Republic of Iraq
Ministry of Higher Education
and Scientific Research
University of Babylon
College of Education for Pure Sciences

# *The Triple Vertex Cycle Graph for Encryption Schemes*

**A Thesis**

**Submitted to the Council of the College of Education for Pure Sciences in University of Babylon as a Partial Fulfillment of the Requirements for the Degree of Master in Education / Mathematics**

**By**

**Manal Hatif Kadhim Sayed**

**Supervised by**

**Asst. Prof. Dr.**

**Ruma Kareem K. Ajeena**

2022 A.D. 1444 A.H

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

(يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَات)

صدق الله العلي العظيم

سورة المجادلة     الآية 11

# Supervisor's Certification

I certify that the thesis entitled **“The Triple Vertex Cycle Graph for Encryption Schemes"** by**"Manal Hatief Kadhim Saeyd*”*** has been prepared under my supervision in Babylon University/ College of Education for Pure Sciences as a partial requirement for the degree of Master in Education / Mathematics.

***Signature :***

***Name : Dr. Ruma Kareem K. Ajeena***

***Title: Assistant Professor***

***Date: / / 2022***

***In view of the available recommendation, I forward this thesis for debate by the examining committee.***

***Signature:***

***Name: Dr. Azal Jaafar Musa***

***Head of Mathematics Department***

***Title: Assistant Professor***

***Date: / / 2022***

# Certification of linguistic Expert

I certify that I have read this thesis entitled **“The Triple Vertex Cycle Graph for Encryption Schemes”** and corrected its grammatical mistakes; therefore, it has qualified for debate.

**Signature:**

**Name: Dr.**

**Title:**

**Address:**

**Date: / /** 2022

# Certification of Scientific Expert

I certify that I have read the scientific content of this thesis **“The Triple Vertex Cycle Graph for Encryption Schemes”** and I have approved this dissertation is qualified for debate*.*

**Signature:**

**Name:** ***Dr. Basim Karim Kadhir Albuohimad***

**Title:** ***Assistant Professor***

**Date:**

# Certification of Scientific Expert

I certify that I have read the scientific content of this thesis **“The Triple Vertex Cycle Graph for Encryption Schemes”** and I have approved this dissertation is qualified for debate*.*

**Signature:**

**Name:** ***Dr. Najlae Falah Hameed***

**Title:** ***Assistant Professor***

**Date:**

# Examining Committee Certification

We certify that we have read the thesis entitled **" The Triple Vertex Cycle Graph for Encryption Schemes "** by**" Manal Hatif Kadhim Sayed**"

**"** and as a committee examined the student in its contents and, according to our opinion, it is accepted as a thesis for the degree of Master in Education / Mathematics.

**Signature:**

**Name: Dr. Hassan Rashed Yassein Ali Omran**

**Title: Professor**

**Date:** / / 2022

**Chairman**

**Signature:**

**Name:Dr. Ahmed Abed**

**Title: Professor**

**Date:** / / 2022

**Member**

**Signature:**

**Name: Dr. Ahmed Sabah Ahmed**

**Title:** ***Assistant Professor***

**Date:** / / 2022

**Member /**

**Signature:**

**Name*****: Ruma Kareem K. Ajeena***

**Title:**

**Date:** / / 2022

**Member / Supervisor**

# شكر وتقدير

من الصعب البوح بكلمة الشكر لأنـها لا تحد عطاء اساتذتي الذين تعلمت على ايديهم واخص فيهم بالذكر الجميل والثناء الوفير الى الدكتورة المشرفة رومى كريم خضر فجزاها الله خير جزاء المحسنين

# الإهداء

**إلهي لا يطيب الليل إلا بشكرك ولا يطيب النهار إلا بطاعتك.. ولا تطيب<br>اللحظات إلا بذكرك. ولا تطيب الآخرة إلا بعفوك.. ولا تطيب الجنة إلا برؤيتك<br>(الله جل جلاله)**

**الى من بلغ الرسالة وأدى الأمانة ..ونصح الأمة ..الى نبي الرحمة ونور<br>العالمين..<br>(سيدنا محمد صلى الله عليه واله وسلم)**

**الى من كلله الله بالهيبة والوقار ..الى من علمني العطاء بدون انتظار ..الى<br>من أحمل أسمه<br>بكل افتخار ..ارجو من الله أن يمد في عمرك لترى ثماراً قد حان قطافها بعد<br>طول انتظار وستبقى كلماتك نجوم أهتدي بها اليوم وفي الغد والى الأبد..<br>(والدي العزيز)**

**الى ملاكي في الحياة ..الى معنى الحب والى معنى الحنان والتفاني ..الى<br>بسمة الحياة وسر الوجود ..الى من كان دعاؤها سر نجاحي وحنانها بلسم جراحي<br>إلى أغلى الحبايب<br>(أمي الحبيبة)**

**الى من بهم أكبر وعليهم أعتمد ..الى شمعة متقدة تنير ظلمة حياتي..<br>(اخوتي)**

**الى من بوجودهم أكتسب قوة ومحبة لا حدود لها ..الى من عرفت معهم<br>معنى الحياة الى من حافظ علينا، الى من وقف الى جانبنا...<br>(زوجي وابني)**

# Abstract

In recent years, many researchers, including Dr. Ruma Ajeena ISD soft graphing method (SG-ISD) through the use of connected sub-graphs of a simple indirect graph formed a soft graph (F, A) generated by a specific value function F. Because of the development of technology, it became easier to access the original text A new version of the polyalphabetic coding scheme (PES) has been proposed based on the triple head cycle graph (TVCG). TVCG is given as a new definition. TVCG is a key point in this work to modify PES and increase the level of security compared to previous symmetric cipher schemes. The plaintext ciphertext, in the proposed PES, is sent to the receiving entity as TVCG. Based on the Cn-cycle graph, TVCG is generated based on the encoded characters of the plaintext bits and secret key rules. The proposed TVCG-PES case study was presented as a new experimental result. Security issues for TVCG-PES are identified. TVCG-PES is a new vision for more secure communications.

On the other hand, three-headed graph (TVDG) and three-headed vector graph (TVCDG) are also defined as new concepts. New versions of symmetric coding schemes (SE) are designed based on TVCDG to give new versions of coding schemes. These graphs are also applied to polyalphabetic, affine and hill schemes that use EAVs and ASCII values to represent plaintext or ciphertext characters. In the proposed schemes, the text encoded as TVCG is transmitted to the receiver by the transmitter.

The new experimental results of the proposed TVCG-SE and TVCDG-SE schemes are discussed. Security considerations for TVPG-SE and TVCDG-SE systems have been identified. The comparison of the proposed TVCG-SE and TVCDG-SE schematics with the original SE schematics is explained. The proposed TVCG-SE and TVCDG-SE schemes are safer new insights compared to previous insights...

# List of Contents

## List of Figures

## List of Tables

# Chapter One

# Introduction

## 1.1 General Introduction

In a communication network, cryptography [1] is the science that is used for protecting the information by converting it for secure communications. Modern technology requires the transmitted information through efficient encryption schemes. These schemes are designed based on some mathematical problems, ones of them those are solved using the basic concepts of graph theory [2] by creating new models.

The connection between graph theory and the encryption schemes is done in this thesis. New definitions have been proposed based on the three vertices of a graph which are called triple vertex graph (TVG) and triple vertex cycle graph (TVCG) as the main point to design new versions of symmetric encryption (SE) schemes based on the English alphabetic values (EAVs) and ASCII values.

On proposed ES schemes, the ciphertexts of the plaintexts are sent to receiver entity as the TVC graph. Two study cases of the proposed TVCG-SE schemes are presented as new experimental results. The security issues of the proposed TVCG-SE schemes are determined. On the other hand, the triple vertex digraph (TVDG) and triple vertex cycle digraph (TVCDG) are defined. Some SE schemes are created using the TVCDG. The TVCG-SE and TVCDG-SE schemes consider as new insights for more secure communications.

## 1.2 Previous Studies

Several researchers employed the concepts of graph theory to modify the encryption schemes and make them more secure for communications. In 2001, Ustimenko [3] presented a study used the graphs as important tools for symmetric encryption. This study treated the vertices of a graph as messages and walks of a certain length as encryption tools.

In 2002, Ustimenko [4], presented a basic idea to use the vertices of a graph as the messages and arcs of a certain length as the encryption tools. His study introduced the quality of an encryption in the case of graphs of high girth.

In 2007, Mittenthal [6], proposed used the directed graphs and their applications to encryption schemes through a method to find the complete Latin squares.

In 2012, Selvakumar and Gupta, [9], proposed their study based on the fundamental circuits and cut-sets in cryptography. They presented an algorithm for encrypting and decrypting through using the connected graphs. In the same year, Yamuna, et al., [10], introduced their study based on Hamilton path properties. They encrypt the data using the Hamilton path, and the complete graph for more secure method.

Also, in 2013, Cheema, et al., [13], proposed a network security using the graph theory. Furthermore, in 2013, Yamuna, et al., [14], introduced the encryption of a binary string using the music notes and graph theory. They proposed two phases of an encryption algorithm for transforming plaintext into musical notes using graph theory.

In 2014, Al Etaiwi, [15], used graph theory to encrypt and decrypt the data. He used graph theory properties such as complete graph and minimum spanning tree.

In 2015, Femina and Antony [16], introduced a study of data encryption standard using graph theory. Her study used the Hamilton’s in 2019, Debajit Sensarma and Sarma [24], introduced a cryptosystem for protecting valuable information. The method is purely based on the properties of matrices, while the second method is based on graphical code. In the same year, Sherin and Maheswar [25], introduced encoding the graph using an instant instanity puzzle and decoding with a Hamiltonian cycle. In their study, they created a puzzle with six cubes and discussed the 5 properties of the Hamiltonian graph. They used an asymmetric key to determine whether the graph contains a Hamiltonian path leading to a Hamiltonian cycle. The receiver computes edge labeling technique to determine the decoded message. Furthermore, in 2019, Sherin and Maheswari [26], proposed an encryption and decryption process using the edge magic labeling. They used edge magic labeling graph to encrypt a message and the longest cycle path technique to decrypt the message by inverse matrix multiplication.

In 2020, Maheswari, et al., [27], proposed a method for secret coding technique on two star graphs. They generalized the concept of super mean labeling on two star graphs and developed a technique for coding secret messages using two star graphs through the super mean labels. Furthermore, Beaula and Venugopal [28] introduced a cryptosystem based on the double vertex graph. For encryption and decryption, an edge labeling of a double vertex graph is used. Also in 2020, Akl, [29], proposed a study to encrypt graphs. A message is considered as a graph and its edge weights are the information. Also, in 2020, Santoso, et al., [30], presented a modified Caesar cryptosystem using the binary vertices colouring.

In 2021, Perera and Wijesiri [31] used a matrix as the secret key, which added more security to the cryptosystem. It converted the plaintext into several graphs and represented these graphs in their matrix form. In 2021, Adnan, et al., [32], proposed securing text messages using the graph theory and steganography. Also, in 2021, Sherin, et al., [33], examined the encryption technique by defined functions on the face wheel graph. As well as, Baizhu, et al in 2021 [34], presented the special corona graphs and bipartite graphs along with some algebraic properties.

In 2021 also, Ruma Ajeena [35] presented the soft graphic ISD (SG-ISD) method through employing the connected sub-graphs of undirected simple graph, which formed a soft graph $(F, A)$ generated by a set valued function $F$. Also, in the same year, Karrar Aljamaly and Ruma Ajeena [36, 37] have presented two studies, first one proposed a new public key cryptosystem based on undirected complete graph (UCG). A new graph based on the scalar multiplication operation on elliptic curve defined over a prime field is defined in second one which is considered as a key point to design a new version of an asymmetric encryption scheme.

## 1.3 Statement of the Problem

In this work, the triple vertex graph (TVG) and triple vertex cycle graph (TVCG) have been defined as new concepts of the graph theory. These concepts consider as the key points to design alternative versions of symmetric encryption (SE) schemes and to increase the level of security compared to previous ES schemes. The TVCG first is used for sending the ciphertext of the plaintext that is converted into numbers based on the English alphabetic values and second based on the ASCII values and decrypt it to recover the original a plaintext. Also, this work includes other symmetric encryption schemes that are modified used TVCG, these are the polyalphabetic, affine and hill schemas. Furthermore, triple vertex digraph (TVDG) and triple vertex path digraph (TVCDG) are also defined as new concepts. The TVPDG are employed for some symmetric encryption schemes as well. On the proposed TVCG-SE and TVCDG-SE schemes, the security considerations are determined. More secure communications with using the TVCG-SE and TVCDG-SE schemes are investigated, so they consider as new insights more secure for communications.

## 1.4 Thesis Objectives

This work proposed new graphs called TV, TVC, TVD and TVDPC graphs. These graphs are employed to design new versions of the symmetric encryption schemes which are more secure communications in compare with original ones.

## 1.5 Thesis Structure

The outline of this study is as follows:

- **chapter 1:** This chapter contains the introduction, previous studies, problem statement, Thesis Objectives, and Thesis organization.
- **Chapter 2:** The first part includes some concepts of graph theory through definitions and examples. The second part includes an explanation of the introduction to cryptography and some kinds of the symmetric encryption schemes.
- **Chapter 3:** It includes the proposition of new definitions of special graphs that are named by TVG and TVCG. Also, it includes the new versions of symmetric encryption schemes, TVPG-SE, that are employed the TVCG graph based on EAVs and ASCII Tables. This chapter also proposed other new versions of the polyalphabetic, affine, and hill encryption schemes that are used the TVCG and

depended on the EAVs and ASCII values. The security considerations for TVCG-SE schemes are determined.

- **Chapter 4:** It includes new definitions of special digraphs that are called TVDG and TVCDG. Also, it includes the new versions of symmetric encryption schemes, TVCDG-SE, that are employed the TVCDG graph based on EAVs and ASCII Tables. This chapter also proposed other new versions of the polyalphabetic, affine, and hill encryption schemes that are used the TVCDG and depended on the EAVs and ASCII values. The security considerations for TVCDG-SE schemes are determined.
- **Chapter 5:** Draws the conclusions and future works.

# Chapter Two

# Mathematical Background

## 2.1 Introduction

This chapter presents some concepts of graph theory and explains them by examples. Also, it discusses the important concepts in cryptography and some symmetric encryption schemes as follows.

## 2.2 Basic Concepts of Graph Theory

In this section, some important concepts of the graph theory are discussed as follows.

**Definition 2.2.1. (Graph)** [38]. The graph $G=(V(G),E(G)$ or simply $G=(V,E)$ is collection of non-empty vertex set $V(G)$ and edge set $E(G)$ can be an empty set. Each element of $V(G)$ is called a vertex of $G$ and each element $(u,v) \in E(G)$ is an unordered pair called an edge of $G$ where $(u,v) \in V(G)$.

**Example 2.2.1** A graph with two sets vertex set V={ a, b, c, d, k, m }

and E={ $e_1$ , $e_2$ , $e_3$ , $e_4$ , $e_5$ , $e_6$ }.

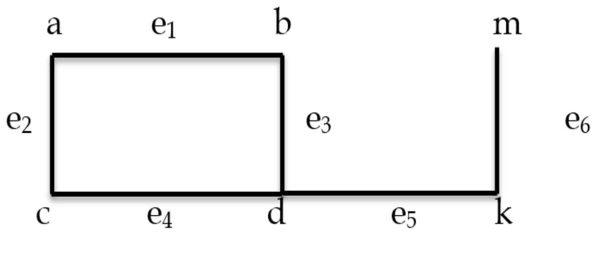


Figure 2.1. G(6.6)

**Proposition 2.2.2. (Adjacent)** [38]. If $e_1=ab$ is an edge of $G$ then $a$ and $b$ are adjacent to each other as shown in Figure (2.1).

**Definition 2.2.3. (Order and Size of *G* )** [39]. A graph *G* is said to be of order *n* if $|V(G)|=n$ and, of  size *m* if $|E(G)|=m$.

Example (2.2.3) , in a graph *G* in Figure (2.2)

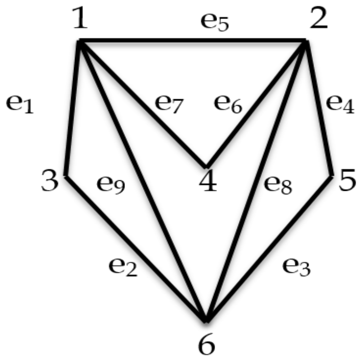


Figure 2.2. A graph *G* with $|V(G)|=6$ and $|E(G)|=9$.

**Definition 2.2.4. (Edge)** [38]. Two vertices *u* and *v* are adjacent if they are connected by an edge, in the word $(u,v)$ is an edge.

**Definition2.2.4. (Parallel)** [38]. In a graph if edges have the same end vertices, then these edges are called parallel whereas, a loop can be formed by edge that has the same begin vertex and end vertex. See Example (2.3), the parallel edges are $e_6$ and $e_2$.

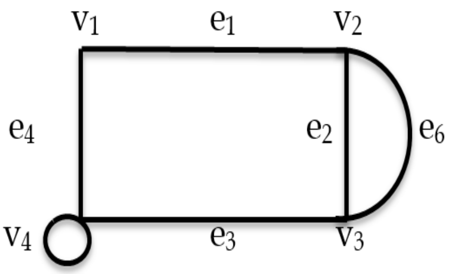


Figure 2.3. A graph *G* with parallel edges.

**Definition 2.2.6. (Simple Graph)** [39]. A graph is simple if it has no parallel edges or loops as shown in Figure (2.4).

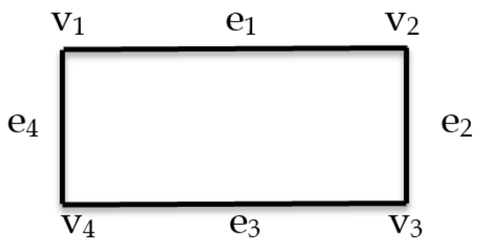
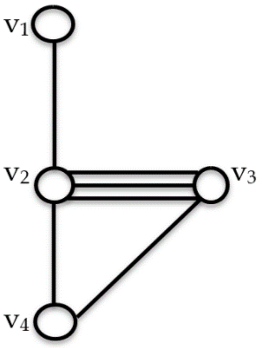


Figure 2.4. A simple graph.

**Definition 2.2.7. (Multi Graph)** [38]. A graph that has multiple edge between a pair of vertices is called Multi Graph.



. Figure 2.5. A simple graph

**Definition 2.2.8. (Complete Graph)** [39]. A graph in which each pair of distinct vertices are adjacent is called a complete graph. A complete graph with *n* vertices is denoted by $K_n$.

**Example 2.2.5.** A complete graph $K_5$ as shown in Figure (2.5).

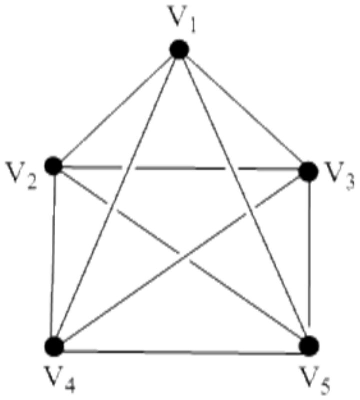


Figure 2.6. A complete graph $K_5$.

**Definition 2.2.9.** (**Bipartite Graph)** [38]. A graph $G$ is called a bipartite graph if the vertex set $V$ of $G$ can be partitioned into two disjoint nonempty sets $V_1$ and $V_2$, both of which are independent.

**Example 2.2.5.** A bipartite graph as shown in Figure (2.6)

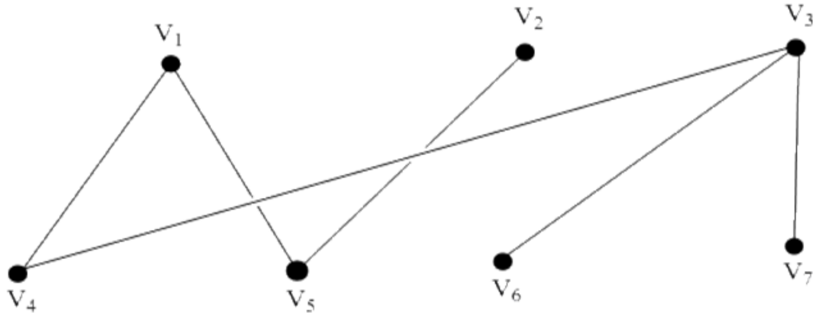


Figure 2.7. A bipartite graph.

$v=\{ v_1,v_2 ,v_3,v_4,v_5,v_6 ,v_7\}$

$U=\{ v_1,v_2 ,v_3\}$

$T=\{v_4,v_5,v_6 ,v_7\}$

No edge between any vertex $U$ and $T$. And $v= U \cup T$.

**Definition 2.2.10. (Complete Bipartite Graph)** [38]. A complete bipartite graph where the two partite sets contains 3 and 4 vertices, respectively. This graph is denoted by $K_{3,4}$. In general, a complete bipartite graph is denoted by $K_{m,n}$ if its two partite sets contain $m$ and $n$ vertices, respectively. One can easily see that $K_{m,n}$ contains $m \times n$ edges.

**Example 2.2.6.** A complete bipartite graph as shown in Figure (2.7).

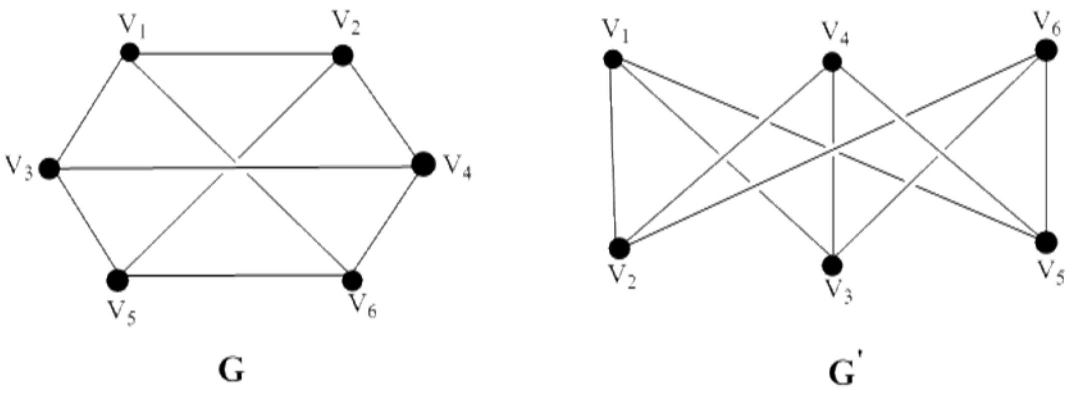


Figure 2.8. A complete bipartite graph.

**Definition 2.2.11**A cycle graph is a number of vertices connected to each other, and all vertices are of the second degree and the number of vertices n≥3

**Example 2.2.7.** A cycle graph as shown in Figure (2.8).

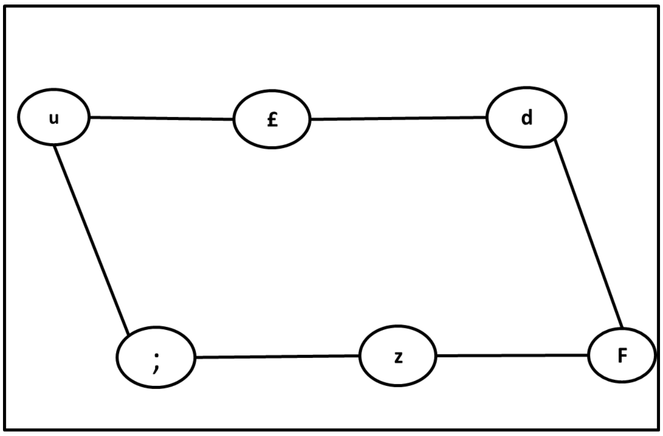


Figure 2.9. A cycle graph.

**Definition 2.2.12. (Walk Graph)** [38]**.** Let *G* be a graph, a walk in *G* is a nonempty list $W=V_0,E_1, V_1,E_2,\ldots,V_{f-1},E_f$ . whose element are alternately vertices and edge of G where for $1 \leq i \leq f,$ the edge $E_i$ has end vertices $V_{i-1}$ and $V_i$ . The vertices $V_0$ and $V_f$ are called the end vertices of *W*. If the end vertices of a walk *W* of a graph *G* are *u* and *v* respectively, *W* is also called an *u, v*-walk in *G*.

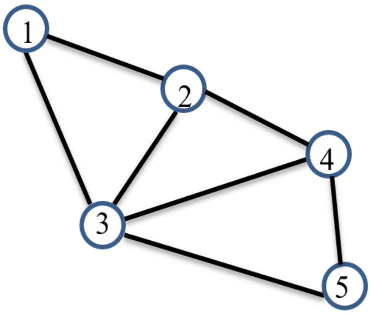


Figure 2.10. A walk is 1-2-3-4-5-3.

**Definition 2.2.13. (Trial Graph)** [39]**.** A graph *G* is a walk in *G* with no repeated edges. That is, in a trail an edge cannot appear more than once.

**Example 2.2.9.** A trial graph

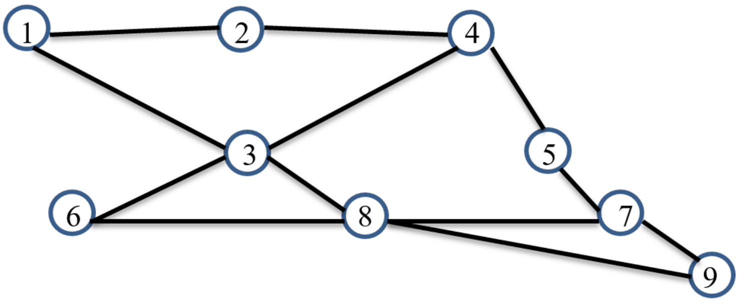


Figure 2.11. A trial graph is 1-3-8-6-3-2.

A closed trail is 1-3-8-6-3-2-1.

**Definition 2.2.14. (Path Graph)** [38]**.** A path graph is a graph *G* that contains a list of vertices $V_1, V_2, \ldots, V_p$ of *G* such that for $1 \le i \le p\text{-}1$, there is an edge ($V_i$ , $V_{i+1}$) in *G* and these are the only edges in *G* the two vertices $V_1$ and $V_2$ are called the end-vertices of *G*, path with *n* vertices $P_n$

**Definition 2.2.15. (Open Path)** [39]**.** An open path in which the first and last vertices are distinct.

A path graph is {u-a-e , d, v} as shown in Figure () vertices and 8 edges.

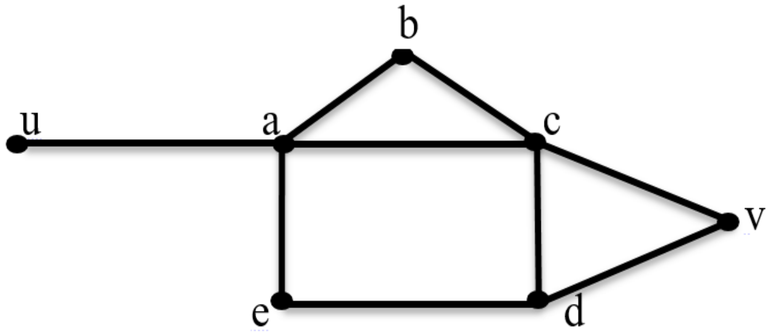


Figure 2.12. A path graph.

**Remark 2.2.1.** A path is with no repeated edje and vertices.

**Remark 2.2.2.** A path is a trial but the converse is not true.

## 2.3 Introduction to Cryptography

Some basic concepts related to cryptography are discussed as follows.

### 2.3.1 Basic Concepts

In this section, some important definitions are presented as follows.

**Definition 2.3.1.1.** [1] Cryptography is the design and analysis of mathematical techniques that enable secure communications in the presence of adversaries.

**Definition 2.3.1.2.** [1] Cryptosystem. A cryptographic system is specifically a set of methods (algorithms) for computing (implementing) the encryption and decryption.

**Definition 2.3.1.3.** [1] Cryptanalysis is the study of analyzing cryptosystem in order to study the hidden aspects of the systems.

**Definition 2.3.1.4.** [40] Plaintext. The information which we want to protect from other people (attackers).

**Definition 2.3.1.5**. [40] Security. It means that the difficulty to know the information which transferred over the channel easily.

## 2.3.2 Basic Communications Model

[1] In Figure (2.11), entities **A** (Alice) and **B** (Bob) are communicating over an unsecured channel. We assume that all communications take place in the presence of an adversary **E** (Eve) whose objective is to defeat any security services being provided to **A** and **B.**

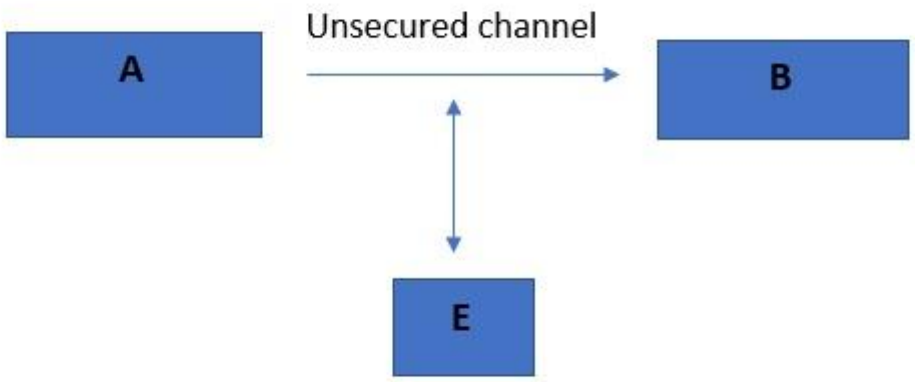


Figure 2.13. Basic communications model.

For example, A and B could be two people communicating over a cellular telephone network, and E is attempting to eavesdrop on their conversation.

## 2.3.3. Important Kinds of Cryptosystems

### 2.3.3.1 Symmetric-Key Cryptosystems.

[1] The cryptosystems which use the same cryptographic keys for both encryption of plaintext and decryption of ciphertext. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information. Basically, symmetric encryption uses a single key for both encryption and description. And it is fast in execution. It is algorithm DES, 3DES, AES, and RC4. The purpose of the symmetric encryption is uses for bulk data transmission.

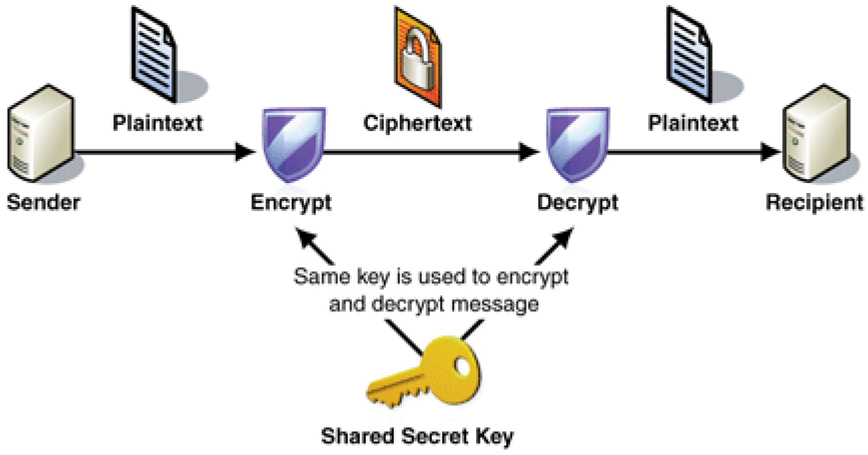


Figure 2.14. Symmetric-Key Cryptosystems.

### 2.3.3.2 Asymmetric-Key Cryptosystems (Public-Key Cryptosystems).

They use public and private keys to encrypt and decrypt data [1]. The keys are not identical (asymmetric). One key in the pair can be shared with everyone; it is called the public key and another one can stay as a secret, which is called a private key. Basically, asymmetric encryption uses a different key for encryption and decryption. And it is slow in execution due to the high computation burden. It is algorithm Diffie-

Hellman, RSA. The purpose of the asymmetric encryption is often used for securely exchanging secret key.

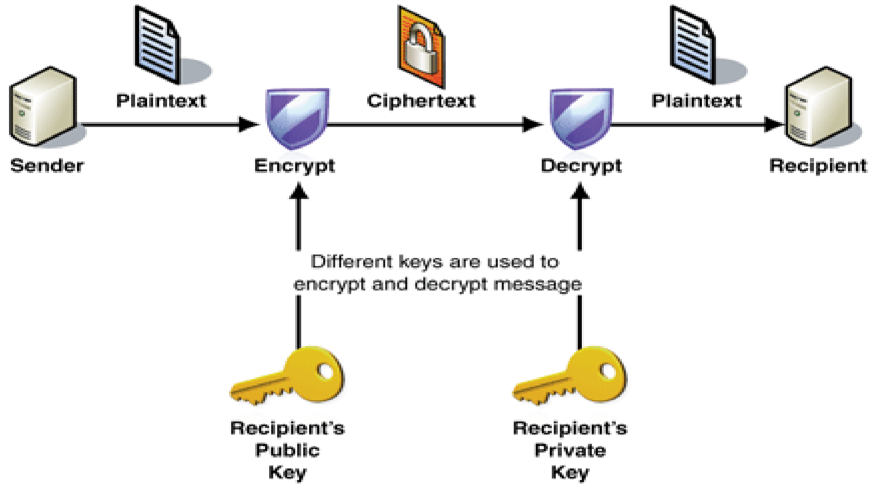


Figure 2.15 Asymmetric-Key Cryptosystems (Public-Key cryptosystems).

## 2.3.4 Symmetric Encryption Schemes

Some examples of the symmetric encryption schemes are discussed as follows.

### 2.3.4.1. The Polyalphabetic Cipher

[1] The polyalphabetic substitution cipher considers as another way to improve the simple monoalphabetic technique. It works with alphabet Tables (English Alphabetic Values as shown in Table (3.1) and ASCII values as given in Table (3.2)) and some rules are putting on the secret key.

### 2.3.4.2. Affine Cipher

An affine cipher (like a shift cipher) [40] is an example of a substitution cipher. In encryption using a substitution cipher, a given letter in the plaintext, it always is replaced by the ciphertext letter. An affine cipher is a generalization to a shift cipher. Shift ciphers are a particular type of affine cipher.

The encryption key of an affine cipher is an ordered pair of integers, both of which come from the set $\{0, \ldots, n-1\}$, where n is the size of the character set being used (for us, the character set is the English alphabet, namely $n = 26$ or n =127 which is the character set of ASCII values). It is important to note that some of the possible pairs of integers from the set $\{0, \ldots, n-1\}$ are not valid as affine encryption keys. On an affine cipher, each letter is enciphered with the function $(ax + b)$ $(mod\ n)$, where $b$ is the magnitude of the shift. Next example explains an affine cipher in more details.

Decription of affine with the function $a^{-1}$ (c-b) m: dn

### 2.3.4.3 Hill Cipher

[40] The Hill cipher is a polygraphic substitution cipher based on linear algebra. It is developed by the mathematician Lester S. Hill. It was the first polygraphic cipher in which it was practical to operate on more than three symbols at once. The encryption on Hill cipher is computed by

$C=K*M\ (mod\ n)$, where K is key, and M is messege matrices.

The decryption is computed by $M=K^{-1}*C\ (mod\ n)$, where $K^{-1}$ and C are square matrices.

To encrypt a message using the Hill cipher, it must first turn the keyword into a key matrix It is also convert the plaintext into matrix and each of these into a column vector. Then a matrix multiplication modulo the length of the alphabet (i.e., 26 or 127) on each vector is computed. These vectors are then converted back into letters to produce the ciphertext.

To encrypt a message, each block of n letters (considered as an n-component vector) is multiplied by an invertible $n \times n$ matrix, against modulus 26 or 127.

In order to decrypt, it requires to convert the ciphertext back into a vector, then simply multiply by the inverse matrix of the key matrix. It must find the inverse matrix, then multiply the inverse by a matrix by the column vectors that the ciphertext is split into, take the results modulo the length of the alphabet, and finally convert the numbers back to letters.

Two complications exist in picking the encrypting matrix:

Not all matrices have an inverse. So,

1. The matrix will have an inverse if and only if its determinant is not zero.
2. The determinant of the encrypting matrix must not have any common factors with the modular base

# Chapter Three

# The Triple Vertex Cycle Graph for Encryption Schemes

## 3.1 Introduction

In this chapter, two new definitions of special graphs which are called a triple vertex graph (TVG) and the triple vertex cycle graph (TVCG). These graphs are used to propose new versions of the symmetric encryption schemes. Several examples are presented and discussed to explain these definitions and symmetric encryption schemes.

## 3.2 The Triple Vertex Graph

In this section, new definition of a special graph is proposed, this graph called a triple vertex graph (TVG) which discusses as follows.

**Definition 3.2.1.** Let $G(V, E)$ be a simple graph. The order of $G$ is equal $n$ with $n \geq 3$. The triple vertex graph $\mathrm{TVG}_3(\mathrm{G})$ is a graph whose vertex x, y , z , u , v , t Є G $\{x,y,z\}\{u,v,t\}$ are adjacent if $x= u$ and $y= v$ there is an arc from z to t

**Example 3.2.1. (The TVG$_3$(*G*)).** Let $G$ be a simple graph has 4 vertices and 4 edges as show in Figure (3.1).

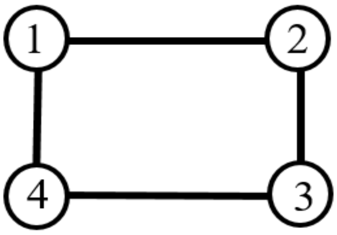


Figure 3.1. A simple graph $G(4,4)$.

Based on Definition (3.2.1), the $TVG_3(G)$ is computed by

$$\{1,2,3\} \cap \{2,3,4\} = \{2,3\} \Rightarrow |\{1,2,3\} \cap \{2,3,4\}| = 2$$

$$\{1,2,3\} \cap \{1,2,4\} = \{1,2\} \Rightarrow |\{1,2,3\} \cap \{1,2,4\}| = 2$$

$$\{2,3,4\} \cap \{3,4,1\} = \{3,4\} \Rightarrow |\{2,3,4\} \cap \{3,4,1\}| = 2$$

$$\{3,4,1\} \cap \{4,1,2\} = \{1,4\} \Rightarrow |\{3,4,1\} \cap \{4,1,2\}| = 2$$

The $TVG_3(G)$ of a graph *G* is given in Figure (3.2).

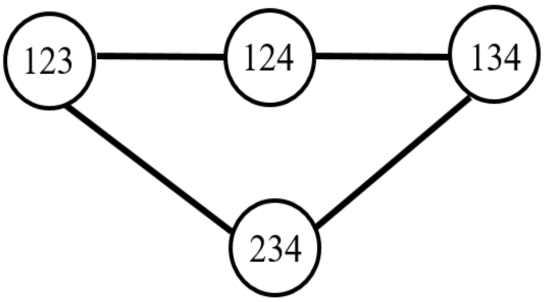


Figure 3.2. The $TVG_3$ G(4,4).

**Example 3.2.2. (The $TVG_3$(*G*)).** Let *G* be a simple graph has 5 vertices and 7 edges as show in Figure (3.3).

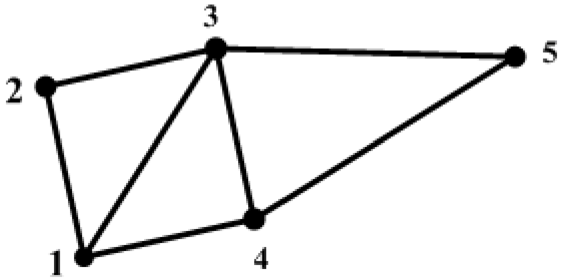


Figure 3.3. A simple graph G (5,7).

Based on Definition (3.2.1), the $TVG_3(G)$ is computed by

$$\{1,2,3\} \cap \{1,2,4\} = \{1,2\} \Rightarrow | \{1,2,3\} \cap \{1,2,4\} | = 2$$

$$\{1,2,3\} \cap \{1,2,5\} = \{1,2\} \Rightarrow | \{1,2,3\} \cap \{1,2,5\} | = 2$$

$$\{1,2,3\} \cap \{2,3,4\} = \{2,3\} \Rightarrow | \{1,2,3\} \cap \{2,3,4\} | = 2$$

$$\{1,2,4\} \cap \{1,3,4\} = \{1,4\} \Longrightarrow |\{1,2,4\} \cap \{1,3,4\}| = 2$$

$$\{1,2,4\} \cap \{1,2,5\} = \{1,2\} \Longrightarrow |\{1,2,4\} \cap \{1,2,5\}| = 2$$

$$\{1,3,4\} \cap \{1,3,5\} = \{1,3\} \Longrightarrow |\{1,3,4\} \cap \{1,3,5\}| = 2$$

$$\{1,3,4\} \cap \{2,3,4\} = \{3,4\} \Longrightarrow |\{1,3,4\} \cap \{2,3,4\}| = 2$$

$$\{1,2,5\} \cap \{1,3,5\} = \{1,5\} \Longrightarrow |\{1,2,5\} \cap \{1,3,5\}| = 2$$

$$\{1,2,5\} \cap \{2,4,5\} = \{2,5\} \Longrightarrow |\{1,2,5\}\cap\{2,4,5\}| = 2$$

$$\{1,3,5\} \cap\{1,4,5\} = \{1,5\} \Longrightarrow |\{1,3,5\}\cap\{1,4,5\}| = 2$$

$$\{1,3,5\}\cap\{2,3,5\} = \{1,2\} \Longrightarrow |\{1,3,5\}\cap\{2,3,5\}| = 2$$

$$\{2,3,4\}\cap\{2,4,5\} = \{2,4\}\Longrightarrow |\{2,3,4\}\cap\{2,4,5\}| = 2$$

$$\{2,3,4\}\cap\{2,3,5\} = \{2,3\}\Longrightarrow|\{2,3,4\}\cap\{2,3,5\}| = 2$$

$$\{1,4,5\}\cap\{2,4,5\}= \{4,5\} \Longrightarrow |\{1,4,5\}\cap\{2,4,5\}| = 2$$

$$\{1,4,5\}\cap\{3,4,5\}=\{4,5\} \Longrightarrow |\{1,4,5\}\cap\{3,4,5\}| = 2$$

$$\{2,3,5\}\cap\{2,4,5\}=\{2,5\} \Longrightarrow |\{2,3,5\}\cap\{2,4,5\}|=2$$

$$\{2,4,5\}\cap\{3,4,5\}=\{4,5\}\Longrightarrow |\{2,4,5\}\cap\{3,4,5\}\}=2$$

The TVG$_3$(G) of a graph *G* is given in Figure (3.4).

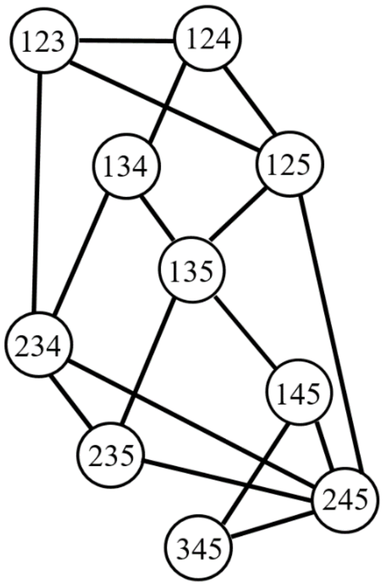


Figure 3.4. The $TVG_3$ G(5,7).

## 3.3 The Triple Vertex Cycle Graph

In this section, a triple vertex cycle graph (TVCG) is defined as a new definition based on the idea of the $TVG_3(G)$ that is given in Definition (3.2.1).

**Definition 3.3.1.** Let $C_n$ be a cycle graph of order $n$. This mean that, this cycle has $n$ vertices and edges. The triple vertex cycle graph $TVG_3(C_n)$ is a graph whose vertex set $V$ of all unordered triples from $V$ such that two vertices $\{x,y,z\}$ and $\{u,v,t\}$ are adjacent if and only if $|\{x,y,z\} \cap \{u,v,t\}| = 2$ and if $x= u$ and $y= v$ then z and t are adjacent in $G$. The triple vertex graph of cycle is denoted by $TVG_3(C_n)$.

**Example 3.3.1.** Let $C_4$ be cycle has 4 vertices and edges, as shown in Figure (3.1). Then, the $TVG_3(C_4)$ of a cycle graph $C_4$ is given in Figure (3.2).

**Example 3.3.2.** Let $C_6$ be cycle graph with 6 vertices and 6 edges as shown in Figure (3.5).

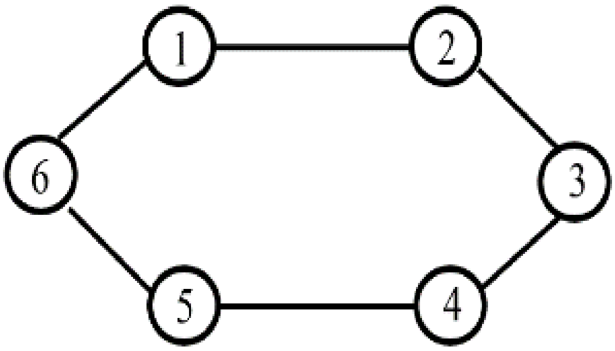


Figure 3.5. A cycle graph $C_6$.

Then, the $TVG_3(C_6)$ is computed based on all intersected vertices that have distances 2.

$$\{1,2,3\}\cap\{1,2,4\} = \{1,2\} \Rightarrow |\{1,2,3\}\cap\{1,2,4\}| = 2$$

$$\{1,2,3\}\cap\{1,2,5\} = \{1,2\} \Rightarrow |\{1,2,3\}\cap\{1,2,5\}| = 2$$

$$\{1,2,3\}\cap\{1,2,6\} = \{1,2\} \Rightarrow |\{1,2,3\}\cap\{1,2,6\}| = 2$$

$$\{1,2,4\}\cap\{1,3,4\} = \{1,4\}\Rightarrow |\{1,2,4\}\cap\{1,3,4\}| = 2$$

$$\{1,2,4\}\cap\{1,2,5\} = \{1,2\} \Rightarrow |\{1,2,4\}\cap\{1,2,5\}| = 2$$

$$\{1,2,4\}\cap\{1,2,6\} = \{1,2\} \Rightarrow |\{1,2,4\}\cap\{1,2,6\}| = 2$$

$$\{1,3,4\}\cap\{1,3,5\} = \{1,3\} \Rightarrow |\{1,3,4\}\cap\{1,3,5\}| = 2$$

$$\{1,3,4\}\cap\{1,3,6\} = \{1,3\} \Rightarrow |\{1,3,4\}\cap\{1,3,6\}| = 2$$

$$\{1,3,5\}\cap\{1,3,6\} = \{1,3\} \Rightarrow |\{1,3,5\}\cap\{1,3,6\}| = 2.$$

The $TVG_3(C_6)$ is given in Figure (3.6).

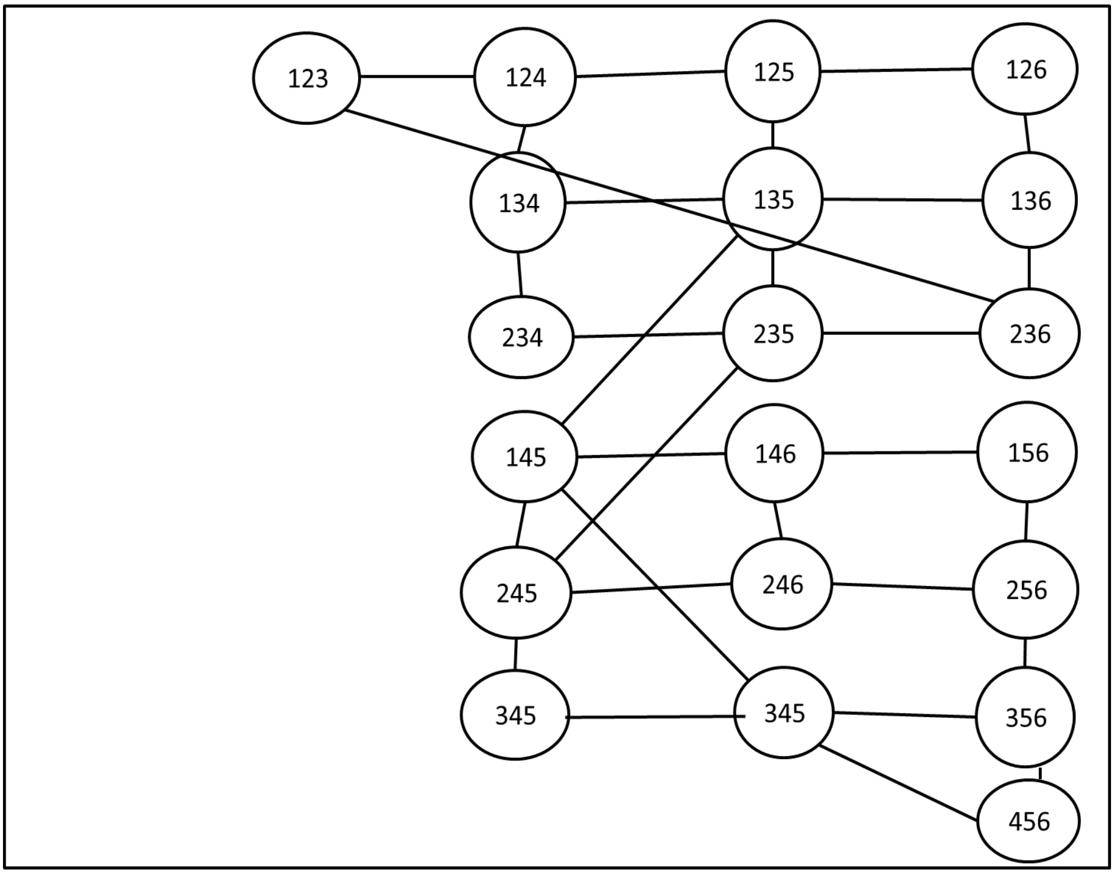


Figure 3.6. The TVG3($C_6$) of a cycle graph $C_6$.

## 3.4 New Versions of Encryption Schemes Using TVG$_3$( cn )

In this section, some encryption schemes have been proposed based on the triple vertex cycle graph (TVG$_3$($C_n$)) which are discussed as follows.

### 3.4.1The Triple Vertex Cycle Graph for Encryption Scheme Based on the English Alphabet Values

The idea to use the TVCG for proposing new version of encryption scheme can be explained as follows.

Suppose $M$ =\{$m_1$,$m_2$,…,$m_n$\} is a plaintext given as an English word or English sentence. A shared secret key $K$ between two users, sender and receiver, is computed using the Diffie-Hellman key exchange [11]. The ciphertext $Cp$ is computed by

$$Cp_i \equiv m_i + K\ (mod\ 26), \text{ with } i=1,2,\dots,n.$$

A cycle graph $C_n$ is formed based on the components of the ciphertext $Cp_i$ for $i$=1,2,…,$n$. In other words, $C_{Cpi}$ = \{$Cp_1$, $Cp_2$, …, $Cp_\mathrm{n}$\}. The triple vertex cycle graph TVG$_3$($C_n$) of a cycle graph $C_n$ is computed and sent to receiver as a ciphertext of $M$ by sender. After the second user (receiver) receives the TVG$_3$($C_n$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices of the TVG$_3$($C_n$). Later on, he/she takes only the letters from vertices (the encrypted letters) without repeating to form correct cycle $C_{Cpi}$. The English alphabetic values (EAVs) are used to convert the encrypted letters of the correct cycle graph into numbers. And since a key is a shard secret key between sender and receiver, so the second user (receiver) can recover the original plaintext easily by $m_i \equiv Cp_i - K$ ($mod$ 26), with $i$=1,2,…,$n$. In other words, \{$m_1$,$m_2$,…,$m_n$\}= $M$

.

**Table 3.1. English alphabet Table.**

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** |
| **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **O** | **p** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **Y** | **Z** | **Y** | |
| **14** | **15** | **16** | **17** | **18** | **19** | **20** | **21** | **21** | **22** | **23** | **24** | **25** |

## Example 3.4.1.1. (The TVCG for Encryption Scheme Based on the EAVs)

Suppose *M* is the plaintext that is given by the word "**grand**". Based on the English alphabet Table (3.1) of the letters, one can convert the letters in the plaintext *M* into numbers. So,

$$g\rightarrow 6,\ r\rightarrow 17,\ a\rightarrow 0,\ n\rightarrow 13,\ d\rightarrow 3.$$

With a shared secret key $K$=6. The ciphertext $Cp_i$ is computed by

$$Cp_i \equiv M + K\ (mod\ 26).$$

In other words,

$$Cp_1 \equiv M_1 + K\ (mod\ 26) \equiv 6+6\ (mod\ 26) \equiv 12\ (mod\ 26) \equiv 12 \rightarrow m$$

$$C_2 \equiv M_2 + K\ (mod\ 26) \equiv 17 + 6\ (mod\ 26) \equiv 23 \rightarrow x$$

$$C_3 \equiv M_3 + K\ (mod\ 26) \equiv 0 + 6\ (mod\ 26) \equiv 6 \rightarrow g$$

$$C_4 \equiv M_4 + K\ (mod\ 26) \equiv 13 + 6\ (mod\ 26) \equiv 19 \rightarrow t$$

$$C_5 \equiv M_5 + K\ (mod\ 26) \equiv 3 + 6\ (mod\ 26) \equiv 9 \rightarrow J$$

So, the ciphertext *Cp* of *M* forms a cycle graph mxgtJ that is shown in Figure (3.7).

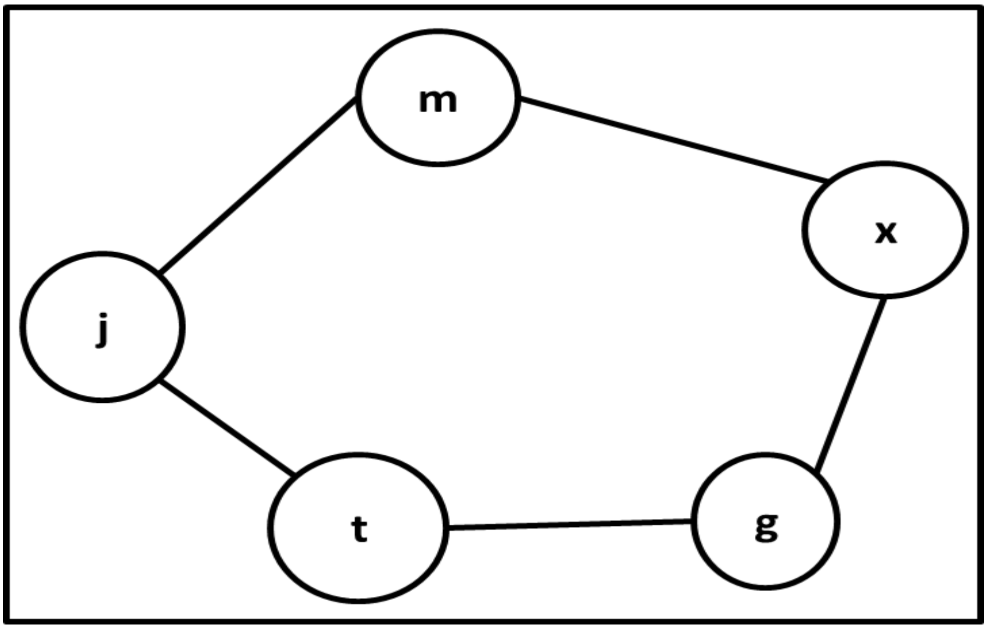


Figure 3.7. The cycle $C_5$ of the ciphertext mxgtj.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.8) and sent to receiver by sender.

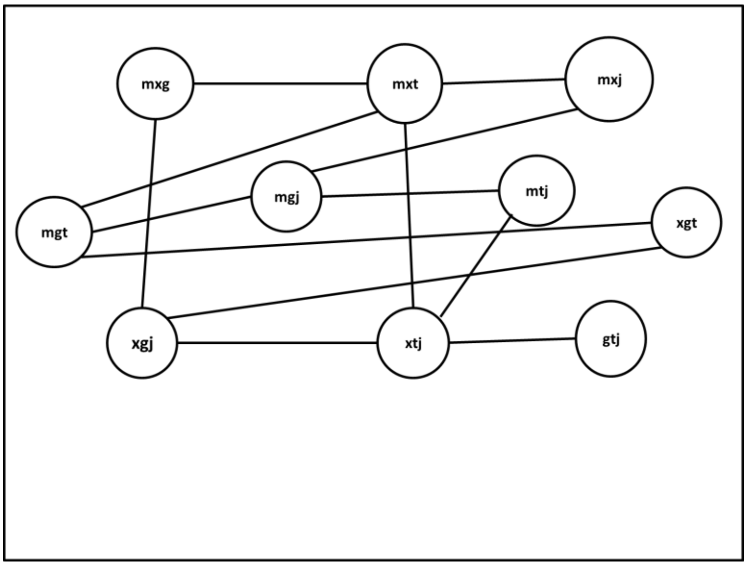


Figure 3.8. The TVCG of the cycle $C_5$ of the ciphertext mxgtj.

After the second user (receiver) receives triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices mxg , mxt, mxg , mgt , mgj , mtj , xgt , xgj , xtj & gtj , based TVG($C_5$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is mxgtj.

Now, based on Table (3.1), these letters converted into numbers

$$m \rightarrow 12, x \rightarrow 23, g \rightarrow 6, t\rightarrow 19, j\rightarrow 9.$$

Since the secret key K =6, so

$$12\text{-}6 \equiv 6 \ (mod\ 26) \equiv 6 \rightarrow g$$

$$23\text{-}6 = 17(mod\ 26) \equiv 17 \rightarrow r$$

$$6\text{-}6 = 0 \ (mod\ 26) \equiv 0 \ \rightarrow a$$

$$19\text{-}6 = 13 \ (mod\ 26) \equiv 13 \ \rightarrow n$$

$$9\text{-}6 = 3 \ (mod\ 26) \equiv 3 \ \rightarrow d$$

Thus, the original message is grand.

## Example 3.4.1.2.

Suppose *M* is the plaintext that is given by the word "**mother"**. Based on the English alphabet Table (3.1) of the letters, one can convert the letters in the plaintext *M* into numbers. So,

$$m \rightarrow 12, o \rightarrow 14, t \rightarrow 19, h\rightarrow 7, e\rightarrow 4,r \rightarrow 17.$$

With a shared secret key *K*= 8. The ciphertext *Cp* is computed by

$$Cp \equiv M + K \ (mod\ 26).$$

In other words,

$$Cp_1 \equiv M_1 + K\ (mod\ 26) \equiv 12 + 8\ (mod\ 26) \equiv 20\ (mod\ 26) \equiv 20 \rightarrow u$$

$$Cp_2 \equiv M_2 + K\ (mod\ 26) \equiv 14 + 8\ (mod\ 26) \equiv 22\ (mod\ 26) \equiv 2\ 2 \rightarrow w$$

$$Cp_3 \equiv M_3 + K\ (mod\ 26) \equiv 19 + 8\ (mod\ 26) \equiv 27\ (mod\ 26) \equiv 1 \rightarrow b$$

$$Cp_4 \equiv M_4 + K\ (mod\ 26) \equiv 7 + 8\ (mod\ 26) \equiv 15 \rightarrow p$$

$$Cp_5 \equiv M_5 + K\ (mod\ 26) \equiv 4 + 8\ (mod\ 26) \equiv 12 \rightarrow m$$

$$Cp_6 \equiv M_6 + K\ (mod\ 26) \equiv 17 + 8\ (mod\ 26) \equiv 25 \rightarrow z.$$

So, the ciphertext *Cp* of *M* forms a cycle graph uwbpmz that is shown in Figure (3.9).

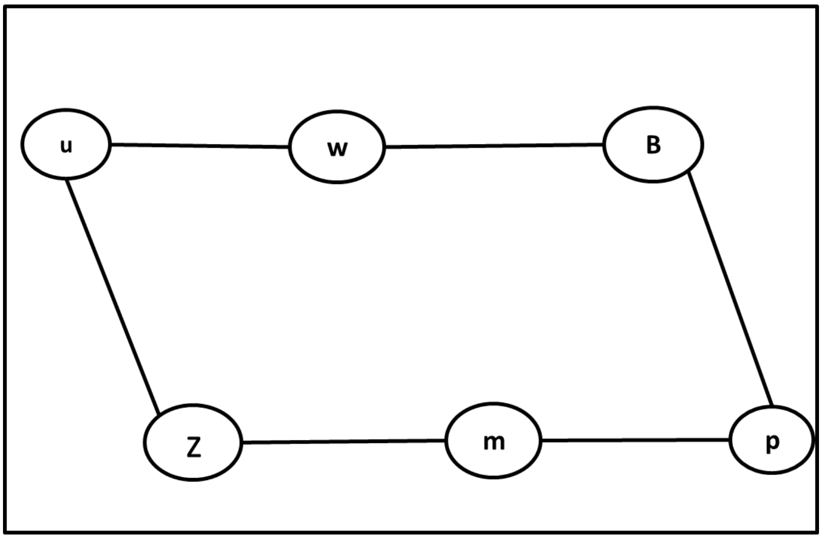


Figure 3.9. The cycle $C_6$ of the ciphertext uwBpmZ.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.10) and sent to receiver by sender.

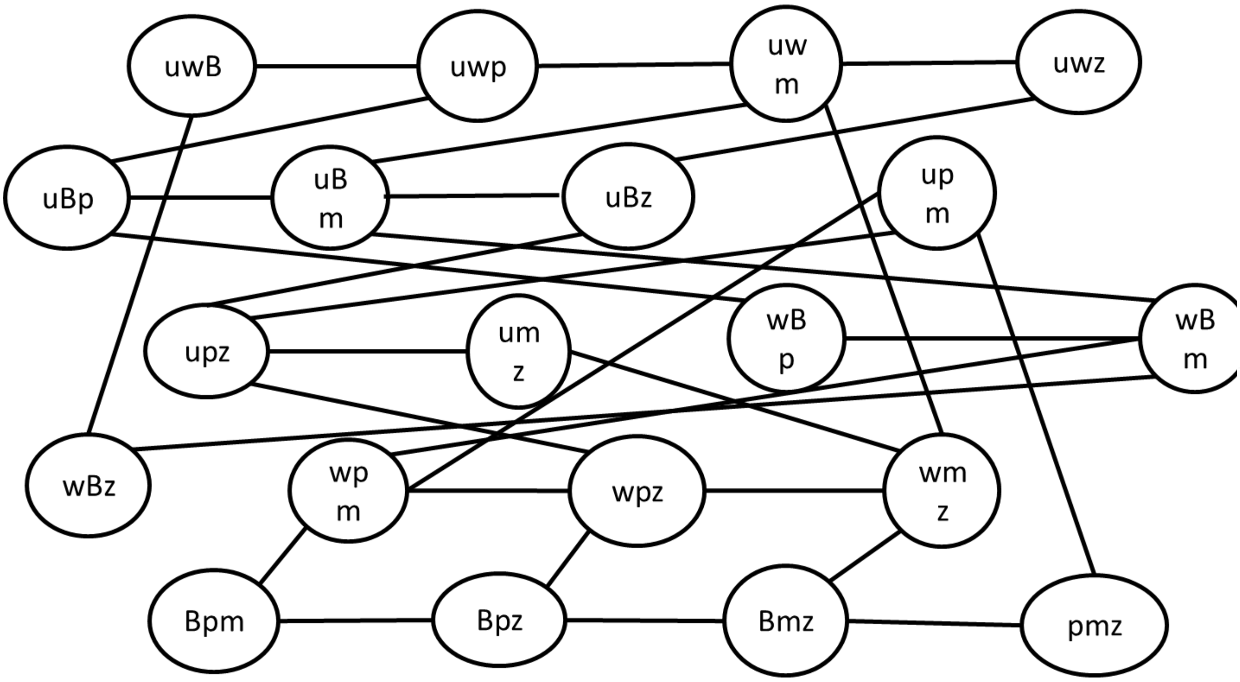


Figure 3.10. The TVCG $C_6$ of the ciphertext: uwbpmz.

After the second user (receiver) receives triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices UWB , UWP , UWM , UWZ , UBP , UBM , UBZ , UBM , UPZ , UMZ , WBP , WBM , WPZ , WPM , WPZ , WMZ , BPM , BPZ , BMZ , & PMZ based on TVG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is UWBPMZ.

Now, based on Table (3.1), these letters converted into numbers

$$U \rightarrow 20, W \rightarrow 22, B \rightarrow 1, P\rightarrow 15, M \rightarrow 12, Z \rightarrow 25.$$

Since the shared secret key K=8 , so

$$20\text{-}8 \equiv 12\ (mod\ 26) \equiv 12 \rightarrow M$$

$$22\text{-}8 \equiv 14\ (mod\ 26) \equiv 14 \rightarrow O$$

$$1\text{-}8 \equiv -7 \ (mod\ 26) \equiv 19 \rightarrow T$$

$$15\text{-}8 \equiv 7 \ (mod\ 26) \equiv 7 \rightarrow H$$

$$12\text{-}8 \equiv 4 (mod\ 26) \equiv 4 \rightarrow E$$

$$25\text{-}8 \equiv 17 \ (mod\ 26) \equiv 17 \rightarrow R$$

Thus, the original message is mother.

### 3.4.2. The Triple Vertex Cycle Graph for Encryption Scheme Based on the ASCII Values

The idea to use the TVCG for proposing new version of encryption scheme with ASCII values can be explained.

The same idea of the TVcG scheme that uses EAVs can be applied to encrypt the plaintext $M$ using the ASCII values [10]. Using ASCII, the numbers of the allowed letters that can be chosen is 127. The ciphertext $Cp$ of a message $M$ is computed by $Cp_i \equiv m_i + K$ ($mod$ 127), with $i$=1,2,…,$k$ and sent to receiver as TVG$_3$($C_n$). Upon second user receives the TVG$_3$($C_n$), he / she determines the label vertices of a graph TVG$_3$($C_n$). Later on, he/she takes only the letters from vertices (the encrypted letters ) without repeating to form correct Cycle $C_{Cpi}$. Using Table (3.2) to convert the encrypted letters in the correct cycle graph into numbers. And since a key is a shared secret key between sender and receiver, so the second user (receiver) can recover the original plaintext easily by $m_i \equiv Cp_i - K$ ($mod$ 127), with $i$=1,2,…,$k$. In other words, $\{m_1,m_2,\ldots,m_k\}= M$.

### Table 3.2 ASCII Values Table.

| Dec. | Char. | Dec. | Char. | Dec. | Char. | Dec. | Char. |
|---|---|---|---|---|---|---|---|
| 0 | Null | 32 | Space | 64 | @ | 96 | ` |
| 1 | Start of heading | 33 | ! | 65 | A | 97 | a |
| 2 | start of text | 34 | " | 66 | B | 98 | b |
| 3 | end of text | 35 | # | 67 | C | 99 | c |
| 4 | end of transmission | 36 | $ | 68 | D | 100 | d |
| 5 | Enquiry | 37 | % | 69 | E | 101 | e |
| 6 | Acknowledge | 38 | & | 70 | F | 102 | f |
| 7 | Bell | 39 | ' | 71 | G | 103 | g |
| 8 | Backspace | 40 | ( | 72 | H | 104 | h |
| 9 | horizontal tab | 41 | ) | 73 | I | 105 | i |
| 10 | NL line feed, new line | 42 | * | 74 | J | 106 | j |
| 11 | vertical tab | 43 | + | 75 | K | 107 | k |
| 12 | NP form feed, new page | 44 | , | 76 | L | 108 | l |
| 13 | carriage return | 45 | - | 77 | M | 109 | m |
| 14 | shift out | 46 | . | 78 | N | 110 | n |
| 15 | shift in | 47 | / | 79 | O | 111 | o |
| 16 | data link escape | 48 | 0 | 80 | P | 112 | p |
| 17 | device control 1 | 49 | 1 | 81 | Q | 113 | q |
| 18 | device control 2 | 50 | 2 | 82 | R | 114 | r |

| Dec. | Char. | Dec. | Char. | Dec. | Char. | Dec. | Char. |
|---|---|---|---|---|---|---|---|
| 19 | device control 3 | 51 | 3 | 83 | S | 115 | s |
| 20 | device control 4 | 52 | 4 | 84 | T | 116 | t |
| 21 | negative acknowledge | 53 | 5 | 85 | U | 117 | u |
| 22 | synchronous idle | 54 | 6 | 86 | V | 118 | v |
| 23 | end of trans. Block | 55 | 7 | 87 | W | 119 | w |
| 24 | Cancel | 56 | 8 | 88 | X | 120 | x |
| 25 | end of medium | 57 | 9 | 89 | Y | 121 | y |
| 26 | Substitute | 58 | : | 90 | Z | 122 | z |
| 27 | Escape | 59 | ; | 91 | [ | 123 | { |
| 28 | file separator | 60 | < | 92 | \ | 124 | \| |
| 29 | group separator | 61 | = | 93 | ] | 125 | } |
| 30 | record separator | 62 | > | 94 | ^ | 126 | ~ |
| 31 | unit separator | 63 | ? | 95 | _ | 127 | Del |

## Example 3.4.2.1.

Suppose $M$ is the plaintext that is given by the word **beauty**. Based on the ASCII Table (3.2) of the letters, one can convert the letters in the plaintext $M$ into numbers. So,

$$b \rightarrow 98,\ e \rightarrow 101,\ a \rightarrow 97,\ u \rightarrow 117,\ t \rightarrow 116,\ y \rightarrow 121.$$

The length $K$ of $M$ is equal to 4. Adding $K$ to all of these numbers one by one respectively gives the ciphertext

$$Cp_1 \equiv 98+4\ (mod\ 127) \equiv 102 \rightarrow f$$

$$Cp_2 \equiv 101{+}4\ (mod\ 127) \equiv 105 \rightarrow i$$

$$Cp_3 \equiv 97{+}4\ (mod\ 127) \equiv 101 \rightarrow e$$

$$Cp_4 \equiv 117{+}4\ (mod\ 127) \equiv 121 \rightarrow \mathrm{y}$$

$$Cp_5 \equiv 116{+}4\ (mod\ 127) \equiv 120 \rightarrow \mathrm{x}$$

$$Cp_6 \equiv 121{+}4\ (mod\ 127) \equiv 125 \rightarrow \}$$

The ciphertext of *M* is

$$Co = fieyx\}$$

A cycle graph of the ciphertext is created as shown in Figure (3.11).

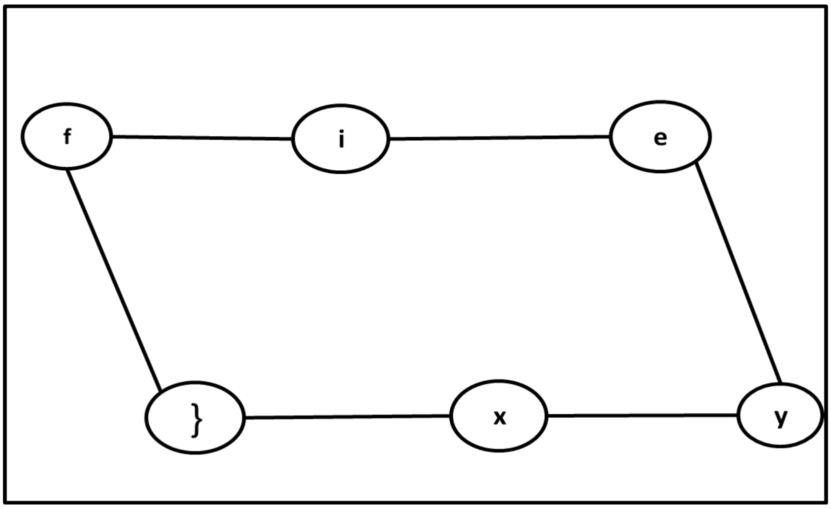


Figure 3.11. The cycle $C_6$ of the ciphertext: *fieyx}*

The ciphertext of a message *M* is considered as the TVG($C_6$) as shown in Figure (3.12) which is sent to receiver by sender.

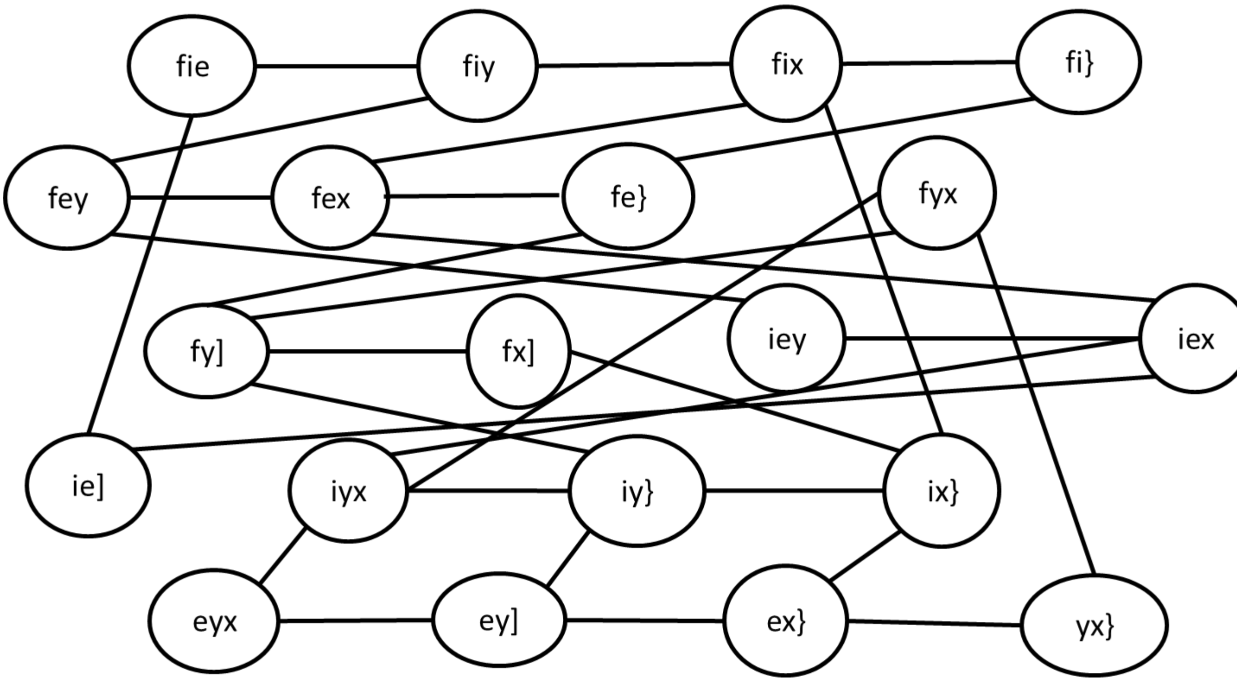


Figure 3.12. The TVG($C_6$) of the ciphertext: *fieyx}*.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices fie , fiy , fix , fi} , fey , fex , fe} , fyx , fy} , fx} , iey , iex , ie} , iyx , iy} , ix}, eyx , ey} , ex} , yx}

Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one in *fieyx}*.

Now, based on Table (3.2), these letters converted into numbers

$$f \rightarrow 102, i \rightarrow 105, e \rightarrow 101 , y \rightarrow 121, x \rightarrow 120, \} \rightarrow 125.$$

Since the number of elements is equal to 6, so

$$102\text{-}4 \equiv 98 \ (mod\ 127) \equiv 98 \rightarrow b$$

$$105\text{-}4 \equiv 101 \ (mod\ 127) \equiv 101 \rightarrow e$$

$$101\text{-}4 \equiv 97 \ (mod\ 127) = 97 \rightarrow a$$

$$121\text{-}4 \equiv 117\ (mod\ 127) \equiv 117 \rightarrow u$$

$$120\text{-}4 \equiv 116\ (mod\ 127) \equiv 116 \rightarrow t$$

$$125\ \text{-}4 \equiv 121\ (mod\ 127) \equiv 121 \rightarrow y$$

Thus, the original message is **beauty**.

**Example 3.4.2.2.** Suppose $M$ is the plaintext that is given by the word **Country**. Based on the ASCII Table (3.2) of the letters, one can convert the letters in the plaintext $M$ into numbers. So,

$$C \rightarrow 67,\ o\rightarrow 111,\ u\rightarrow 117,\ n \rightarrow 110,\ t \rightarrow 116,\ r\rightarrow 114,\ y\rightarrow 121.$$

The length $K$ of $M$ is equal to 5. Adding $K$ to all of these numbers one by one respectively gives the ciphertext

$$Cp_1 \equiv 67+5\ (mod\ 127) \equiv 72 \rightarrow H$$

$$Cp_2 \equiv 111+5\ (mod\ 127) \equiv 116 \rightarrow t$$

$$Cp_3 \equiv 117+5\ (mod\ 127) \equiv 122 \rightarrow z$$

$$Cp_4 \equiv 110+5\ (mod\ 127) \equiv 115 \rightarrow s$$

$$Cp_5 \equiv 116+5\ (mod\ 127) \equiv 121 \rightarrow y$$

$$Cp_6 \equiv 114+5\ (mod\ 127) \equiv 119 \rightarrow w$$

$$Cp_7 \equiv 121+5\ (mod\ 127) \equiv 126 \rightarrow \sim$$

The ciphertext of $M$ is

$$C = Htzsyw\sim$$

A cycle graph of the ciphertext is created as shown in Figure (3.13).

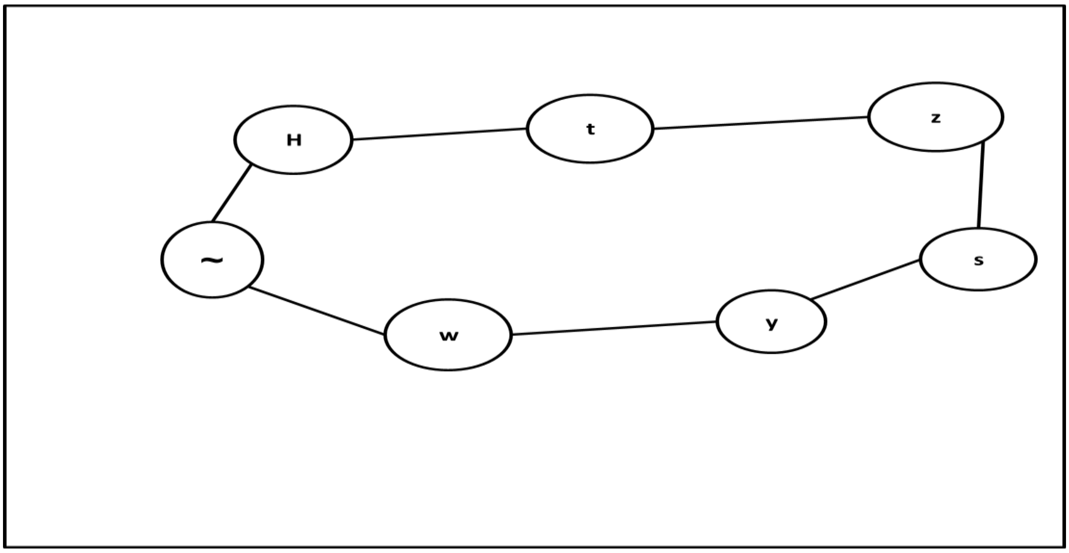


Figure 3.13. The cycle $C_7$ of the ciphertext: *Htzsyw~*.

The ciphertext of a message *M* is considered as the TVCG as shown in Figure (3.14) which is sent to receiver by sender.

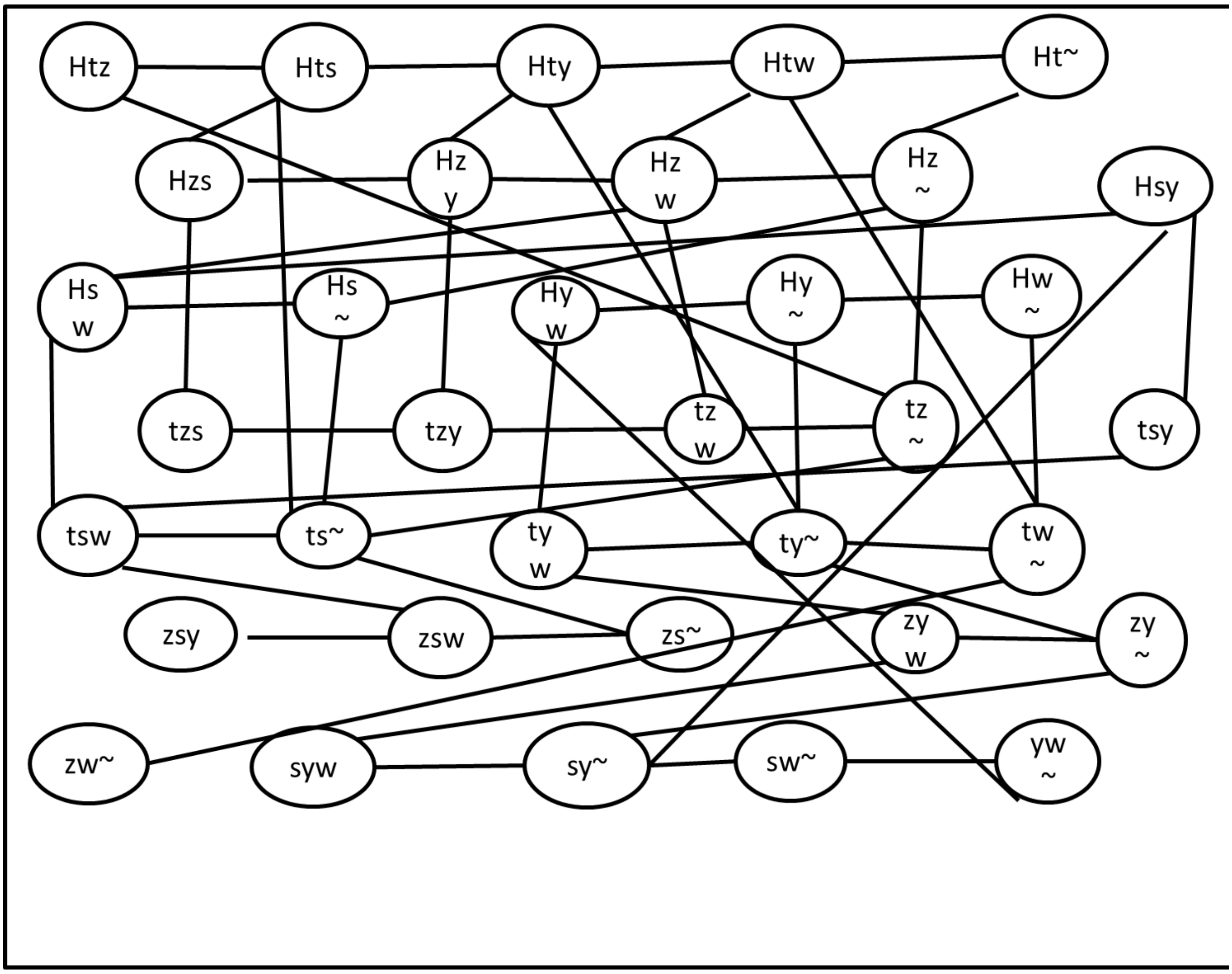


Figure 3.14. The TVG($C_7$) of the ciphertext: *Htzsyw~*

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices Htx , Hts , Hty , Htw , Ht ~ , Hzs , Hzy , Hzw , Hz ~ , Hsy , Hsw , Hs ~ , Hyw , Hy ~ , Hw ~ , tzs , tzy , tzw , tz ~ , tsy , tsw , ts ~ , tyw , ty ~ , tw ~ , zsy , zsw , zs ~ , zyw , zy ~ , zw ~ , syw , sy ~ , sw ~ ,  & yw ~ . Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct Htzsyw ~

Now, based on Table (3.2), these letters converted into numbers

*H*→*72 , t* →116, *z* →122, *s*→115, *y*→121, *w* →119, ~ → 126.

Since the shared secret key K= 5, so

$$72\text{-}5 \equiv 67 \ (mod\ 127) \equiv 67 \ \rightarrow C$$

$$116\text{-}5 \equiv 111 \ (mod\ 127) \equiv \ 111 \rightarrow o$$

$$122\text{-}5 \equiv 117 \ (mod\ 127) \equiv 117 \rightarrow u$$

$$115\text{-}5 \equiv 110 \ (mod\ 127) \equiv 110 \rightarrow n$$

$$121\text{-}5 \equiv 116 \ (mod\ 127) \equiv 116 \rightarrow t$$

$$119 \text{ -}5 \equiv 114 \ (mod\ 127) \equiv 114 \rightarrow r$$

$$126\text{-}5 \equiv 121 \ (mod\ 127) \equiv 121 \rightarrow \text{y}$$

Thus, the original message is **Country**.

## 3.5 The TVCG for Polyalphabetic Encryptions Schemes

In this section, new proposed polyalphabetic encryptions schemes using the TVCG have been discussed as follows.

### 3.5.1 The TVCG for Polyalphabetic Encryption Scheme Based on the EAVs

A new proposed polyalphabetic encryption scheme using the TVCG has been discussed as follows. Let *M* be a plaintext which is chose as an English word or English sentence. The letter of *M* are converted into numbers using EAVs as given in Table (3.1). Some rules are determined of the secret key that give the possibility to shift the letters of the plaintext certain number of the positions into right or left. Using these rules, a plaintext has been encrypted into the ciphertext *C*. The ciphertext *C* represented by cycle graph $C_n$. The TVG$_3$($C_n$) is created based on a cycle graph $C_n$. A graph TVG$_3$($C_n$) is sent to receiver as a ciphertext.

Upon receiver receives the ciphertext that is a graph TVG$_3$($C_n$). He/She wants to decrypt it and recover the original plaintext. He/She determines the label vertices of the TVG$_3$($C_5$). Now, he/she takes only the letters form vertices without repeating to form a list. One case is correct from many cases that are determined A list is formed based on the correct case. Next, the inverse rules of secret key and English alphabetic values are used to decrypt the encrypted letters in the correct formed list. So, an original plaintext is recovered.

This scheme can be explained in the following example.

**Example 3.5.1.1.** Suppose a plaintext M is a plaintext with a word is **apple**. Using polyalphabetic cipher based on the English alphabet values of the letters and the rules of the key:

i. Shift first letter two positions into the right.
ii. Shift second letter five positions into the right.
iii. Shift third letter four positions into the right.

results

$$\text{app le} \rightarrow \text{cut nj}.$$

So the encrypted letters are cutni. Now, the ciphertext cutnj represented by cycle graph as shown in Figure (3.15).

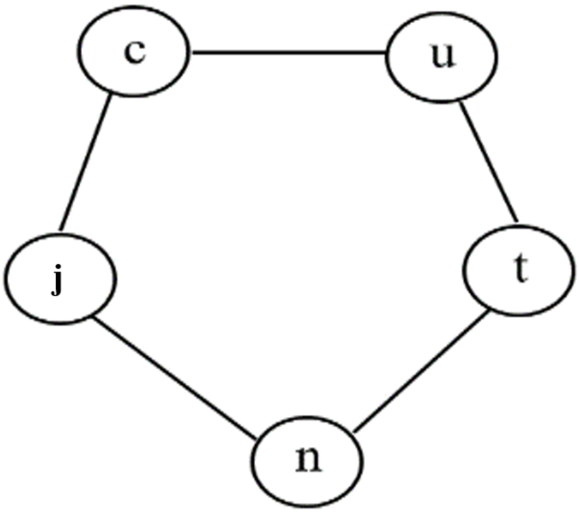


Figure 3.15. The cycle $C_5$ of the ciphertext cutnj.

This cycle $C_5$ is represent as a triple vertex graph that computed by

$$\{c,u,t\}\cap\{c,u,n\}=\{c,u\} \rightarrow |\{c,u,t\}\cap\{c,u,n\}| = 2$$

$$\{c,u,n\}\cap\{c,u,j\}=\{c,u\} \rightarrow |\{c,u,n\}\cap\{c,u,j\}| = 2$$

In similar way, one can compute all other intersections.

The ciphertext is represented by $TVG_3(C_5)$ is shown in next figure and sent to receiver by sender.

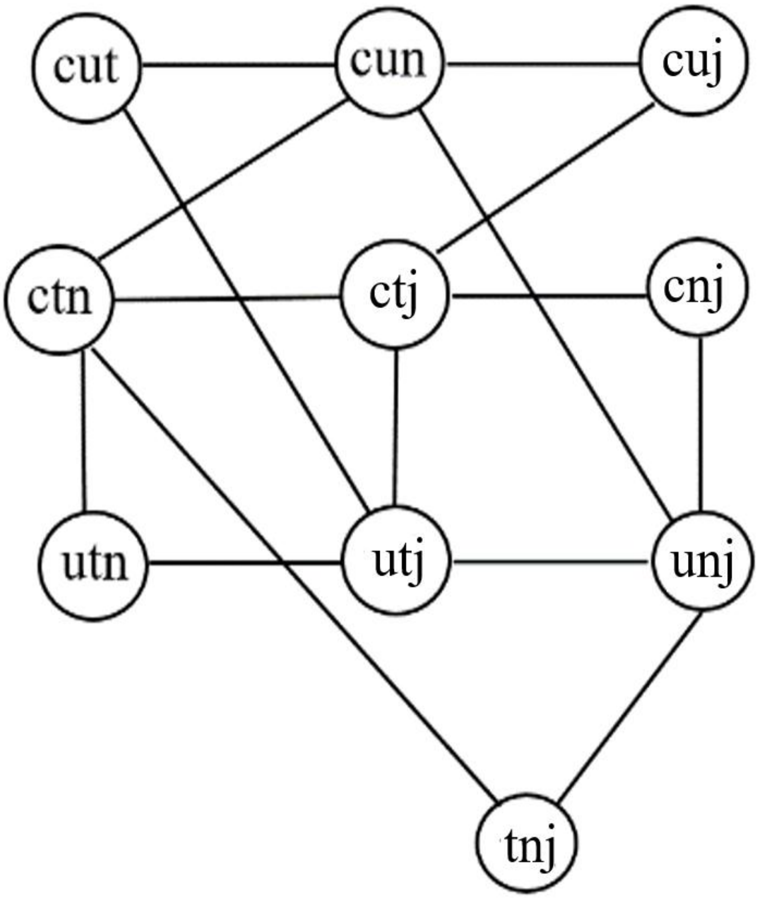


Figure 3.16. The TVG$_3$($C_5$) of the ciphertext cutnj.

After the second user (receiver) receives the TVG$_3$($C_5$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/She first determines the label vertices cut, cun, cuj, ctn, ctj, cnj, utn, utj, unj, and tnj of the TVG$_3$($C_5$). Now, he/she takes only the letters form vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is cutnj. Based on the inverse rules of secret key and English alphabetic values:

j. Shift first letter two positions into the left.
jj. Shift second letter five positions into the left.
jjj. Shift third letter four positions into the left.

The ciphertext cut nj converted into app le. Thus, the original plain text is **apple**.

**Example 3.5.1.2.** Suppose *M* is the plaintext that is given by the word "**Modren"**. Using polyalphabetic cipher based on the EAVS in Table (3.1) of the letters, and the rules on secret key that is given by:

1- Shift first letter five positions into its right,

2- Shift second letter three positions into its left.

One can obtain

Mo dr en

Mo→rl , dr→io and en→ jk.

The ciphertext of *M* *C* = rliojk

So, the ciphertext *Cp* of *M* forms a cycle graph rliojk that is shown in Figure (3.17).

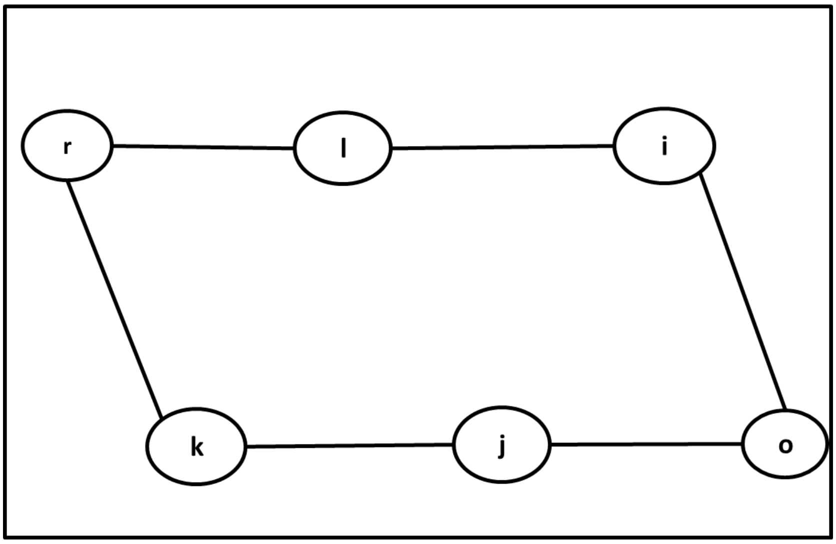


Figure 3.17. The cycle $C_6$ of the ciphertext rliojk.

This path is represented as the triple vertex graph (TVG) that is given in Figure (3.18) and sent to receiver by sender.

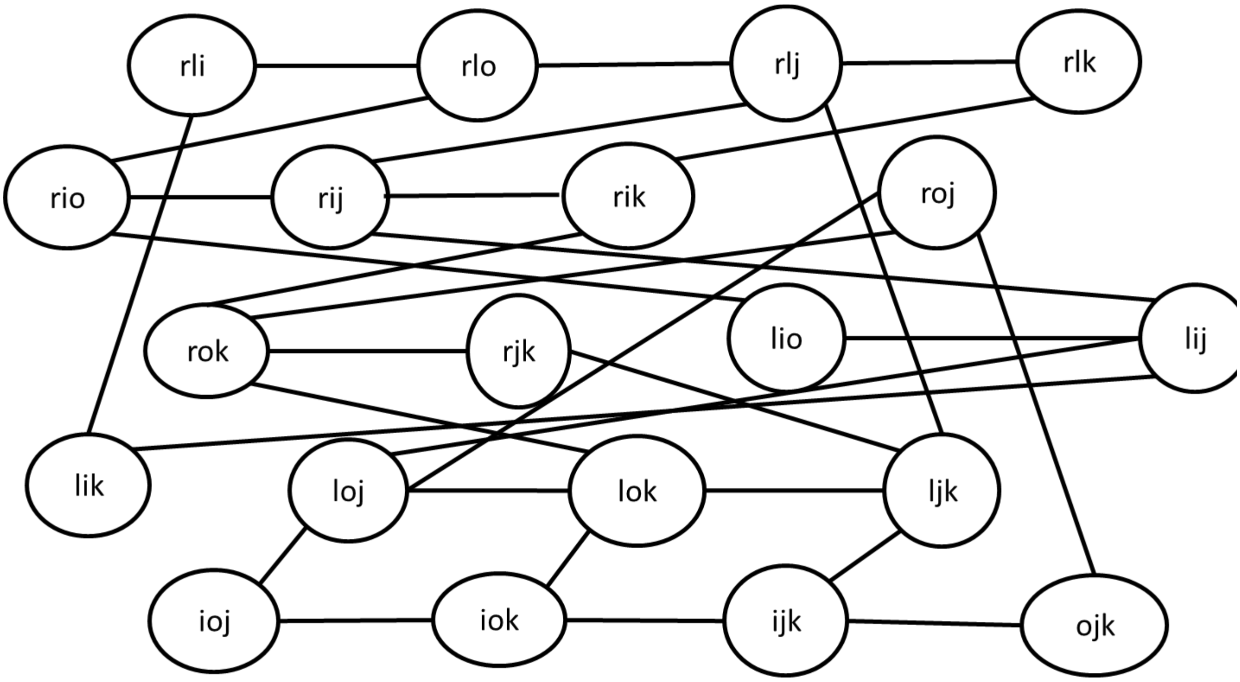


Figure 3.18. The TVG($C_6$) of the ciphertext rliojk.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices rli , rlo , rlj , rlk , rio , rij , rik , roj , rok , rjk , lio , lij , lik , loj , lok , ljk , ioj , iok , ijk ,& ojk , based on the TVPG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is rliojk.

Now, using the invers rules of a secret key that are

1- Shift first letter five positions into its left,
2- Shift second letter three positions into its right.

So, rl→ mo, io→dr and jk→ en.

Thus, the original message is Modren.

### 3.5.2 The TVCG for Polyalphabetic Encryption Scheme Based on the ASCII Values.

This section discusses the same idea of TVCG for polyalphabetic with EAVs that is explained in section (3.5.1) but here with the ASCII values that are given in the Table (3.2). A plaintext *M* is chosen in the similar way as well the ruler the secret key are determined. The ciphertext C of M and sent to receiver as TVCG. Receiver after got the *C*, he/she recovers the original plaintext *M* in same way that is used with EAVs. Some examples are presented and discussed to explain this type of encryption scheme.

**Example 3.5.2.1.** Suppose *M* is the plaintext that is given by the word "**garden**". Using polyalphabetic cipher based on the ASCII Table (2.3) of the letters, and the rules on secret key that is given by:

1- Shift first letter eight positions into its right,
2- Shift second letter twelve positions into its right,
3- Shift third letter twenty positions into its left.

One can obtain

gar den

gar→ om^ and den→ lqZ

The ciphertext of *M* is

*C* =om^lqZ.

So, the ciphertext *C* of *M* forms a cycle graph om^lqZ that is shown in Figure (3.19).

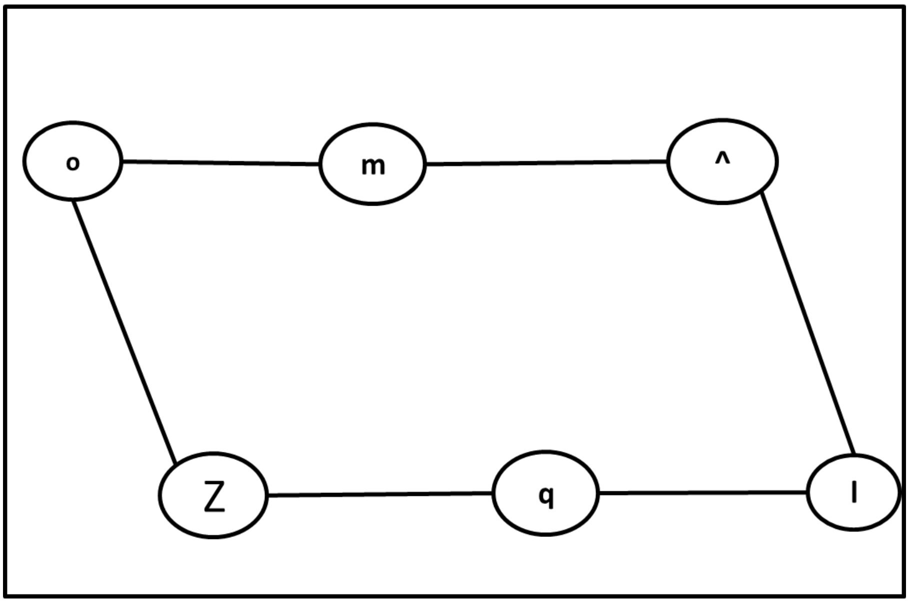


Figure 3.19. The cycle $C_6$ of the ciphertext om^lqZ.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.20) and sent to receiver by sender.

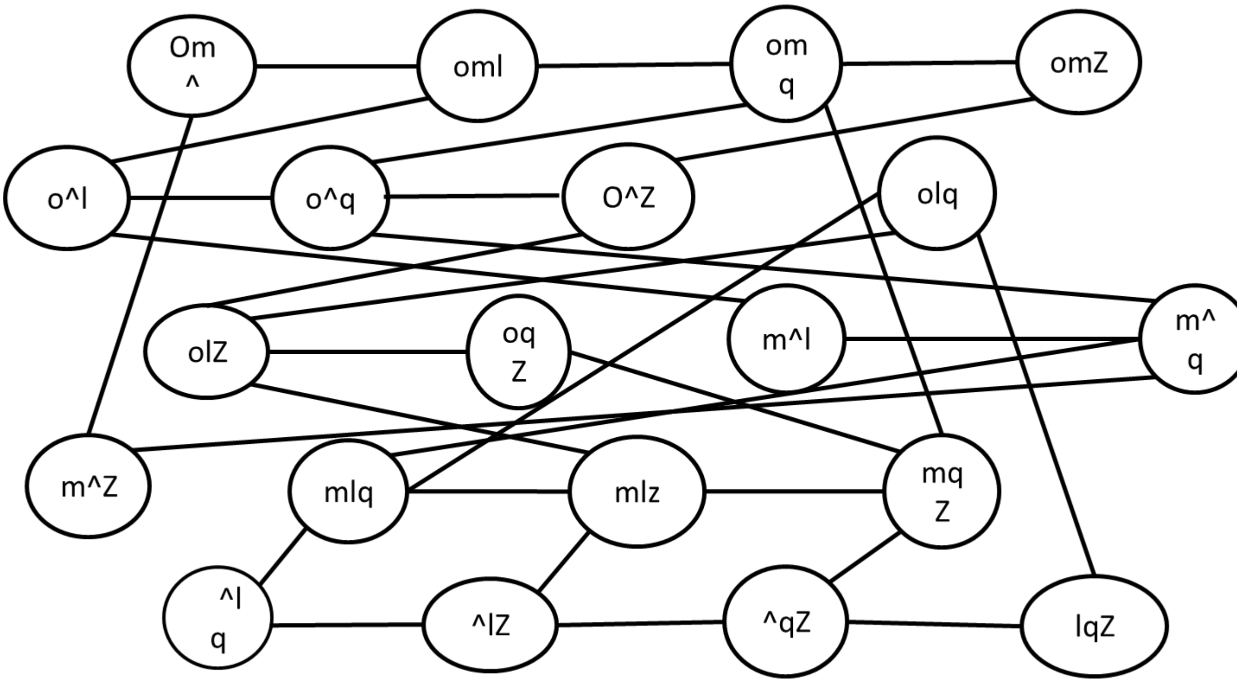


Figure 3.20. The TVG($C_6$) of the ciphertext om^lqZ.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices om^ , oml , omq , omZ , o^l , o^q , o^Z , olq , olZ , oqZ , m^l , m^q , m^Z , mlq , mlZ , mqZ , ^lq , ^lZ , ^qZ ,& lqZ , . Based on the TVCG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is om^lqZ.

Now, using the inverse rules of a secret key that are

1- Shift first letter eight positions into its left,
2- Shift second letter twelve positions into its left,
3- Shift third letter twenty right positions into its right,

So

Om^→ gar and lqZ→ den

Thus, the original message is garden.

**Example 3.5.2.2.** Suppose *M* is the plaintext that is given by the word "**Butcher**". Using Polyalphabetic Cipher based on the ASCII Table (3.2) of the letters, and the rules on secret key that is given by:

1- Shift first letter twelve positions into its right,
2- Shift second letter ten positions into its right,
3- Shift third letter eleven positions into its left.

One can obtain

But che r

But → Nki, che → o^Z and r → ~.

The ciphertext of *M* is

$C$ = **NkiO^Z~.**

So, the ciphertext of $M$ forms a cycle graph **NkiO^Z~** that is shown in Figure (3.21).

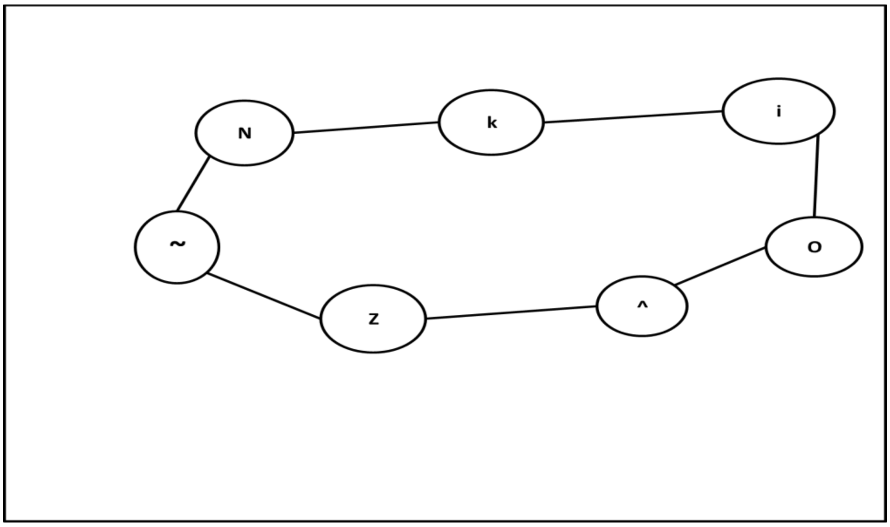


Figure 3.21. The cycle $C_7$ of the ciphertext **NkiO^Z**.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.22) and sent to receiver by sender

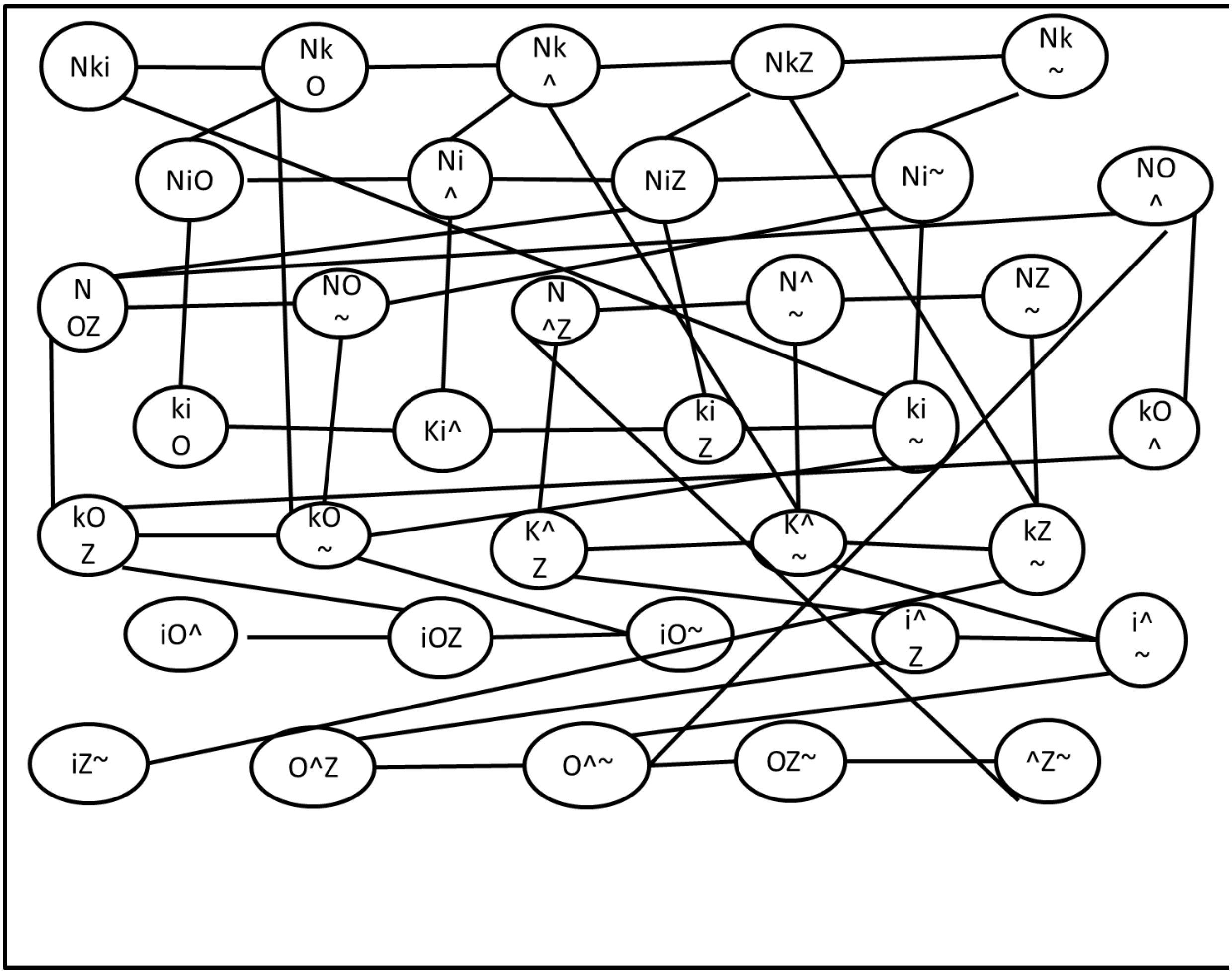


Figure 3.22. The TVG($C_7$) of the ciphertext **NkiO^Z**.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices Nki , Nko , Nk^ , NkZ , Nk~ , Nio , Ni^ , NiZ , Ni~ , No^ , NoZ , No~ , N^Z , N^~ , NZ~ , kio , ki^ , kiZ , ki~ , ko^ , Koz , ko~ ,k^Z , k^~ ,  kZ~ , io^ , ioZ , io~ , i^Z , i^~ , iZ~ , o^Z , o^~ , oZ~ ,& ^Z~ ,   based on the TVCG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **NkiO^Z~.**

Now, using the inverse rules of a secret key that are

So

$$\text{Nki} \rightarrow \text{But,O\^{}Z} \rightarrow \text{che and} \sim \rightarrow \text{r.}$$

Thus, the original message is **Butcher**.

## 3.6 The TVCG for an Affine Encryptions scheme

New versions of a hybrid TVCG – affine encryption scheme are proposed in two cases. First one used the EAVs and second one used ASCII values. These cases are explained as follows.

### 3.6.1. The TVCG for an Affine Encryptions Scheme Based on the EAVs.

The TVCG is employed on an affine encryption scheme to give new hybrid version of a symmetric encryption scheme which is called TVCG-AE scheme. Next examples explain the idea of this scheme.

**Example 3.6.1.1.** Suppose $M$ is the plaintext that is given by the word "**Lion**". Using affine cipher based on the EAVs in Table (3.1) of the letters, with the key is (a,b) = (15,3), one can compute the ciphertext by

$$E \equiv (ax+b)(mod\ 26) \equiv (15x+3)(mod\ 26)$$

$E(l) \equiv (15\times11+3)(mod\ 26) \Longrightarrow 168\ (mod\ 26) \equiv 12 \rightarrow M$

$E(i) \equiv (15\times8+3)(mod\ 26) \Longrightarrow 123\ (mod\ 26) \equiv 19 \rightarrow T$

$E(o) \equiv (15\times14+3)(mod\ 26) \Longrightarrow 213\ (mod\ 26) \equiv 5 \rightarrow F$

$E(n) \equiv (15\times13+3)(mod\ 26) \Longrightarrow 198\ (mod\ 26) \equiv 16 \rightarrow Q$

The ciphertext of $M$ is

$$Cp = \text{MTFQ.}$$

So, the ciphertext *Cp* of *M* forms a cycle graph MTFQ that is shown in Figure (3.23).

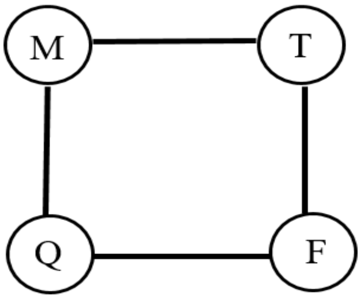


Figure 3.23. The cycle $C_4$ of the ciphertext MTFQ.

This cycle graph is represented as the triple vertex graph (TVG) based on the correct intersection of the vertices which have distance 2 as given in Definition () that is computed by

{M,T,F}∩{T,F,Q} = {T,F} ⟹ |{M,T,F}∩{T,F,Q}| = 2. So, there is an edge between M and Q. Thus, there exists an edge between the vertices MTF and TFQ in the $\text{TVG}_3(C_4)$.

{M,T,F}∩{F,Q,M} = {F,M} ⟹ |{M,T,F}∩{F,Q,M}| = 2. So, there is no an edge between M and Q. Thus, there is no an edge between the vertices MTF and FQM in the $\text{TVG}_3(C_4)$.

In similar way, all other intersected vertices are computed and determined which ones are created the edges of the $\text{TVG}_3(C_4)$.The $\text{TVG}(\text{C}_4)$ is shown in Figure (3.24) and sent to receiver by sender as a ciphertext.

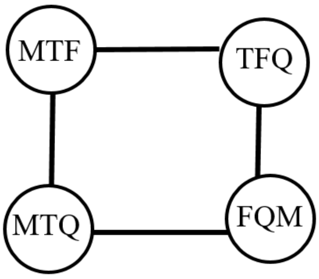


Figure 3.24. The TVG($C_4$) of the ciphertext MTFQ.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices MTF, TFQ, FQM, and MTQ. Based on the TVG($C_4$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is MTFQ.

Now, the decrypt the ciphertext: MTFQ is done using an affine cipher based on Table (3.1) as follows.

$$D \equiv a^{-1}(E-b)\ (mod\ 26).$$

So,

$D(M) \equiv 15^{-1}(12\text{-}3)(mod\ 26) \Longrightarrow 63\ (mod\ 26) \equiv 11 \rightarrow L$, where $15^{-1}(mod\ 26) \equiv 7$.

$D(T) \equiv 7(19\text{-}3)(mod\ 26) \Longrightarrow 112\ (mod\ 26) \equiv 8 \rightarrow I$

$D(F) \equiv 7(5\text{-}3)(mod\ 26) \Longrightarrow 14\ (mod\ 26) \equiv 14 \rightarrow O$

$D(Q) \equiv 7(16\text{-}3)(mod\ 26) \Longrightarrow 91\ (mod\ 26) \equiv 13 \rightarrow N$

Thus, the original message is Lion.

**Example 3.6.1.2.** Suppose *M* is the plaintext that is given by the word "**English**". Using Affine Cipher based on the EAVs in Table (3.1) of the letters, with the key is (a,b) = (3,5), one can compute the ciphertext by

$$E = (ax+b)(mod\ 26)$$

$E(E) \equiv (3\times4+5)(mod\ 26) \Longrightarrow 17\ (mod\ 26) \equiv 17 \rightarrow r$

$E(n) \equiv (3\times13+5)(mod\ 26) \Longrightarrow 44\ (mod\ 26) \equiv 18 \rightarrow s$

$E(g) \equiv (3\times6+5)(mod\ 26) \Longrightarrow 23\ (mod\ 26) \equiv 23 \rightarrow x$

$E(l) \equiv (3\times11+5)(mod\ 26) \Longrightarrow 38\ (mod\ 26) \equiv 12 \rightarrow m$

$E(i) \equiv (3\times8+5)(mod\ 26) \Longrightarrow 29\ (mod\ 26) \equiv 3 \rightarrow d$

$E(s) \equiv (3\times18+5)(mod\ 26) \Longrightarrow 59\ (mod\ 26) \equiv 7 \rightarrow h$

$E(h) \equiv (3\times7+5)(mod\ 26) \Longrightarrow 26\ (mod\ 26) \equiv 0 \rightarrow a$

The ciphertext of $M$ is

$$Cp = \text{rsxmdha}.$$

So, the ciphertext $Cp$ of $M$ forms a cycle graph rsxmdha that is shown in Figure (3.25).

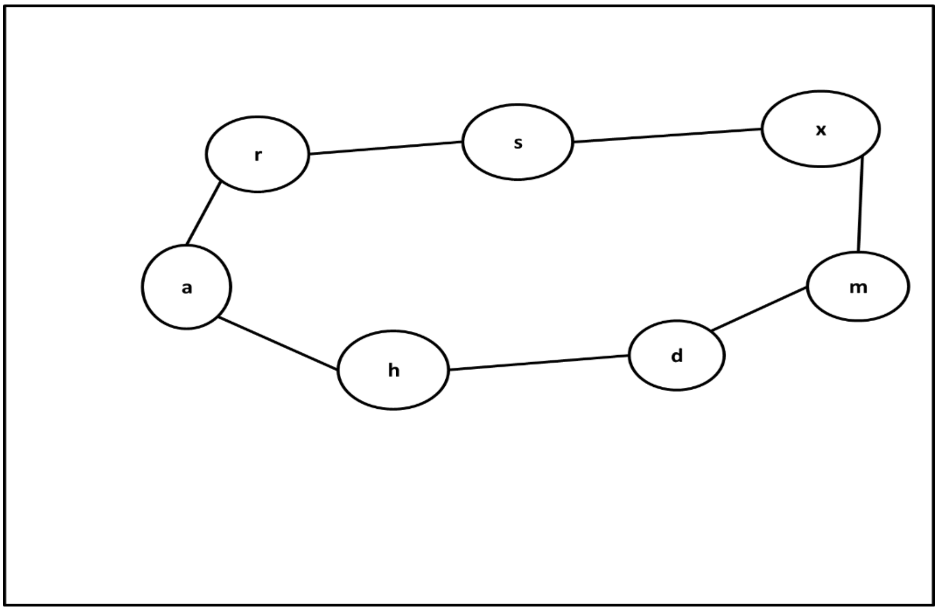
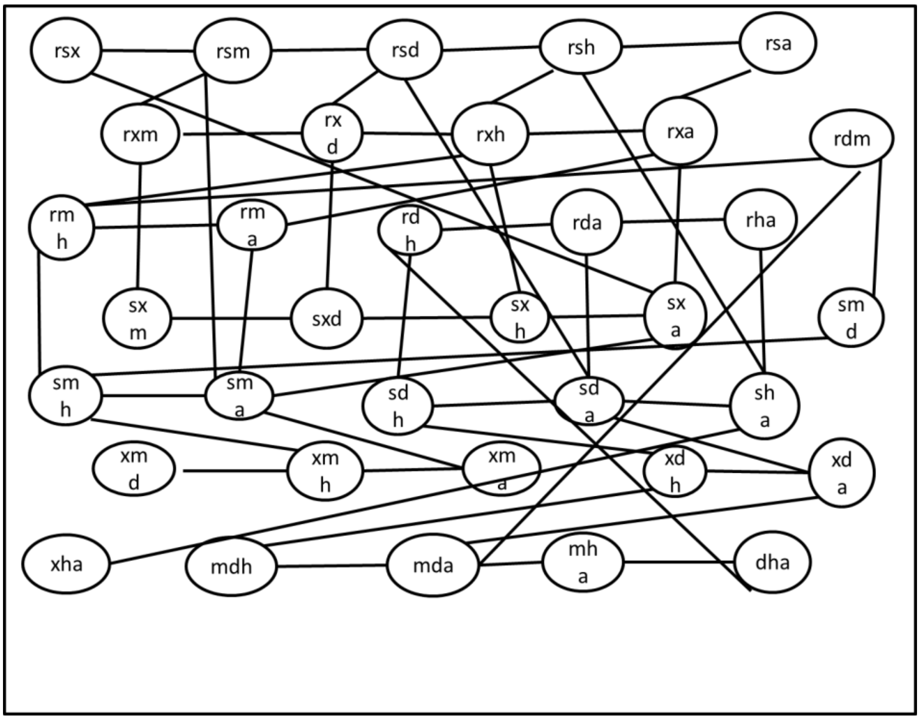


Figure 3.25. The cycle $C_7$ of the ciphertext rsxmdha.

This cycle is represented as the triple vertex graph TVG($C_7$) that is



given in Figure (3.26) and sent to receiver by sender.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices rsx, rsm, rsd, rsh, rsa, rxm, rxd, rxh, rxa, rmd, rmh, rma, rdh, rda, rha, sxm, sxd, sxh, sxa, smd, smh, sma, sdh, sda, sha, xmd, xmh, xma, xdh, xda, xha, mdh, mda, mha and dha based on the TVCG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is rsxmdha.

Now, the decrypt the ciphertext rsxmdha is done using an affine cipher based on Table (3.1) as follows.

$$D \equiv a^{-1}(E\text{-}b)(mod\ 26).$$

$D(r) \equiv 3^{-1}(17\text{-}5)(mod\ 26) \Longrightarrow 108\ (mod\ 26) \equiv 4 \rightarrow$ E, where $3^{-1}(mod\ 26) \equiv 9$.

$D(s) \equiv 9(18-5)(mod\ 26) \Longrightarrow 117\ (mod\ 26) \equiv 13 \rightarrow \mathrm{n}$

$D(x) \equiv 9(23-5)(mod\ 26) \Longrightarrow 162\ (mod\ 26) \equiv 6 \rightarrow \mathrm{g}$

$D(m) \equiv 9(12-5)(mod\ 26) \Longrightarrow 63\ (mod\ 26) \equiv 11 \rightarrow \mathrm{l}$

$D(d) \equiv 9(3-5)(mod\ 26) \Longrightarrow -18\ (mod\ 26) \equiv 8 \rightarrow \mathrm{i}$

$D(h) \equiv 9(7-5)(mod\ 26) \Longrightarrow 18\ (mod\ 26) \equiv 18 \rightarrow \mathrm{s}$

$D(a) \equiv 9(0-5)(mod\ 26) \Longrightarrow -45\ (mod\ 26) \equiv 7 \rightarrow \mathrm{h}$

Thus, the original message is English.

### 3.6.2. The TVCG for Affine Encryption Scheme Based on the ASCII Values.

The TVCG can be applied for an affine encryptions scheme using the ASCII values that are given in Table (3.2). This application are done with two simple examples to give new hybrid version of an affine cipher and TVCG. This version is discussed through the following examples.

**Example 3.6.2.1.** Suppose *M* is the plaintext that is given by the word "**Number**". using affine cipher based on the ASCII Table (3.2) of the letters, with the key is (a,b) = (9,2), one can compute the ciphertext by

$$E \equiv (ax+b)(mod\ 127)$$

$E(N) \equiv (9\times78+2)(mod\ 127) \Longrightarrow 704\ (mod\ 127) \equiv 69 \rightarrow E.$

$E(u) \equiv (9\times117+2)(mod\ 127) \Longrightarrow 1055\ (mod\ 127) \equiv 39 \rightarrow$ '

$E(m) \equiv (9\times109+2)(mod\ 127) \Longrightarrow 983\ (mod\ 127) \equiv 94 \rightarrow$ ^

$E(b) \equiv (9\times98+2)(mod\ 127) \Longrightarrow 884\ (mod\ 127) \equiv 122 \rightarrow z$

$E(e) \equiv (9\times101+2)(mod\ 127) \Longrightarrow 911\ (mod\ 127) \equiv 22 \rightarrow$ *synchornousidle* =¥

$E(r) \equiv (9\times114+2)(mod\ 127) \Longrightarrow 1028\ \ (mod\ 127) \equiv 12 \rightarrow$ *Np from feed new page*→ £

The ciphertext of *M* is

*C* = ***E,^Z¥£***

So, the ciphertext *Cp* of *M* forms a cycle graph ***E,^Z¥£*** as shown in Figure (3.27).

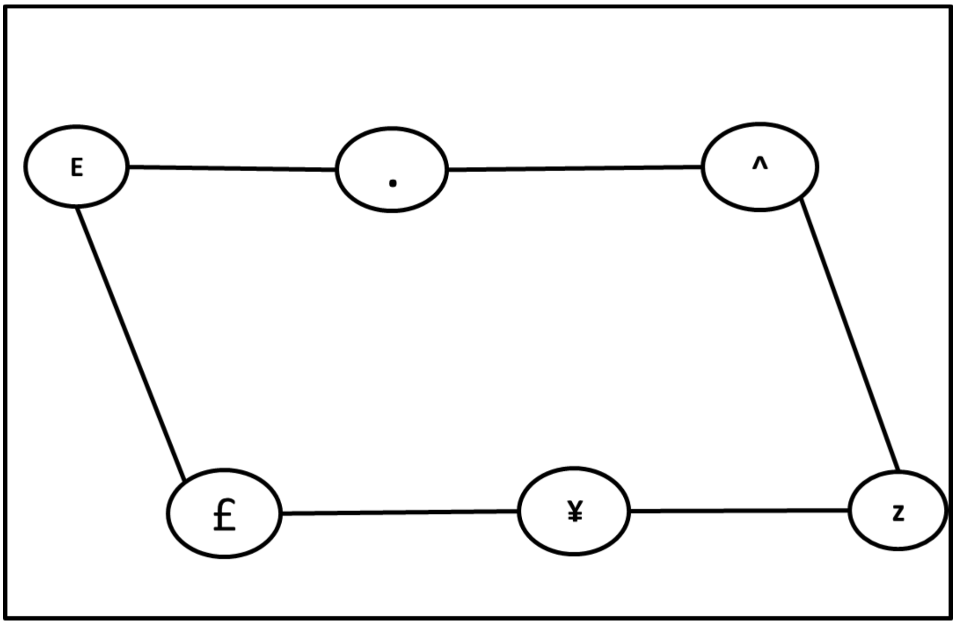


Figure 3.27. The TVG($C_6$) of the ciphertext ***E,^Z¥£***.

After the second user (receiver) receives triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices *E,^* , *E,Z* , *E,¥* , *E,£* , *E^Z* , *E6¥* , *E6£* , *EZ¥* , *EZ£* , *E¥£* , **,**^Z , **,**^¥ , ,^£ , **,**Z¥ , ,Z£ , **,**¥£ , ^Z¥ , ^Z£ , ^¥£ , & Z¥£ . Based on the TVCG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is ***E,^Z¥£***.

Now, the decrypt the ciphertext ***E,^Z¥£***is done using an affine cipher based on the ASCII Table (3.2) of the letters as follows.

$$D \equiv a^{-1}(E\text{-}b)(mod\ 127).$$

So,

$D(E)$ ≡$9^{-1}$(69-2)($mod$ 127) ⟹ 7571 ($mod$ 127) ≡ 78 → $N$, where $9^{-1}$ ($mod$ 127) ≡ 113.

$D($ **,** $)$ ≡ 113(39-2)($mod$ 127) ⟹ 4181 ($mod$ 127) ≡ 117 → $u$

$D(\text{^})$ ≡ 113(94-2)($mod$ 127) ⟹ 10396 ($mod$ 127) ≡ 109 → $m$

$D(Z)$ ≡ 113(122-2)($mod$ 127) ⟹ 13560 ($mod$ 127) ≡ 98 → $b$

$D$(¥) ≡ 113(22-2)($mod$ 127) ⟹ 2260 ($mod$ 127) ≡ 101 → $e$

$D$(£) ≡ 113(12-2)($mod$ 127) ⟹ 1130 ($mod$ 127) ≡ 114 → $r$

Thus, the original message is Number.

**Example 3.6.2.2.** Suppose $M$ is the plaintext that is given by the word **mistake** using affine cipher based on the ASCII Table (3.2) of the letters, with the key is (a,b) = (5,29), one can compute the ciphertext by

$$E \equiv (ax+b)(mod\ 127)$$

$E(m)$ ≡ (5×77+29)($mod$ 127) ⟹ 414 ($mod$ 127) ≡ 33→*!*

$E$(i) ≡= (5×105+29)($mod$ 127) ⟹ 554 ($mod$ 127) ≡ 46 → **,**

$E(s)$ ≡ (5×115+29)($mod$ 127) ⟹ 604 ($mod$ 127) ≡ 96 → **.**

$E(t)$ ≡ (5×116+29)($mod$ 127) ⟹ 609 ($mod$ 127) ≡ 101 → $e$

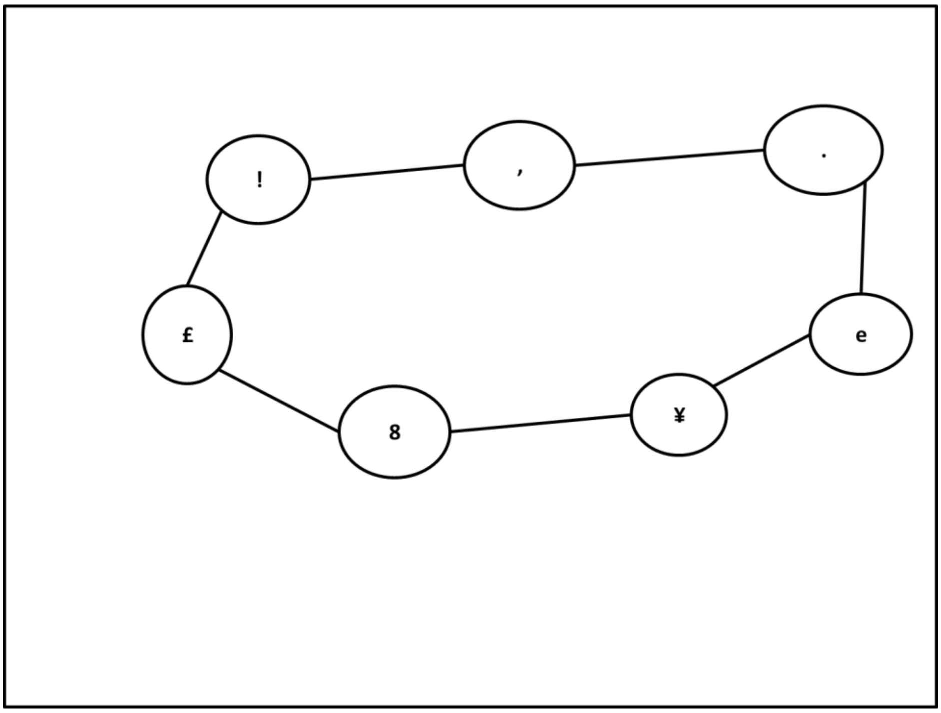
$E(a)$ ≡ (5×97+29)($mod$ 127) ⟹ 514 ($mod$ 127) ≡ 6 → *Acknowlege=¥*

$E(k)$ ≡ (5×107+29)($mod$ 127) ⟹ 564 ($mod$ 127) ≡ 56 → *8*

$E(e)$ ≡ (5×101+29)($mod$ 127) ⟹ 534 ($mod$ 127) ≡ 26 → *substitute=£*

The **!,. e¥8£** of *M* is

$$Cp = \textbf{!,.e¥8£} \; .$$



So, the ciphertext *Cp* of M forms a cycle graph **!,.e¥8£** that is shown in Figure (3.29).

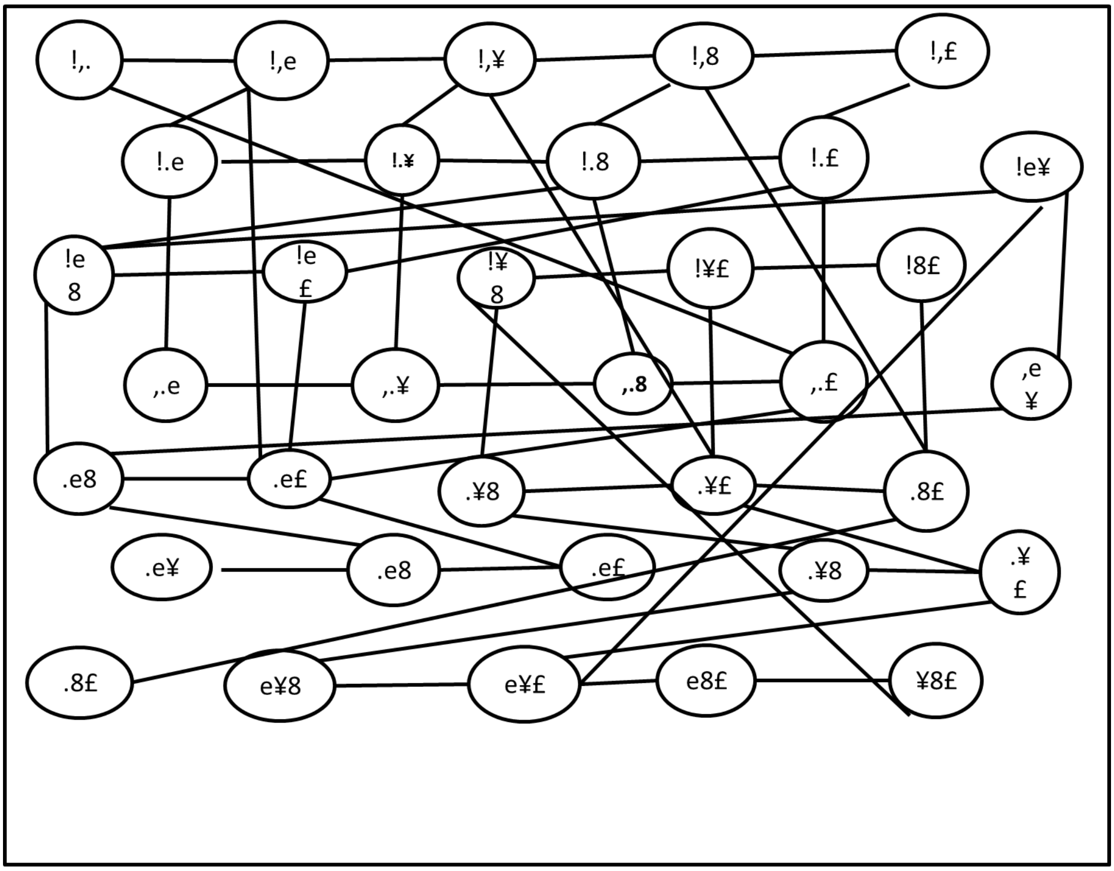


Figure 3.30. The cycle $C_7$ of the ciphertext **!,.e¥8£.**

This cycle is represented as the triple vertex graph TVG($C_7$) that is given in Figure (3.30) and sent to receiver by sender.

Figure 3.30. The TVG($C_7$) of the ciphertext **!,.e¥8£.**

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices !,. , !,e , !,¥ , !,8 , !,£ , !.e , !.¥ , !.8 , !.£ , !e¥ , !e8 , !e£ , !¥8 , !¥£ , !8£ , ,.e , ,.¥ , ,.8 , ,.£ , ,e¥ , ,e8 , ,e£ , .e¥ , .e8 , .e£ , .¥8 , .8£ , e¥8 , e¥£ , & ¥8£ . Based on the TVPG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **!,.e¥8£**.

Now, the decrypt the ciphertext **!,.e¥8£** is done using an affine cipher based on the ASCII Table (3.2) of the letters as follows.

$$D(x) \equiv a^{-1}(E\text{-}b)(mod\ 127)$$

Invers of ($a$): $5^{-1}\ (mod\ 127) \equiv 51$

$D(!) \equiv 5^{-1}(33\text{-}29)(mod\ 127) \Longrightarrow 204\ (mod\ 127) \equiv 77 \rightarrow M$ where $5^{-1}$ $(mod\ 127) \equiv 51$.

$\mathrm{D}(') \equiv 51(46\text{-}29)(mod\ 127) \Longrightarrow 867\ (mod\ 127) \equiv 105 \rightarrow i$

$D(.\ ) \equiv 51(96\text{-}29)(mod\ 127) \Longrightarrow 3417\ (mod\ 127) \equiv 115 \rightarrow s$

$D(e) \equiv 51(101\text{-}29)(mod\ 127) \Longrightarrow 3672\ (mod\ 127) \equiv 116 \rightarrow t$

$D($ ¥ =Acknowledge$) \equiv 51(6\text{-}29)(mod\ 127) \Longrightarrow -1173\ (mod\ 127) \equiv\ \rightarrow 97\ a$

$D(8) \equiv 51(56\text{-}29)(mod\ 127) \Longrightarrow 1377\ \ (mod\ 127) \equiv 107 \rightarrow k$

$D($£= substitute$) \equiv 51(26\text{-}29)(mod\ 127) \Longrightarrow -\ 153(mod\ 127) \equiv 101 \rightarrow e$

Thus, the original message is mistake.

## 3.7 The TVCG - Hill Encryption Scheme

In this section, two encryption schemes are proposed based on the TVCG for Hill encryption scheme. These schemes are explained follows.

### 3.7.1. The TVCG - Hill Encryption Scheme Based on the EAVs

The idea of the proposed hybrid TVCG-HE scheme using the EAVs is explained in the following examples.

**Example 3.7.1.1.** Suppose *M* is the plaintext that is given by the word "**help**". Let K be a secret key which is given by 2×2 matrix, namely

$$K=\begin{bmatrix}3 & 5\\ 7 & 6\end{bmatrix}.$$

Now, it is easy to divide *M* into 2 blocks, each block has length 2. In other words, it can write *M* by

$$M = (M_1,M_2) = \text{(he lp)}.$$

Using the EAVs in Table (3.1), one can represent the letters in

$$(M_1,M_2) = \text{(he lp)}$$

Into

$$\binom{h}{e}\binom{7}{4},\binom{l}{p}\binom{11}{15}.$$

The ciphertext *Cp* is computed by $Cp \equiv K \cdot M_i$ (*mod* 26). With more details for *i*=1,2,3, then

$$\begin{bmatrix}3 & 5\\ 7 & 6\end{bmatrix}\binom{7}{4} \equiv \binom{41}{73}(mod\ 26) \equiv \binom{15}{21} \longrightarrow \binom{P}{V}$$

$$\begin{bmatrix}3 & 5\\ 7 & 6\end{bmatrix}\binom{11}{15} \equiv \binom{108}{167}(mod\ 26) \equiv \binom{4}{11} \longrightarrow \binom{E}{L}$$

The ciphertext of *M* is

$$Cp = \text{PVEL}.$$

So, the ciphertext *Cp* of *M* forms a cycle graph PVEL that is shown in Figure (3.32).

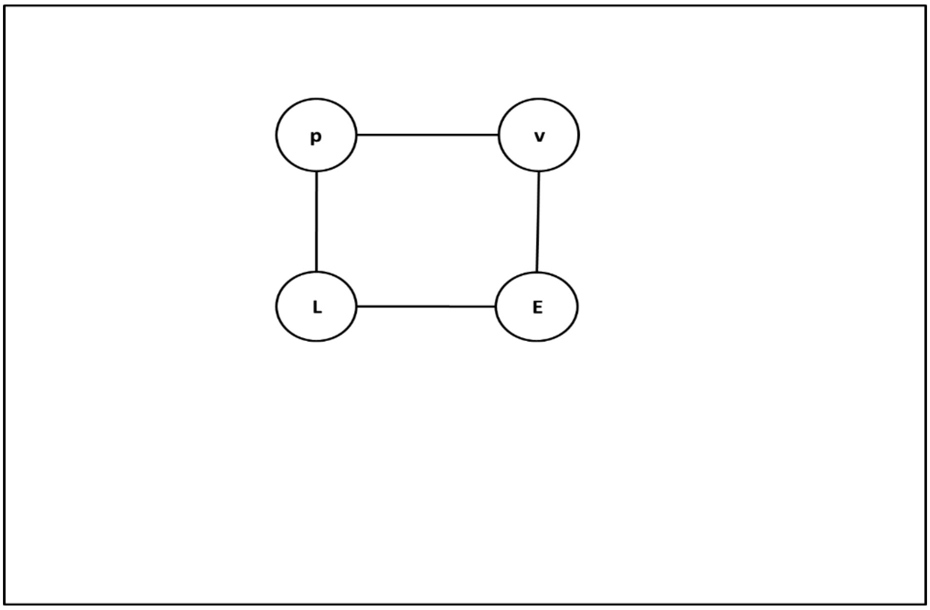


Figure 3.32. The cycle $C_4$ of the ciphertext PVEL.

This cycle is represented as the triple vertex cycle graph (TVCG) that is given in Figure (3.32) and sent to receiver by sender.

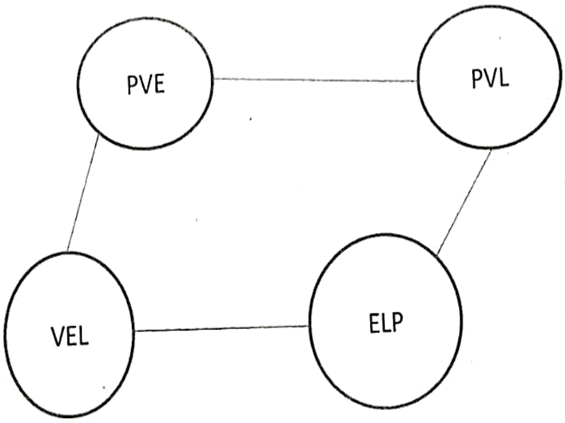


Figure 3.32. The TVG of the cycle $C_4$ of the ciphertext PVEL.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices PVE, PVL, PEL and VEL based on the TVCG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is PVEL.

Now, the decryption of the ciphertext is done using Hill cipher based on Table (3.1). Second user computes the determinate of the key $K$ by

Det($K$) = 18-35($mod$ 26) $\Longrightarrow$ 9. The inverse element of Det($K$) is computed by $9^{-1}$ ($mod$ 26) ≡ 3. The key inverse $(K)^{-1}$ has been computed by

$$(K)^{-1} \equiv 3\begin{bmatrix} 6 & 21 \\ 19 & 3 \end{bmatrix} \Longrightarrow \begin{bmatrix} 18 & 63 \\ 57 & 9 \end{bmatrix} (mod 26) \equiv \begin{bmatrix} 18 & 11 \\ 5 & 9 \end{bmatrix},$$

Since the ciphertext is given by

$$\begin{pmatrix} P \\ V \end{pmatrix} = \begin{pmatrix} 15 \\ 21 \end{pmatrix}, \begin{pmatrix} E \\ L \end{pmatrix} = \begin{pmatrix} 4 \\ 11 \end{pmatrix}.$$

So,

$$\begin{bmatrix} 18 & 11 \\ 5 & 9 \end{bmatrix}\begin{pmatrix} 15 \\ 21 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 501 \\ 264 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 7 \\ 4 \end{pmatrix} \longrightarrow \begin{pmatrix} H \\ E \end{pmatrix}$$

$$\begin{bmatrix} 18 & 11 \\ 5 & 9 \end{bmatrix}\begin{pmatrix} 4 \\ 11 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 193 \\ 119 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 11 \\ 15 \end{pmatrix} \longrightarrow \begin{pmatrix} L \\ P \end{pmatrix}.$$

Then the plaintext is Help.

**Example 3.7.1.2.** Suppose $M$ is the plaintext that is given by the sentence “**We are safe”**. Let K be a secret key which is given by 3×3 matrix, namely

$$K= \begin{bmatrix} 0 & 11 & 15 \\ 7 & 0 & 1 \\ 4 & 19 & 0 \end{bmatrix}.$$

Now, it is easy to divide $M$ into 3 blocks, Each block has length 2. In other words, it can write $M$ by

$$M = (M_1,M_2,M_3) = \text{(Wea} \quad \text{res} \quad \text{afe).}$$

Using the EAVs in Table (3.1), one can represent the letters

$(M_1,M_2,M_3)$ = (Wea res afe).

into

$$\begin{pmatrix} W \\ e \\ a \end{pmatrix} = \begin{pmatrix} 22 \\ 4 \\ 0 \end{pmatrix}, \begin{pmatrix} r \\ e \\ s \end{pmatrix} = \begin{pmatrix} 17 \\ 4 \\ 18 \end{pmatrix}, \begin{pmatrix} a \\ f \\ e \end{pmatrix} = \begin{pmatrix} 0 \\ 5 \\ 4 \end{pmatrix}.$$

The ciphertext *Cp* is computed by $C \equiv K{*}M_i$ (*mod 26*), in the more details for *i*=1,2,3, then

$$\begin{bmatrix} 0 & 11 & 15 \\ 7 & 0 & 1 \\ 4 & 19 & 0 \end{bmatrix} \begin{pmatrix} 22 \\ 4 \\ 0 \end{pmatrix} \equiv \begin{pmatrix} 44 \\ 154 \\ 164 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 18 \\ 24 \\ 8 \end{pmatrix} \longrightarrow \begin{pmatrix} S \\ Y \\ I \end{pmatrix}$$

$$\begin{bmatrix} 0 & 11 & 15 \\ 7 & 0 & 1 \\ 4 & 19 & 0 \end{bmatrix} \begin{pmatrix} 17 \\ 4 \\ 18 \end{pmatrix} \equiv \begin{pmatrix} 314 \\ 137 \\ 144 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 2 \\ 7 \\ 14 \end{pmatrix} \longrightarrow \begin{pmatrix} C \\ H \\ O \end{pmatrix}$$

$$\begin{bmatrix} 0 & 11 & 15 \\ 7 & 0 & 1 \\ 4 & 19 & 0 \end{bmatrix} \begin{pmatrix} 0 \\ 5 \\ 4 \end{pmatrix} \equiv \begin{pmatrix} 115 \\ 4 \\ 95 \end{pmatrix} (mod\ 26) \equiv \begin{pmatrix} 11 \\ 4 \\ 17 \end{pmatrix} \longrightarrow \begin{pmatrix} L \\ E \\ R \end{pmatrix}.$$

The ciphertext of *M* is

*Cp* = SYICHOLER.

So, the ciphertext *Cp* of *M* forms a cycle graph SYICHOLER that is shown in Figure (3.33)

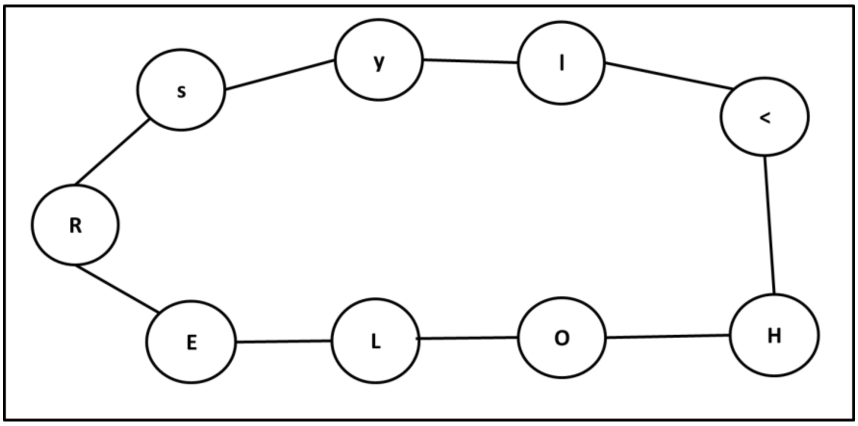


Figure 3.33. The cycle $C_6$ of the ciphertext SYICHOLER.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.34) and sent to receiver by sender.

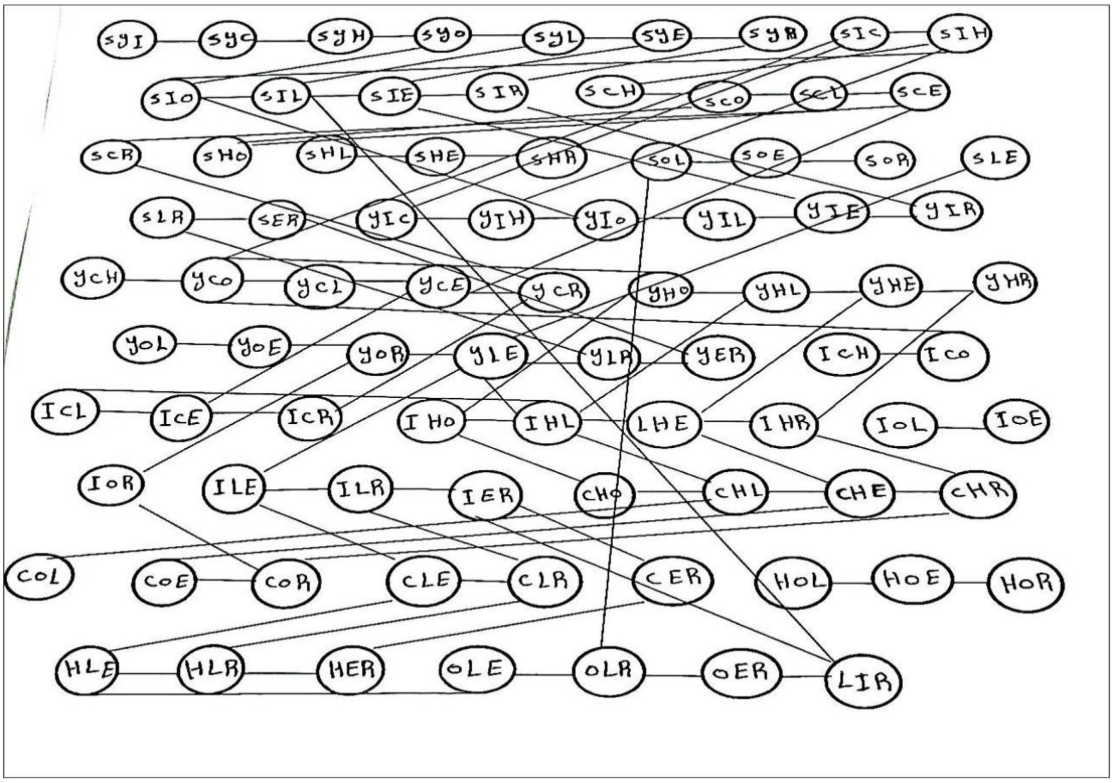


Figure 3.34. The TVG($C_6$) of the ciphertext SYICHOLER.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices SYI, …. based on the TVPG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is SYICHOLER.

Now, the decryption of the ciphertext SYICHOLER is done using Hill cipher based on Table (3.1). Second user computes the determinate at the key $K$ by

Det($K$) = 11. The inverse of Det($K$) is computed by $11^{-1}$ ($mod$ 26) ≡ 19. The key inverse $(K)^{-1}$ has been compute by

$$(K)^{-1} \equiv 19\begin{bmatrix} -19 & 285 & 11 \\ 4 & -60 & 105 \\ 133 & 44 & -77 \end{bmatrix}(mod\ 26) \Longrightarrow \begin{bmatrix} 3 & 7 & 1 \\ 24 & 4 & 19 \\ 5 & 4 & 19 \end{bmatrix}.$$

Since the ciphertext is given by

$$\begin{pmatrix} S \\ Y \\ I \end{pmatrix} = \begin{pmatrix} 18 \\ 24 \\ 8 \end{pmatrix}, \begin{pmatrix} C \\ H \\ O \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 14 \end{pmatrix}, \begin{pmatrix} L \\ E \\ R \end{pmatrix} = \begin{pmatrix} 11 \\ 4 \\ 17 \end{pmatrix}.$$

$$\begin{bmatrix} 3 & 7 & 1 \\ 24 & 4 & 19 \\ 5 & 4 & 19 \end{bmatrix}\begin{pmatrix} 18 \\ 24 \\ 8 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 230 \\ 680 \\ 338 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 22 \\ 4 \\ 0 \end{pmatrix} = \begin{pmatrix} W \\ e \\ a \end{pmatrix},$$

$$\begin{bmatrix} 3 & 7 & 1 \\ 24 & 4 & 19 \\ 5 & 4 & 19 \end{bmatrix}\begin{pmatrix} 2 \\ 7 \\ 14 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 69 \\ 342 \\ 304 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 17 \\ 4 \\ 18 \end{pmatrix} = \begin{pmatrix} r \\ e \\ s \end{pmatrix},$$

$$\begin{bmatrix} 3 & 7 & 1 \\ 24 & 4 & 19 \\ 5 & 4 & 19 \end{bmatrix}\begin{pmatrix} 11 \\ 4 \\ 17 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 78 \\ 603 \\ 394 \end{pmatrix}(mod\ 26) \equiv \begin{pmatrix} 0 \\ 5 \\ 4 \end{pmatrix} = \begin{pmatrix} a \\ f \\ e \end{pmatrix}.$$

The plaintext is the sentence “**We are safe**”.

### 3.7.2. The TVCG for Hill Encryption Scheme Based on the ASCII Values.

The idea of the proposed hybrid TVCG-HE scheme using the ASCII values is explained in the following examples.

**Example 3.7.2.1.** Suppose $M$ is the plaintext that is given by the word "**Office**". Let K be a secret key which is given by 2×2 matrix, namely

$$K= \begin{bmatrix} 17 & 27 \\ 2 & 5 \end{bmatrix}.$$

Now, it is easy to divide $M$ into 3 blocks, Each block has length 2. In other words, it can write $M$ by

$$M = (M_1,M_2,M_3) = (\text{Of} \quad \text{fi} \quad \text{ce}).$$

Using the ASCII Table (3.2), one can represent the letters

$$(M_1,M_2,M_3) = (\text{Of} \quad \text{fi} \quad \text{ce})$$

into

$$\begin{pmatrix} O \\ f \end{pmatrix}\begin{pmatrix} 79 \\ 102 \end{pmatrix} \Bigg| \begin{pmatrix} f \\ i \end{pmatrix}\begin{pmatrix} 102 \\ 105 \end{pmatrix} \Bigg| \begin{pmatrix} c \\ e \end{pmatrix}\begin{pmatrix} 99 \\ 101 \end{pmatrix}.$$

The ciphertext $Cp$ is computed by $Cp \equiv K \cdot M_i$ $(mod$ 127). In more details, for $i$=1,2,3, then

$$\begin{bmatrix} 17 & 27 \\ 2 & 5 \end{bmatrix}\begin{pmatrix} 79 \\ 102 \end{pmatrix} \equiv \begin{pmatrix} 17 \times 79 + 27 \times 102 \\ 2 \times 79 + 5 \times 102 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 4097 \\ 668 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 33 \\ 33 \end{pmatrix} \longrightarrow \begin{pmatrix} ! \\ ! \end{pmatrix}$$

$$\begin{bmatrix} 17 & 27 \\ 2 & 5 \end{bmatrix}\begin{pmatrix} 102 \\ 105 \end{pmatrix} \equiv \begin{pmatrix} 17 \times 102 + 27 \times 105 \\ 2 \times 102 + 5 \times 105 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 4569 \\ 729 \end{pmatrix} (mod 127) \equiv \begin{pmatrix} 124 \\ 94 \end{pmatrix} \rightarrow \begin{pmatrix} | \\ \wedge \end{pmatrix}$$

$$\begin{bmatrix} 17 & 27 \\ 2 & 5 \end{bmatrix}\begin{pmatrix} 99 \\ 101 \end{pmatrix} \equiv \begin{pmatrix} 17 \times 99 + 27 \times 101 \\ 2 \times 99 + 5 \times 101 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 4410 \\ 703 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 92 \\ 68 \end{pmatrix} \longrightarrow \begin{pmatrix} \backslash \\ D \end{pmatrix}$$

The ciphertext of *M* is

$Cp$ =!!|^\D.

So, the ciphertext *Cp* of *M* forms a cycle graph !!|^\D that is shown in Figure (3.35).

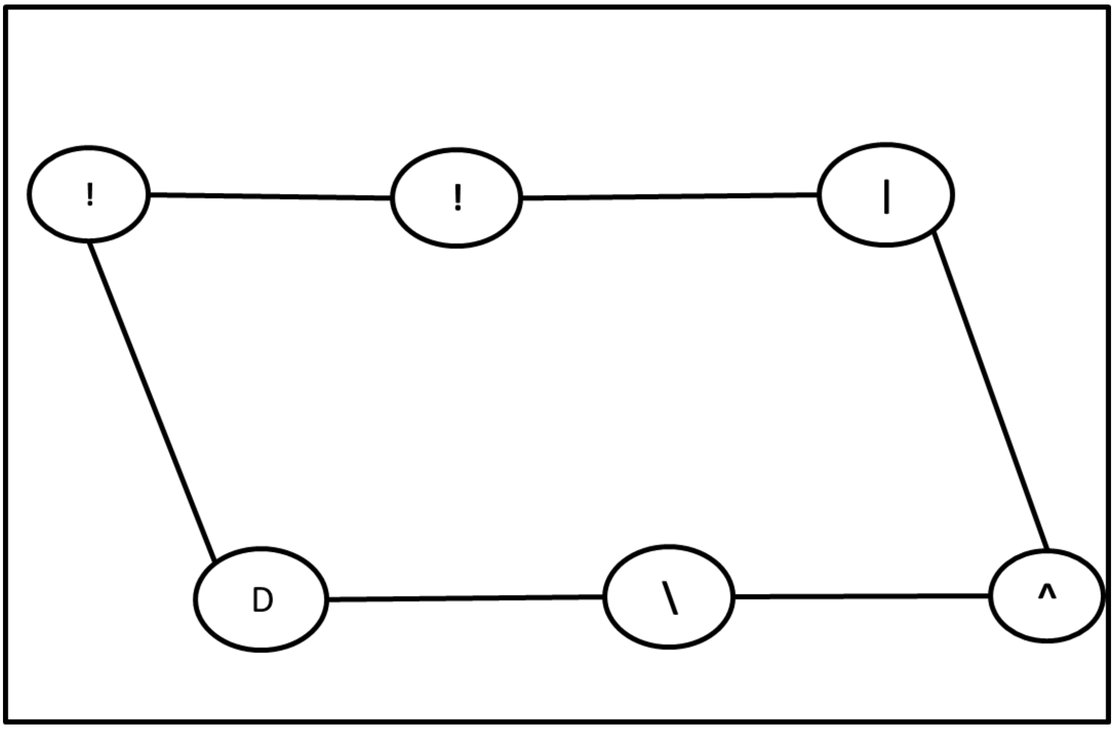


Figure 3.35. The cycle $C_6$ of the ciphertext !!|^\D.

This cycle is represented as the triple vertex graph (TVG) that is given in Figure (3.36) and sent to receiver by sender.

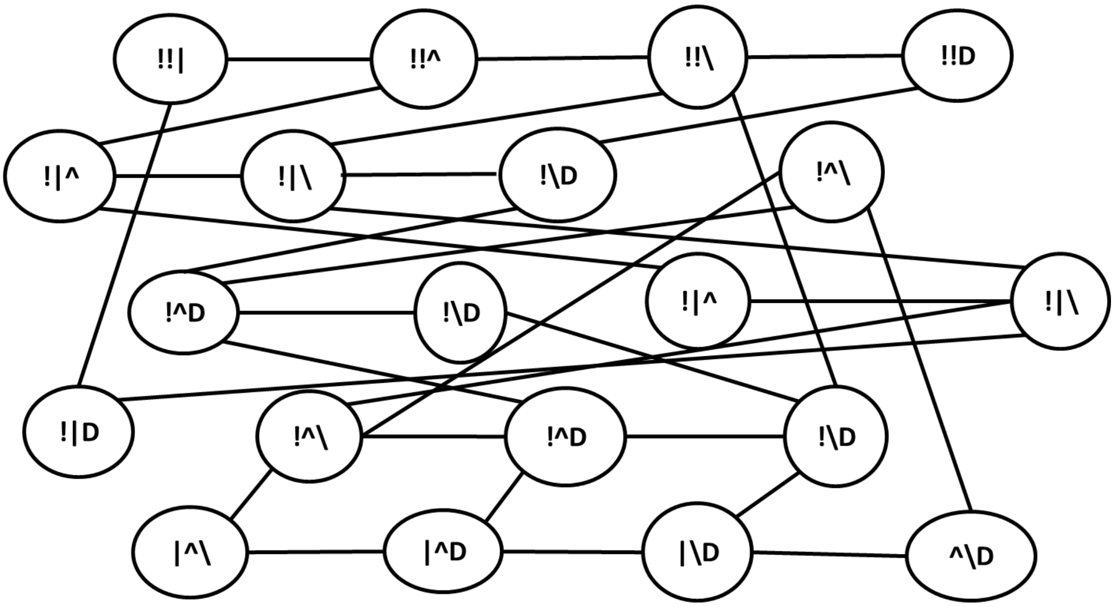


Figure 3.36. The TVCG of the cycle $C_6$ of the ciphertext !!|^\D.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices !!|, !!^, !!\, !!D, !|^, !|\, !\D, !^\, !^D, !\D, !|^, !|\, !|D, !^\, !^D, !\D, |^\, |^\, |\D, and ^\D. based on the TVPG. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is !!|^\D.

Now, the decryption of the ciphertext !!|^\D is done using Hill cipher based on the ASCII Table (3.2). Second user computes the determinate of the key *K* by

Det($K$) = $(17\times5)-(2\times27)$ ($mod$ 127) $\Longrightarrow$ 31($mod$ 127) $\equiv$ 31. The inverse of Det($K$) is computed by $31^{-1}$ ($mod$ 127) $\equiv$ 41. The key inverse $(K)^{-1}$ has been compute by

$(K)^{-1} \equiv 41\begin{bmatrix} 5 & -27 \\ -2 & 17 \end{bmatrix} \Longrightarrow \begin{bmatrix} 205 & -1107 \\ -82 & 697 \end{bmatrix} (mod\ 127) \equiv \begin{bmatrix} 78 & 36 \\ 45 & 62 \end{bmatrix}$,

where $\begin{bmatrix} 5 & -27 \\ -2 & 17 \end{bmatrix}$ is adjunct matrix of $K$.

Since the ciphertext is given by

$$\begin{pmatrix} ! \\ ! \end{pmatrix}\begin{pmatrix} 33 \\ 33 \end{pmatrix} \mid \begin{pmatrix} | \\ \wedge \end{pmatrix}\begin{pmatrix} 124 \\ 94 \end{pmatrix} \mid \begin{pmatrix} \backslash \\ D \end{pmatrix}\begin{pmatrix} 92 \\ 68 \end{pmatrix}.$$

$$\begin{bmatrix} 78 & 36 \\ 45 & 62 \end{bmatrix}\begin{pmatrix} 33 \\ 33 \end{pmatrix} \equiv \begin{pmatrix} 78 \times 33 + 36 \times 33 \\ 45 \times 33 + 62 \times 33 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 3762 \\ 3531 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 79 \\ 102 \end{pmatrix} \longrightarrow \begin{pmatrix} O \\ f \end{pmatrix}$$

$$\begin{bmatrix} 78 & 36 \\ 45 & 62 \end{bmatrix}\begin{pmatrix} 124 \\ 94 \end{pmatrix} \equiv \begin{pmatrix} 78 \times 124 + 36 \times 94 \\ 45 \times 124 + 62 \times 94 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 13056 \\ 11408 \end{pmatrix} (mod 127) \equiv \begin{pmatrix} 102 \\ 105 \end{pmatrix} \rightarrow \begin{pmatrix} f \\ i \end{pmatrix}$$

$$\begin{bmatrix} 78 & 36 \\ 45 & 62 \end{bmatrix}\begin{pmatrix} 92 \\ 68 \end{pmatrix} \equiv \begin{pmatrix} 78 \times 92 + 36 \times 68 \\ 45 \times 92 + 62 \times 68 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 9624 \\ 8356 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 99 \\ 101 \end{pmatrix} \longrightarrow \begin{pmatrix} c \\ e \end{pmatrix}$$

Then the original plaintextt is Office.

**Example 3.7.2.2.** Suppose *M* is the plaintext that is given by the word "**Twenty** Let K be a secret key which is given by 3×3 matrix, namely :

K = $\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}$

Now, it is easy to divide M into 3 blocks, Each block has length 2. In other words, it can write M by = $(M_1,M_2)$

M = (Twe nty)

Using the ASCII Table (3.2) one can represent the litters in

$(M_1, M_2)$ = (Twe nty)

$$\begin{pmatrix} T \\ w \\ e \end{pmatrix}\begin{pmatrix} 84 \\ 119 \\ 101 \end{pmatrix}\begin{pmatrix} n \\ t \\ y \end{pmatrix}\begin{pmatrix} 110 \\ 116 \\ 121 \end{pmatrix}$$

The ciphertext C is computed by $C \equiv K{*}P_i$ ($mod$ 127), in the more details for $i$=1,2,3.

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}\begin{pmatrix} 84 \\ 119 \\ 101 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 84 + 2 \times 119 + 3 \times 101 \\ 0 \times 84 + 1 \times 119 + (-1) \times 101 \\ 2 \times 84 + 2 \times 119 + 2 \times 101 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 625 \\ 18 \\ 608 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 117 \\ 18 \\ 100 \end{pmatrix} \longrightarrow \begin{pmatrix} u \\ \text{device control 2} \\ d \end{pmatrix}$$

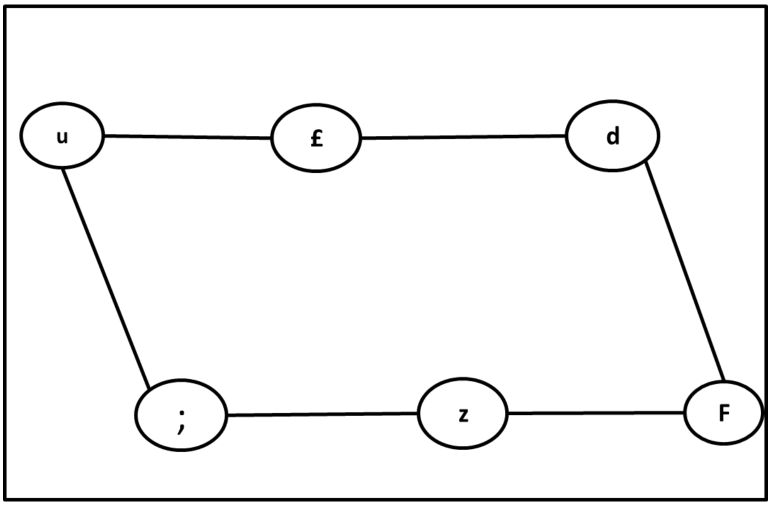
$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}\begin{pmatrix} 110 \\ 116 \\ 121 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 110 + 2 \times 116 + 3 \times 121 \\ 0 \times 110 + 1 \times 116 + (-1) \times 121 \\ 2 \times 121 + 2 \times 116 + 2 \times 121 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 705 \\ -5 \\ 694 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 70 \\ 122 \\ 59 \end{pmatrix} \longrightarrow \begin{pmatrix} F \\ z \\ ; \end{pmatrix}$$

The ciphertext of M is

$C$ = u device control 2 dFz;.

So, the cipher text C of M forms a cycle graph udevicecontrol2dFz; that



is shown in Figure (3.37).

This is represented as the triple vertex graph (TVG) that is given in

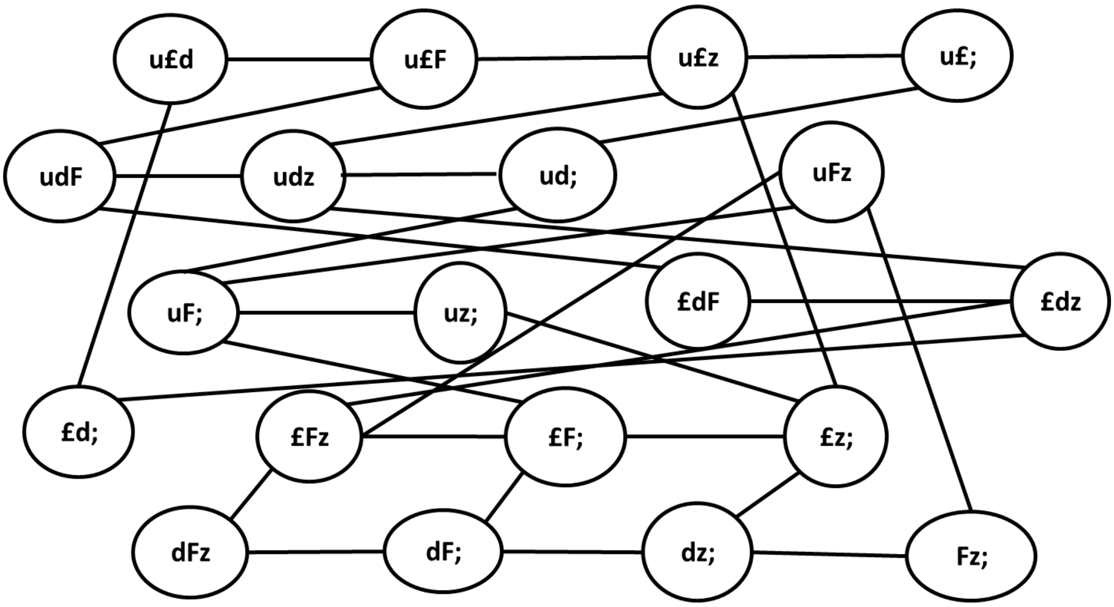


Figure (3.38) and sent to receiver by send.

After the second user (receiver) receives a triple vertex graph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices u£d , u£F, u£z , u£; , udF , udz , ud; , uFz , uF; , uz; , £dF , £dz , £d; , £Fz , £F; , £z; , dFz , dF; , dz; ,& Fz;

## 3.8 3.8 Security Considerations

On a new proposed PES, the security considerations are determined based on secret generating of a cycle graph $C_n$ which the attackers want to know it. If they determine the ciphertext that is computed as TVCG. So, Eve needs to guess the vertices of a cycle graph $C_n$ and also the rules of a secret key. Therefore, in study case, there are 26 possible probabilities $P$ to form the correct cycle graph. Thus, the total probability can be determined by

$$C_5^{26} = \frac{26!}{5!\,(26-5)!} = 65780.$$

where $P$ = 26 and the vertices of $C_n$ is 5. So, there are 65780 case are corresponded to the plaintext, one of them is correct one. So, more secure communication with PES using the TVCG. If one apply the proposed version with ASCII values, then there are 254231775 cases, one of them is correct to recover the original plaintext because

$$C_5^{127} = \frac{127!}{5!\,(127-5)!} = 254231775.$$

# Chapter Four

# The Triple Vertex Cycle Digraph for Encryption Schemes

## 4.1 Introduction

In this chapter, two new definitions of special graphs which are called a triple vertex digraph (TVDG) and the triple vertex cycle digraph (TVCDG). These graphs are used to propose new versions of the symmetric encryption schemes. Several examples are presented and discussed to explain these definitions and symmetric encryption schemes.

## 4.2 The Triple Vertex Digraph

In this section, new definition of a special digraph is proposed, this digraph called a triple vertex digraph (TVDG) which discusses as follows.

Definition 4.2.1. Let $G(V,E)$ be a simple digraph. The order of $G$ is equal $n$ with $n \geq 3$. The triple vertex digraph TVDG$_3$($G$) is a digraph whose vertex set $V$ such that two vertices $\{x,y,z\}$ and $\{u,v,t\}$ are adjacent if and only if $|\{x,y,z\}\cap\{u,v,t\}| = 2$ and if $x= u$ and $y= v$ then z and t are adjacent in $G$ and there exist directed edge between them.

Example 4.2.1. (The TVDG$_3$($G$)). Let $G$ be a simple digraph has 5 vertices and 6 edges as shown in Figure (4.1).

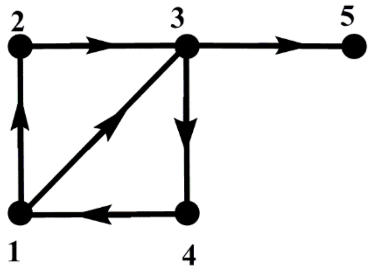


Figure 4.1. A simple digraph G(5,6).

Based on Definition (4.2.1), all vertices of the TVDG$_3$(G) are determined by

{1,2,3}, {2,3,5}, {2,3,4}, {3,4,1}, {4,1,2}, {4,1,3}, {1,3,4}, {1,3,5}.

Now, the intersections between an these vertices are given by

$\{1,2,3\}\cap\{2,3,5\ \} = \{2,3\} \Rightarrow |\{1,2,3\}\cap\{2,3,5\}| = 2$. There is no a directed edge between 1 and 5 in digraph G. So, there is no a directed edge between 123 and 235 in TVDG$_3$(G).

$\{1,2,3\}\cap\{2,3,4\} = \{2,3\} \Rightarrow |\{1,2,3\}\cap\{2,3,4\}| = 2$. There is no a directed edge between 1 and 4 in digraph G. So, there is no a directed edge between 123 and 234 in TVDG$_3$(G).

$\{1,2,3\}\cap\{3,4,1\} = \{1,3\} \Rightarrow |\{1,2,3\}\cap\{3,4,1\}| = 2$. There is no a directed edge between 2 and 4 in digraph G. So, there is no a directed edge between 123 and 341 in TVDG$_3$(G).

$\{1,2,3\}\cap\{4,1,2\} = \{1,2\} \Rightarrow |\{1,2,3\}\cap\{4,1,2\}| = 2$. There is a directed edge from 3 into 4. So, there exists a directed edge from 3 into 4 in TVDG$_3$(G).

$\{1,2,3\}\cap\{4,1,3\} = \{1,3\} \Longrightarrow |\{1,2,3\}\cap\{4,1,3\}| = 2$. There is no a directed edge between 2 and 4 in digraph G. So, there is no a directed edge between 123 and 413 in $TVDG_3(G)$.

$\{1,2,3\}\cap\{1,3,4\} = \{1,3\} \Longrightarrow |\{1,2,3\}\cap\{1,2,4\}| = 2$. There is no a directed edge between 2 and 4 in digraph G. So, there is no a directed edge between 123 and 134 in $TVDG_3(G)$.

$\{1,2,3\}\cap\{1,3,5\} = \{1,3\} \Longrightarrow |\{1,2,3\}\cap\{1,3,5\}| = 2$. There is no a directed edge between 2 and 5 in digraph G. So, there is no a directed edge between 123 and 135 in $TVDG_3(G)$.

$\{2,3,5\}\cap\{2,3,4\} = \{2,3\} \Longrightarrow |\{2,3,5\}\cap\{2,3,4\}| = 2$. There is no a directed edge between 4 and 5 in digraph G. So, there is no a directed edge between 235 and 234 in $TVDG_3(G)$.

$\{2,3,5\}\cap\{1,3,5\} = \{3,5\} \Longrightarrow |\{2,3,5\}\cap\{1,3,5\}| = 2$. There is no a directed edge between 2 and 1 in digraph G. So, there is no a directed edge between 235 and 135 in $TVDG_3(G)$.

$\{2,3,4\}\cap\{3,4,1\} = \{3,4\} \Longrightarrow |\{2,3,4\}\cap\{3,4,1\}| = 2$. There is no a directed edge between 2 and 1 in digraph G. So, there is no a directed edge between 234 and 341 in $TVDG_3(G)$.

$\{2,3,4\}\cap\{4,1,2\} = \{2,4\} \Longrightarrow |\{2,3,4\}\cap\{4,1,2\}| = 2$. . There is no a directed edge between 3 and 1 in digraph G. So, there is no a directed edge between 234 and 412 in $TVDG_3(G)$.

$\{2,3,4\}\cap\{4,1,3\} = \{3,4\} \Longrightarrow |\{2,3,4\}\cap\{4,1,3\}| = 2$. There is no a directed edge between 2 and 1 in digraph G. So, there is no a directed edge between 234 and 413 in $TVDG_3(G)$.

$\{2,3,4\}\cap\{1,3,4\} = \{3,4\} \Longrightarrow |\{2,3,4\}\cap\{1,3,4\}| = 2$. There is no a directed edge between 2 and 1 in digraph G. So, there is no a directed edge between 234 and 134 in $TVDG_3(G)$.

$\{3,4,1\}\cap\{4,1,2\} = \{4,1\} \Longrightarrow |\{3,4,1\}\cap\{4,1,2\}| = 2$. There is no a directed edge between 3 and 2 in digraph G. So, there is no a directed edge between 341 and 412 in $TVDG_3(G)$.

$\{3,4,1\}\cap\{1,3,5\} = \{1,3\} \Longrightarrow |\{3,4,1\}\cap\{1,3,5\}| = 2$. There is no a directed edge between 4 and 5 in digraph G. So, there is no a directed edge between 341 and 135 in $TVDG_3(G)$.

$\{4,1,2\}\cap\{4,1,3\} = \{1,4\} \Longrightarrow |\{4,1,2\}\cap\{4,1,3\}| = 2$ . There is a directed edge from 2 into 3. So, there exists a directed edge from 2 into 3 in $TVDG_3(G)$.

$\{4,1,2\}\cap\{1,3,4\} = \{1,4\} \Longrightarrow |\{4,1,2\}\cap\{1,3,4\}| = 2$ . There is a directed edge from 2 into 3. So, there exists a directed edge from 2 into 3 in $TVDG_3(G)$.

$\{4,1,2\} \cap \{3,4,1\} = \{4,1\} \Longrightarrow |\{3,4,1\}\cap\{4,1,2\}| = 2$. There is a directed edge from 2 into 3. So, there exists a directed edge from 2 into 3 in $TVDG_3(G)$.

$\{4,1,2\} \cap \{2,3,4\} = \{2,4\} \Longrightarrow |\{2,3,4\}\cap\{4,1,2\}| = 2$. There is a directed edge from 1 into 3. So, there exists a directed edge from 1 into 3 in $TVDG_3(G)$.

$\{4,1,3\}\cap\{1,3,5\} = \{1,3\} \Longrightarrow |\{4,1,3\}\cap\{1,3,5\}| = 2$. There is no a directed edge between 4 and 5 in digraph G. So, there is no a directed edge between 413 and 135 in $TVDG_3(G)$.

$\{1,3,4\} \cap \{1,3,5\} = \{1,3\} \Longrightarrow |\{1,3,4\} \cap \{1,3,5\}| = 2$. There is no a directed edge between 4 and 5 in digraph G. So, there is no a directed edge between 134 and 135 in $TVDG_3(G)$.

The $TVDG_3(G)$ of a digraph *G* is given in Figure (4.2).

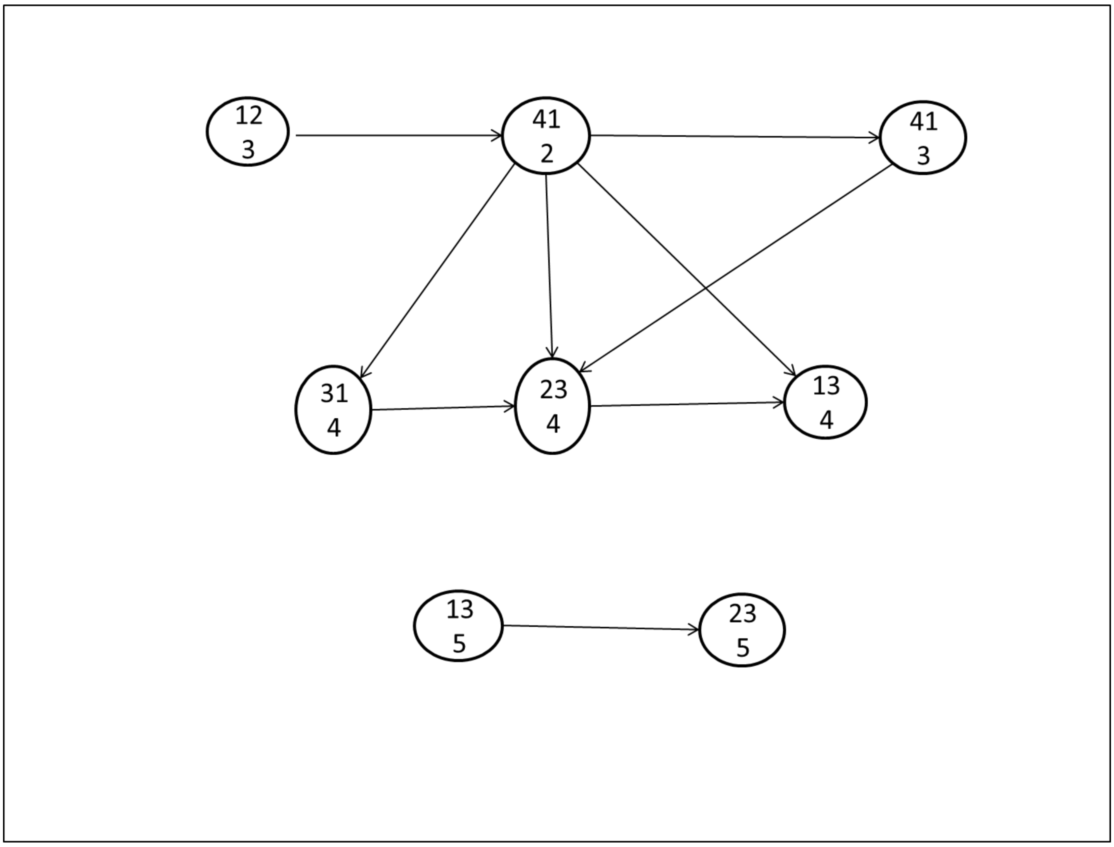


Figure 4.2. The $TVDG_3(G(5,6))$.

## 4.3 The Triple Vertex Cycle Digraph

In this section, a triple vertex cycle digraph $TVCDG_3$ is defined as a new definition based on the idea of the $TVCDG_3(G)$ that is given in Definition (4.2.1).

Definition 4.3.1. Let $C_n$ be a cycle digraph of order *n*. This mean that, this cycle has *n* vertices and n edges 1. The triple vertex digraph of cycle is denoted by $TVDG_3(C_n)$.

Example 4.3.1. Let $C_6$ be a cycle has 6 vertices and 6 edges, as show in Figure (4.3).

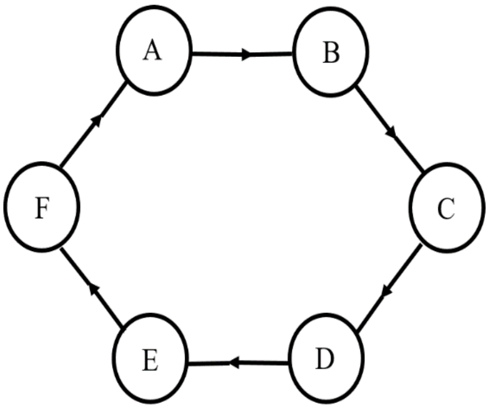


Figure 4.3. The cycle digraph $C_6$.

To form the TVDG$_3$($C_6$), first one determines the vertices of TVDG$_3$($C_6$) as follows:

$\{A,B,C\}$, $\{A,B,D\}$, $\{A,B,E\}$, $\{A,B,F\}$, $\{A,C,D\}$, $\{A,C,E\}$, $\{A,C,F\}$, $\{A,D,E\}$, $\{A,D,F\}$, $\{A,E,F\}$, $\{B,C,D\}$, $\{B,C,E\}$, $\{B,C,F\}$, $\{B,D,E\}$, $\{B,D,F\}$, $\{B,E,F\}$, $\{C,D,E\}$, $\{\mathrm{C},D,F\}$, $\{C,E,F\}$, $\{D,E,F\}$.

The TVDG$_3$($C_6$) is computed by

Let $\{A,B,C\}$ and $\{A,B,D\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,B,C\}\cap\{A,B,D\} = \{\mathrm{A,B}\}$. So, $|\{A,B,C\}\cap\{A,B,D\}| = 2$. Since, there is a directed edge from $C$ into $D$. So, there exists directed edge from $ABC$ into $ABD$ in the TVDG$_3$($C_6$).

Let $\{A,B,D\}$ and $\{A,B,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,B,D\}\cap\{A,B,E\} = \{A,B\}$. So, $|\{A,B,D\}\cap\{A,B,E\}| = 2$. Since, there is a directed edge from $D$ into $E$. So, there exists directed edge from *ABC* into *ABD* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,B,D\}$ and $\{A,C,D\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,B,D\}\cap\{A,C,D\} = \{A,D\}$. So, $|\{\mathrm{A},B,D\}\cap\{\mathrm{A},C,D\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from *ABD* into *ACD* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,B,E\}$ and $\{A,B,F\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,B,E\}\cap\{A,B,F\} = \{\mathrm{A},B\}$ So, $|\{A,B,E\}\cap\{A,B,F\}| = 2$. Since, there is a directed edge from $E$ into $F$. So, there exists directed edge from *ABE* into *ABF* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,B,E\}$ and $\{A,C,E\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,B,E\}\cap\{A,C,E\} = \{\mathrm{A},E\}$ So, $|\{A,B,E\}\cap\{A,C,E\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from *ABE* into *ACE* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,B,F\}$ and $\{A,C,F\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,B,F\}\cap\{A,C,F\} = \{A,F\}$ So, $|\{A,B,F\}\cap\{A,C,F\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from *ABF* into *ACF* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,C,D\}$ and $\{A,C,E\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,C,D\}\cap\{A,C,E\} = \{A,C\}$ So, $|\{A,C,D\}\cap\{A,C,E\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from *ABD* into *ACD* in the $\mathrm{TVDG}_3(C_6)$.

Let $\{A,C,D\}$ and $\{B,C,D\}$ are vertices in $\mathrm{TVDG}_3(C_6)$ then

$\{A,C,D\}\cap\{B,C,D\} = \{C,D\}$ So, $|\{A,C,D\}\cap\{B,C,D\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ACD$ into $BCD$ in the TVDG$_3$($C_6$).

Let $\{A,C,E\}$ and $\{A,C,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,E\}\cap\{A,C,F\} = \{A,C\}$ So, $|\{A,C,E\}\cap\{A,C,F\}| = 2$. Since, there is a directed edge from $E$ into $F$. So, there exists directed edge from $ACE$ into $ACF$ in the TVDG$_3$($C_6$).

Let $\{A,D,E\}$ and $\{A,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,D,E\}\cap\{A,D,F\} = \{A,D\}$ So, $|\{A,D,E\}\cap\{A,D,F\}| = 2$. Since, there is a directed edge from $E$ into $F$. So, there exists directed edge from $ADE$ into $ADF$ in the TVDG$_3$($C_6$).

Let $\{A,D,F\}$ and $\{A,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,D,F\}\cap\{A,E,F\} = \{A,F\}$ So, $|\{A,D,F\}\cap\{A,D,E\}| = 2$. Since, there is a directed edge from $D$ into $E$. So, there exists directed edge from $ADF$ into $AEF$ in the TVDG$_3$($C_6$).

Let $\{A,E,F\}$ and $\{B,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{\mathrm{A},E,F\}\cap\{B,E,F\} = \{E,F\}$ So, $|\{A,E,F\}\cap\{B,E,F\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $AEF$ into $BEF$ in the TVDG$_3$($C_6$).

Let $\{A,D,F\}$ and $\{B,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,D,F\}\cap\{B,D,F\} = \{D,F\}$ So, $|\{A,D,F\}\cap\{B,D,F\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ADF$ into $BDF$ in the TVDG$_3$($C_6$)

Let $\{A,D,E\}$ and $\{B,D,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,D,E\}\cap\{B,D,E\} = \{D,E\}$ So, $|\{A,D,E\}\cap\{B,D,E\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ADE$ into $BDE$ in the TVDG$_3$($C_6$).

Let $\{A,C,F\}$ and $\{B,C,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,F\}\cap\{B,C,F\} = \{C,F\}$ So, $|\{A,C,F\}\cap\{B,C,F\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ACF$ into $BCF$ in the TVDG$_3$($C_6$).

Let $\{A,C,E\}$ and $\{A,D,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,E\}\cap\{A,D,E\} = \{A,E\}$ So, $|\{A,C,E\}\cap\{A,D,E\}| = 2$. Since, there is a directed edge from $C$ into $D$. So, there exists directed edge from $ACE$ into $ADE$ in the TVDG$_3$($C_6$).

Let $\{A,C,F\}$ and $\{A,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,F\}\cap\{A,D,F\} = \{A,F\}$ So, $|\{A,C,F\}\cap\{A,D,F\}| = 2$. Since, there is a directed edge from $C$ into $D$. So, there exists directed edge from $ACF$ into $ADF$ in the TVDG$_3$($C_6$).

Let $\{A,C,E\}$ and $\{B,C,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,E\}\cap\{B,C,E\} = \{C,E\}$ So, $|\{A,C,E\}\cap\{B,C,E\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ACE$ into $BCE$ in the TVDG$_3$($C_6$).

Let $\{A,C,D\}$ and $\{B,C,D\}$ are vertices in TVDG$_3$($C_6$) then

$\{A,C,D\}\cap\{B,C,D\} = \{C,D\}$ So, $|\{A,C,D\}\cap\{B,C,D\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $ACD$ into $BCD$ in the TVDG$_3$($C_6$).

Let $\{B,C,D\}$ and $\{B,C,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{B,C,D\} \cap \{B,C,E\} = \{B,C\}$ So, $|\{B,C,D\} \cap \{B,C,E\}| = 2$. Since, there is a directed edge from *A* into *B*. So, there exists directed edge from *AEF* into *BEF* in the TVDG$_3$($C_6$).

Let $\{B,C,E\}$ and $\{B,C,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{\mathrm{B},C,E\} \cap \{B,C,F\} = \{B,C\}$ So, $|\{B,C,E\} \cap \{B,C,F\}| = 2$. Since, there is a directed edge from *E* into *F*. So, there exists directed edge from *BCE* into *BCF* in the TVDG$_3$($C_6$).

Let $\{B,C,E\}$ and $\{B,D,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{\mathrm{B},C,E\} \cap \{B,D,E\} = \{B,E\}$ So, $|\{B,C,E\} \cap \{B,D,E\}| = 2$, Since, there is a directed edge from *C* into *D*. So, there exists directed edge from *BCE* into *BDE* in the TVDG$_3$($C_6$).

Let $\{B,C,F\}$ and $\{B,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{\mathrm{B},C,F\} \cap \{B,D,F\} = \{B,F\}$ So, $|\{B,C,E\} \cap \{B,C,F\}| = 2$. Since, there is a directed edge from *C* into *D*. So, there exists directed edge from *BCF* into *BDF* in the TVDG$_3$($C_6$).

Let $\{B,D,E\}$ and $\{B,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{B,D,E\} \cap \{B,D,F\} = \{B,D\}$ So, $|\{B,D,E\} \cap \{B,D,F\}| = 2$. Since, there is a directed edge from *E* into *F*. So, there exists directed edge from *BDE* into *BDF* in the TVDG$_3$($C_6$).

Let $\{B,D,F\}$ and $\{B,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{\mathrm{B},D,F\} \cap \{B,E,F\} = \{B,F\}$ So, $|\{B,D,F\} \cap \{B,E,F\}| = 2$. Since, there is a directed edge from *D* into *E*. So, there exists directed edge from *BDF* into *BEF* in the TVDG$_3$($C_6$).

Let $\{B,E,F\}$ and $\{C,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{B,E,F\}\cap\{C,E,F\} = \{E,F\}$ So, $|\{B,E,F\}\cap\{C,E,F\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from $BEF$ into $CEF$ in the TVDG$_3$($C_6$).

Let $\{B,D,F\}$ and $\{C,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{B,D,F\}\cap\{C,D,F\} = \{D,F\}$ So, $|\{B,D,F\}\cap\{C,D,F\}| = 2$. Since, there is a directed edge from $A$ into $B$. So, there exists directed edge from $AEF$ into $BEF$ in the TVDG$_3$($C_6$).

Let $\{B,D,E\}$ and $\{C,D,E\}$ are vertices in TVDG$_3$($C_6$) then

$\{B,D,E\}\cap\{C,D,E\} = \{D,E\}$ So, $|\{B,D,E\}\cap\{C,D,E\}| = 2$. Since, there is a directed edge from $B$ into $C$. So, there exists directed edge from $BDE$ into $CDE$ in the TVDG$_3$($C_6$).

Let $\{C,D,E\}$ and $\{C,D,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{C,D,E\}\cap\{C,D,F\} = \{C,D\}$ So, $|\{C,D,E\}\cap\{C,D,F\}| = 2$. Since, there is a directed edge from $E$ into $F$. So, there exists directed edge from $CDE$ into $CDF$ in the TVDG$_3$($C_6$).

Let $\{C,D,F\}$ and $\{C,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{C,D,F\}\cap\{C,E,F\} = \{C,F\}$ So, $|\{C,D,F\}\cap\{C,E,F\}| = 2$. Since, there is a directed edge from $D$ into $E$. So, there exists directed edge from $CDF$ into $CEF$ in the TVDG$_3$($C_6$).

Let $\{C,E,F\}$ So, $\{D,E,F\}$ are vertices in TVDG$_3$($C_6$) then

$\{C,E,F\}\cap\{D,E,F\} = \{E,F\}$ So, $|\{C,E,F\}\cap\{D,E,F\}| = 2$. Since, there is a directed edge from $C$ into $D$. So, there exists directed edge from $CEF$ into $DEF$ in the TVDG$_3$($C_6$).

Now, the TVDG$_3$($C_6$) is shown in Figure (4.4).

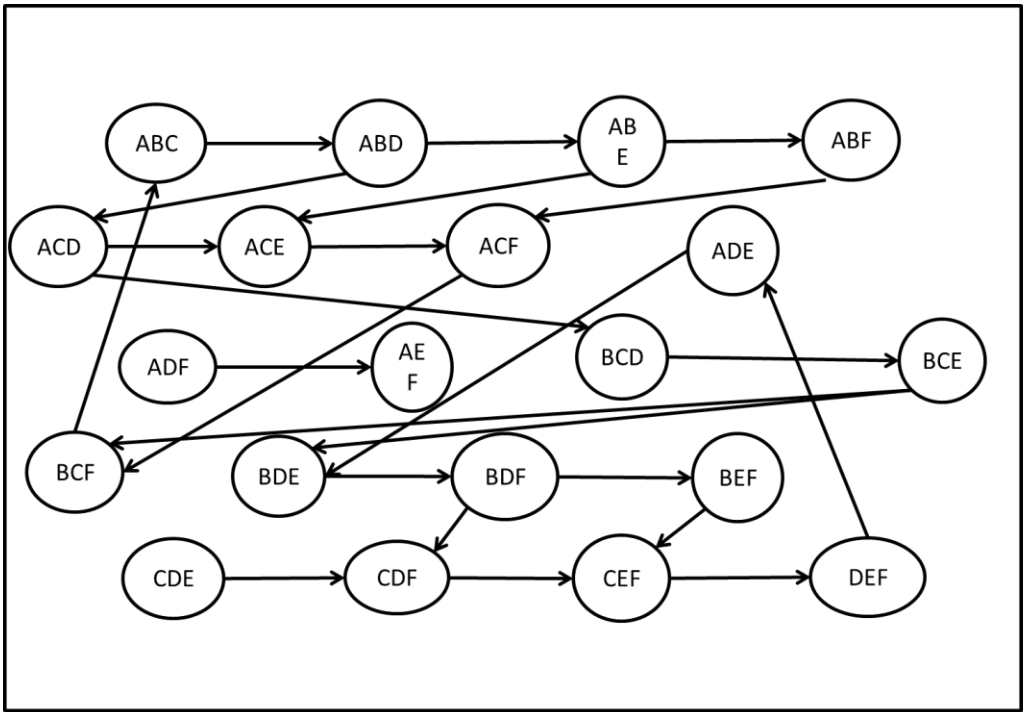


Figure 4.4. the TVDG$_3$($C_n$) of path $C_6$ in Figure (4.3).

## 4.4 The Triple Vertex Cycle Digraph for Encryption Schemes

In this section, some encryption schemes have been proposed based on the triple vertex cycle digraph (TVCDG$_3$($C_n$)) which are discussed as follows.

### 4.4.1 The Triple Vertex Cycle Digraph for Encryption Scheme Based on the EAVs

This section discusses the application of the TVDG$_3$($C_n$) to encrypt the plaintext *M*. Next example is used to explain the idea of the proposed encryption scheme that is used the TVDG$_3$($C_n$) based on the EAVs.

**Example 4.4.1.1. (The TVCDG for Encryption Scheme Based on the EAVs)**

Suppose *M* is the plaintext that is given by the word Nurse. Based on the English alphabet Table (3.1) of the letters, one can convert the letters of *M* into numbers. So,

$$N\rightarrow 13,\ u\rightarrow 20\ ,\ r\rightarrow 17,\ s\rightarrow 18,\ e\rightarrow 4.$$

With a shared secret key *K*=6, the ciphertext *Cp* is computed by

$$Cp \equiv M+ K\ (mod\ 26).$$

In other words,

$$Cp_1 \equiv M_1 + K\ (mod\ 26) \equiv 13 +6\ (mod\ 26) \equiv 19 \rightarrow T$$

$$Cp_2 \equiv M_2 + K\ (mod\ 26) \equiv 20 + 6(mod\ 26) \equiv 26(\ \mathrm{mod}\ 26) \equiv 0 \rightarrow a$$

$$Cp_3 \equiv M_3 + K\ (mod\ 26) \equiv 17+ 6\ (mod\ 26) \equiv 23 \rightarrow x$$

$$Cp_4 \equiv M_4 + K\ (mod\ 26) \equiv 18+ 6\ (mod\ 26) \equiv 24 \rightarrow y$$

$$Cp_5 \equiv M_5 + K\ (mod\ 26) \equiv 4+ 6\ (mod\ 26) \equiv 10 \rightarrow k$$

So, the ciphertext *Cp* of *M* forms a cycle digraph **taxyk** that is shown in Figure (4.5).

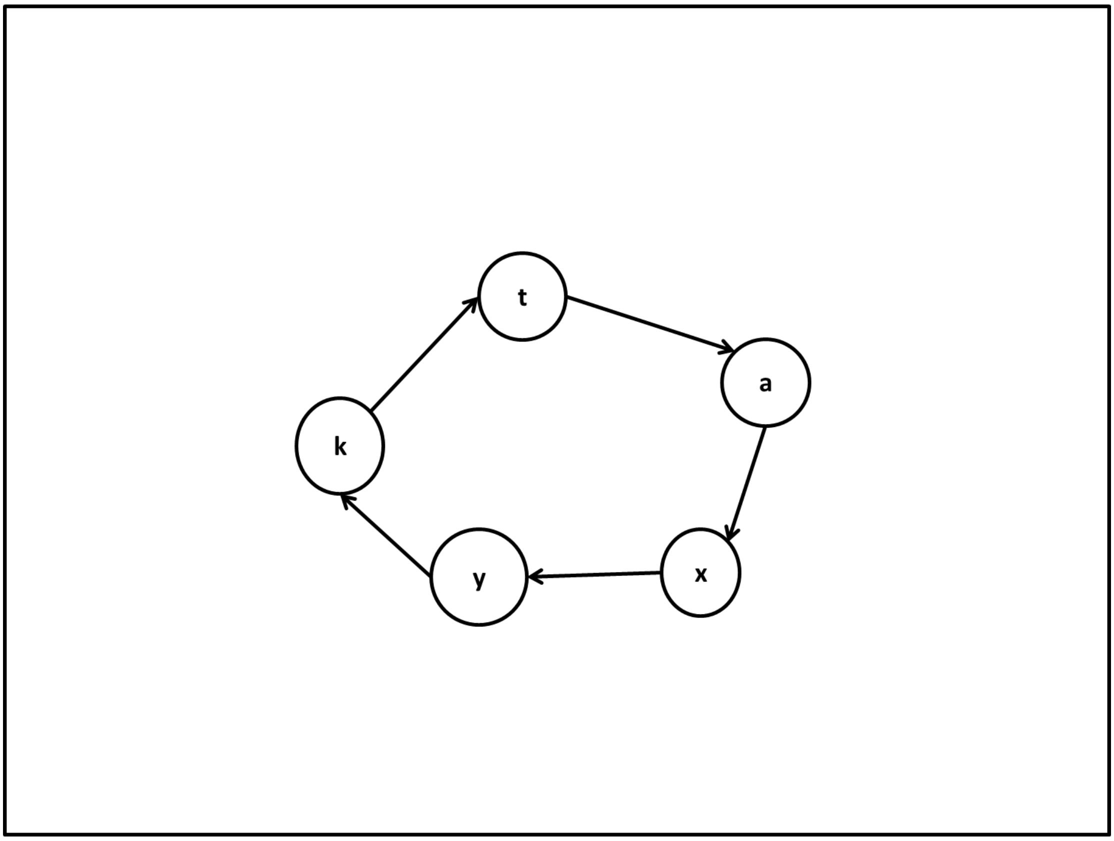


Figure 4.5. The cycle digraph $C_5$ of the ciphertext **taxyk**.

This cycle digraph is represented as the triple cycle vertex digraph $\mathrm{TVDG}_3(C_5)$ that is given in Figure (4.6) and sent to receiver by sender.

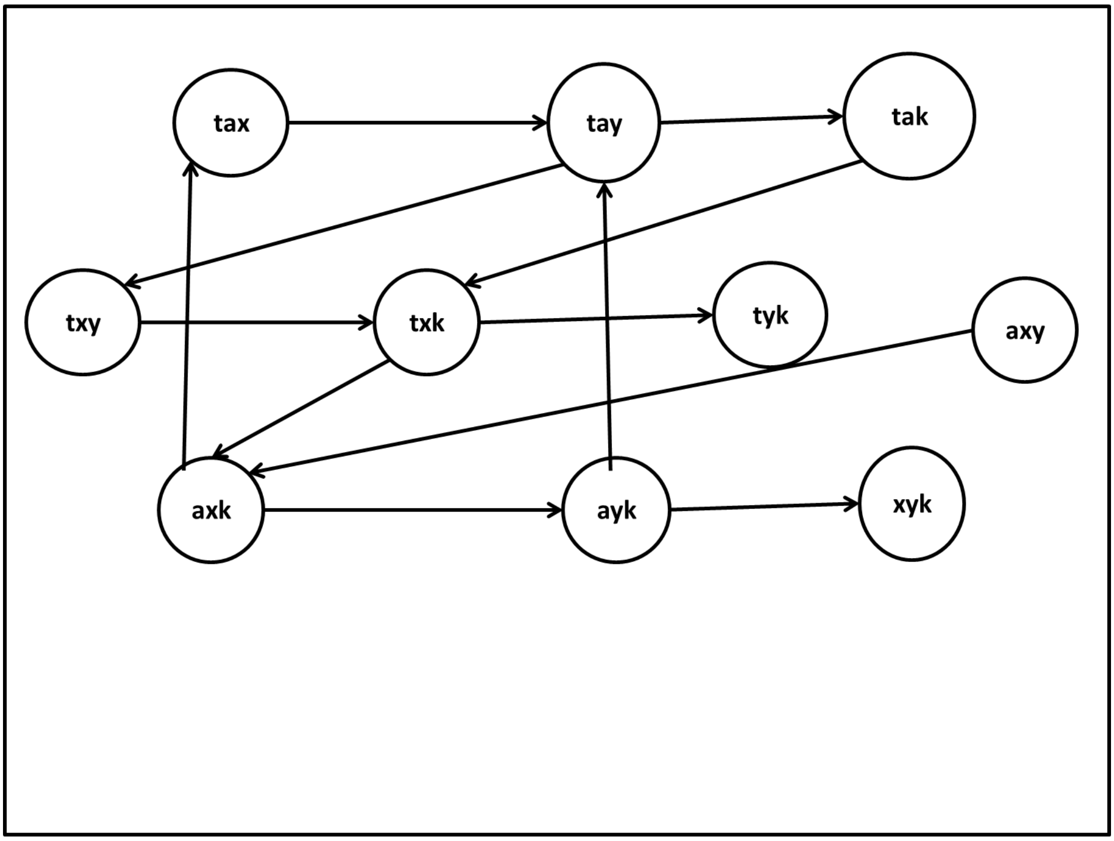


Figure 4.6. The TVDG$_3$($C_5$) of the cycle digraph $C_5$ of the ciphertext **taxvk**.

After the second user (receiver) receives TVDG$_3$($C_5$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices tax , tav , tak , txv , txk , tvk , axv , axk , avk , & xvk , based on TVDG$_3$($C_5$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is Taxvk.

Now, based on Table (3.1), these letters converted into numbers

$$T \rightarrow 19,\ a \rightarrow 0,\ x \rightarrow 23, v \rightarrow 24,\ k \rightarrow 10.$$

Now, the second user uses his/ her shared secret key $K = 6$ to find the following computations

$$19\text{-}6 \equiv 13\ (mod\ 26) \equiv 13 \rightarrow N$$

$$0\text{-}6 = \text{-}6\ (mod\ 26) \equiv 20 \rightarrow u$$

$$23\text{-}6 = 17\ (mod\ 26) \equiv 17 \rightarrow r$$

$$24\text{-}6 = 18\ (mod\ 26) \equiv 18 \rightarrow s$$

$$10\text{-}6= 4\ (mod\ 26) \equiv 4 \rightarrow e$$

Thus, the original message is **Nurse**.

### 4.4.2. The Triple Vertex Cycle Digraph for Encryption Scheme Based on the ASCII Values

This section discusses the application of the TVDG$_3$($C_n$) to encrypt the plaintext *M*. Following example explains the idea of the proposed encryption scheme that is used the TVDG$_3$($C_n$) based on the ASCII Values.

**Example 4.4.2.1.** Suppose *M* is the plaintext that is given by the word **family** . Based on the ASCII Table (3.2) of the letters, one can convert the letters in the plaintext M into numbers. So,

f→ 102, a→ 97, m→109, i→105, l→108, y→121.

Let *K* = -4 be shared secret key. Adding K to all of these numbers one by one respectively gives the ciphertext

$$Cp_1 \equiv 102+(\text{-}4)\ (mod\ 127) \equiv 98 \rightarrow b$$

$$Cp_2 \equiv 97+(\text{-}4)\ (mod\ 127) \equiv 93 \rightarrow ]$$

$$Cp_3 \equiv 109+(\text{-}4)\ (mod\ 127) \equiv 105 \rightarrow i$$

$$Cp_4 \equiv 105+(-4)\ (mod\ 127) \equiv 101 \rightarrow e$$

$$Cp_5 \equiv 108+(-4)\ (mod\ 127) \equiv 104 \rightarrow h$$

$$Cp_6 \equiv 121+(-4)\ (mod\ 127) \equiv 117 \rightarrow u$$

The ciphertext of $M$ is

$$Cp = b]iehu$$

A cycle digraph of the ciphertext is created as shown in Figure (4.7).

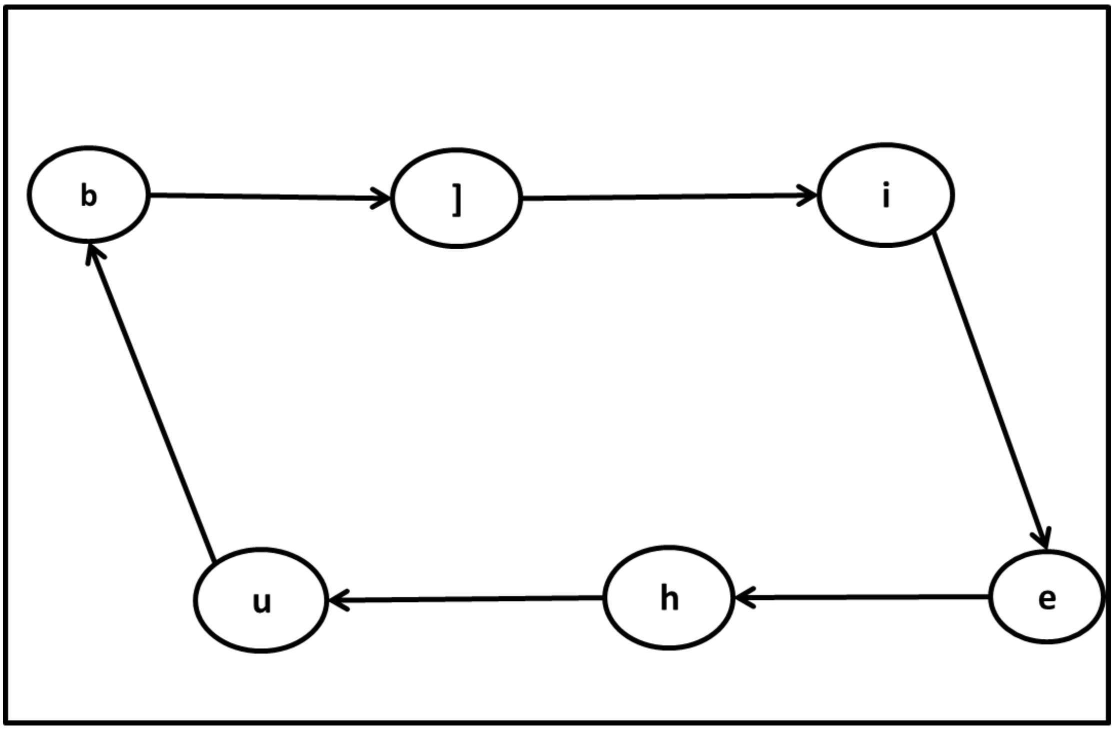


**Figure 4.7. The cycle digraph $C_6$ of the ciphertext *b]iehu***

The ciphertext of a message $M$ is considered as the TVDG$_3$($C_6$) as shown in Figure (4.8) which is sent to receiver by sender.

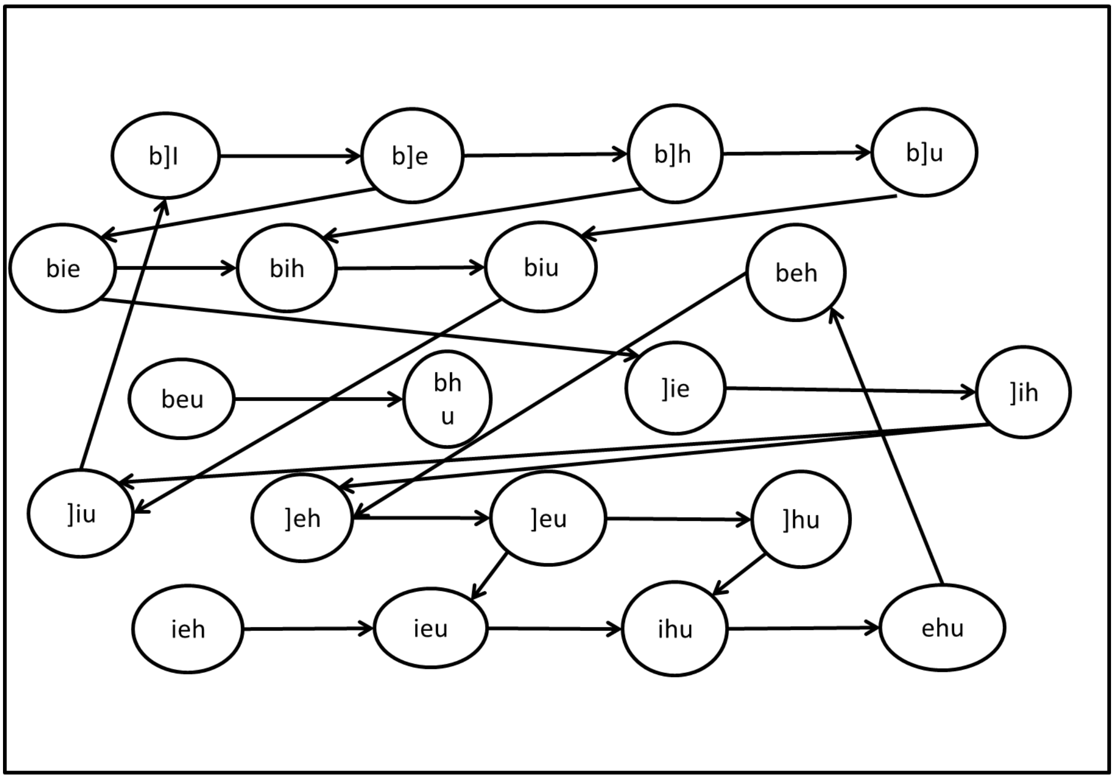


Figure 4.8. The TVDG$_3$($C_6$) of the ciphertext *b]iehu*

After the second user (receiver) receives the TVDG$_3$($C_6$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices b]I , b]e , b]h , b]u , bie , bih , biu , beh , beu , bhu , ]ie , ]ih , ]iu , ]eh , ]eu , ]hu , ieh , ieu , ihu ,& ehu ,

based on TVDG$_3$($P_6$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **b]iehu**

Now, based on Table (3.2), these letters converted into numbers

b →98, ] → 93, i → 105, e → 101, h → 104, u → 117.

Second user uses a shared secret key $K$ = -4 to recover the original plaintext as following.

$$98 +(4) \equiv 102 \ (mod\ 127) \equiv 102 \rightarrow \text{f}$$

$$93+(4)= 97 \ (mod\ 127) \equiv 97\rightarrow \text{a}$$

$$105+(4) = 109 \ (mod \ 127) \equiv 109 \rightarrow \mathrm{m}$$

$$101+(4) = 105 \ (mod \ 127) \equiv 105 \rightarrow \mathrm{i}$$

$$104+(4) = 108 \ (mod \ 127) \equiv 108 \rightarrow \mathrm{l}$$

$$117 +(4) = 121 \ (mod \ 127) \equiv 121 \rightarrow \mathrm{y}$$

Thus, the original message is **family**.

## 4.5 The TVCDG for Polyalphabetic Encryptions Schemes

This section presents two encryption schemes that are depending on the polyalphabetic cipher. These schemes are discussed as follows.

### 4.5.1 The TVCDG for Polyalphabetic Encryption Scheme Based on the EAVs

This section discusses the application of the TVDG$_3$($C_n$) to encrypt the plaintext $M$. Following example explains the idea of the proposed polyalphabetic encryption scheme based on the TVDG$_3$($C_n$) using the EAVs.

Example 4.5.1.1. Suppose $M$ is the plaintext that is given by the word **tiger**. Using polyalphabetic cipher based on the EAVS in Table (3.1) of the letters, and the rules on secret key that is given by:

1- Shift first letter six positions into its left,

2- Shift second lettr four positions into its right,

So, one can obtain

ti ge r

ti→nm , ge→ ai and, r→ l

Thus, the ciphertext: **nmail**

The ciphertext $C$ of $M$ forms a path digraph **nmail** that is shown in Figure (4.9).

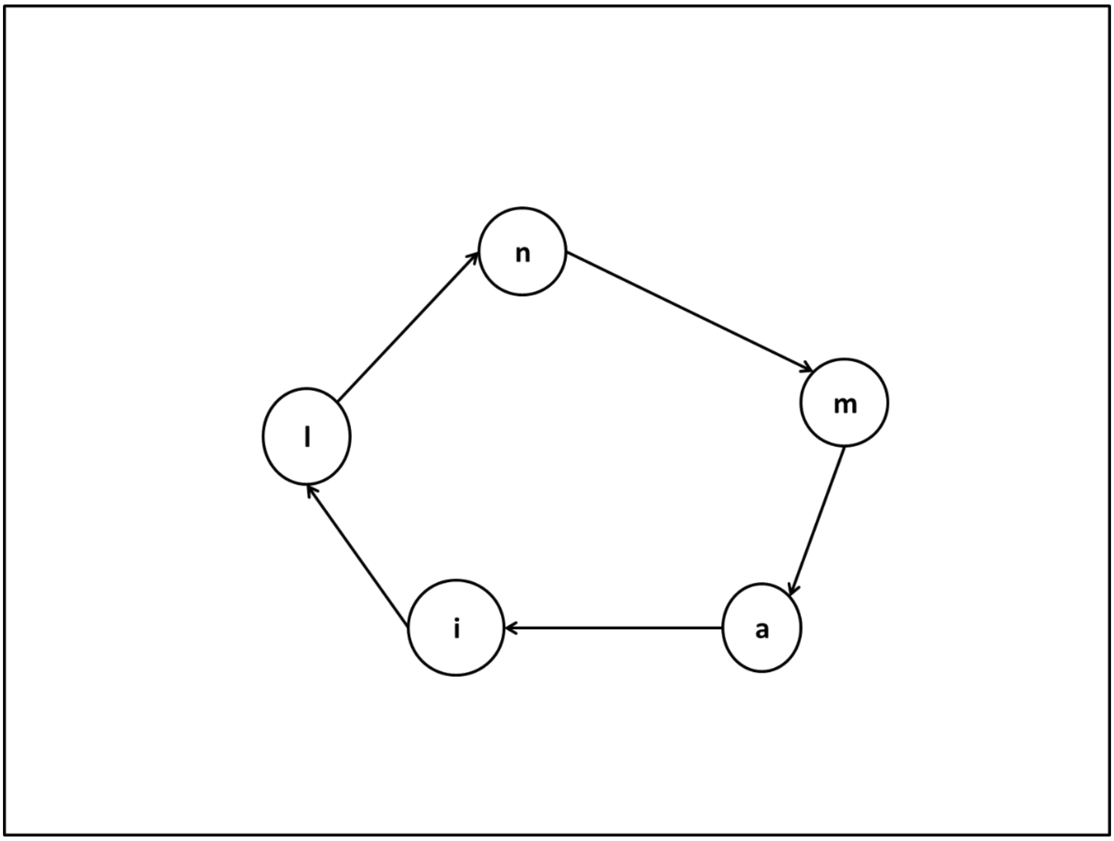


Figure 4.9. The cycle digraph $C_5$ of the ciphertext **nmail**.

This cycle is represented as the TVDG$_3$($C_5$) that is given in Figure (4.10) and sent to receiver by sender.

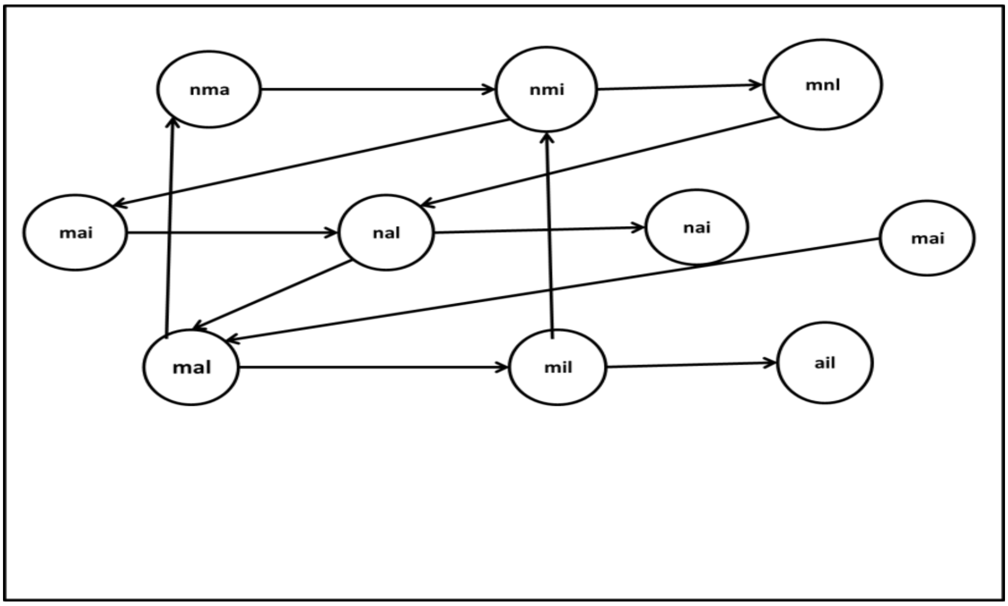


Figure 4.10. The TVDG$_3$($C_5$) of the ciphertext **nmail**.

After the second user (receiver) receives the TVDG$_3$($C_5$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices nma , nmi , nml , nai , nal , nil , mai , mal , mil ,& ail , based on TVDG$_3$(C$_5$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **nmail**.

Now, using the inverse a rules of a secret key that are

1- Shift first letter six positions into its right,
2- Shift second letter four positions into its left.

So

nm→ti, ai → geand, l→ r

Thus, the original message is **tiger**.

## 4.5.2 The TVCDG for Polyalphabetic Cipher Encryption Scheme Based on the ASCII Values.

This section discusses the application of the TVDG$_3$($C_n$) to encrypt the plaintext *M*. Following example explains the idea of the proposed polyalphabetic encryption scheme based on the TVDG$_3$($C_n$) using the ASCII values.

**Example 4.5.2.1.** Suppose *M* is the plaintext that is given by the word **second**. Using polyalphabetic cipher based on the ASCII Table (3.2) of the letters, and the rules on secret key that is given by:

4- Shift first letter six positions into its left,
5- Shift second letter ten positions into its right,

6- Shift third letter thir teen positions into its right.

Using the key, one can obtain

Sec ond

Sec→ mob and, ond → ixq

The ciphertext of M is

*C* = **mobixq**

The ciphertext *C* of *M* forms a cycle digraph **mobixq** that is shown in Figure (4.11).

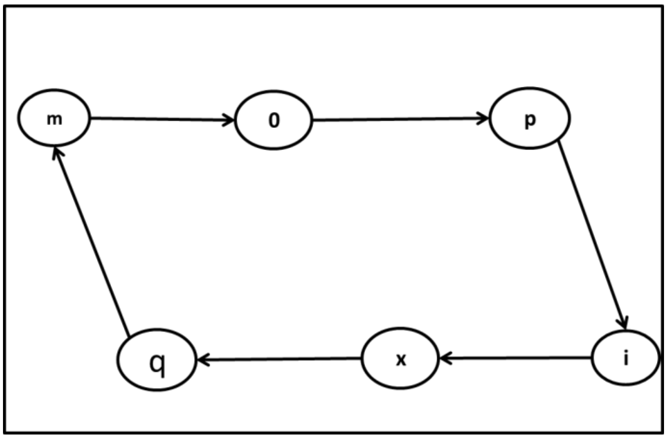


Figure 4.11. The cycle digraph $C_6$ of the ciphertext **mobixq**.

This cycle is represented as the (TVDG$_3$($C_6$)) that is given in Figure (4.12) and sent to receiver by sender.

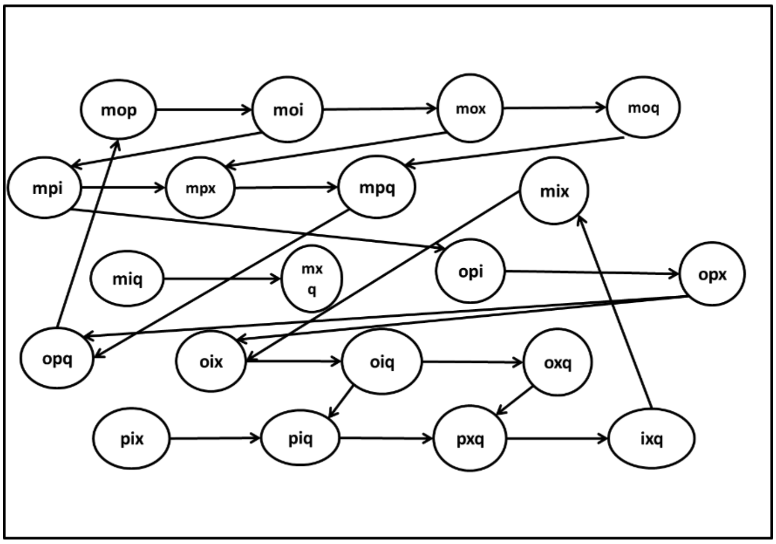


Figure 4.12. The path digraph

$C_6$ of the ciphertext **mobixq**.

After the second user (receiver) receives the TVDG$_3$($C_6$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices mob , moi , mox , moq , mbi , mbx , mbq , mix , miq , mxq , obi , obx , obq , oix , oiq , oxq , bix , biq , bxq , & ixq , based on TVDG(C$_6$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **mpbixq**.

Now, using the invers a rules of a secret key that are

1- Shift first letter six positions into its right,
2- Shift second letter ten positions into its left,
3- Shift third letter thir teen positions into its left.

So mob→ sec and, ixq → ond

Thus, the original message is **second**.

## 4.6 The TVCDG for an Affine Encryption Schemes

The TVCDG is used for an affine encryption schemes to give new versions of affine encryption scheme. First scheme uses the EAVs and second one uses the ASCII values. These scheme are discussed as follows.

### 4.6.1 The TVCDG for an Affine Encryptions Scheme Based on the EAVs.

The proposed affine encryptions scheme using the TVCDG based on the representation of the EAVs is explained in the following example.

**Example 4.6.1.1**. Suppose *M* is the plaintext that is given by the word **honey**. Using affine cipher based on the EAVs in Table (3.1) of the letters, with the secret key is (a,b) = (15,3), one can compute the ciphertext by

$$E \equiv (ax+b)(mod\ 26)$$

$E(h) \equiv (15\times 7+3)(mod\ 26) \Longrightarrow 108\ (mod\ 26)= 4 \rightarrow e$

$E(\mathrm{o}) \equiv (15\times 14+3)(mod\ 26) \Longrightarrow 213\ (mod\ 26)= 5 \rightarrow f$

$E(\mathrm{n}) \equiv (15\times 13+3)(mod\ 26) \Longrightarrow 198\ (mod\ 26)= 16 \rightarrow q$

$E(\mathrm{e}) \equiv (15\times 4+3)(mod\ 26) \Longrightarrow 63\ (mod\ 26)=11 \rightarrow l$

$E(\mathrm{y}) \equiv (15\times 24+3)(mod\ 26) \Longrightarrow 363\ (mod\ 26)= 25 \rightarrow z$

The ciphertext of *M* is

*C* =**efqlz**.

The ciphertext *C* of *M* forms a cycle digraph **efqlz** that is shown in Figure (4.13).

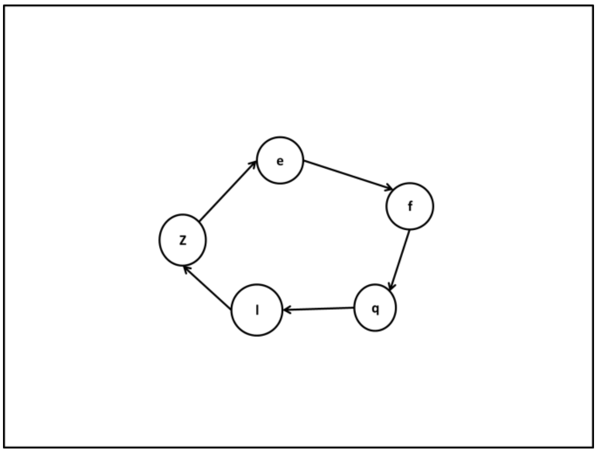


Figure 4.13. The cycle digraph $C_5$ of the ciphertext **efqlz.**

This cycle is represented as the triple vertex digraph (TVDG$_3$($C_5$)) that is given in Figure (4.14) and sent to receiver by sender.

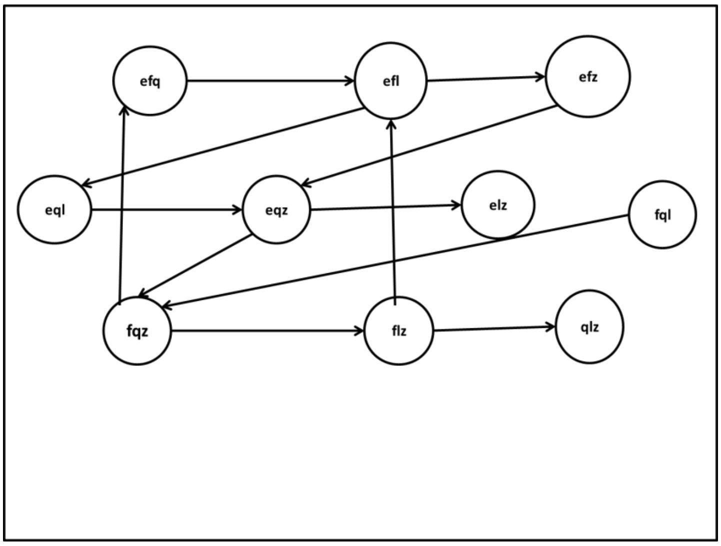


Figure 4.14. The cycle digraph $C_5$ of the ciphertext **efqlz**.

After the second user (receiver) receives the TVDG$_3$($C_5$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices efq , efl , efz , eql , wqz , elz , fql , fqz , flz , & qlz, based on TVDG$_3$($C_5$). Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is **efqlz**.

Now, the decrypt of the ciphertext **efqlz** is done using an affine cipher based on Table (3.1) as follows.

$$D(x) \equiv a^{-1}(E\text{-}b)(mod26)$$

$D(e) \equiv 7(4\text{-}3)(mod\ 26) \Longrightarrow 7\ (mod\ 26) \equiv 7 \rightarrow h$ , where $15^{-1}\ (mod\ 26) = 7$,

$D(f) \equiv 7(5\text{-}3)(mod\ 26) \Longrightarrow 14\ (mod\ 26) \equiv 14 \rightarrow o$

$D(q) \equiv 7(16\text{-}3)(mod\ 26) \Longrightarrow 13\ (mod\ 26) \equiv 13 \rightarrow n$

$D(l) \equiv 7(11\text{-}3)(mod\ 26) \Longrightarrow 4\ (mod\ 26) \equiv\ 4 \rightarrow e$

$D(z) \equiv 15(25\text{-}3)(mod\ 26) \Longrightarrow 24\ (mod\ 26) \equiv 24 \rightarrow y$

Thus, the original message is **honey**.

## 4.6.2 The TVCG for Affine Encryption Scheme Based on the ASCII Values.

The proposed affine encryptions scheme using the TVCG based on the representation of the ASCII values is explained in the following example.

Example 4.6.2.1. Suppose $M$ is the plaintext that is given by the word **listen** . Using an affine cipher based on the ASCII Table (3.2) of the letters, with the key is (a,b) = (9, 15), one can compute the ciphertext by

$$E \equiv (ax+b)(mod\ 127)$$

$E(S) \equiv (9\times76+15)(mod\ 127) \Longrightarrow 699\ (mod\ 127) \equiv 64 \rightarrow @$

$E(i) \equiv (9\times105+15)(mod\ 127) \Longrightarrow 960\ \ (mod\ 127) \equiv 71 \rightarrow G$

$E(s) \equiv (9\times115+15)(mod\ 127) \Longrightarrow 1050\ (mod\ 127) \equiv 34 \rightarrow ''$

$E(t) \equiv (9\times116+15)(mod\ 127) \Longrightarrow 1059\ (mod\ 127) \equiv 43 \rightarrow +$

$E(e) \equiv (9\times101+15)(mod\ 127) \Longrightarrow 924\ (mod\ 127) \equiv 35 \rightarrow \#$

$E(n) \equiv (9\times110+15)(mod\ 127) \Longrightarrow 1005\ (mod\ 127) \equiv 116 \rightarrow t$

The ciphertext of *M* is

*C* **=@G"+#t.**

The ciphertext *C* of *M* forms a cycle digraph ***@G"+#t*** that is shown in Figure (4.15).

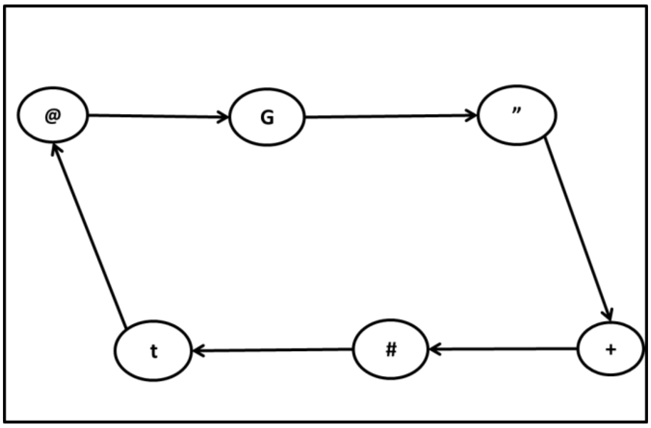


Figure 4.15. The cycle digraph $C_6$ of the ciphertext ***@G"+#t***

This cycle is represented as the triple vertex digraph (TVDG$_3$($C_6$)) that is given in Figure (4.16) and sent to receiver by sender.

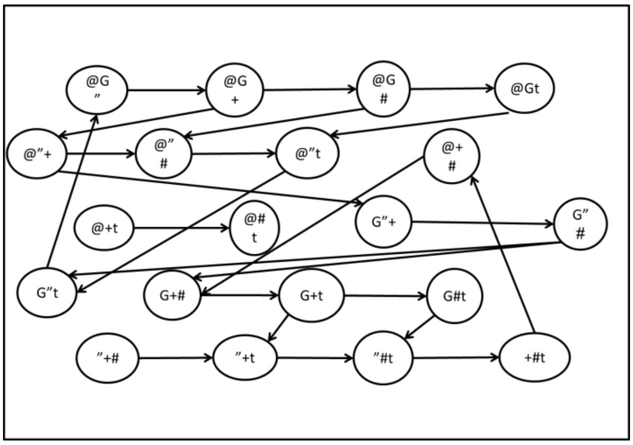


Figure 4.16. The TVDG$_3$($C_6$) of the ciphertext ***@G"+#t***

After the second user (receiver) receives the TVDG$_3$($C_6$), he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices based on TVDG($C_6$). Later on, he/she takes only the letters from vertices without repeating to form a list @G" , @G+ , @G# , @Gt , @"+ , @"# , @"t , @+# , @+t , @#t , G"+ , G"+ , G"t , G+# , G+t , G"t , "+# , "+t , "#t , & +#t , . There are many cases to determine the correct choice of this list. The correct one is ***@G"+#t***.

Now, the decrypt the ciphertext: ***@G"+#t*** is done using an affine cipher based on the ASCII Table (3.2) of the letters as follows.

$$D(x) \equiv a^{-1}(E\text{-}b)(mod\ 127)$$

$D(@) \equiv 113\ (64\text{-}15)(mod\ 127) \Longrightarrow 5537\ (mod\ 127) \equiv 76 \rightarrow l,$ where

$9^{-1}\ (mod\ 127) \equiv 113,$

$D(G) \equiv 113(71\text{-}15)(md3\ 127) \Longrightarrow 6328\ (mod\ 127) \equiv 105 \rightarrow i$

$D('') \equiv 113(34\text{-}15)(mod\ 127) \Longrightarrow 2147\ \ (mod\ 127) \equiv\ 115 \rightarrow s$

$D(+){\equiv}113(43\text{-}15)(mod127){\Longrightarrow}3164(mod127){\equiv}116{\rightarrow}t$

$D(\#) \equiv 113(35\text{-}15)(mod\ 127) \Longrightarrow 2260\ (mod\ 127) \equiv 101 \rightarrow e$

$D(t) \equiv 113(116\text{-}15)(mod\ 127) \Longrightarrow 11413\ (mod\ 127) \equiv 110\ \rightarrow n$

Thus, the original message is **listen**.

## 4.7 The TVCDG -Hill Cipher Encryption schemes

In this section, two encryption schemes are proposed based on the TVCDG for Hill encryption scheme. These schemes are used the EAVs and ASCII values respectively. These scheme are discussed as follows.

### 4.7.1 The TVCDG - Hill Encryption Scheme Based on the EAVs

The proposed Hill encryptions scheme using the TVCDG based on the representation of the EAVs is explained in the following example.

**Example 4.7.1.1.** Suppose *M* is the plaintext that is given by the word "**Twenty** Let K be a secret key which is given by 3×3 matrix, namely :

$$K = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}$$

Now, it is easy to divide M into 3 blocks, Each block has length 2. In other words, it can write M by = $(M_1,M_2)$

M = (Twe nty)

Using the ASCII Table (3.2) one can represent the litters in

$(M_1,M_2)$ = (Twe nty)

$$\begin{pmatrix} T \\ w \\ e \end{pmatrix}\begin{pmatrix} 84 \\ 119 \\ 101 \end{pmatrix}\begin{pmatrix} n \\ t \\ y \end{pmatrix}\begin{pmatrix} 110 \\ 116 \\ 121 \end{pmatrix}$$

The ciphertext C is computed by $C \equiv K{*}P_i$ (*mod* 127), in the more details for *i*=1,2,3.

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}\begin{pmatrix} 84 \\ 119 \\ 101 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 84 + 2 \times 119 + 3 \times 101 \\ 0 \times 84 + 1 \times 119 + (-1) \times 101 \\ 2 \times 84 + 2 \times 119 + 2 \times 101 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 625 \\ 18 \\ 608 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 117 \\ 18 \\ 100 \end{pmatrix} \longrightarrow \begin{pmatrix} u \\ \text{device control 2} \\ d \end{pmatrix}$$

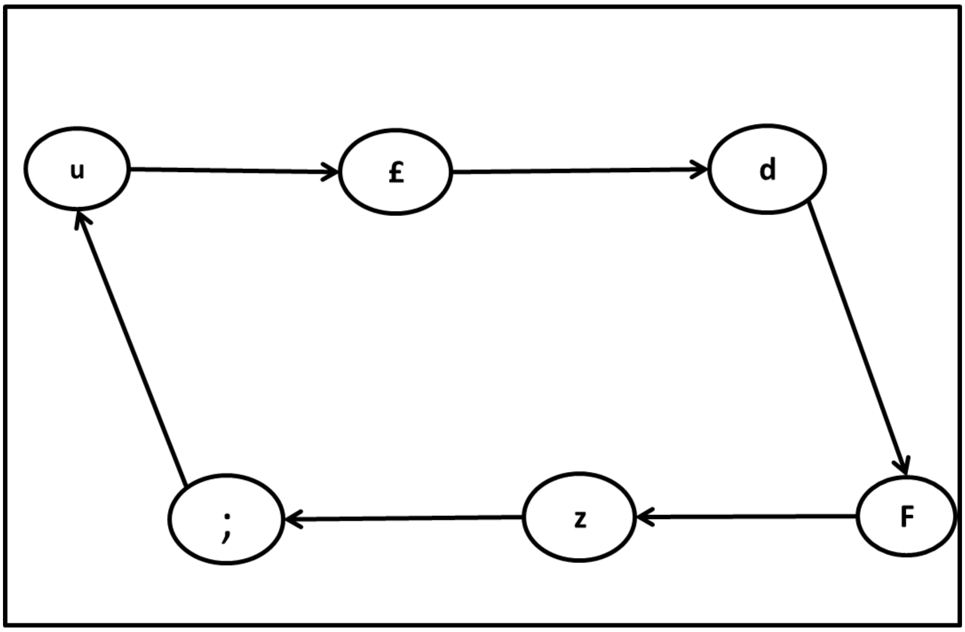
$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 2 & 2 & 2 \end{bmatrix}\begin{pmatrix} 110 \\ 116 \\ 121 \end{pmatrix} \equiv \begin{pmatrix} 1 \times 110 + 2 \times 116 + 3 \times 121 \\ 0 \times 110 + 1 \times 116 + (-1) \times 121 \\ 2 \times 121 + 2 \times 116 + 2 \times 121 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 705 \\ -5 \\ 694 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 70 \\ 122 \\ 59 \end{pmatrix} \longrightarrow \begin{pmatrix} F \\ z \\ ; \end{pmatrix}$$

The ciphertext of M is

*C* = u device control 2 dFz;.

So, the cipher text C of M forms a cycle digraph udevicecontrol2dFz; that



is shown in Figure (4.17).

This is represented as the triple vertex digraph (TVG) that is given in

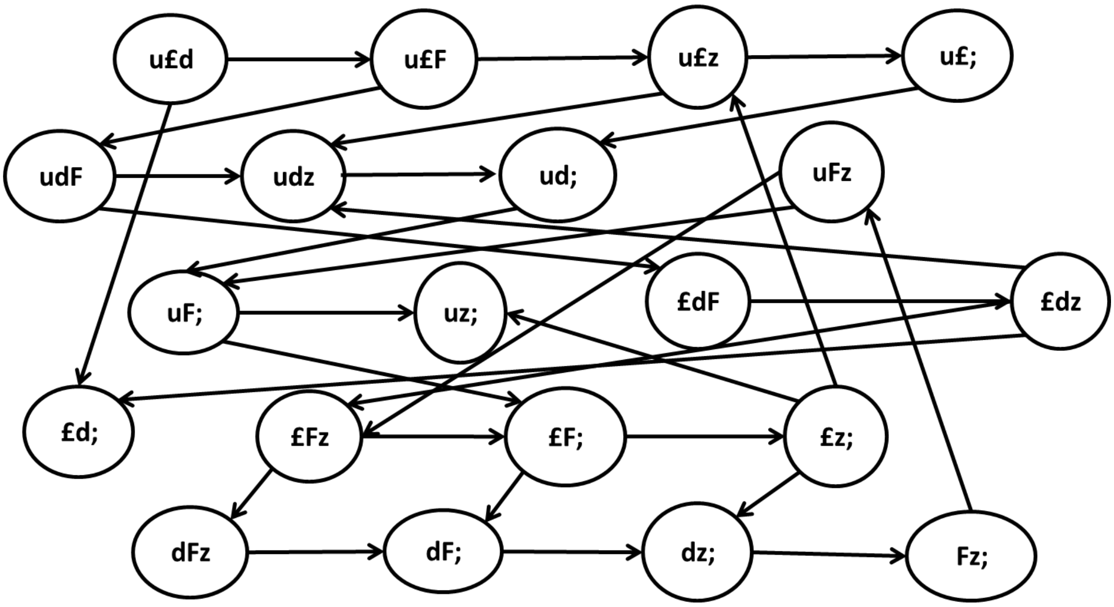


Figure (4.18) and sent to receiver by send.

After the second user (receiver) receives a triple vertex digraph, he/ she wants to decrypt the cipher text and recover the original plaintext. He/ She first determines the label vertices u£d , u£F, u£z , u£; , udF , udz , ud; , uFz , uF; , uz; , £dF , £dz , £d; , £Fz , £F; , £z; , dFz , dF; , dz; ,& Fz;

## 4.7.2 The TVCG for Hill Encryption Scheme Based on the ASCII Values.

The proposed Hill encryption schemes using the TVCDG based on the representation of the ASCII values is explained in the following example.

**Example 4.7.2.1.** Suppose *M* is the plaintext that is given by the word Silver. Let K be a secret key which is given by 2×2 matrix, namely

$$K=\begin{bmatrix}9 & 14\\8 & 19\end{bmatrix}$$

Now, it is easy to divide *M* into 3 blocks, Each block has length 2. In other words, it can write *M* by

$$(M_1,M_2,M_3) = (\text{Si} \quad \text{lv} \;\; \text{er}).$$

Using the ASCII Table (3.2) one can represent the letters in $(M_1,M_2,M_3)$ = (Si lv er) by

$$\begin{pmatrix}S\\i\end{pmatrix}\begin{pmatrix}83\\105\end{pmatrix} \mid \begin{pmatrix}l\\v\end{pmatrix}\begin{pmatrix}108\\118\end{pmatrix} \mid \begin{pmatrix}e\\r\end{pmatrix}\begin{pmatrix}101\\114\end{pmatrix}$$

The ciphertext Cp is computed by $Cp \equiv K*M_i$ (*mod* 127), in the more details for *i*=1,2,3, then

$$\begin{bmatrix}9 & 14\\8 & 19\end{bmatrix}\begin{pmatrix}83\\105\end{pmatrix} \equiv \begin{pmatrix}9\times 83+14\times 105\\8\times 83+19\times 105\end{pmatrix}$$

$$\equiv \begin{pmatrix} 2217 \\ 2659 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 58 \\ 119 \end{pmatrix} \longrightarrow \begin{pmatrix} : \\ w \end{pmatrix}$$

$$\begin{bmatrix} 9 & 14 \\ 8 & 19 \end{bmatrix} \begin{pmatrix} 108 \\ 118 \end{pmatrix} \equiv \begin{pmatrix} 9 \times 108 + 14 \times 118 \\ 8 \times 108 + 19 \times 118 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 2624 \\ 3106 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 84 \\ 58 \end{pmatrix} \longrightarrow \begin{pmatrix} T \\ : \end{pmatrix}$$

$$\begin{bmatrix} 9 & 14 \\ 8 & 19 \end{bmatrix} \begin{pmatrix} 101 \\ 114 \end{pmatrix} \equiv \begin{pmatrix} 9 \times 101 + 14 \times 114 \\ 8 \times 101 + 19 \times 114 \end{pmatrix}$$

$$\equiv \begin{pmatrix} 2505 \\ 2974 \end{pmatrix} (mod\ 127) \equiv \begin{pmatrix} 92 \\ 53 \end{pmatrix} \longrightarrow \begin{pmatrix} \backslash \\ 5 \end{pmatrix}$$

The ciphertext of *M* is

*C* = :wT:\5.

The ciphertext *C* of *M* forms a cycle digraph :wT:\5 that is shown in Figure (4.19).

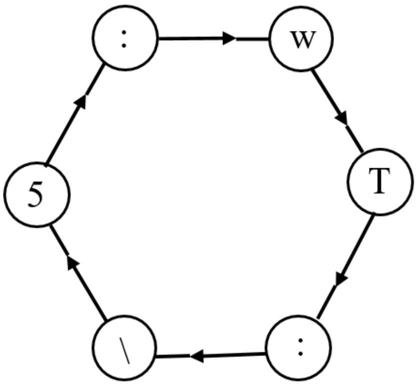


Figure 4.19. The cycle digraph $C_6$ of the ciphertext :wT:\5.

This cycle is represented as the triple vertex digraph (TVDG$_3$($C_6$)) that is given in Figure (4.20) and sent to receiver by sender.

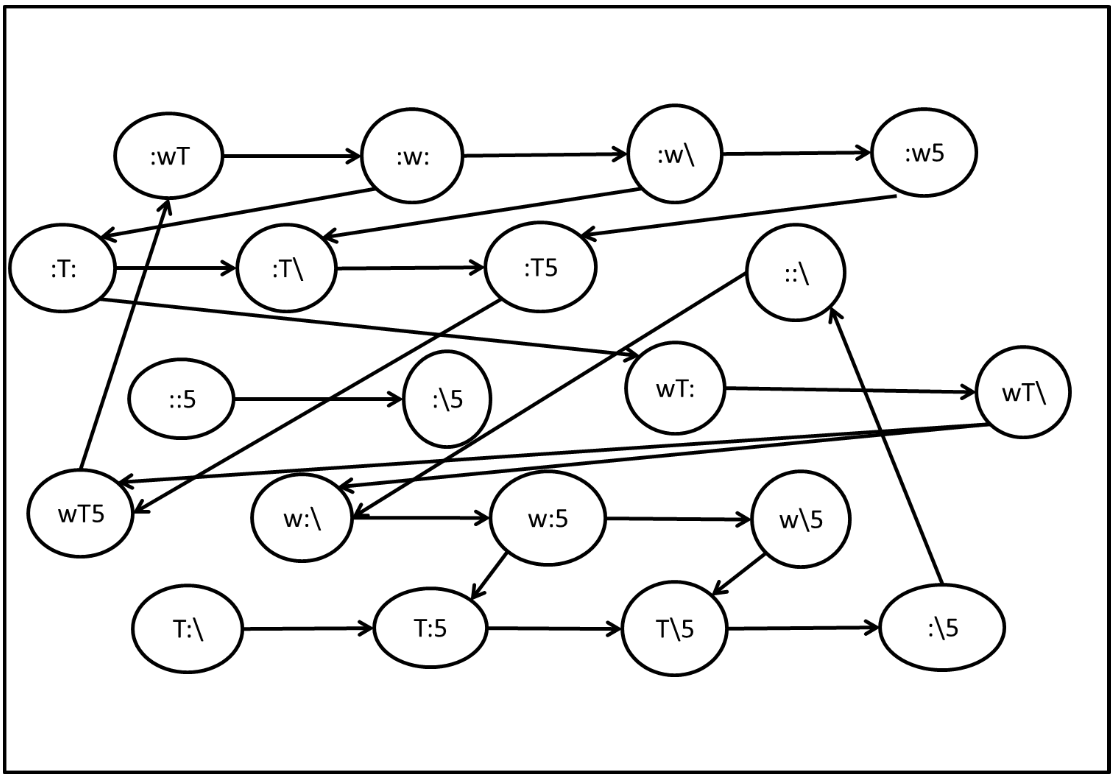


Figure 4.20. The $TVDG_3(C_6)$ of the ciphertext :wT:\5.

After the second user (receiver) receives The $TVDG_3(C_6)$, he/ she wants to decrypt the ciphertext and recover the original plaintext. He/ She first determines the label vertices :wT , :w: , :w\, :w5 , :T: , :T\, :T5, ::\, ::5, :\5, wt:, wT\, wT5, w:\, w:5, w\5, T:\, T:5, T\5, and :\5, based on $TVDG_3(C_6)$. Later on, he/she takes only the letters from vertices without repeating to form a list. There are many cases to determine the correct choice of this list. The correct one is :wT:\5.

Now, the decryption of the Ciphertext :wT:\5, is done using Hill cipher based on the ASCII Table (3.2). Second user computes the determinate of the key $K$ by

$\text{Det}(K) = (9\times19)-(8\times14)$ ($mod$ 127) $\Longrightarrow$ 59($mod$ 127) $\equiv$ 59, the inverse of $\text{Det}(K)$ is computed by $59^{-1}$ ($mod$ 127) $\equiv$ 28. Now, the key inverse $(K)^{-1}$ has been computed by

$K^{-1} \quad \equiv 28\begin{bmatrix}19 & -14\\ -8 & 9\end{bmatrix} \Longrightarrow \begin{bmatrix}532 & -392\\ -224 & 252\end{bmatrix}(mod\ 127) \equiv \begin{bmatrix}24 & 116\\ 30 & 125\end{bmatrix},$

where $\begin{bmatrix}5 & -27\\ -2 & 17\end{bmatrix}$ is an adjacent matrix of $K$

Since the ciphertext, is given by

$$\binom{:}{w}\binom{58}{119} \mid \binom{T}{:}\binom{84}{58} \mid \binom{\backslash}{5}\binom{92}{53}$$

$$\begin{bmatrix}24 & 116\\ 30 & 125\end{bmatrix}\binom{58}{119} \equiv \binom{24\times 58+116\times 119}{30\times 58+125\times 119}$$

$$\equiv \binom{15196}{16615}(mod\ 127) \equiv \binom{83}{105} \longrightarrow \binom{S}{i}$$

$$\begin{bmatrix}24 & 116\\ 30 & 125\end{bmatrix}\binom{84}{58} \equiv \binom{24\times 84+116\times 58}{30\times 84+125\times 58}$$

$$\equiv \binom{8744}{9770}(mod\ 127) \equiv \binom{108}{118} \longrightarrow \binom{l}{v}$$

$$\begin{bmatrix}24 & 116\\ 30 & 125\end{bmatrix}\binom{92}{53} \equiv \binom{24\times 92+116\times 53}{30\times 92+125\times 53}$$

$$\equiv \binom{8356}{9385}(mod\ 127) \equiv \binom{101}{114} \longrightarrow \binom{e}{r}$$

So, the plaintext $M$ is Silver.

## 4.8 The Security Considerations on the Proposed Symmetric Encryption Schemes

The security considerations of new proposed encryption schemes depended on secret generating of a cycle graph CG that the attackers want to know it if they determine the ciphertext is computed as TVC digraph. So, Eve needs to guess the vertices of digraph CG and also the shared secret key, with Polyalphabetic, Affine and Hill modified cipher systems.

Therefore, in case I, there are 26 possible probabilities to form the correct cycle graph. Thus, the total probability can be determined by

$$\mathrm{C}_{\mathrm{k}}^{\mathrm{n}} = \binom{\mathrm{n}}{\mathrm{k}} = \frac{\mathrm{n}!}{\mathrm{k}!\,(\mathrm{n}-\mathrm{k})!}$$

Based on the result of Case (4.4.1), with n = 26 and K =5, then $C_5^{26}$= 65780 paths are existed, one of them is correct one. Whereas, the probability of all possible to the cycle graphs, on study case (4.4.2), is computed by $C_5^{127}$= 254231775cycle , one is the correct that gives the correct original plaintext. So, if the adversaries know a ciphertext of a plaintext m is computed by CG and represented and sent as the TVC digraph, they need to guess more and more probability cases to generate the graphs CG. Thus, it is more secure to recover the original message among all possible probabilities cases.

## 4.9 Comparison on the TVCG-SE and TVCDG-SE

Previous symmetric encryption schemes, for example, polyalphabetic, affine and Hill cipher schemes are less reliable according to the developments in technology today, since few possibilities are available to recover the secret key and it is easy to access the plaintext. But, with the proposed definitions, namely TVCG and TVCDG, new versions of the symmetric encryption schemes. These schemes are more secure. To recover the original plaintext, it requires a lot of possibilities to determine correct cycle of the ciphertext based on the TVCG and correct directed cycle of the ciphertext based on the TVCDG. This indicates to the security of the transmitted information has become higher than before and it has become difficult to break through. Compared with the results obtained from the graphs of the triple vertex cycle and the triple vertex directed cycle of, the possibilities to find the correct cycle of ciphertext are many.

For example, to find the plaintext of five letters, it needs 65,780 possible cases. With a plaintext that has more than five letters, there are largest chances of finding the original plaintext, and this makes the proposed encryption schemes based on TVCG and TVCDG are more secure than the previous encryption schemes

# Chapter Five

# Conclusions and Future Works

The conclusions and future work of this work can be discussed as follows.

## 5.1 Conclusions

New graphs which are called TVG, TVCG, TVDG, and TVCDG are defined as main points in this work. These graphs are used to design new versions of symmetric encryption schemes using the EAVs and ASCII values. Some kinds of symmetric encryption schemes are proposed based on the merged the proposed graphs and the polyalphabetic or an affine or the hill encryption schemes that are depended on the EAVs and ASCII values. On the TVCG-SE and TVCDG-SE schemes, the security considerations have been determined. TVCG-SE and TVCDG schemes are more secure for communication in compare to the previous symmetric encryption schemes.

## 5.2 Future Works

In this work, new graphs are suggested to improve the security of symmetric encryption techniques. In order to improve these schemes and make them more secure communication systems, it is also possible to suggest different types of graphs. It is possible to create more symmetric encryption schemes using the TVCG and TVCDG. Based on the suggested graphs, new iterations of the asymmetric encryption techniques can be made. The TVC and TVCD.

# References

1. S. Cheema, J. Kohli, K. Arora, S. Gupta, and S. S. Ahmed. Network security using graph theory. International Journal of Innovation in Engineering and Technology, 2:131–138, 2013.
2. Aljamaly, Karrar Taher R., and Ruma Kareem K. Ajeena. "The elliptic scalar multiplication graph and its application in elliptic curve cryptography." Journal of Discrete Mathematical Sciences and Cryptography 24.6 (2021): 1793-1807.
3. Aljamaly, Karrar Taher R., and Ruma Kareem K. Ajeena. "The kr-elliptic curve public key cryptosystem." Journal of Physics: Conference Series. Vol. 1879. No. 3. IOP Publishing, 2021.
4. Ni, R. Qazi, S. U. Rehman, and G. Farid. Some graph-based encryption schemes. Journal of Mathematics, 2021, 2021.
5. Bellare, Mihir, and Phillip Rogaway. "Introduction to modern cryptography." Ucsd Cse 207 (2005): 207.
6. Beaula and P. Venugopal. Cryptosystem using double vertex graph. Indian Journal of Science and Technology, 13(44):4483–4489, 2020.
7. A. Sherin and V. Maheswari. Encoding the graph using instant insanity puzzle and decoding with hamiltonian cycle. The International Journal of analytical and modal analysis, ISSN-08869-9367. P, (167-175).
8. A. Sherin and V. Maheswari. Encryption and decryption process using edge magic labeling. In Journal of Physics: Conference Series, volume 1362, page 012024. IOP Publishing, 2019.
9. D. A. Sherin, V. Maheswari, and V. Balaji. Encryption of dual numbers using edge injective labeling. 2021.
10. D. Jao, S. D. Miller, and R. Venkatesan. Expander graphs based on grh with an application to elliptic curve cryptography. Journal of Number Theory, 129(6):1491–1504, 2009.
11. D. Sensarma and S. S. Sarma. Application of graphs in security. Int. J. Innov. Technol. Explor. Eng., 8(10):2273–2279, 2019.
12. Essam, John W., and Michael E. Fisher. "Some basic definitions in graph theory." Reviews of Modern Physics 42.2 (1970): 271.
13. Amounas. An innovative approach for enhancing the security of amazigh text using graph theory based ecc. 2016.

14. Samid. Denial cryptography based on graph theory, Nov. 23 2004. US Patent 6,823,068.
15. U. Maheswari, J. Arthy, and S. J. Obaiys. A method of secret coding technique on two star graphs. International Journal of Computer Applications, 177(39):11–15, 2020.
16. O. Abdullah and M. Eftekhari. Cryptanalysis and improvements on some graph-based authentication schemes. Journal of Discrete Mathematical Sciences and Cryptography, 16(4-5):297–306, 2013.
17. Santoso, I. Agustin, and R. Prihandini. The modification of caesar cryptosystem based on binary vertices colouring. In Journal of Physics: Conference Series, volume 1538, page 012006. IOP Publishing, 2020.
18. Mittenthal. Sequencings and directed graphs with applications to cryptography. In Sequences, Subsequences, and Consequences, pages 70–81. Springer, 2007.
19. A. Ahmed and J. B. Babujee. Encryption through labeled graphs using strong face bimagic labeling. In International Mathematical Forum, volume 12, pages 151–158, 2017.
20. Yamuna, A. Sankar, S. Ravichandran, and V. Harish. Encryption of a binary string using music notes and graph theory. International Journal of Engineering and Technology, 5(3):2920–2925, 2013.
21. Yamuna, M. Gogia, A. Sikka, and M. J. H. Khan. Encryption using graph theory and linear algebra. International Journal of Computer Application, 5(2):102–107, 2012.
22. Amudha, A. C. Sagayaraj, and A. S. Sheela. An application of graph theory in cryptography. International Journal of Pure and Applied Mathematics, 119(13):375–383, 2018.
23. Femina and D. A. David. A study of data encryption standard using graph theory. In 2nd International Conference on Science, Technology and Management, University of Delhi, New Delhi, India, 2015.
24. Kedia and S. Agrawal. Encryption using venn-diagrams and graph. International Journal of Advanced Computer Technology, 4(1):94–99, 2015.
25. P. Perera and G. Wijesiri. Encryption and decryption algorithms in symmetric key cryptography using graph theory. Psychology and Education Journal, 58(1):3420–3427, 2021.

26. P. Priyadarsini and R. Ayyagari. Ciphers based on special graphs. In 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pages 460–465. IEEE, 2013.
27. J. Wilson. Introduction to graph theory. Pearson Education India, 1979.
28. K. K. Ajeena. Integer sub-decomposition (ISD) method for elliptic curve scalar multiplication. PhD thesis, Universiti Sains Malaysia, 2015.
29. R. K. K. Ajeena. The graph and its role for speeding up the elliptic scalar multiplication algorithms. In 2019 2nd International Conference on Engineering Technology and its Applications (IICETA), pages 227–228. IEEE, 2019.
30. R. K. K. Ajeena. The graphs for elliptic curve cryptography. In Applied Mathematics. IntechOpen, 2019.
31. R. Selvakumar and N. Gupta. Fundamental circuits and cut-sets used in cryptography. Journal of Discrete Mathematical Sciences and Cryptography, 15(4-5):287–301, 2012.
32. Rahman, Md Saidur. Basic graph theory. Vol. 9. India: Springer, 2017.
33. A. Abdul-Ghani, R. D. Abdul-Wahhab, and E. W. Abood. Securing text messages using graph theory and steganography. Baghdad Science Journal, 19(1):0189–0189, 2021.
34. S. Agarwal and A. S. Uniyal. Prime weighted graph in cryptographic system for secure communication. International Journal of Pure and Applied Mathematics, 105(3):325–338, 2015.
35. S. G. Akl. How to encrypt a graph. International Journal of Parallel, Emergent and Distributed Systems, 35(6):668–681, 2020.
36. Stinson, Douglas R. Cryptography: theory and practice. Chapman and Hall/CRC, 2005.
37. A. Ustimenko. Graphs with special arcs and cryptography. Acta Applicandae Mathematica, 74(2):117–153, 2002.
38. Ustimenko. Cryptim: Graphs as tools for symmetric encryption. In International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes, pages 278–286. Springer, 2001.

39. M. Al Etaiwi. Encryption algorithm using graph theory. Journal of Scientific Research and Reports, pages 2519–2527, 2014.

40. Z. Jiang, R. Zhong, and B. Zheng. A software watermarking method based on public-key cryptography and graph coloring. In 2009 Third International Conference on Genetic and Evolutionary Computing, pages 433–437. IEEE, 2009.

# المـلخص

تم تعريف الرسوم البيانية للرأس الثلاثية (TV) والدورة الثلاثية (TVC) على أنها مفاهيم جديدة لنظرية الرسم البياني. هذه المفاهيم هي نقطة أساسية في هذا العمل لتصميم إصدارات جديدة من أنظمة التشفير المتماثل .(SE) في مخططات SE المقترحة ، يتم زيادة مستوى الأمان مقارنة بمخططات SE السابقة. ويتم ايظاً تطبيق الرسوم البيانية الجديدة مع أنواع مختلفة من مخططات SE مثل polyalphabetic, affine and hill schemes التي تعتمد على قيم EAVs و ASCIIلتمثيل أحرف النص العادي أو النص المشفر. في المخططات المقترحة ، يتم إرسال النص المشفر على شكل TVCG من قبل المرسل إلى المستقبل.

من ناحية أخرى ، يُعرَّف الرسم البياني ذو الرأس الثلاثي (TVDG) والرسم البياني المتجه ذو الرأس الثلاثي (TVCDG) على أنهما مفاهيم جديدة أيضًا. تم تصميم إصدارات جديدة من مخططات التشفير المتماثل (SE) بناءً على TVCDGلإعطاء إصدارات جديدة من مخططات التشفير. يتم تطبيق هذه الرسوم البيانية أيضًا على مخططات polyalphabetic, affine and hill schemes التي تستخدم قيم EAVs و ASCII لتمثيل أحرف النص العادي أو النص المشفر. في المخططات المقترحة ، يتم إرسال النص المشفر كـ TVCG إلى المتلقي بواسطة المرسل.

تمت مناقشة النتائج التجريبية الجديدة لمخططات TVCG-SE و TVCDG-SE المقترحة. تم تحديد الاعتبارات الأمنية لأنظمة TVPG-SE و .TVCDG-SE تم شرح المقارنة بين مخططات TVCG-SE و TVCDG-SE المقترحة مع مخططات SE الأصلية. تعد مخططات TVCG-SE و TVCDG-SE المقترحة رؤى جديدة أكثر أمانًا مقارنة بالرؤى السابقة.

وزارة التعليم العالي والبحث العلمي

جامعة بابل

كلية التربية للعلوم الصرفة

# بيان الرأس الثلاثي الدوار لانظمة التشفير

رسالة

مقدمة الى مجلس كلية التربية للعلوم الصرفة / جامعة بابل وهي جزء

من متطلبات نيل درجة الماجستير في التربيه \ الرياضيات

**من قبل الطالبة**

**منال هاتف كاظم سعيد**

**أشراف**

**أ.م.د. رومى كريم عجينه**

٢٠٢٢ م ١٤٤٤ هـ