple (U, P, R, γ^{-1}) is called an **FC-cipher decryption** on U.

2.2 FC-cipher Encryption Algorithm

In this subsection, we give the algorithm of FC-cipher.

Assume that U is a character set and X is an indexed character set. Then, an algorithm of the FC-cipher encryption is set up as follows:

Algorithm of FC-cipher Encryption:

Step 1: Find the fragment number p and the $P = (a_1, a_2, ..., a_p)$,
Step 2: Choose the $R = (r_1, r_2, ..., r_p)$ according to the P,
Step 3: Find the $\{X_i : i \in I_p\}$ and the module m_i for each X_i,
Step 4: Find the $\beta_i(x_t)$ for $x_t \in X_i$ $t \in I_{a_i}$ and $i \in I_p$,
Step 5: Find the $\alpha^{-1}(x_t)$ for $x_t \in X_i$, $t \in I_{a_i}$ and $i \in I_p$,
Step 6: Find the $\gamma(x)$ for $x \in U$.

Example 19. Let

U = {ç, d, e, f, g, ğ, h, ı, a, b, c, m, n, i, j, k, l, u, ü, o, ö, p,r, s, ş, t, z, v, y}

and

$$X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, ..., x_{29}\}$$

Then,

Step 1: By using the algorithm of fragmentation, we can obtain the fragment number $p = 4$ and the $P = (11, 6, 9, 3)$ where $a_1 = 11$, $a_2 = 6$, $a_3 = 9$ and $a_4 = 3$.

Step 2: For $a_1 = 11$, $a_2 = 6$, $a_3 = 9$ and $a_4 = 3$ the rotation key is obtained as $R = (3, 4, 7, 2)$ since $0 < r_1 = 3 < a_1 = 11$, $\quad 0 < r_2 = 4 < a_2 = 6$, $\quad 0 < r_3 = 7 < a_3 = 9$, $\quad 0 < r_1 = 2 < a_4 = 3$.

Step 3: For a_i $(i = 1, 2, 3, 4)$ the fragments of X, X_i, are obtained as,

$$\begin{array}{ll}
\text{for } a_1 = 11, & X_1 = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}\} \\
\text{for } a_2 = 6, & X_2 = \{x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}\} \\
\text{for } a_3 = 9, & X_3 = \{x_{18}, x_{19}, x_{20}, x_{21}, x_{22}, x_{23}, x_{24}, x_{25}, x_{26}\} \\
\text{for } a_4 = 3, & X_4 = \{x_{27}, x_{28}, x_{29}\}
\end{array}$$

and value of m_i $(i = 0, 1, 2, 3, 4)$ can also choose as,

$$\begin{array}{ll}
\text{for } i = 0, & m_0 = 0 \\
\text{for } i = 1, & m_1 = (m_0 = 0) + (a_1 = 11) = 11 \\
\text{for } i = 2, & m_2 = (m_1 = 11) + (a_2 = 6) = 17 \\
\text{for } i = 3, & m_3 = (m_2 = 17) + (a_3 = 9) = 26 \\
\text{for } i = 4, & m_4 = (m_3 = 26) + (a_4 = 3) = 29.
\end{array}$$

Step 4: For $i = 1, 2, 3, 4$ values of the X_i-rotation function β_i are obtained as follows. Here, we first obtain the values of β_1 as,

$$\begin{array}{ll}
\beta_1(x_1) = x_4, & \text{since } 1 + r_1 = 1 + 3 = 4 \text{ because of } 1 + 3 < 11 \\
\beta_1(x_2) = x_5, & \text{since } 2 + r_1 = 1 + 3 = 5 \text{ because of } 2 + 3 < 11 \\
\beta_1(x_3) = x_6, & \text{since } 3 + r_1 = 1 + 3 = 6 \text{ because of } 3 + 3 < 11 \\
\beta_1(x_4) = x_7, & \text{since } 4 + r_1 = 1 + 3 = 7 \text{ because of } 4 + 3 < 11 \\
\beta_1(x_5) = x_8, & \text{since } 5 + r_1 = 1 + 3 = 8 \text{ because of } 5 + 3 < 11 \\
\beta_1(x_6) = x_9, & \text{since } 6 + r_1 = 1 + 3 = 9 \text{ because of } 6 + 3 < 11 \\
\beta_1(x_7) = x_{10}, & \text{since } 7 + r_1 = 1 + 3 = 10 \text{ because of } 7 + 3 < 11 \\
\beta_1(x_8) = x_{11}, & \text{since } 8 + r_1 = 1 + 3 = 11 \text{ because of } 8 + 3 = 11 \\
\beta_1(x_9) = x_1, & \text{since } (9 + 3)(mod\,11) + 0 = 1 \text{ because of } 9 + 3 > 11 \\
\beta_1(x_{10}) = x_2, & \text{since } (10 + 3)(mod\,11) + 0 = 2 \text{ because of } 10 + 3 > 11 \\
\beta_1(x_{11}) = x_3, & \text{since } (11 + 3)(mod\,11) + 0 = 3 \text{ because of } 11 + 3 > 11
\end{array}$$

and for $i = 2, 3, 4$ the values of β_i are obtained similarly. Hence

$$\begin{array}{r|lllllllllll}
X_1 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\
\beta_1 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_1 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} & x_1 & x_2 & x_3
\end{array}$$

$$\begin{array}{r|llllll}
X_2 & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} & x_{17} \\
\beta_2 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_2 & x_{16} & x_{17} & x_{12} & x_{13} & x_{14} & x_{15}
\end{array}$$

$$\begin{array}{r|lllllllll}
X_3 & x_{18} & x_{19} & x_{20} & x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & x_{26} \\
\beta_3 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_3 & x_{25} & x_{26} & x_{18} & x_{19} & x_{20} & x_{21} & x_{22} & x_{23} & x_{24}
\end{array}$$

$$\begin{array}{r|lll}
X_4 & x_{27} & x_{28} & x_{29} \\
\beta_4 & \downarrow & \downarrow & \downarrow \\
X_4 & x_{29} & x_{27} & x_{28}
\end{array}$$

and therefore,

X	x_1	x_2	...	x_{11}	x_{12}	x_{13}	...	x_{17}	x_{18}	x_{19}	...	x_{26}	x_{27}	x_{28}	x_{29}
β	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
X	x_4	x_5	...	x_3	x_{16}	x_{17}	...	x_{15}	x_{25}	x_{26}	...	x_{24}	x_{29}	x_{27}	x_{28}

Step 5: Values of the characterization function α^{-1} are obtained as following list:

X	x_4	x_5	...	x_3	x_{16}	x_{17}	...	x_{15}	x_{25}	x_{26}	...	x_{24}	x_{29}	x_{27}	x_{28}
α^{-1}	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
U	ç	d	...	c	m	n	...	l	u	ü	...	t	z	v	y

Step 6: Values of the encryption function γ are obtained as following list:

U	a	b	c	ç	d	e	f	g	ğ	h	ı	i	j	k	l	m	n	...	v	y	z
γ	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$		$\downarrow$	$\downarrow$	$\downarrow$
U	ç	d	e	f	g	ğ	h	ı	a	b	c	m	n	o	i	j	k	...	z	v	y

In this example we showed that the plaintext "ankara" is encrypted as "çliçöç" according to the method which can be seen in Figure 1.

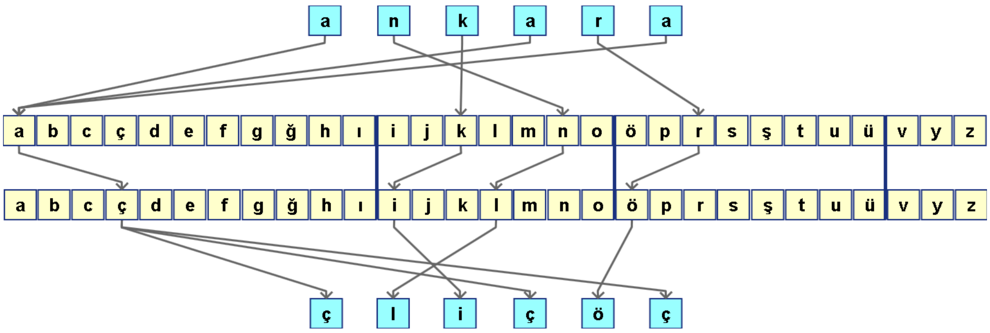

Figure 1: Encryption of "ankara" by FCEA

2.3 FC-cipher Decryption Algorithm

In this subsection, we give the algorithm of FC-cipher decryption method.

Assume that U is a character set and X is an indexed character set. Then, an algorithm of the FC-cipher decryption is set up as follows:

Algorithm of FC-cipher Decryption:

Step 1: Use the $\{X_i : i \in I_p\}$ and the module m_i for each X_i,
Step 2: Find the $\beta_i^{-1}(x_t)$ for $x_t \in X_i$, $t \in I_{a_i}$ and $i \in I_p$,
Step 3: Find the $\alpha^{-1}(x)$ for $x \in U$,
Step 4: Find the $\gamma^{-1}(x)$ for $x \in U$.

Example 20. Let us consider the result of Example 19 where

$$U = \{\text{ç, d, e, f, g, ğ,h, ı, a, b, c, m, n, i, j, k, l, u, ü, o, ö, p, r, s, ş, t, z, v, y}\}$$

and

$$X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, ..., x_{29}\}$$

Then,

Step 1 : In Example 19, for a_i $(i = 1, 2, 3, 4)$ the fragments of X, X_i, and was obtained as

$$\begin{array}{ll}
\text{for } a_1 = 11, & X_1 = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}\} \\
\text{for } a_2 = 6, & X_2 = \{x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}\} \\
\text{for } a_3 = 9, & X_3 = \{x_{18}, x_{19}, x_{20}, x_{21}, x_{22}, x_{23}, x_{24}, x_{25}, x_{26}\} \\
\text{for } a_4 = 3, & X_4 = \{x_{27}, x_{28}, x_{29}\}
\end{array}$$

and value of m_i $(i = 0, 1, 2, 3, 4)$ was also choosen as,

$$\begin{array}{ll}
\text{for } i = 0, & m_0 = 0 \\
\text{for } i = 1, & m_1 = (m_0 = 0) + (a_1 = 11) = 11 \\
\text{for } i = 2, & m_2 = (m_1 = 11) + (a_2 = 6) = 17 \\
\text{for } i = 3, & m_3 = (m_2 = 17) + (a_3 = 9) = 26 \\
\text{for } i = 4, & m_4 = (m_3 = 26) + (a_4 = 3) = 29
\end{array}$$

Step 2 : For $i = 1, 2, 3, 4$ values of the X_i-rotation function β_i^{-1} are obtained as follows. Here, we first obtain the values of β_1^{-1} as,

$$\begin{array}{ll}
\beta_1^{-1}(x_1) = x_9, & \text{since } 1 + m_1 - r_1 = 1 + 11 - 3 = 9 \text{ because of } 1 + 11 - 3 < 11 \\
\beta_1^{-1}(x_2) = x_{10}, & \text{since } 2 + m_1 - r_1 = 2 + 11 - 3 = 10 \text{ because of } 2 + 11 - 3 < 11 \\
\beta_1^{-1}(x_3) = x_{11}, & \text{since } 3 + m_1 - r_1 = 3 + 11 - 3 = 11 \text{ because of } 3 + 11 - 3 = 11 \\
\beta_1^{-1}(x_4) = x_1, & \text{since } (4 + 11 - 3)(mod\, 11) + 0 = 1 \text{ because of } 4 + 11 - 3 > 11 \\
\beta_1^{-1}(x_5) = x_2, & \text{since } (5 + 11 - 3)(mod\, 11) + 0 = 2 \text{ because of } 5 + 11 - 3 > 11 \\
\beta_1^{-1}(x_6) = x_3, & \text{since } (6 + 11 - 3)(mod\, 11) + 0 = 3 \text{ because of } 6 + 11 - 3 > 11 \\
\beta_1^{-1}(x_7) = x_4, & \text{since } (7 + 11 - 3)(mod\, 11) + 0 = 4 \text{ because of } 7 + 11 - 3 > 11 \\
\beta_1^{-1}(x_8) = x_5, & \text{since } (8 + 11 - 3)(mod\, 11) + 0 = 5 \text{ because of } 8 + 11 - 3 > 11 \\
\beta_1^{-1}(x_9) = x_6, & \text{since } (9 + 11 - 3)(mod\, 11) + 0 = 6 \text{ because of } 9 + 11 - 3 > 11 \\
\beta_1^{-1}(x_{10}) = x_7, & \text{since } (10 + 11 - 3)(mod\, 11) + 0 = 7 \text{ because of } 10 + 11 - 3 > 11 \\
\beta_1^{-1}(x_{11}) = x_8, & \text{since } (11 + 11 - 3)(mod\, 11) + 0 = 8 \text{ because of } 11 + 11 - 3 > 11
\end{array}$$

and for $i = 2, 3, 4$ the values of β_i^{-1} are obtained similarly. Hence,

$$\begin{array}{r|lllllllllll}
X_1 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} \\
\beta_1^{-1} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_1 & x_9 & x_{10} & x_{11} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8
\end{array}$$

$$\begin{array}{r|llllll}
X_2 & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} & x_{17} \\
\beta_2^{-1} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_2 & x_{14} & x_{15} & x_{16} & x_{17} & x_{12} & x_{13}
\end{array}$$

$$\begin{array}{r|lllllllll}
X_3 & x_{18} & x_{19} & x_{20} & x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & x_{26} \\
\beta_3^{-1} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
X_3 & x_{20} & x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & x_{26} & x_{18} & x_{19}
\end{array}$$

$$\begin{array}{r|lll}
X_4 & x_{27} & x_{28} & x_{29} \\
\beta_4^{-1} & \downarrow & \downarrow & \downarrow \\
X_4 & x_{29} & x_{27} & x_{28}
\end{array}$$

and therefore,

$$\begin{array}{r|llll|llll|llll|lll}
X & x_1 & x_2 & \dots & x_{11} & x_{12} & x_{13} & \dots & x_{17} & x_{18} & x_{19} & \dots & x_{26} & x_{27} & x_{28} & x_{29} \\
\beta^{-1} & \downarrow & \downarrow & & \downarrow & \downarrow & \downarrow & & \downarrow & \downarrow & \downarrow & & \downarrow & \downarrow & \downarrow & \downarrow \\
X & x_9 & x_{10} & \dots & x_8 & x_{14} & x_{15} & \dots & x_{13} & x_{20} & x_{21} & \dots & x_{19} & x_{29} & x_{27} & x_{28}
\end{array}$$

Step 3 : Values of the characterization function α^{-1} are obtained as following list:

X	x_9	x_{10}	...	x_8	x_{14}	x_{15}	...	x_{13}	x_{20}	x_{21}	...	x_{19}	x_{29}	x_{27}	x_{28}
α^{-1}	↓	↓		↓	↓	↓		↓	↓	↓		↓	↓	↓	↓
U	a	b	...	ı	i	j	...	n	o	ö	...	ü	v	y	z

Step 4 : Values of the decryption function γ^{-1} are obtained as following list:

U	ç	d	e	f	g	ğ	h	ı	a	b	c	m	n	o	i	j	k	...	z	v	y
γ^{-1}	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓		↓	↓	↓
U	a	b	c	ç	d	e	f	g	ğ	h	ı	i	j	k	l	m	n	...	v	y	z

In this example we showed that the ciphertext ”çliçöç” is decrypted as ”ankara”.

3 Fragmented Polyalphabetic Cipher

In this section, we define a new cipher method which is called fragmented polyalphabetic cipher (FP-cipher) based on the FC-cipher. In the FP-cipher, the plaintext is encrypted by multiple encryption alphabets which are obtained by using the FC-cipher.

From now on, we use $k \in \mathbb{N}$ as a number of encrypted alphabets that are obtained by using the FC-cipher.

3.1 Mathematical Model of FP-cipher

In this subsection, we first give a mathematical model of the FP-cipher. We then write an algorithm of the FP-cipher to make a computer program.

Definition 21. Let U be a character set and $\gamma_i : U \to U$ be an encryption function for all $i \in I_k$. If all of the characters in the plaintext are indexed as $y_1 y_2 ... y_q$ for $q \in \mathbb{N}$, then a k-**multiple encryption function** on U is defined by

$$\delta_k : U \to U, \quad \delta_k(y_t) = \begin{cases} \gamma_i(y_t), & t \equiv i (mod\, k) \\ \gamma_k(y_t), & t \equiv 0 (mod\, k) \end{cases}$$

for all $t \in I_q$ and $i \in I_k$.

Definition 22. Let U be a character set and $\gamma_i^{-1} : U \to U$ be an encryption function for all $i \in I_k$. If all of the characters in the ciphertext are indexed as $s_1 s_2 ... s_q$ for $q \in \mathbb{N}$, then a k-**multiple decryption function** on U is defined by

$$\delta_k^{-1} : U \to U, \quad \delta_k^{-1}(s_t) = \begin{cases} \gamma_i^{-1}(s_t), & t \equiv i (mod\, k) \\ \gamma_k^{-1}(s_t), & t \equiv 0 (mod\, k) \end{cases}$$

for all $t \in I_q$ and $i \in I_k$.

The Definitions 21 and 22 are demonstrated in Figure 2.

Definition 23. Let U be a character set and P_i be a fragment key, R_i be a rotation key for all $i \in I_k$. Then a k-**multiple fragment key** and k-**multiple rotation key** are defined as follow, respectively

$$P_k = (P_1, P_2, ..., P_k), \quad R_k = (R_1, R_2, ..., R_k).$$

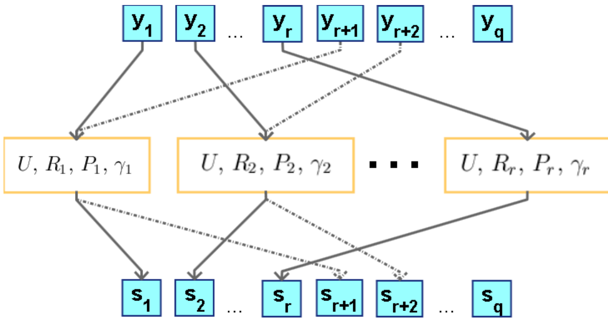

Figure 2: FP-cipher

Definition 24. Let (U, P_i, R_i, γ_i) be an FC-cipher encryption and $(U, P_i, R_i, \gamma_i^{-1})$ be an FC-cipher decryption on U for all $i \in I_k$. Then, each five tuple

$$(U, k, P_k, R_k, \delta_k), \quad (U, k, P_k, R_k, \delta_k^{-1})$$

is called an **FP-cipher encryption** and **FP-cipher decryption** on U, respectively.

3.2 FP-cipher Encryption Algorithm

In this subsection, we give an algorithm of the FP-cipher encryption method.

Assume that all of characters in a plaintext are be indexed as $y_1y_2...y_q$ and k be a number of encrypted alphabets that are obtained by using the FC-cipher. Then, an algorithm of the FP-cipher encryption is set up as follow:

Algorithm of FP-cipher Encryption

Step 1 : Find the $P_k = (P_1, P_2, ..., P_k)$ and $R_k = (R_1, R_2, ..., R_k)$,
Step 2 : Find the values of γ_i for $i \in I_k$,
Step 3 : Find the values $\delta(y_t)$ for all $t \in I_q$.

Example 25. Let

$$U = \{\text{a, b, c, ç, d, e, f, g, ğ, h, ı, i, j, k, l, m, n, o, ö, p, r, s, ş, t, u, ü, v, y, z}\}$$

be a character set and "ankara" be a plaintext. Assume that this plaintext is indexed as $y_1y_2y_3y_4y_5y_6$ and encrypted by 3-FP-cipher encryption. Then,

Step 1 : The P_i and R_i can be obtained by using the FC-cipher as follow,

i	P_i	R_i
1	(11,6,10,2)	(4,2,5,1)
2	(10,9,5,5)	(3,4,2,3)
3	(5,6,4,10,4)	(2,2,1,3,1)

Step 2 : For $i = 1, 2, 3$, the values $\gamma_i(x)$ are obtained as follow,

x	a	b	c	ç	d	e	f	g	ğ	h	ı	i	j	k	l
$\gamma_1(x)$	d	e	f	g	ğ	h	ı	a	b	c	ç	m	n	i	j
$\gamma_2(x)$	ç	d	e	f	g	ğ	h	a	b	c	l	m	n	o	ö
$\gamma_3(x)$	c	ç	d	a	b	g	ğ	h	ı	e	f	j	k	l	i

	m	n	o	ö	p	r	s	ş	t	u	ü	v	y	z
	k	l	ş	t	u	ü	v	o	ö	p	r	s	z	y
	ı	i	j	k	s	ş	t	p	r	y	z	u	ü	v
	ö	p	r	s	ş	t	u	m	n	o	v	y	z	ü

Step 3 : For all $t \in I_6$, the values $\delta(y_t)$ are obtained as follow,

$$\begin{array}{llll}
\text{for } i=1, & \delta(y_1)=\gamma_1(a)=\text{d} & \text{because of} & 1 \equiv 1(mod\,3) \\
\text{for } i=2, & \delta(y_2)=\gamma_2(n)=\text{i} & \text{because of} & 2 \equiv 2(mod\,3) \\
\text{for } i=3, & \delta(y_3)=\gamma_3(k)=\text{l} & \text{because of} & 3 \equiv 0(mod\,3) \\
\text{for } i=4, & \delta(y_4)=\gamma_1(a)=\text{d} & \text{because of} & 4 \equiv 1(mod\,3) \\
\text{for } i=5, & \delta(y_5)=\gamma_2(r)=\text{ş} & \text{because of} & 5 \equiv 2(mod\,3) \\
\text{for } i=6, & \delta(y_6)=\gamma_3(a)=\text{c} & \text{because of} & 6 \equiv 0(mod\,3)
\end{array}$$

Therefore,

y_t	a	n	k	a	r	a
$\delta(y_t)$	d	i	l	d	ş	c

This example is demonstrated in Figure 3.

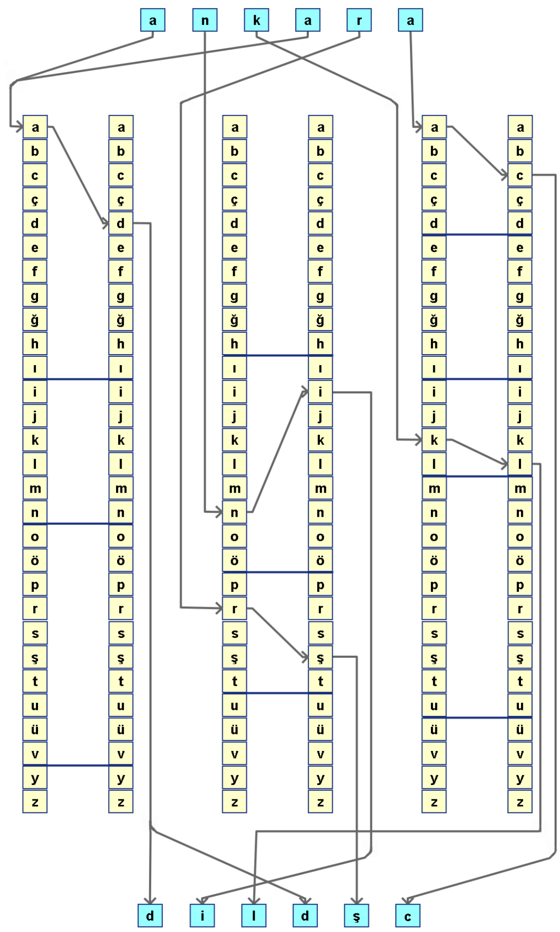

Figure 3: Encrypting the word of "ankara" by FP-cipher

3.3 FP-cipher Decryption Algorithm

In this subsection, we give an algorithm of the FP-cipher decryption method.

Assume that all of characters in a ciphertext are indexed as $s_1 s_2 ... s_q$. Here, we have to use the same values of P_i and R_i are obtained in the encryption. Then, an algorithm of the FP-cipher decryption is set up as follow.

Algorithm of FP-cipher Decryption

Step 1 : Find the values of γ_i^{-1} for $i \in I_k$,
Step 2 : Find the values $\delta^{-1}(s_t)$ for all $t \in I_q$.

Example 26. Let us consider Example 25 where "ankara" was encrypted as "dildşc". Now, the cipher text "dildşc" is decrypted. Here, we have to use same values of P_i and R_i in Example 25. Assume that this ciphertext is indexed as $s_1 s_2 s_3 s_4 s_5 s_6$ and decrypted by 3-FP-cipher decryption. Then,

Step 1 : For $i = 1, 2, 3$, the values $\gamma_i^{-1}(x)$ are obtained as follow,

x	a	b	c	ç	d	e	f	g	ğ	h	ı	i	j	k	l	m	n	o	ö
$\gamma_1^{-1}(x)$	d	e	f	g	ğ	h	ı	a	b	c	ç	m	n	i	j	k	l	ş	t
$\gamma_2^{-1}(x)$	ç	d	e	f	g	ğ	h	a	b	c	l	m	n	o	ö	ı	i	j	k
$\gamma_3^{-1}(x)$	c	ç	d	a	b	g	ğ	h	ı	e	f	j	k	l	i	ö	p	r	s

$\gamma_1^{-1}(x)$	u	ü	v	o	ö	p	r	s	z	y
$\gamma_2^{-1}(x)$	s	ş	t	p	r	y	z	u	ü	v
$\gamma_3^{-1}(x)$	ş	t	u	m	n	o	v	y	z	ü
x	p	r	s	ş	t	u	ü	v	y	z

Step 3 : For all $t \in I_6$, the values $\delta^{-1}(s_t)$ are obtained as follow,

$$
\begin{array}{llll}
\text{for } i = 1, & \delta^{-1}(s_1) = \gamma_{(1)}^{-1}(s_1) = \text{a} & \text{because of} & 1 \equiv 1 (mod\, 3) \\
\text{for } i = 2, & \delta^{-1}(s_2) = \gamma_{(2)}^{-1}(s_2) = \text{n} & \text{because of} & 2 \equiv 2 (mod\, 3) \\
\text{for } i = 3, & \delta^{-1}(s_3) = \gamma_{(3)}^{-1}(s_3) = \text{k} & \text{because of} & 3 \equiv 0 (mod\, 3) \\
\text{for } i = 4, & \delta^{-1}(s_4) = \gamma_{(1)}^{-1}(s_4) = \text{a} & \text{because of} & 4 \equiv 1 (mod\, 3) \\
\text{for } i = 5, & \delta^{-1}(s_5) = \gamma_{(2)}^{-1}(s_5) = \text{r} & \text{because of} & 5 \equiv 2 (mod\, 3) \\
\text{for } i = 6, & \delta^{-1}(s_6) = \gamma_{(3)}^{-1}(s_6) = \text{a} & \text{because of} & 6 \equiv 0 (mod\, 3)
\end{array}
$$

Therefore,

s_t	d	i	l	d	ş	c
$\delta^{-1}(s_t)$	a	n	k	a	r	a

3.4 FP-cipher Program Codes

In this subsection, FP-cipher is programmed by using C# as follows:

```
private void btn_alfabe_olustur_Click(object sender, EventArgs e)
{
    //txtanahtar.Text = "";
    if (rdsifre.Checked)
```

```
{
    txtanahtar.Text = "";
    //anahtar oluşturma
    for (int i = 0; i < trczorluk.Value; i++)
    {
        System.Threading.Thread.Sleep(500);
        string anahtar = _anahtar(alf_metin().Length);
        txtanahtar.Text += anahtar.Substring(0,
                                    anahtar.Length - 1) + "*";
    }
}
//sanal matris oluşturma
DataTable matris = new DataTable();
for (int i = 0; i <alf_metin().Length; i++)
{
    matris.Columns.Add(alf_metin()[i]);
}
string[] key = txtanahtar.Text.Substring(0,
                        txtanahtar.Text.Length - 1).Split('*');
for (int i = 0; i < key.Length; i++)
{
    int alfabe_sayac = 1;
    string[] parca = key[i].Substring(0, key[i].Length).
                                        ToString().Split('-');
    string[][] U = new string[parca.Length][];//(Açık U)
    string[][] SU = new string[parca.Length][];//(Şifreli U)
    //alfabenin kümelere bölnmesi
    for (int j = 0; j < parca.Length; j++)
    {
        string[] parca_a = parca[j].ToString().Split(',');
        int P = int.Parse(parca_a[0]); //parça anahtarı
        U[j] = new string[alf_metin().Length+1];
        SU[j] = new string[alf_metin().Length+1];
        for (int x = 0; x < P; x++)
        {
            U[j][alfabe_sayac] = alf_metin()[alfabe_sayac-1];
            alfabe_sayac++;
        }
    }
    //alfabenin şifrelenmesi
    int m = 0;
    int indis = 1;
    for (int j = 0; j < U.Length; j++)
    {
        string[] parca_a = parca[j].ToString().Split(',');
        int P = int.Parse(parca_a[0]); //parça anahtarı
        int R = int.Parse(parca_a[1]); //öteleme anahtarı
```

```
            m += P;
            for (int x = 0; x < P; x++)
            {
                int k = 0;
                if ((indis+R)<=m) //öteleme fonksiyonu
                {
                    k = indis + R;
                }
                else if ((indis + R) > m)
                {
                    k = ((indis + R) % m)+(m-P);
                }
                SU[j][indis] = U[j][k];
                indis++;
            }
        }
        //matrise değerlerin eklenmsi
        matris.Rows.Add();
        int alsayac = 0;
        for (int j = 0; j < U.Length; j++)
        {
            string[] parca_a = parca[j].ToString().Split(',');
            int P = int.Parse(parca_a[0]); //parça anahtarı
            for (int x = 0; x < P; x++)
            {
                matris.Rows[i][alsayac] = SU[j][alsayac + 1];
                alsayac++;
            }
        }
        dataGridView1.DataSource = matris;
    }
}
```

4 Conclusions

In this work, we defined the FP-cipher that is a generalization of the Vigenere cipher. The Vigenere cipher is a type of polyalphabetic cipher in which different shift ciphers are used to encryption. In the FP-cipher, plaintext is encrypted by multiple encryption alphabets which are obtained by using the FC-cipher. Therefore, the key space of FP-cipher cipher has more possibility then the Vigenere cipher. We then constructed a mathematical modeling and make a computer program of the method.

References

[1] Y. Aydoğan, N. Çağman, I. Şimşek, Fragmented Caesar Cipher, Journal of New Theory 14 (2016) 46-57.

[2] Y. Aydoğan, Multi Fragmented Caesar Cipher Method and its Applications (in Turkish), MSc Thesis, Gaziosmanpasa University, Graduate School of Natural and Applied Science, 2014.

[3] J. Hoffstein, J. Pipher and J. H. Silverman, An Introduction to Mathematical Cryptograph, Springer-Newyork, 2008.

[4] C. Paar, J. Pelzl, Understanding Cryptograph, Springer-Verlag, 2010.

[5] A. Sinkov, Elementary Cryptanalysis - A Mathematical Approach, New Mathematical Library, No. 22, Mathematical Association of America, 1966.