Available online at http://scik.org
J. Math. Comput. Sci. 2026, 16:5
https://doi.org/10.28919/jmcs/9638
ISSN: 1927-5307

CRYPTANALYSIS ON MODIFIED VIGENERE USING BIPARTITE GRAPH

KOUSHANI ROY, SUPRIYO MAZUMDER*

Department of Mathematical Sciences, Adamas University, Barasat - Barrackpore Road, Jagannathpur, Kolkata, India

Abstract. In the recent digital era of data transmission and network operations, data security and safety are some of the biggest challenges faced by the mankind. Cryptography is the study of encryption and decryption, the procedure of converting data into a format which prevents the unauthorized parties to access, analyse or change the shared data. An increasing number of studies demonstrating new approaches of Cryptography can be found in this area. Among these, one of the most burgeoning approaches of Cryptography is Graph Theory. Various graphs with their unique properties have been implemented in the cryptographic algorithms which have effectively resisted the unwanted parties from interfering. Vigenere Cipher is one of the most secured encryptions which adapts the concept of substitution cipher. An algorithm with the concept of Vigenere Cipher along with Bipartite Graphs has been proposed in this paper. The existing Vigenere Cipher has only 26 alphabets which limits the eligibility of the data types with only the uppercase alphabets. In this algorithm a modified Vigenere table has been proposed with all the characters with ASCII values, covering a wide range of all kinds of texts. This work enables security, authenticity and confidentiality of the shared information with the bipartite graph characteristics like vertex partitioning, edge weighting etc. The XOR operations used in this work ensures more security. All the security procedures ensures to prevent some of the most threatening third party attacks (for example, Brute Force Attack, Plaintext attack, ciphertext only attack etc.).

Keywords: cryptography; bipartite graph; Vigenere cipher; shift cipher; encryption; ASCII codes.

2020 AMS Subject Classification: 05C90, 05C99, 94A60.

*Corresponding author

E-mail address: supriyo88@gmail.com

Received October 09, 2025

1. INTRODUCTION

Graph theory has become one of the key components of encryption and decryption over time [19] [21] [29] [31]. In 2018, P. Amudha et al. [3] discovered an encryption method by determining the text's ASCII values, locating the matching binary values, and using the XOR operation on them. This research got an enhancement by Joseph et al. [13] introducing a key in cryptography algorithms. In this area, Kedia et al. [16] worked with the binary and hexadecimal representations of the plain text's ASCII values by transforming them into a graph that used the concept of graph connections. According to research by W. Etaiwi [10] and U. Dixit [9], a simple text can be transformed into the vertices of a weighted graph with pre-assigned values. A few algorithms also make use of the adjacency matrix concept [25] [22].

A symmetrical cryptographical algorithm involving infinite families of graphs with directions of huge girth was created in 2007 by V. Ustimenko [30]. Network security has additionally benefited significantly from the matrix approach of graph theory [20]. In the work of M. H. R. et al. [12], the idea of a graph's complement to produce the cipher graph has been covered. Agarwal [1] built a Java application and described how an algorithm can work with a prime weighted graph. Research conducted by Beaula [5] shows how cryptography and double vertex graphs are integrated. The writer furthermore illustrated a strong cryptosystem using 'Graph labelling' on Turan graphs [6]. Kuppan [17] used face antimagic tagging in a tree graph to encode and decode thirteen secret digits.

For a last few years, the concept of Bipartite Graph in Data security has been emerged extensively in the field of Cryptography [7] [18] [32]. Gurjar et al. [11] incorporated the application of Graph Labelling with Bipartite graphs in his paper in 2021.

A demand of encrypting all characters including the alphabets, special characters and numbers, can be found in this research area. To work on this research gap and also to introduce more confidentiality in encryption, this paper has been designed with all the 256 characters with ASCII values. The proposed algorithm not only deals with the security but also enables all the characters for encryption.

2. PRELIMINARIES

In this section we take a glance at some ideas of Graph Theory [8] and Cryptography [27] which are used to construct the proposed algorithm.

Definition 2.1. A graph $G(V,E)$ is a set of two sets E of edges and V of vertices such that each edge is defined as a joining side between a pair of vertices. If two vertices are joined by an edge then those vertices are said to be connected.

Definition 2.2. A Bipartite Graph can be a directed or undirected graph where the vertices can be divided into two disjoint sets such that no two vertices of the same set are connected (Figure 1).

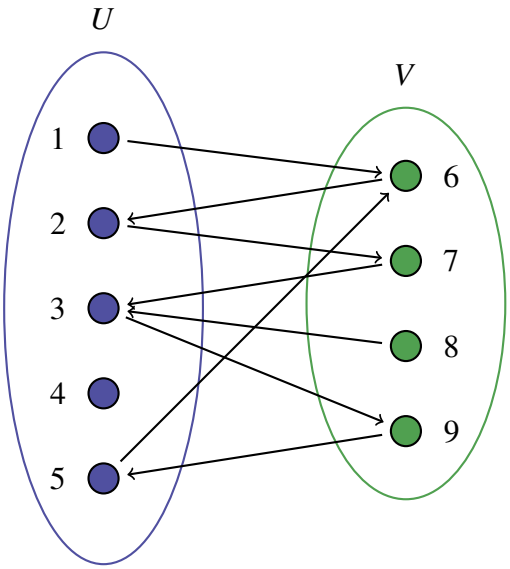

FIGURE 1. Example of a Directed Bipartite Graph

Definition 2.3. Cryptography is the study of encryption and decryption of messages. We find a code, known as a cipher text which is the encrypted form of a plain text and can be decrypted to find the plain text and this process may follow one or more algorithms.

Definition 2.4. Encryption is the procedure of getting a pseudo text from a given plain text to maintain the authenticity and the confidentiality of the plain text shared by the sender. Decryption is the procedure to get back to the shared plain text from the pseudo text sent by the receiver.

ASCII control characters			ASCII printable characters						Extended ASCII characters							
00	NULL	(Null character)	32	space	64	@	96	`	128	Ç	160	á	192	└	224	Ó
01	SOH	(Start of Header)	33	!	65	A	97	a	129	ü	161	í	193	┴	225	ß
02	STX	(Start of Text)	34	"	66	B	98	b	130	é	162	ó	194	┬	226	Ô
03	ETX	(End of Text)	35	#	67	C	99	c	131	â	163	ú	195	├	227	Ò
04	EOT	(End of Trans.)	36	$	68	D	100	d	132	ä	164	ñ	196	─	228	õ
05	ENQ	(Enquiry)	37	%	69	E	101	e	133	à	165	Ñ	197	┼	229	Õ
06	ACK	(Acknowledgement)	38	&	70	F	102	f	134	å	166	ª	198	ã	230	µ
07	BEL	(Bell)	39	'	71	G	103	g	135	ç	167	º	199	Ã	231	þ
08	BS	(Backspace)	40	(	72	H	104	h	136	ê	168	¿	200	╚	232	Þ
09	HT	(Horizontal Tab)	41	)	73	I	105	i	137	ë	169	®	201	╔	233	Ú
10	LF	(Line feed)	42	*	74	J	106	j	138	è	170	¬	202	╩	234	Û
11	VT	(Vertical Tab)	43	+	75	K	107	k	139	ï	171	½	203	╦	235	Ù
12	FF	(Form feed)	44	,	76	L	108	l	140	î	172	¼	204	╠	236	ý
13	CR	(Carriage return)	45	-	77	M	109	m	141	ì	173	¡	205	═	237	Ý
14	SO	(Shift Out)	46	.	78	N	110	n	142	Ä	174	«	206	╬	238	¯
15	SI	(Shift In)	47	/	79	O	111	o	143	Å	175	»	207	¤	239	´
16	DLE	(Data link escape)	48	0	80	P	112	p	144	É	176	░	208	ð	240	≡
17	DC1	(Device control 1)	49	1	81	Q	113	q	145	æ	177	▒	209	Ð	241	±
18	DC2	(Device control 2)	50	2	82	R	114	r	146	Æ	178	▓	210	Ê	242	‗
19	DC3	(Device control 3)	51	3	83	S	115	s	147	ô	179	│	211	Ë	243	¾
20	DC4	(Device control 4)	52	4	84	T	116	t	148	ö	180	┤	212	È	244	¶
21	NAK	(Negative acknowl.)	53	5	85	U	117	u	149	ò	181	Á	213	ı	245	§
22	SYN	(Synchronous idle)	54	6	86	V	118	v	150	û	182	Â	214	Í	246	÷
23	ETB	(End of trans. block)	55	7	87	W	119	w	151	ù	183	À	215	Î	247	¸
24	CAN	(Cancel)	56	8	88	X	120	x	152	ÿ	184	©	216	Ï	248	°
25	EM	(End of medium)	57	9	89	Y	121	y	153	Ö	185	╣	217	┘	249	¨
26	SUB	(Substitute)	58	:	90	Z	122	z	154	Ü	186	║	218	┌	250	·
27	ESC	(Escape)	59	;	91	[	123	{	155	ø	187	╗	219	█	251	¹
28	FS	(File separator)	60	<	92	\	124	|	156	£	188	╝	220	▄	252	³
29	GS	(Group separator)	61	=	93	]	125	}	157	Ø	189	¢	221	¦	253	²
30	RS	(Record separator)	62	>	94	^	126	~	158	×	190	¥	222	Ì	254	■
31	US	(Unit separator)	63	?	95	_			159	ƒ	191	┐	223	▀	255	nbsp
127	DEL	(Delete)														

TABLE 1. ASCII Table

Definition 2.5. A Plain Text is the main text sent by the sender.

Definition 2.6. A Cipher text is the Encrypted text that is received by the receiver.

Definition 2.7. The American Standard Code for Information Interchange(ASCII) is a World wide used encoding system that gives the codes of alphabets, characters, numbers etc. to be used and accessed by the computer and other electronic devices (Table 1).

Definition 2.8. A Vigenere Cipher is a poly-alphabetic symmetric cipher. A Vigenere table includes all 26 alphabets written row wise and column wise in a 26×26 matrix. Each alphabet is shifted in a cyclic manner to the right so that a table is formed like the Table 2

Definition 2.9. A security attack occurs when a third party, other than the sender and receiver, tries to access, change or modify a confidential data. here are different kinds of attacks in Cryptography like: Brute Force attack(when the attacker tries the trial and error method with all possibilities), plaintext attack(when a part of the plaintext is known and the attacker cracks the whole text) etc.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

TABLE 2. Vigenere Table for 26 characters

Definition 2.10. A key is a string of characters shared only between the sender and the receiver and helps to encrypt and decrypt the data with more security.

Definition 2.11. A symmetric key encryption is an encryption technique that only takes one single key for encryption and that key is shared between the sender and the receiver.

Definition 2.12. A binary number system is a system used vastly mainly in the computers where all the existing numbers can be represented with only two digits 0(zero) and 1(one).

Input 1	Input 2	Output
T	T	F
T	F	T
F	T	T
F	F	F

TABLE 3. Truth Table for Binary XOR Operation

Definition 2.13. The XOR Gate, also known as the Exclusive OR gate operates on two binary inputs and the output has the maximum length of the highest length input. The truth table for the XOR gate is Table 3.

In the next section we have proposed one encryption scheme using Modified Vigenere Table and the Bipartite Graph. Also in section 3 we have shown one example to highlight the validity of this scheme. The advantages that come with using this proposed technique are covered in the last section.

3. ENCRYPTION SCHEME USING THE MODIFIED VIGENERE TABLE AND THE BIPARTITE GRAPH:

In this section we have used the concepts and properties of the Graph theory to find a feasible algorithm that can be used to encrypt and decrypt messages.

3.1. Encryption Algorithm. The encryption technique of the proposed algorithm follows the following steps:

(1) We do a modification on the Vigenere Table where instead of only 26 existing alphabets, we take all the characters with ASCII codes. Their ASCII values are the elements taken in rows and columns. Each row starts with the next value of the code in a cyclic manner. (Table. 4).

(2) Find the ASCII values of the plain text from the ASCII table (Table 1). Let these values be a_i where, $i = 1, ..., n$ for $n \in \mathbb{N}$.

	0	1	2	3	4	5	6	7	8	9	10	.	.	.	250	251	252	253	254	255
0	1	2	3	4	5	6	7	8	9	10	11	.	.	.	251	252	253	254	255	0
1	2	3	4	5	6	7	8	9	10	11	12	.	.	.	252	253	254	255	0	1
2	3	4	5	6	7	8	9	10	11	12	13	.	.	.	253	254	255	0	1	2
3	4	5	6	7	8	9	10	11	12	13	14	.	.	.	254	255	0	1	2	3
4	5	6	7	8	9	10	11	12	13	14	15	.	.	.	255	0	1	2	3	4
5	6	7	8	9	10	11	12	13	14	15	16	.	.	.	0	1	2	3	4	5
6	7	8	9	10	11	12	13	14	15	16	17	.	.	.	1	2	3	4	5	6
7	8	9	10	11	12	13	14	15	16	17	18	.	.	.	2	3	4	5	6	7
8	9	10	11	12	13	14	15	16	17	18	19	.	.	.	3	4	5	6	7	8
9	10	11	12	13	14	15	16	17	18	19	20	.	.	.	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19	20	21	.	.	.	5	6	7	8	9	10
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
250	251	252	253	254	255	0	1	2	3	4	5	.	.	.	244	245	246	247	248	249
251	252	253	254	255	0	1	2	3	4	5	6	.	.	.	245	246	247	248	249	250
252	253	254	255	0	1	2	3	4	5	6	7	.	.	.	246	247	248	249	250	251
253	254	255	0	1	2	3	4	5	6	7	8	.	.	.	247	248	249	250	251	252
254	255	0	1	2	3	4	5	6	7	8	9	.	.	.	248	249	250	251	252	253
255	0	1	2	3	4	5	6	7	8	9	10	.	.	.	249	250	251	252	253	254

TABLE 4. Modified Vigenere Table with 256 ASCII Values

(3) We introduce a string of any length as the 'key' which is to be shared between the sender and the receiver. If this 'key' has a length smaller than the plain text, then we keep on repeating the characters of the key one by one in a cyclic way till the end of the plain text character.

(4) We find the corresponding ASCII values of the characters of the shared key string and consider them as b_i's.

(5) We locate the a_i's along the column headings and the b_i's along the row headings and find the value at the intersection of that particular a_ith column and b_ith row. These values are considered as c_i's.

(6) We convert the b_i's and the c_i's to binary numbers and name these as p_i's and q_i's respectively.

(7) We perform the XOR (Exclusive OR) operation taking the p_i's and q_i's as inputs and consider the outputs as m_i's. As the maximum possible ASCII value is 255, and the corresponding binary number is 11111111 which is of 8 digits, therefore, the maximum possible length of binary m_i's is also 8.

(8) We convert the m_i's to the corresponding decimals. Consider these decimals as s_i's. The highest s_i is 255 and hence the maximum possible digits of these values is 3.

(9) The last digit, which is also the tens digit of these s_i's are taken as the elements of the set v_2 and the remaining previous part of the numbers are taken as the elements of the set v_1. If any number has the length of less than 3, then we consider 0 in the missing digit places before the existing digits.

(10) Construct a Bipartite Graph taking v_1 and v_2 as the two vertex sets and connect the pair of the vertices if they are making any of the numbers of s_i.

(11) Assign some random values(also known as the 'weights') w_i's to the edges in an ascending order maintaining the order of the characters of the plain text one by one such that no two edges will have equal weights, i.e. $w_1 < w_2 < w_3 < ....$ This weighted Bipartite graph is the required Cipher Text.

3.2. Decryption algorithm: The recipient receives the labeled bipartite graph as the cipher text and the key.

(1) At first, we take the weights of the edges of the cipher text bipartite graph and arrange them in an increasing order.

(2) Find the corresponding edges of the weights and the nodes they are connected to. The set of vertices having three digit numbers are considered as v_1 and that of one digit numbers as v_2.

(3) Construct the numbers with connected edges taking the one digit numbers as the unit's place and the three digit numbers as the previous part of the unit's place. Consider these constructed numbers as s_i's.

(4) Find the Binary number corresponding to the values of s_i's. Name these numbers as m_i's.

(5) Find the ASCII values of the characters of the shared key string from Table 1. Consider these values as b_i's.

(6) convert the b_i's to corresponding Binary numbers. Name these as p_i's.

(7) Consider the m_i's and the p_i's as the inputs and perform XOR operation between them and the result output is considered as q_i's. Then convert these q_i's to corresponding decimals c_i's.

(8) For the final step, we refer to the modified 256×256 Vigenere Table (Table 4). We locate the position of the c_i's in the b_ith row. The column where that particular c_i is found, that column is a_i and these are the ASCII values of the required plain text.

(9) At the last step, we convert the obtained ASCII values to the corresponding characters from the ASCII table (Table 1. These characters together make the string of the plain text.

4. Example of the Proposed Algorithm

Example 4.1. Let us take the plain text "I have done $5+8=13$." and the key is "Homework" The following are the required steps for the Encryption.

Step 1: The following table (Table 5) finds the final s_i values for the given plain text with the shared key. In the given table,

a_i= ASCII values of the plain text shared by the sender.

b_i= ASCII values of the characters of the key string.

c_i= Elements at the intersection of a_i'th column and b_i'th row.

Step 2: Form the set v_2 (unit's digits of each number) as $v_2 = \{2,3,4,5,7,8,9\}$ and the set v_1 taking the remaining section. Therefore, $v_1 = \{21,24,14,17,22,18,03,20,19\}$

Step 3: Form the Bipartite graph with these vertex sets. Connect the vertices forming the numbers of s_i's and assign random weights to the edges in an ascending order. This weighted bipartite graph is the required cipher text (Figure. 2).

Plain Text	a_i	**Key String**	b_i	c_i	p_i	q_i	m_i	s_i
I	73	H	72	145	1001000	10010001	11011001	217
’	39	o	111	150	1101111	10010110	11111001	249
v	118	m	109	227	1101101	11100011	10001110	142
e	101	e	101	202	1100101	11001010	10101111	175
space	32	w	119	151	1110111	10010111	11100000	224
d	100	o	111	211	1101111	11010011	10111100	188
o	111	r	114	225	1110010	11100001	10010011	147
n	110	k	107	217	1101011	11011001	10110010	178
e	101	.	46	147	101110	10010011	10111101	189
spaec	32	H	72	104	1001000	01101000	100000	32
5	53	o	111	164	1101111	10100100	11001011	203
+	43	m	109	152	1101101	10011000	11110101	245
8	56	e	101	157	1100101	10011101	11111000	248
=	61	w	119	180	1110111	10110100	11000011	195
1	49	o	111	160	1101111	10100000	11001111	207
3	51	r	114	165	1110010	10100101	11010111	215
.	46	k	107	153	1101011	10011001	11110010	242

TABLE 5. Table to find the s_i’s corresponding to the plain text

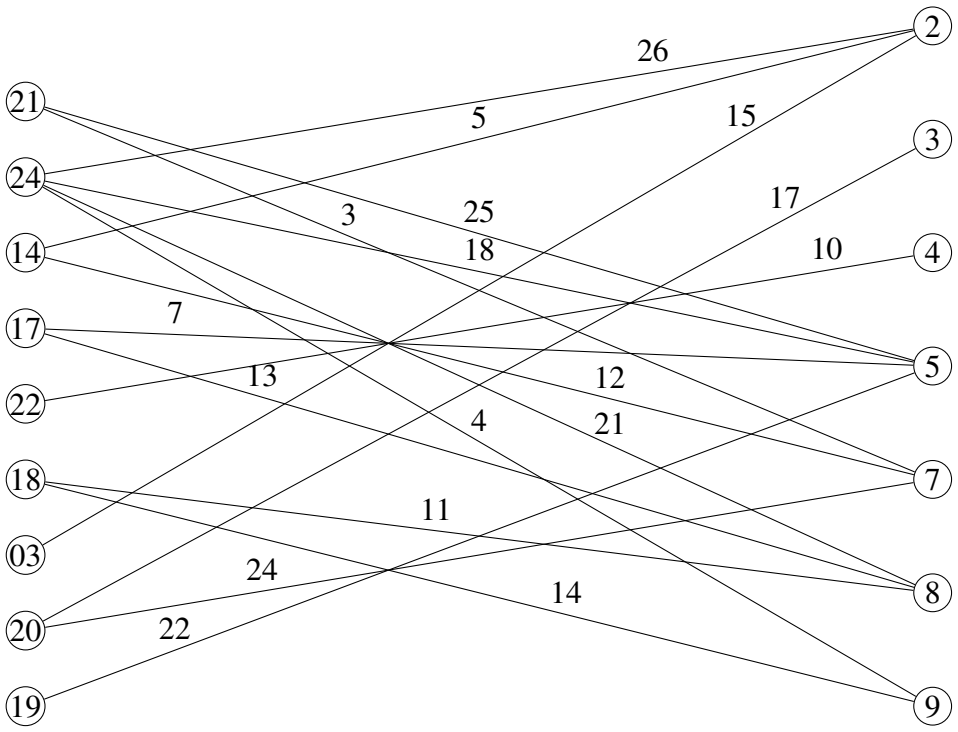

FIGURE 2. Bipartite Graph as the Cipher Text

For decryption, we have the bipartite graph as the cipher text (Figure 2) and the shared key string: "Homework.". The decryption of this cipher text can be done in the following steps.

Step 1: Arrange all the weights in an ascending order and we get,

$3,4,5,7,10,11,12,13,14,15,17,18,19,21,22,24,25,26.$

Step 2: Finding the corresponding edges related to these weights, we get,

$(21,7),(24,9),(14,2),(17,5),(22,4),(18,8),(14,7),(17,8),(18,9),(03,2),(20,3),$
$(24,5),(24,8),(19,5),(20,7),(21,5),(24,2)$

Step 3: The s_i numbers are formed as:

$217,249,142,175,224,188,147,178,189,032,203,245,248,195,207,215,242$

Step 4: A table(Table 6) is generated to get back the plain text from these s_i values following the mentioned decryption algorithm converting the numbers to corresponding binary numbers and performing XOR accordingly. The shared key which the receiver already has is "Homework."

Thus the Plain text is obtained as "I've done 5+8=13."

s_i	m_i	p_i	q_i	c_i	b_i	Key String	a_i	Plain Text
217	11011001	1001000	10010001	145	72	H	73	I
249	11111101	1101111	11010110	150	111	o	39	,
142	10001110	1101101	11100011	227	109	m	118	v
175	10101111	1100101	11001010	202	101	e	101	e
224	11100000	1011111	11101111	151	119	w	32	space
188	10111100	1101111	11010011	211	111	o	100	d
147	10010001	1101100	11100010	225	114	r	111	o
178	10110110	1101011	11011001	217	107	k	110	n
189	11011101	1011000	10001111	147	46	.	101	e
32	1000000	1000000	01101000	104	72	H	32	space
203	11101111	1100100	10110101	164	111	o	53	5
245	11111001	1101011	10110101	152	109	m	43	+
248	11110000	1011000	11101111	157	101	e	56	8
195	11000011	1100101	11100011	180	119	w	61	=
207	11000111	1101111	11000010	160	111	o	49	1
215	11101001	1100101	11100001	165	114	r	51	3
242	11110100	1101000	11011001	153	107	k	46	.

TABLE 6. Table to find the final plain text

5. ADVANTAGES OF THE PROPOSED ALGORITHM

5.1. Different kinds of attacks prevention: The large 256×256 Vigenere table with the large keys make it computationally infeasible for any attacker to try all possible keys in a given time frame, which ensures prevention of **Brute Force Attack**. Even if a part of the plaintext (along with the key) is known to the attacker, it's near to impossible for that one to crack the whole text as there are a large number of possible characters with unknown length of the plaintext. This assures the prevention of **Plaintext attack**. Even if the attacker knows the ciphertext and cracks the key somehow, he will be unable to get back the plaintext because of

some long and complecated mathematical deductions and for this, the **Ciphertext-only attack** does not occur.

5.2. Confidentiality, authenticity and privacy: This proposed algorithm works to maintain the security and confidentiality of data with multiple computational steps.

5.3. Acceptance of a wide range of data: The proposed work deals with all of the existing 256 characters which implies a text of any length including any of these characters can be secured with this algorithm.

6. PERFECT SECRECY OF THE ALGORITHM

Definition 6.1. An Encryption scheme with the algorithms **GEN**, **ENC**, **DEC** with message space M is said to be perfect secret if for every probability distribution over M, every message $m \in M$ and every ciphertext $c \in C$, for which $\Pr[C = c] > 0$ (as there should be a ciphertext to keep the decryption process ongoing):

$$\Pr[M = m \mid C = c] = \Pr[M = m]$$

Equivalent formula for the same is:

$$\Pr[\mathrm{Enc}_K(m) = c] = \Pr[\mathrm{Enc}_K(m') = c]; \quad \text{for every } m, m' \in M$$

As the distribution of the ciphertext does not depend on the plaintext. This implies, there can be chances of getting a single ciphertext from different plaintexts [15]

Now, with the definition of the perfectly secret algorithm, we can prove that the algorithm shown here is perfectly secret.

For the proof, we will consider the subpart of the algorithm where we are getting the a_i's. Although it is not the ciphertext, but the mid way text. For this particular algorithm, for the proof we will address that as the ciphertext as the secrecy of the algorithm lies on the values of this only.

We can have the key space as, $K = \{0, 1, 2, \ldots, 24, 25\}$, with probability, $\Pr[K = k] = \frac{1}{26}$

Note 6.2. In case the message space is limited, that is, the adversary already knows that particularly some alphabets are not used, in that case it is not necessary to assign the uniform

distribution, we can define our own distribution assigning probabilistic values to each alphabet such that for the message space M, $\sum_i \Pr[M = m_i] = 1$]

Method 1: When all the alphabets are equally likely to occur: For the message space M, we can have all the 26 alphabets which are equally likely to occur.

Therefore,

$$\Pr[M = A] = \frac{1}{26}, \quad \Pr[M = B] = \frac{1}{26}, \quad \Pr[M = C] = \frac{1}{26}, \ldots, \quad \Pr[M = Z] = \frac{1}{26}$$

Now, let us take the ciphertext B. How many ways we can have B as a ciphertext from the spaces K and M?

There can be 26 ways of getting it: $\{B+0, A+1, Z+2, Y+3, \ldots, D+24, C+25\}$, where $K = \{0, 1, 2, \ldots, 24, 25\}$ and $M = \{A, B, C, \ldots, Y, Z\}$

$$\Pr[M = B \wedge K = 0] = \Pr[M = B] \cdot \Pr[K = 0] = \frac{1}{26} \cdot \frac{1}{26}$$

Similarly,

$$\Pr[M = A \wedge K = 1] = \Pr[M = A] \cdot \Pr[K = 1] = \frac{1}{26} \cdot \frac{1}{26}$$
$$\Pr[M = C \wedge K = 25] = \Pr[M = C] \cdot \Pr[K = 25] = \frac{1}{26} \cdot \frac{1}{26}$$

Therefore,

$$\Pr[C = B] = \Pr[M = B \wedge K = 0] + \Pr[M = A \wedge K = 1] + \Pr[M = Z \wedge K = 2] + \ldots + \Pr[M = C \wedge K = 25]$$
$$= 26 \cdot \left(\frac{1}{26} \cdot \frac{1}{26}\right) = \frac{1}{26}$$

Now, we can also calculate the conditional probability using the Bayes' Theorem:

$$\Pr[M = A \mid C = B] = \frac{\Pr[C = B \mid M = A] \cdot \Pr[M = A]}{\Pr[C = B]}$$
$$= \frac{\Pr[C = B \mid M = A] \cdot \Pr[M = A]}{\Pr[C = B \mid M = A] \cdot \Pr[M = A] + \ldots + \Pr[C = B \mid M = Z] \cdot \Pr[M = Z]}$$
$$= \frac{1}{26} = \frac{1}{26} \cdot \left(\frac{1}{26} + \frac{1}{26} + \cdots + \frac{1}{26}\right) = \frac{1}{26}$$

To be noted that,

$$\Pr[C = B \mid M = A] = \frac{1}{26}$$

since if $M = A$ then the only possible way to have $C = B$ is if $K = 1$.

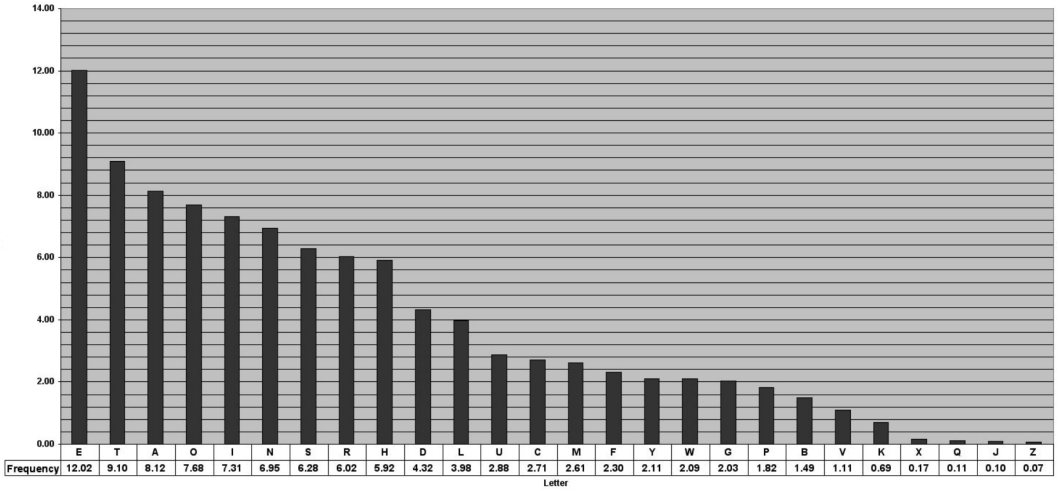

FIGURE 3. Existing Frequency Table for all alphabets

So, from the definition of perfect secrecy, the elaborated algorithm is secured.

Method 2: When all the alphabets follow the existing frequency of occurrence: We have the existing frequency of occurrence of alphabets as follows: [33]

Now, let us take the ciphertext B. How many ways we can have B as a ciphertext from the spaces K and M?

There can be 26 ways of getting it: $\{B+0, A+1, Z+2, Y+3, \ldots, D+24, C+25\}$, where $K = \{0,1,2,\ldots,24,25\}$ and $M = \{A,B,C,\ldots,Y,Z\}$.

$$\Pr[M = B \wedge K = 0] = \Pr[M = B] \cdot \Pr[K = 0] = (0.0149) \cdot \frac{1}{26}$$
$$\Pr[M = A \wedge K = 1] = \Pr[M = A] \cdot \Pr[K = 1] = (0.0812) \cdot \frac{1}{26}$$
$$\Pr[M = C \wedge K = 25] = \Pr[M = C] \cdot \Pr[K = 25] = (0.0271) \cdot \frac{1}{26}$$

Therefore,

$$\begin{aligned}
&\Pr[C = B] \\
&= \Pr[M = B \wedge K = 0] + \Pr[M = A \wedge K = 1] \\
&\quad + \Pr[M = Z \wedge K = 2] + \ldots + \Pr[M = C \wedge K = 25] \\
&= (0.0149 + 0.0812 + 0.0007 + 0.0211 + 0.0017 + 0.0020 + 0.0196 + 0.0111 + 0.1086 + 0.0911 \\
&\quad + 0.0268 + 0.0602 + 0.0011 + 0.0182 + 0.0768 + 0.0695 + 0.0261 + 0.0398 + 0.0098 + 0.0001
\end{aligned}$$

$$+0.0731+0.0592+0.0003+0.0230+0.1202+0.0432+0.0271)\cdot\frac{1}{26}=1\cdot\frac{1}{26}=\frac{1}{26}$$

[As, sum of all the probability frequencies of the alphabets is 1.]

Now, we can also calculate the conditional probability using the Bayes' Theorem. And we will get:

$$\Pr[M=A\mid C=B]=\frac{0.0812\cdot\frac{1}{26}}{\frac{1}{26}}=\Pr[M=A]$$

So, from the definition of perfect secrecy, the elaborated algorithm is secured.

7. Conclusion

Cryptography is the study of Encryption and Decryption where a given text is modified in such a manner that it's not accessible by anyone except for the sender and the receiver. Vigenere Cipher is a cyclic shift cipher that is commonly defined with a 26×26 order table that encrypts only the 26 alphabets where as, this proposed algorithm in this paper deals with all the 256 existing characters having ASCII values. For this purpose the constructed 256×256 order Vigenere table also broadens the way of confidentiality and security. The properties of the XOR Gate and the Bipartite graph ensures more security and authenticity. The objective of this research work is fulfilled as we can encrypt any text with all the existing characters including uppercase, lower case alphabets, punctuation, special characters, space etc. with high security and confidentiality as discussed in the previous section. Three main kinds of attacks can be prevented, a wide range of characters can be encrypted and any length of text can be secured with this proposed algorithm. The key string shared between the sender and the receiver works as a pivotal security enhancement. The current research adds to the expanding area of mathematics and information security by providing an authoritative basis for Cryptographic mechanisms with a broad spectrum of applications.

Conflict of Interests

The authors declare that there is no conflict of interests.

References

[1] S. Agarwal, Prime weighted graph in cryptographic system for secure communication, Int. J. Pure Appl. Math. 105 (2015), 325–338.

[2] S. Agarwal, A novel corona graph based proof-of-work algorithm for public blockchains, Int. J. Intell. Syst. Appl. Eng. 12 (2024), 450–455.

[3] P. Amudha, A.C. Charles Sagayaraj, A.C. Shantha Sheela, An application of graph theory in cryptography, Int. J. Pure Appl. Math. 119 (2018), 375–383.

[4] H. Anwar, Z. Shams, Algorithms of encryption using graph theory, Math. Sci. Appl. 2 (2023), 73–85.

[5] B. Charles, Cryptosystem using double vertex graph, Indian J. Sci. Technol. 13 (2020), 4483–4489.

[6] B. Charles, P. Venugopal, B. Praba, Block encryption and decryption of a sentence using decomposition of the Turan graph, J. Math. 2023 (2023), Article ID 7588535, 9 pp.

[7] V.M. Chandrasekaran, B. Praba, A. Manimaran, G. Kailash, Data transfer using complete bipartite graph, IOP Conf. Ser. Mater. Sci. Eng. 263 (2017), 042120.

[8] N. Deo, Graph theory with applications to engineering and computer science, Prentice Hall, 2010.

[9] U. Dixit, Cryptography: a graph theory approach, Int. J. Adv. Res. Sci. Eng. 6 (2017).

[10] W. Etaiwi, Encryption algorithm using graph theory, J. Sci. Res. Rep. 3 (2014), 2519–2527.

[11] D. Gurjar, A. Krishnaa, Labeled paths in cryptography, Adv. Math. Res. (2021).

[12] H.R. Medini, S. Dsouza, D.C. Nayak, P. Bhat, Encoding and decoding of messages using graph labeling and complement of a graph, Glob. Stoch. Anal. 11 (2024).

[13] B. Joseph, B. Thomas, Malaya J. Mat. 9 (2021), 470–473.

[14] R. Kareem, F. Abdullatif, S. Mohson, The corona graph for symmetric encryption schemes, Proc. 5th Int. Conf. Electr. Comput. Commun. Technol. (2023), 1–5.

[15] J. Katz, Y. Lindell, Introduction to modern cryptography, CRC Press, 2025.

[16] P. Kedia, S. Agarwal, Encryption using Venn diagrams and graph, Int. J. Adv. Comput. Technol. 4 (2015), 94–99.

[17] R. Kuppan, L. Shobana, I.N. Cangl, Encrypting and decrypting algorithms using strong face graph of a tree, Int. J. Comput. Math. Comput. Syst. Theory 5 (2020), 225–233.

[18] B. Ni, R. Qazi, S. Rehman, G. Farid, Some graph-based encryption schemes, J. Math. 2021 (2021), 1–8.

[19] A. Paszkiewicz, A. Górska, K. Górski, Z. Kotulski, K. Kulesza, J. Szczepanski, Proposals of graph-based ciphers: theory and implementations, 2001.

[20] S. Perera, S. Wijesiri, Encryption and decryption algorithms in symmetric key cryptography using graph theory, Psychology Education 58 (2021), 3420–3427.

[21] P.L.K. Priyadarsini, A survey on some applications of graph theory in cryptography, J. Discrete Math. Sci. Cryptogr. 18 (2015), 209–217.

[22] A. Razaq, G. Alhamzi, S. Abbas, M. Ahmad, A. Razzaque, Secure communication through reliable S-box design: a proposed approach using coset graphs and matrix operations, Heliyon 9 (2023), e15902.

[23] P.S.K. Reddy, P.S. Hemavathi, Generalization of bipartite graphs, J. Discrete Math. Sci. Cryptogr. 23 (2020), 787–793.

[24] K. Roy, S. Mazumder, Encryption scheme of modified Caesar cipher using bipartite graph, J. Inform. Optim. Sci. 46 (2025), 1645–1658.

[25] D. Sensarma, S. Sarma, Application of graphs in security, Int. J. Innov. Technol. Explor. Eng. 8 (2019), 2273–2279.

[26] C.R. Sharmila, S. Meenakshi, A fuzzy graph theory approach to symmetric key cryptography, J. Propuls. Technol. 45 (2024), 467–477.

[27] W. Stallings, Cryptography and network security, 7th ed., Pearson Education Pvt. Ltd., 2017.

[28] S.A. Swadi, A.A. Najim, The generalized k-connectivity of equally complete bipartite graphs and their line graphs, J. Discrete Math. Sci. Cryptogr. 27 (2024), 1567–1573.

[29] N. Tokareva, Connections between graph theory and cryptography, G2C2: Graphs and Groups, Cycles and Coverings, Novosibirsk, Russia, 2014.

[30] V. Ustimenko, On graph-based cryptography and symbolic computations, Serdica J. Comput. 1 (2007), 131–156.

[31] M. Yamuna, M. Gogia, A. Sikka, Md.J.H. Khan, Encryption using graph theory and linear algebra, Int. J. Comput. Appl. 5 (2012).

[32] M. Yamuna, K. Karthika, Data transfer using bipartite graphs, Int. J. Adv. Res. Sci. Eng. 4 (2015).

[33] Frequency table, available at https://pi.math.cornell.edu/~mec/2003-2004/cryptography/subs/frequencies.html.