Some Graph Based Encryption Techniques

Dhanvanth Narayan H, Surekha Ravishankar Bhat, Ravishankar Bhat and Smitha Ganesh Bhat*

Abstract—**In today's fast-evolving technological environment, ensuring confidentiality is of utmost importance. Cryptography stands as a critical discipline in safeguarding information from unauthorized access. It employs various encryption algorithms to secure data effectively. As digital threats evolve, there's a growing demand for unconventional encryption methods to counter traditional cyber-attacks. This paper introduces innovative encryption algorithms leveraging special graphs and public key cryptography techniques, enhancing security through modular arithmetic properties and enabling more robust communication safeguards.**

A partition $V_1, V_2, \dots, V_k$ of the vertex set V is called a chromatic partition of G if each V_i, $1 \leq i \leq k$ is an independent set of G. The minimum order of a chromatic partition of G is called chromatic number $\chi(G)$. A chromatic partition of G is called an ordered partition if $|V_1| = \beta_0$ and $|V_i| = \beta_0(V - \cup_{j=1}^{i} V_j)$. The order of a minimum ordered chromatic partition of G is called ordered chromatic number $\chi_1(G)$. It is immediate that $\chi_1(G) \geq \chi(G)$. In this paper we extend Nordhaus Gaddum results to ordered chromatic number.

Index Terms—**Bipartite graphs, Corona of two graphs, Star Graphs, Encryption, Decryption.**

I. Introduction

ALGORITHMS in graphs are methods used to solve various problems related to graph theory, which involves the study of graphs as mathematical structures. A graph consists of vertices (or nodes) and edges (or links) that connect pairs of vertices. Graph algorithms are fundamental in computer science and are used in many practical applications, such as network design, social network analysis, and biological data modeling.

In the past, secure communications were primarily used by military personnel to ensure the safe transfer of messages. However, in the modern world, the widespread use of the internet, mobile phones, and computer technology has made every individual concerned about the security of their personal data and information. As technology for data security advances, new methods to compromise confidential communications also emerge.

Cryptographic algorithms are the cornerstone of modern secure communication and data protection. These algorithms are mathematical constructs designed to encrypt and decrypt data, ensuring confidentiality, integrity, authentication, and non-repudiation in digital transactions. At its core, a cryptographic algorithm involves a set of procedures and rules for manipulating data to make it unintelligible to anyone without the appropriate decryption key.

Cryptographic algorithms play a vital role in various applications, including secure communication over the internet, digital signatures, secure e-commerce transactions, and authentication protocols. Their effectiveness depends not only on the complexity of the algorithm itself but also on the strength of the keys used and the implementation of the cryptographic protocols. As technology evolves, cryptographic algorithms continue to adapt to meet the growing challenges of securing digital information in an increasingly interconnected world.

Graphs can be used to design different encryption algorithms. The interaction between graph theory and cryptography is fascinating, with recent developments showing growing interest in using graphs to propose new methodologies in various areas of cryptography. In this context, new algorithms have been defined using existing public key cryptographic methods and graph structures like bipartite graphs, star graphs, and the corona of two graphs. The proposed algorithms can securely send and receive messages of any length using these graphical structures and some algebraic properties.

Surekha et al. [14], [15], [16], [17] and Tana et al. [18] have conducted a comprehensive investigation into the characteristics of cliques in graph structures. This implies a detailed study focusing on understanding various properties and behaviors of cliques within graph theory. Isabel Cristina Lopes et al. [6] have also explored the topic of cliques in graph structures, indicating another independent study on this subject.

II. Secure Data Transfer using Bipartite Graphs

In this section, we introduce an encryption algorithm aimed at ensuring secure and confidential communication between two entities. This algorithm leverages bipartite graphs, modular arithmetic, and the properties of co-prime numbers. We begin by generating both public and private keys.

- Select two random prime number i.e p and q.
- Multiply p and q, i.e n.
- Find the Euler phi function (ϕ) for n.
- $\phi(n) = (p-1) * (q-1)$.
- Choose a random number e that is coprime to $\phi(n)$.
- compute modular inverse d of e modulo $\phi(n)$ that is $(d * e)\ mod\ \phi(n) = 1$.
- So, (n, e) is the public key and (n, d) is the private key.

Using the above algorithm, both public and private keys have been generated. Next, we proceed with the encryption process, as outlined in the formulated algorithm below.

- To encrypt a message m using the public key (n,e), we first apply the chosen encoding method to convert

Manuscript received April 15, 2024; revised October 4, 2024.

Dhanvanth Narayan H is Postgraduate student at Department of Mathematics, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India-576104 (e-mail: h.dhanvanth@gmail.com)

Surekha Ravishankar Bhat is Professor at Department of Mathematics, Milagres College, Kallianpur, Udupi, Karnataka, India-574111, (e-mail: surekharbhat@gmail.com)

Ravishankar Bhat is Former Professor at Department of Mathematics, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India-576104 (e-mail: ravishankar.bhats@gmail.com)

Smitha Ganesh Bhat is Assistant Professor-Senior Scale at Department of Mathematics, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India-576104 (corresponding author to provide phone: 9844061970, e-mail: smitha.holla@manipal.edu)

the message into numerical form. Then, we use the encryption function $C(m) = m^{e}mod\ n$.

- Constructing a path graph using $C(m)$.
- Seperate the graph labels as V_1 and V_2.
- Form a bipartite graph with this vertex set G(V_1,V_2).
- Assign some random numbers to the edges as weight in increasing order.

The receiver then obtains the bipartite graphical structure and the private key, which are used for the decryption process according to the algorithm detailed below.

- Arrange the weights of edges in increasing order.
- Arrange the vertices of edges as ordered pairs with respect to the weights.
- Construct a path graph using the ordered pairs.
- Decrypt the message C using the private key (n, d) by applying the decryption function $D(C) = C^{d}mod\ n$.
- The necessary alphabets are retrieved using the encoding table.

A	B	C	D	E	F	G	H	I	J	K	L
1	2	3	4	5	6	7	8	9	10	11	12
M	N	O	P	Q	R	S	T	U	V	W	X
13	14	15	16	17	18	19	20	21	22	23	24
Y	Z										
25	26										

TABLE I: Numerical Representation

Let's consider an example to illustrate this algorithm.

To demonstrate the described technique, let's take an example that follows the outlined steps. We'll use the word **M I T** and generate public and private keys to encrypt and decrypt the message.

Generating Public and Private key's:

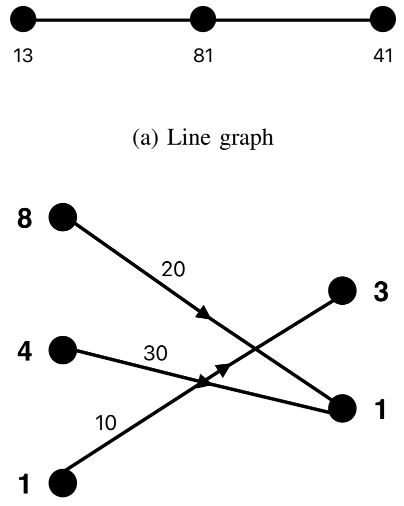
- Let us take two prime numbers $p = 7$ and $q = 13$.
- Computing the product of p and q, $n = p * q$, $n = 91$.
- Computing the Euler's totient/phi function of n, $\phi(n) = (p - 1) * (q - 1) = 6 * 12 = 72$.
- Let us take $e = 23$, where 23 is co-prime to 72.
- Computing the private key d. The multiplicative inverse of 23 modulo 72 is 47. Hence $d = 47$.
- The public key (n, e) is (91,23) and the private key (n, d) is (91,47).

Encryption using Public key(n,e):

- The message to be encrypted is **MIT**.
- Representing the message using numerical representation table i.e. m_1= M = 13, m_2= I = 9, m_3 = T = 20,
- Applying the encryption function:
  - C(m_1)= 13^{23} mod 91 = 13.
  - C(m_2)= 9^{23} mod 91 = 81.
  - C(m_3)= 20^{23} mod 91 = 41.
- Constructing a path graph(Fig. (1a)) using C(m_1), C(m_2) and C(m_3).
- Seperating the graph lables as V_1 = (1,8,4) and V_2 = (3,1).
- Forming the Bipartite graph(Fig. (1b)).

Decryption using Private key(n.d):

- Arranging the weights in increasing order, 10, 20, 30.
- Arranging the vertices of edges as ordered pairs with respect to the weights, (1,3) (8,1) (4,1).
- Constructing a path graph using the ordered pairs, (Fig. (1a)).
- Decrypting the message,
  - D(C_1)= 13^{47} mod 91 = 13.
  - D(C_2)= 81^{47} mod 91 = 9.
  - D(C_3)= 41^{47} mod 91 = 20.
- Decoding D(C_1), D(C_2) and D(C_3) using the Numerical representation table, i.e.$13 = M,\ 9 = I,\ 20 = T$.
- Hence the decrypted message using the Bipartite graph (Fig. (1b)) is **MIT**.



(a) Line graph

(b) Bipartite Graph

Fig. 1: Secure Data Transfer using Bipartite Graph

III. Secure Data Transfer Using Corona of two Graphs

In this section, we introduce an encryption algorithm intended to enable secure and confidential communication between two entities. This algorithm utilizes bipartite graphs, modular arithmetic, and the properties of co-prime numbers. We begin by generating both public and private keys.

- Select two random prime numbers, i.e p and q.
- Multiply p and q, i.e n.
- Find the Euler phi function (ϕ) for n.
- $\phi(n) = (p - 1) * (q - 1)$.
- Choose a random number e that is coprime to $\phi(n)$.
- Compute modular inverse d of e modulo $\phi(n)$ that is $(d * e)\ mod\ \phi(n) = 1$.
- So, (n, e) is the public key and (n, d) is the private key.

The algorithm used for generating both public and private keys operates similarly to the bipartite graph approach.

- Number the alphabets using Table I
- Next, to encrypt the message m using the public key (n, e), we apply the encryption formula, a_i=$m^{e}mod\ n$.
- Select random integers b_i in increasing order where $gcd(a_i,\ b_i) = 1$.
- Consider a corona graph $C_x \odot K_1$ with $2x$ vertices and allot weights $b_1, b_2, b_3, \ldots, b_n$ to the vertices, adjacent to pendent vertices randomly.

- Find the inverse of $a_i \pmod{b_i}$ for all i and denote them by c_i, that is $c_i = (a_i)^{-1} \pmod{b_i}\ \forall\ i$.
- Give the numeric values of $c_1, c_2, c_3, \ldots, c_n$ to pendent vertices. Send this corona graph $C_x \odot K_1$ to the receiver.

Then, the recipient receives the graphical structure and the private key, which are used for the decryption process following the algorithm outlined below.

- To decrypt, arrange all main vertices in ascending order using the labeled graph.
- Calculate the inverses of the weights of pendent vertices c_i modulo their adjacent vertices b_i, denoted as a_i for each i.
- Decrypt the corresponding values using the decryption formula with the private key (n, d) .
- $m = a_i^d mod\ n.$
- The necessary letters are retrieved from the encoding table.

Let's explore an example of this algorithm.

To illustrate the described technique, we'll use an example to demonstrate each step. Let's take the word **MAHE**, generate public and private keys, and proceed to encrypt and decrypt using these keys.

Generating Public and Private key's:

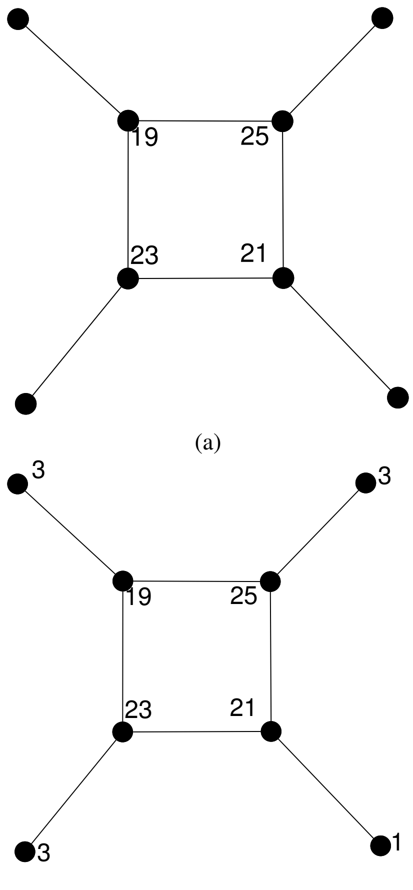
- Let us take two prime numbers $p = 3$ and $q = 7$.
- Computing the product of p and q, $n = p * q$, $n = 21$.
- Computing the Euler's totient/phi function of n, $\phi(n) = (p - 1) * (q - 1) = 2 * 6 = 12$.
- Let us take $e = 17$, where 17 is co-prime to 12.
- Computing the private key d. The multiplicative inverse of 17 modulo 12 is 5. Hence $d = 5$.
- The public key (n, e) is $(21, 17)$ and the private key (n, d) is $(21, 5)$.

Encryption using Public key (n, e):

- The message to be encrypted is ***MAHE***
- Representing the Message using Table I, i.e., m_1= M = 13, $m_2 = A = 1$, $m_3 = H = 8$, $m_4\ = E = 5$.
- Applying the encryption function:
  - a_1= 13^{17} mod 21 = 13.
  - a_2= 1^{17} mod 21 = 1.
  - a_3= 8^{17} mod 21 = 8.
  - a_4= 5^{17} mod 21 = 17.
- Selecting random integers b_i in increasing order, i.e., $19 < 21 < 23 < 25$.
- Build the corona graph $C_4 \odot K_1$ and assign random values to b_i on the main vertices, illustrated in Fig. 2a.
- Calculating the inverse of $a_i \pmod{b_i}$ for all i.
  - c_1= 13^{-1} mod 19 = 3.
  - c_2= 1^{-1} mod 21 = 1.
  - c_3= 8^{-1} mod 23 = 3.
  - c_4= 17^{-1} mod 25 = 3.
- Assign these inverse values to the adjacent pendant vertices in Fig. (2a), depicted as shown in Fig. (2a).
- Send this labelled graph to the reciever Fig. (2a).

Decryption using Private key $(n.d)$:

- The recipient, after receiving that labeled graph, arranges the main vertices in ascending order such that $19 < 21 < 23 < 25$ and consider these values as b_i for all $i = 4$.
- Taking inverse of corresponding pendent vertices with respect to the value of each b_i
  - a_1= 3^{-1} mod 19 = 13.
  - a_2= 1^{-1} mod 21 = 1.
  - a_3= 3^{-1} mod 23 = 8.
  - a_4= 3^{-1} mod 25 = 17.
- Decrypting the message
  - m_1= 13^5 mod 21 = 13.
  - m_2= 1^5 mod 21 = 1.
  - m_3= 8^5 mod 21 = 8.
  - m_4= 17^5 mod 21 = 5.
- Decoding DC_1, DC_2, C_3 and m_4 using Table I, i.e. 13 = M, 1 = A, 8 = H, 5 = E.
- Hence the decrypted message using Corona graph $C_4 \odot K_1$ (Fig. (2b)) is **MAHE**



(a)

(b)

Fig. 2: Secure Data Transfer using Corona of two Grapha

IV. Secure Data Transfer Using Star graphs

In this section, we introduce an encryption algorithm aimed at enabling secure and confidential communication between two entities. This algorithm utilizes star graphs, modular arithmetic, and properties of co-prime numbers. We begin by generating both public and private keys to facilitate the encryption process.

- Select two random prime number i.e p and q.
- Multiply p and q i.e n
- Find the Euler phi function(ϕ) for n.
- $\phi(n) = (p - 1) * (q - 1)$.
- Choose a random number e that is coprime to $\phi(n)$.
- Compute modular inverse d of e modulo $\phi(n)$ that is $(d * e)\ mod\ \phi(n) = 1$.

- So, (n, e) is the public key and (n, d) is the private key.

The algorithm used for generating both public and private keys follows a similar approach as that used for bipartite graphs.

- Number the Alphabets using Table I.
- Then to encrypt the message m using the public key (n, e) we apply the encryption formula $C(m) = m^e mod\ n$.
- Take a star graph $S_{(x+1)} = (K_1 \odot \overline{K_x})$, such that the number of corner vertices is equal to the length of message.
- Label the edges $e_1, e_2, e_3, \ldots, e_n$.
- Assign labels to each vertex using the values obtained from the public key cryptographic method.
- Give the weights $w_1, w_2, w_3, \ldots, w_n$ to each edge $e_1, e_2, e_3, \ldots, e_n$
- Subtract increasing power of 10 from each vertex label.
- $V_1 - 10^1$, $V_2 - 10^2$, ... , $V_n - 10^n$, where the resulting values becomes the weights of the corresponding edges.
- The final graph is the star graph with edge's weights.

Next, the recipient receives the graphical structure and the private key, which are utilized for decryption using the specified algorithm below.

- To decrypt, organize the edge weights in ascending order.
- Sum up the increasing powers of 10 accordingly.
- Apply the decryption formula $D(C) = C^d mod\ n$ to the resulting number.
- Decode the characters from the encoding table to retrieve the desired text.

Here's an example illustrating this algorithm.

To demonstrate the described technique, we'll use an example to illustrate each step. Let's take the word **Manipal**, generate public and private keys, and proceed to encrypt and decrypt using these keys.

Generating Public and Private key's:

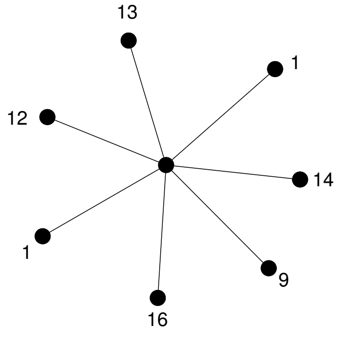
- Let us take two prime numbers $p = 3$ and $q = 11$.
- Computing the product of p and q, $n = p * q$, $n = 33$.
- Computing the Euler's totient/phi function of n, $\phi(n) = (p - 1) * (q - 1) = 2 * 10 = 20$.
- Let us take $e = 13$, where 13 is co-prime to 20.
- Computing the private key d. The multiplicative inverse of 13 modulo 20 is 17. Hence $d = 17$.
- The public key (n, e) is $(33, 13)$ and the private key (n, d) is $(33, 17)$.

Encryption using Public key (n, e):

- The Message to be encrypted is ***Manipal***
- Representing the message using Table 1 table i.e. m_1= M = 13, m_2= a = 1, m_3 = n = 14, m_4 = i = 9, m_5 = p = 16, m_6 = a = 1, m_7 = l = 12.
- Applying the encryption function:
  - C(m_1)= 13^{13} mod 33 = 19.
  - C(m_2)= 1^{13} mod 33 = 1.
  - C(m_3)= 14^{13} mod 33 = 5.
  - C(m_4)= 9^{13} mod 33 = 3.
  - C(m_5)= 16^{13} mod 33 = 4.
  - C(m_6)= 1^{13} mod 33 = 1.
  - C(m_7)= 12^{13} mod 33 =12.
- Taking a star graph $S_{(7+1)} = (K_1 \odot \overline{K_7})$ Fig. (3a) in such a way that edges are labeled as $e_1, e_2, e_3, \ldots, e_n$.
- Assigning labels to each vertex using values obtained from the public key cryptographic method, the resulting graph appears as depicted in Fig. (4a)
- Afterwards, assign weights w_i, to the edges of the vertices, where for all $i \in \{1, 2, 3, 4, 5, 6, 7\}$ with the weights ordered as follows: $w_1(19) < w_2(1) < w_3(5) < w_4(3) < w_5(4) < w_6(1) < w_7(12)$.
- The weights are derived by subtracting the increasing powers of 10 from each corresponding numeric value adjacent in Fig. (4a):
  - weight of edge $e_1 = w_1 = 19 - 10 = 9$,
  - weight of edge $e_2 = w_2 = 1 - 10^2 = -99$,
  - weight of edge $e_3 = w_3 = 5 - 10^3 = -995$,
  - weight of edge $e_4 = w_4 = 3 - 10^4 = -9997$,
  - weight of edge $e_5 = w_5 = 4 - 10^5 = -99996$,
  - weight of edge $e_6 = w_6 = 1 - 10^6 = -999999$,
  - weight of edge $e_7 = w_7 = 12 - 10^7 = -9999988$,
- The resulting star graph is depicted in Fig. (5a). This graph, labeled accordingly, is then transmitted to the second authority. Now, let's outline the decryption process. Initially, the recipient receives the labeled graph, as illustrated in Fig. (5a).

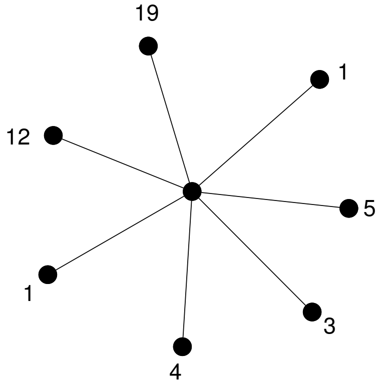
Decryption using Private key (n, d):

- The first step involves arranging the edge weights in ascending order based on their modulo values. i. e. $|9| < | - 99| < | - 995| < | - 9997| < | - 99996| < | - 999999| < | - 9999988|$
- Add the increasing powers of 10 to each adjacent value so that $|9 + 10| < | - 99 + 10^2| < | - 995 + 10^3| < | - 9997 + 10^4| < | - 99996 + 10^5| < | - 999999 + 10^6| < | - 9999988 + 10^7|$
- Through this mod operation, we get the values: 19,1,5,3,4,1,12
- Applying the decryption formula
  - D(m_1)= 19^{17} mod 33 = 13
  - D(m_2)= 1^{17} mod 33 = 1
  - D(m_3)= 5^{17} mod 33 =14
  - D(m_4)= 3^{17} mod 33 = 9
  - D(m_5)= 4^{17} mod 33 = 16
  - D(m_6)= 1^{17} mod 33 = 1
  - D(m_7)= 12^{17} mod 33 = 12
- Finally, we get the values 13,1,14,9,16,1,12, Through the encoding table, we get their respective letters as ***M a n i p a l***. Get the required hidden text.



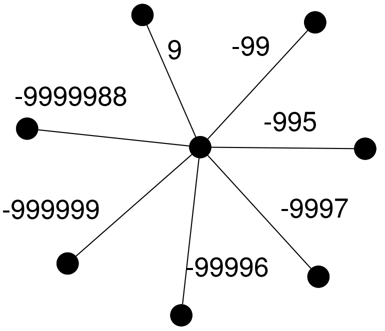
(a)

Fig. 3: Secure Data Transfer using Star Graphs



(a)

Fig. 4: Secure Data Transfer using Star Graphs



(a)

Fig. 5: Secure Data Transfer using Star Graphs

Painting the vertices of a graph G is called a colouring of G. A colouring is called a proper colouring of G if no two adjacent vertices receive the same colour. The minimum number of colours required to colour all the vertices of a graph G is called the chromatic number $\chi(G)$ of G.

A set D is said to be independent if no two vertices D are adjacent. The maximum order of an independent set is called independence number $\beta_0(G)$.

We can also define the chromatic number in another way as follows. Any proper colouring of G partition the vertex set into independent sets. Thus a partition $V_1, V_2, \dots, V_k$ of the vertex set V is called a chromatic partition of G if each V_i, $1 \leq i \leq k$ is an independent set of G. Then the minimum order of a chromatic partition of G is called chromatic number $\chi(G)$.

In this partition of vertex set in to independent sets, it is not necessary that the partition contain a maximum independent set. This fact made us to define ordered chromatic partition and ordered chromatic number as follows.

V. Ordered Chromatic Number

A chromatic partition of G is called an ordered partition if $|V_1| = \beta_0$ and $|V_i| = \beta_0(V - \cup_{j=1}^{i} V_j)$, $i = 2, 3, \dots, k$. The order of a minimum ordered chromatic partition of G is called ordered chromatic number $\chi_1(G)$. It is immediate that $\chi_1(G) \geq \chi(G)$.

Example V.1. *The minimum chromatic partition of any cycle C_n and any complete graph K_n are examples of chromatically ordered partitions.*

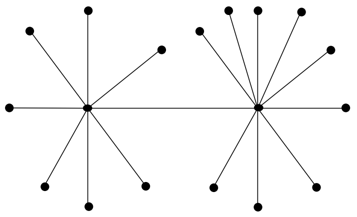


Fig. 6: Double Star

The minimum chromatic partition of any double star $K_{1,n} * K_{1,m}$ is not chromatically ordered. Let v_1 and v_2 are two supports of a double star and S_1={$v_1 \cup$ set of all pendant vertices adjacent to v_2} and S_2={$v_2 \cup$ set of all pendant vertices adjacent to v_1}. Then S_1 and S_2 form a minimum chromatic partition of the double star which is not an ordered partition as $|S_1| \neq \beta_0(K_{1,n} * K_{1,m})$. Let S_3={set of all pendant vertices}, $S_4 = \{v_1\}$ $S_5 = \{v_2\}$. Then S_3, S_4 and S_5 is an ordered minimum chromatic partition of the double star. Therefore $\chi_1(K_{1,n} * K_{1,m}) = 3 > 2 = \chi(K_{1,n} * K_{1,m})$.

We quote another example.

Example V.2. *The graph G shown in the Fig 7 is an example of a graph which is not tree for which $\chi_1(G) = 3 > 2 = \chi(G)$.*

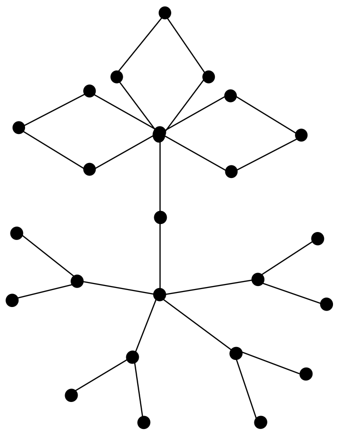


Fig. 7: A graph for which $\chi_1(G) \geq \chi(G)$.

Proposition V.1. *For any graph G,*

$$\begin{aligned} & \chi_1 \geq \chi \geq \bar{\beta}_0 \\ \textit{and} \quad & \bar{\chi}_1 \geq \bar{\chi} \geq \beta_0 \end{aligned}$$

Proof: By definition we have $\chi_1 \geq \chi$ and it is well known that $\chi(G) \geq \omega(G) = \bar{\beta}_0$. Thus we have $\chi_1 \geq \chi \geq \bar{\beta}_0$. Complimenting this inequality (that is applying the first inequality to $\bar{G}$) we get the second inequality. ■

A. Nordhaus - Gaddum Type Result

Bounds for sum and product of chromatic numbers of a graph and its compliment were developed by Nordhaus and Gaddum [10]

Proposition V.2. *[10] For any graph G*

$$2\sqrt{p} \leq \chi + \bar{\chi} \leq p + 1$$
$$p \leq \chi\bar{\chi} \leq (\frac{p+1}{2})^2$$

We now extend this inequality to sum and product of independence numbers of a graph and its compliment.

Proposition V.3. *For any graph G*

$$p\frac{(\chi_1 + \bar{\chi_1})}{\chi_1\bar{\chi}_1} \leq \beta_0 + \bar{\beta}_0 \leq \chi + \bar{\chi} \leq \chi_1 + \bar{\chi}_1 \leq p + 1$$
$$\frac{p^2}{\chi_1\bar{\chi}_1} \leq \beta_0\bar{\beta}_0 \leq \chi\bar{\chi} \leq \chi_1\bar{\chi}_1 \leq \left(\frac{p+1}{2}\right)^2$$

Proof: From Proposition V.1, it follows that

$$\beta_0 + \bar{\beta}_0 \leq \chi + \bar{\chi} \leq \chi_1 + \bar{\chi}_1$$

and

$$\beta_0\bar{\beta}_0 \leq \chi\bar{\chi} \leq \chi_1\bar{\chi}_1.$$

Let $\{V_1, V_2, \ldots, V_k\}$ denote a minimum ordered chromatic partition of G such that $\chi_1(G) = k$. Given $|V_1| = \beta_0$ and $\beta_0 \geq |V_i|$ for $2 \leq i \leq k$, it follows that

$$\chi_1\beta_0 \geq p,$$

leading to

$$\beta_0 \geq \frac{p}{\chi_1}.$$

By a similar argument, we find that

$$\bar{\beta}_0 \geq \frac{p}{\bar{\chi}_1}.$$

Thus, we have

$$\beta_0\bar{\beta}_0 \geq p,$$

establishing the lower bound for the second inequality.

Since the geometric mean of two positive numbers does not exceed their arithmetic mean, we obtain

$$\beta_0 + \bar{\beta}_0 \geq \frac{p}{\chi_1} + \frac{p}{\bar{\chi}_1} = p\left(\frac{\chi_1 + \bar{\chi}_1}{\chi_1\bar{\chi}_1}\right),$$

providing the lower bound for the second inequality.

To show that

$$\chi_1 + \bar{\chi}_1 \leq p + 1,$$

we apply induction on p. The base case holds for $p = 1$. Assume $\chi_1 + \bar{\chi}_1 \leq p$ is true for graphs with $p - 1$ vertices. Let H be a graph with p vertices and $\bar{H}$ its complement. Removing a vertex v from H gives graphs $G = H - v$ and $\bar{G} = \bar{H} - v$ with $p - 1$ vertices. We find that

$$\chi_1(H) \leq \chi_1(G) + 1$$

and

$$\bar{\chi}_1(H) \leq \bar{\chi}_1(G) + 1.$$

Case 1: If $\chi_1(H) < \chi_1(G) + 1$ or $\bar{\chi}_1(\bar{H}) < \bar{\chi}_1(\bar{G}) + 1$, then

$$\chi_1(H) + \bar{\chi}_1(H) \leq p + 2,$$

leading to

$$\chi_1(H) + \bar{\chi}_1(H) \leq p + 1.$$

Case 2: If $\chi_1(H) = \chi_1(G) + 1$ and $\bar{\chi}_1(\bar{H}) = \bar{\chi}_1(\bar{G}) + 1$, we find that the removal of v from H reduces the chromatic number, so

$$d \geq \chi_1(G)$$

and $d(v)$ in $\bar{H}$ is

$$p - d - 1 \geq \bar{\chi}_1(\bar{G}).$$

Hence,

$$\chi_1(G) + \bar{\chi}_1(\bar{G}) \leq p - 1,$$

leading to

$$\chi_1(H) + \bar{\chi}_1(H) \leq p.$$

In all cases, we conclude that

$$\chi_1(H) + \bar{\chi}_1(H) \leq p + 1.$$

Finally, since the geometric mean of two positive numbers does not exceed their arithmetic mean, we obtain

$$4\chi_1\bar{\chi}_1 \leq (\chi_1 + \bar{\chi}_1)^2 \leq \left(\frac{p+1}{2}\right)^2,$$

providing the desired upper bound for the second inequality. This result can also be extended to domination and independent domination numbers.

■

Corollary V.3.1. *For any graph G*

$$\gamma + \bar{\gamma} \leq i + \bar{i} \leq \beta_0 + \bar{\beta}_0 \leq \chi + \bar{\chi} \leq \chi_1 + \bar{\chi}_1 \leq p + 1$$
$$\gamma + \bar{\gamma} \leq i + \bar{i} \leq \beta_0\bar{\beta}_0 \leq \chi\bar{\chi} \leq \chi_1\bar{\chi}_1 \leq \left(\frac{p+1}{2}\right)^2$$

VI. Vertex Clique Domination

The partition number $\theta(G)$ of a graph, introduced by Berge [2], represents the minimum number of cliques needed to cover all the vertices. Choudam et al. [4] referred to it as the vertex clique covering number and explored its edge counterpart, the edge clique covering number $\theta_1(G)$, which is the least number of cliques required to cover all edges in the graph.

Building on these concepts, a new type of domination was defined: vertex clique domination. In this context, a vertex v is said to clique-dominate a clique l if v is connected to l. A subset D of vertices is termed a vertex clique dominating set (VCD-set) if every clique in the graph is dominated by at least one vertex from D. The vertex clique domination number, denoted as $\gamma_{vc}(G)$, indicates the size of the smallest VCD-set in the graph.

Conversely, a set F of cliques in $K(G)$ is defined as a clique vertex dominating set (CVD-set) if every vertex in G is dominated by at least one clique from F. The clique vertex domination number, $\gamma_{cv}(G)$, is the size of the smallest CVD-set in G.

Example VI.1. *For the graph G of Fig. 8, $\gamma_{vc} = 7$ and $\gamma_{cv} = 13$*

Note VI.1.

1) *For any triangle free graph G, $\gamma_{vc}(G) = \alpha_0(G)$.*
2) *We observe that*
   (i) *$n_0 = \gamma_{vc}$.*
   (ii) *If G has no isolates, then $\gamma_{ve} = n_0 = \gamma_{vc}$.*
   (iii) *$\gamma_{cv} = \theta_0$.*

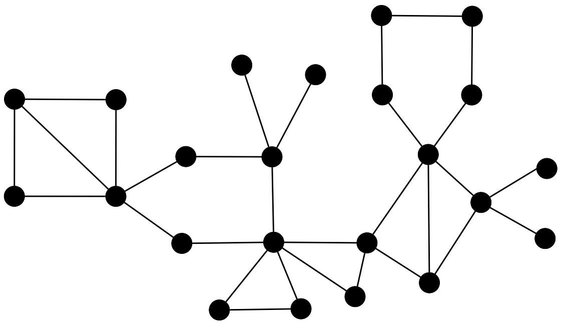


Fig. 8

E Sampathkumar and Prabha S Neeralagi [12] obtained the following bounds.

Proposition VI.2. *For any graph G,* $\gamma_{vc} = n_0 \leq p - \Delta$

R. S. Bhat et. al. [3] obtained the following bounds.

Proposition VI.3. *For any graph G,* $\dfrac{q}{\Delta_{ve}} \leq \gamma_{ve} = \gamma_{vc}$

Proposition VI.4. *For any graph G,* $\dfrac{p}{\Delta_{cv}} = \dfrac{p}{\omega} \leq \theta_0 = \gamma_{cv}$

Proof: We know that $\dfrac{p}{\beta_0} \leq \chi(G)$. But $\chi(\bar{G}) = \theta_0$ and $\beta_0(\bar{G}) = \omega$. Thus $\dfrac{p}{\beta_0(\bar{G})} \leq \chi(\bar{G})$ gives $\dfrac{p}{\omega} \leq \theta_0$. ■

We obtain some new bounds for neighborhood number using the new definitions.

Proposition VI.5. *For any graph G,* $\gamma_{vc} \leq p_c$*. Equality holds if G is a clique star.*

Proof: Since the set of all polycliqual vertices is a VCD set of G we have $\gamma_{vc} \leq p_c$. ■

Now we give vertex clique domination number of newly defined graphs.

Proposition VI.6.

1) *If G is a clique cycle with k cliques,* $\gamma_{vc}(G) = \left\lceil \dfrac{k}{2} \right\rceil$.
2) *If G is a clique path with k cliques,* $\gamma_{vc}(G) = \left\lceil \dfrac{k}{2} \right\rceil$.
3) *If G has a vertex of degree* $p - 1$, $\gamma_{vc}(G) = 1$.
4) *If G is a clique star with c cut-vertices,* $\gamma_{vc}(G) = c$.

Proposition VI.7. *Let G be any graph with k cliques and maximum VC degree* Δ_{vc}*, then*

$$\left\lceil \frac{k}{\Delta_{vc}} \right\rceil \leq \gamma_{vc} \tag{1}$$

Further the bound is sharp.

Proof: Since any polycliqual vertex can clique dominate atmost Δ_{vc} cliques, we need atleast $\left\lceil \dfrac{k}{\Delta_{vc}} \right\rceil$ vertices to clique dominate all the cliques of G. This yields the bound in (1).

The bound is sharp for any clique complete graph. ■

Example VI.2. *For the graph G of Figure 9, number of cliques* $k = 8$*, maximum vc-degree* $\Delta_{vc} = 3$ *and* $\gamma_{vc} = 3$*. Thus* $\left\lceil \dfrac{k}{\Delta_{vc}} \right\rceil = \left\lceil \dfrac{8}{3} \right\rceil = 3 = \gamma_{vc}$

Surekha et al. define clique-free sets in their work [16]. We present a result similar to Gallai's Theorem concerning the vertex clique domination number. To begin, we will discuss the following Proposition.

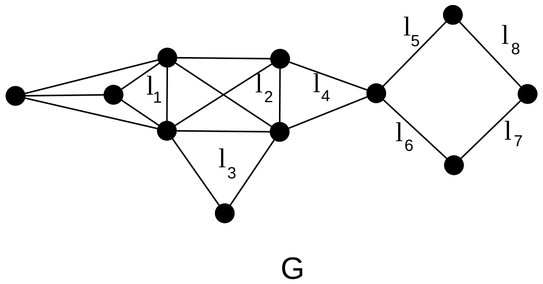


Fig. 9

Proposition VI.8. *Let G be a (p,q) graph and* $S \subseteq V$*, then S is a VCD set of G if and only if V-S is a clique free set of G.*

Proof: Let S be a vertex clique dominating (VCD) set of G. Since each clique in G is dominated by at least one vertex from S, it follows that at least one vertex from every clique must belong to S. Consequently, the set $V - S$ cannot form any cliques in G, which means $V - S$ is clique-free.

Conversely, suppose S is a clique-free set of G. If $V - S$ were not a VCD set, there would exist at least one clique $k \in G$ that is not dominated by any vertex in $V - S$. This implies that all vertices of k must be included in S. Therefore, $\langle S \rangle$ would contain the clique k, contradicting the assumption that S is a clique-free set of G.

■

Proposition VI.9. *For any graph G with p vertices,*

$$\gamma_{vc} + \beta_{vc} = p$$

Proof: Let S be a γ_{vc}-set of G. According to Proposition VI.8, $V - S$ is a clique-free set of G. Thus, we have $\beta_{vc} \geq |V - S| = p - \gamma_{vc}$, leading to $\gamma_{vc} + \beta_{vc} \geq p$. (i)

Conversely, let D be a β_{vc}-set of G. Again, by Proposition VI.8, $V - D$ is a VCD set of G, giving us $\gamma_{vc} \leq |V - D| = p - \beta_{vc}$ and therefore $\gamma_{vc} + \beta_{vc} \leq p$. (ii)

The result then follows from (i) and (ii).

■

REFERENCES

[1] Baizhu Ni, Rabiha Qazi, Shafiq Ur Rehman and Ghulam Farid, "Some Graph Based Encryption Schemes", Hindawi Journal of Mathematics, vol. 01, pp1-8, 2021.

[2] C. Berge, "Theory of Graphs and its Applications", Methuen, London, 1962.

[3] R. S. Bhat, S. S. Kamath and Surekha R. Bhat, "Strong/Weak Edge Vertex Mixed Domination Number of a Graph", IJMS, vol.11, pp433-444, 2012.

[4] S. A. Choudam, K. R. Parthasarathy and G. Ravindra, "Line-clique cover number of a graph", Proceedings of INSA, Vol. 41 Part A, No. 3, pp289-293, 1975

[5] Frank Harary, "Graph Theory", Addison Wesley, 1969

[6] Isabel Cristina Lopes, J.M. Valerio de Carvalho, "Minimization of Open Orders Using Interval Graphs", IAENG International Journal of Applied Mathematics, vol. 40, no.4, pp297-306, 2010

[7] Jonathan Katz and Yehuda Lindell, "Indroduction to Modern Cryptography", Second Edition, 2016.

[8] W. Mahmoud and A. Etaiwi, "Encryption algorithm using graph theory," Journal of Scientific Research and Reports, vol. 3, no. 19, pp. 2519–2527, 2014.

[9] Narsingh Deo, “Graph theory with applications to engineering and Computer science”, PHI, 1974

[10] E.A. Nordhaus and J.W.Gaddum, “On complimentary graphs”, American Mathematical Monthly, vol.63, pp175 -177, 1956

[11] Rolf Oppliger, “Cryptography 101 from theory to practice”, Artech House Publishers, 2021.

[12] E. Sampathkumar and Prabha S. Neeralagi, “The Neighbourhood Number of a Graph”, Indian J. pure and appl. Math. Vol. 16, pp126-132, 1985.

[13] R. Selvakumar and N. Gupta, “Fundamental circuits and cut- sets used in cryptography,” Journal of Discrete Mathematical Sciences and Cryptography, vol. 15, no. 4-5, pp287–301, 2012.

[14] Surekha Ravi shankar Bhat, Ravi shankar Bhat, Smitha Ganesh Bhat, and Sayinath Udupa Nagara Vinayaka, “A Counter Example for Neighbourhood Number Less Than Edge Covering Number Of a Graph”, IAENG International Journal of Applied Mathematics, vol. 52, no.2, pp500-506, 2022

[15] Surekha Ravishankar Bhat, Ravishankar Bhat, Smitha Ganesh Bhat, “Properties of n- Independent Sets and n-Complete Sets of a Graph”, IAENG International Journal of Computer Science, vol. 50, no. 4, pp1354-1358, 2023

[16] Surekha Ravishankar Bhat, Ravishankar Bhat, Smitha Ganesh Bhat, “Clique Free Number of a Graph”, Engineering Letters, vol.31, no.4, pp1832-1836, 2023

[17] Surekha Ravishankar Bhat, Ravishankar Bhat, Smitha Ganesh Bhat, “A Comprehensive Analysis of Total and Semi-Total Graphs”, Engineering Letters, vol.32, no.1, pp21-29, 2024

[18] Tana and Nobuo Funabiki, “A Proposal of Graph-based Blank Element Selection Algorithm for Java Programming Learning with Fill-in-Blank Problems”, Proceedings of the International MultiConference of Engineers and Computer Scientists 2015, vol. 1, pp448-453, 2015

[19] M. Yamuna and K. Karthika, “Data transfer using bipartite graphs,” International Journal of Advance Research in Science and Engineering, vol. 4, no. 02, pp128–131, 2015.