International Journal of Applied Mathematics

Volume 38 No. 4s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

CONSTRUCTION OF FIFTH-ROOT BIQUADRATE MEAN CORDIAL GRAPH AND ITS CODE-BASED VALIDATION.

[1]A. Netto Mertia, [2]M. Sudha

[1]Research scholar, Noorul Islam Centre for Higher Education, Kumaracoil
Tamil Nadu, India, 629180

[2]Assistant Professor, Noorul Islam Centre for Higher Education, Kumaracoil,
Tamil Nadu, India, 629180

Abstract

In this work, we introduce a new graph-based method called the fifth-root biquadrate mean cordial technique. A graph that satisfies the conditions of this technique is referred to as a fifth-root biquadrate mean cordial graph. We investigate the applicability of the proposed fifth root biquadrate mean cordial technique to structured graph classes, specifically the cycle graphs C_n, the Corona graph $P_m \odot C_{n,}$and the Corona graph $C_n \odot C_{m,} where\ m > 2$ and n is congruent to zero mod 6. Furthermore, the theoretical results are also obtained through computational methods.

Keywords: Graph labeling, Cycle Graph, Corona Graph, Triangular snake graph, Crown graph, and Python programming.

1. Introduction

In the middle of the 1960s, the first graph labeling paper was published. This was followed by approximately 3,147 investigations conducted using more than 200 graph labeling techniques, which motivated our work. A labelled graph is a type of graph where vertices or edges are assigned labels, such as numbers or symbols, to represent additional information or functions associated with them. To solve some mathematical models, the graph's label needs to be assigned. Graph labeling can be used to automatically detect routes in a network. The design of good types of codes, synch-set codes, missile guidance codes, and convolutional codes with optimal auto-correlation features are only a few of the many applications for which labelled graphs provide efficient mathematical models. Labelled graphs have also been used to solve radio astronomy problems, database management, and ambiguities in x-ray crystallographic analysis. They make it easier to find the best nonstandard encodings of integers. J.A. Bondy and U.S.R. Murty [3] present basic material of graph theory together with a wide variety of applications. Jingxiang Jin and Zhuojie Tu [4] delivered a survey on graph antimagic labeling. M. Ponraj et al. [6] introduced the concept of difference cordial labeling of graphs. Adarsh

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Kumar Handa et al. [1] investigated local distance antimagic labelings across various families of graphs, including paths, cycles, wheel graphs, friendship graphs, complete multipartite graphs, and certain special caterpillars. Arumugam.S et al. [2] have explored fundamental results concerning a new parameter known as the local antimagic chromatic number, which is derived from local antimagic labeling. Joseph A. Gallian has been collecting all of the published studies in one book [5]. Each labeling technique has a unique set of criteria and applications; the type of labels chosen will rely on the problem being investigated. Studying symmetry, balance, and other combinatorial aspects of a graph is made easier with the aid of graph labeling. Among the various labeling schemes we introduced, the fifth-root biquadrate mean cordial technique. In the present study, we investigate the fifth-root biquadrate mean cordial technique for particular graph classes, which provides new perspectives on how they behave structurally under this technique. These labeled graphs may also be used for various other tasks, depending on the viewer's perspective and background expertise. In addition, we utilized a Python program to compute our findings. For detailed study of various graph labeling, we refer [7, 8].

Definition: 2.1 Let $G = (V, E),$ be a graph, where each vertex $v \in V$ of the graph G is assigned by randomly selecting a value from the set {0, 1, 2}, ensuring that the total number of vertices labeled 0, 1, and 2 varies by no more than one. Define an edge labeling function $\Omega: E(G) \rightarrow \{0,1\}$ such that for each edge $e = uv \in E(G)$, the label assigned to e is given by:

$$\Omega(e) = \begin{cases} 1, & if \left\lfloor \sqrt[5]{\dfrac{\Omega(u)^4 + \Omega(v)^4}{2}} \right\rfloor = 1 \\ 0, & otherwise \end{cases}$$

where u and v are the values assigned to the vertices incident to edge e. This labeling technique is termed a fifth root biquadrate mean cordial technique if the condition$|e_\Omega\ (1) - e_\Omega(0)| \leq 1$ is satisfied, e_Ω (1)denotes the cardinality of the set of edges labeled with 1, ande_Ω(0 represents the count of edges assigned the label 0.A graph G that permits such a labeling technique is referred to as a fifth root biquadrate mean cordial graph.

Example 1: The crown graph $C_3 \odot K_1$ is a fifth-root-biquadrate mean cordial as shown below

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

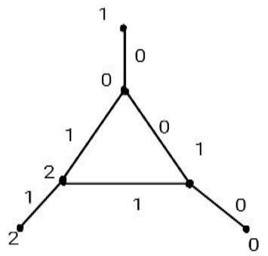

Fig. 1 Crown Graph $C_3 \odot K_1$

Theorem 2.2. The cycle graph C_n is fifth root biquadrate mean cordial, when $n \equiv 0(mod\ 6)$.

Proof : Let $G = (V, E)$depict the cycle graph C_n . Let $\{v_1, v_2, \ldots, v_n\}$ represent the vertices of cycle C_n and $\{e_1, e_2, \ldots, e_n\}$ symbolize its edges. Hence $|V(G)| = n$ and $|E(G)| = n$.

Vertex labeling $\psi: V(G) \to \{0,1,2\}$ is required to be defined as follows:

$\psi(v_s) = 2\ if\ s \equiv 1,2\ (mod\ 6)$; $\psi(v_s) = 1\ if\ s \equiv 3,5\ (mod\ 6)$;

$\psi(v_s) = 0$ if $s \equiv 4,0$ (mod 6);

The aforementioned labeling pattern fulfills the conditions.

Hence, the total number of vertices labeled with 0, 1, and 2 differ pairwise by at most one and $|e_\Omega(0) - e_\Omega(1)| \leq 1$.Therefore, we effectively concluded that the cycle graph, where $n \equiv 0(mod\ 6)$, is the fifth root biquadrate mean cordial from all of the above-mentioned situations.

2.3 Python Algorithm for Fifth Root Mean Cordial Technique of Cycle Graph C_n where $n \equiv 0(mod\ 6)$.

The Python code is given as

```
import networkx as nx
import matplotlib.pyplot as plt
import math
defcreate_graph(num_nodes):
  graph = nx.Graph()
  for i in range(num_nodes):
graph.add_edge(i, (i+1)%(num_nodes))
  return graph
deffunc(fu, fv):
val = (((fu**4) + (fv**4))/2)**(1/5)
  return math.floor(val)
defvisualize_graph(graph, num_nodes):
pos = nx.circular_layout(graph)
```

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
labeldict = {}
edgedict = {}
nodeVal = {}
  for i in range(num_nodes):
labeldict[i] = "v"+str(i+1)
    m = (i+1) % 6
    if m == 1 or m == 2:
nodeVal[i] = 2
elif m == 3 or m == 5:
nodeVal[i] = 1
    else:
nodeVal[i] = 0
x,y=pos[i]
plt.text(x,y+0.2,s=labeldict[i],bbox=dict(facecolor='red',alpha=0.5),horizontalalignment='cen
ter')
  for i in range(num_nodes):
edgedict[(i,(i+1)%num_nodes)] = func(nodeVal[i], nodeVal[(i+1)%num_nodes])
nx.draw(graph, pos, labels=nodeVal, with_labels=True, node_size=1000,
node_color='skyblue', font_size=10)
nx.draw_networkx_edge_labels(graph, pos, edge_labels=edgedict, font_color='red')
plt.show()
num_nodes = 6
graph = create_graph(num_nodes)
visualize_graph(graph, num_nodes)
```

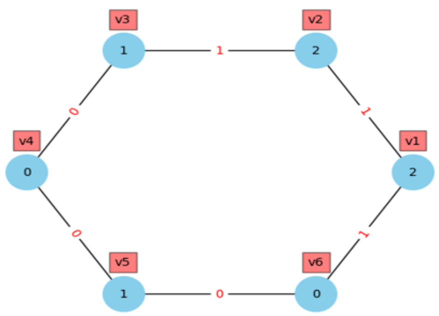

Figure 2.

Received: August 06, 2025

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The preceding figure depicts the output generated by the Python program for the fifth root biquadrate mean cordial of the cycle graph C_6.

Hence, the Python algorithm has been used to prove the theorem successfully.

Theorem 2.4 . The corona graph $P_m \odot C_n$ isfifth root biquadrate mean cordial, where $n \equiv 0(mod\ 6)$ and m is any positive integer.

Proof: Let's say that the vertices of P_m are represented by $\{v_{1_1}, v_{2_1}, v_{3_1}, \dots, v_{m_1}\}$and that the vertices of the i^{th}copy of C_n are denoted by $\{v_{i_1}, v_{i_2}, v_{i_3}, \dots \dots, v_{i_n}\}$where $n \equiv 0(mod\ 6)$. By definition, the corona graph of $P_m \odot C_n$is generated by joining one copy of P_m and m copies of C_n, and connecting the i^{th} vertex of $P_m$with the vertex v_{i_1} of the i^{th}copy of C_n, it gives the corona graph $P_m \odot C_n$. Let $G = (V, E)$ be the corona graph $P_m \odot C_n$. Hence $|V(G)| = mn$ and $|E(G)| = mn + m - 1$.

Construct a mapping $\psi: V(G) \to \{0,1,2\}$ as follows

Case (1): $i \equiv 1\ (mod\ 4)$

$\psi\left(v_{i_j}\right) = 2\ if\ \ j \equiv 1,2\ (mod\ 6); \psi\left(v_{i_j}\right) = 0\ if\ \ j \equiv 3,5\ (mod\ 6)$

$\psi\left(v_{i_j}\right) = 1\ if\ \ j \equiv 0,4\ (mod\ 6),\ where\ 1 \le i \le m\ and\ 1 \le j \le n$

Case (2):$i \equiv 0,2\ (mod\ 4)$

$\psi\left(v_{i_j}\right) = 2\ if\ \ j \equiv 0,5\ (mod\ 6); \psi\left(v_{i_j}\right) = 0\ if\ \ j \equiv 2,4\ (mod\ 6)$

$\psi\left(v_{i_j}\right) = 1\ if\ \ j \equiv 1,3\ (mod\ 6),\ where\ 1 \le i \le m\ and\ 1 \le j \le n$

Case (3):$\mathrm{i} \equiv 3\ (\mathrm{mod}\ 4)$

$\psi\left(v_{i_j}\right) = 2\ if\ \ j \equiv 4,5\ (mod\ 6); \psi\left(v_{i_j}\right) = 0\ if\ \ j \equiv 1,3\ (mod\ 6)$

$\psi\left(v_{i_j}\right) = 1\ if\ \ j \equiv 0,2\ (mod\ 6)$, where $1 \le i \le mand\ 1 \le j \le n$.

Furthermore, the total quantity of edges assigned by 0 and 1 and, overall, the quantity of vertices assigned by 0, 1, and 2 do not vary by more than one.

As an outcome, we have demonstrated that the corona graph $P_m \odot C_n$, $n \equiv 0(mod\ 6)$, is fifth root biquadrate mean cordial.

2.5 Python Algorithm for Fifth Root Mean Cordial Technique of Cycle Graph C_n Coronagraph $P_m \odot C_n$, where $n \equiv 0(mod\ 6)$ and m is any positive integer.

The Python code is given as

```
defcorona_graph(n,m):
```

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
  G = nx.cycle_graph(m)
  H = nx.path_graph(n)
  C = nx.rooted_product(H, G, 0)
pos = nx.nx_pydot.graphviz_layout(C)
nodeVal = {}
edgedict = {}
  for i in range(n):
    for j in range(m):
      if (i+1)%n == 1:
        if (j+1)%6 == 1 or (j+1)%6 == 2:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 3 or (j+1)%6 == 5:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
elif (i+1)%n == 0 or (i+1)%n == 2:
        if (j+1)%6 == 0 or (j+1)%6 == 5:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 2 or (j+1)%6 == 4:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
      else:
        if (j+1)%6 == 4 or (j+1)%6 == 5:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 1 or (j+1)%6 == 3:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
  for i in range(n):
    for j in range(m):
edgedict[((i,j),(i,(j+1)%m))] = func(nodeVal[(i,j)], nodeVal[(i,(j+1)%m)])
  for i in range(n-1):
```

Received: August 06, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
edgedict[((i,0),(i+1,0))] = func(nodeVal[(i,0)], nodeVal[(i+1,0)])
nx.draw(C, pos, labels=nodeVal, with_labels=True, node_size=100, node_color='skyblue',
font_size=6)
nx.draw_networkx_edge_labels(graph, pos, edge_labels=edgedict, font_color='red')
corona_graph(4, 12)
```

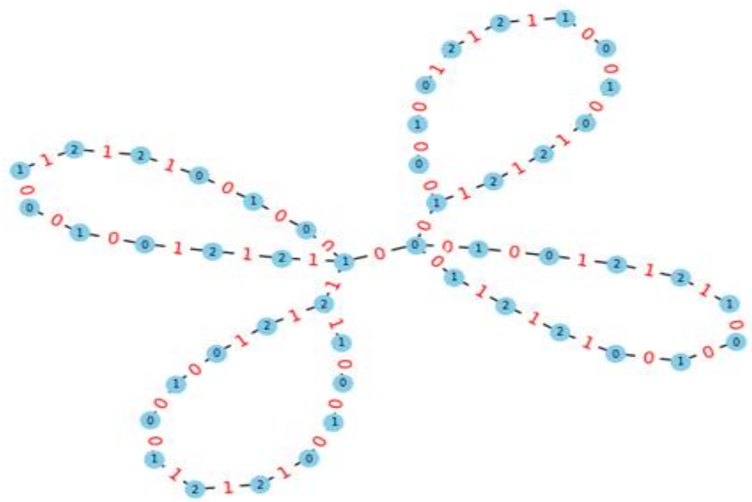

Figure 3

Figure shown above depicts the output generated by the Python program for the fifth root biquadtrate mean cordial of the coronagraph $P_m \odot C_n$, where $n \equiv 0(\mathrm{mod}\ 6)$ and mis any positive integer.

Hence, the Python algorithm has been used to prove theorem successfully.

Theorem 2.6. The coronagraph $C_n \odot C_m$ is the fifth root biquadrate mean cordial labeling, where m > 2 and n is congruent to zero mod 6

Proof: Let C_n and C_m have the sets of vertices $\{v_{1_1}, v_{2_1}, v_{3_1}.., v_{n_1}\}$ and $\{v_{i_1}, v_{i_2}, v_{i_3}, ..., v_{i_m}\}$ respectively. By the definition of corona graph taking one copy of $C_n$and n copies of C_m, connect the i^{th} vertex of vertex of $C_n(i = 1,2,3, ..., n)$with the vertex $v_{i_1}$of the i^{th}copy of C_m, provides the corona graph $C_n \odot C_m$. Let $G = (V, E)$denote the corona graph $C_n \odot C_m$. Hence $|V(G)| = nm$ and $|E(G)| = nm + n$.

Define vertex labeling $\psi: V(G) \to \{0,1,2\}$in the following ways:

Case 1: $i \equiv 1,3\ (mod\ 6)$

$\psi(v_{ij}) = 2\ if\ ;\ j \equiv 5,0\ (mod\ 6);\psi(v_{ij}) = 0\ if\ ;\ j \equiv 2,4\ (mod\ 6);$

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$\psi(v_{ij}) = 1\ if\ ;\ j \equiv 1{,}3\ (mod\ 6)$ *and* $1 \leq j \leq m$

Case 2: $i \equiv 2{,}4\ (mod\ 6)$

$\psi(v_{ij}) = 2\ if\ j \equiv 0{,}5\ (mod\ 6)$;$\psi(v_{ij}) = 1\ if\ j \equiv 2{,}4\ (mod\ 6)$

$\psi(v_{ij}) = 0\ if\ ;\ j \equiv 1{,}3\ (mod\ 6)$ *and* $1 \leq j \leq m$

Case 3: $i \equiv 0{,}5\ (mod\ 6)$

$\psi(v_{ij}) = 2\ if\ ;\ j \equiv 1{,}2\ (mod\ 6)$;$\psi(v_{ij}) = 0\ if\ ;\ j \equiv 3{,}5\ (mod\ 6)$

$\psi(v_{ij}) = 1\ if\ ;\ j \equiv 0{,}4\ (mod\ 6)$ *and* $1 \leq j \leq m$

Thus, there should be a maximum of 1 variation among the overall edges marked as 0 or 1 and the overall vertices assigned as 0, 1, and 2.

We have demonstrated that the corona graph $C_n \odot C_m,\ n \equiv 0(mod\ 6)$ where m and n are both multiples of 6 can be classified as a fifth root biquadrate mean cordial after analysing the results.

2.7: Python Algorithm for Fifth Root Mean Cordial Technique of Cycle Graph C_n Coronagraph $C_n \odot C_m$, where m > 2 and n is congruent to zero mod 6.

The Python code is given as

```
defcorona_graph_n(n):
  G = nx.cycle_graph(n)
  H = nx.cycle_graph(n)
  C = nx.rooted_product(H, G, 0)
pos = nx.nx_pydot.graphviz_layout(C)
nodeVal = {}
edgedict = {}
  for i in range(n):
    for j in range(n):
      if (i+1)%n == 1:
        if (j+1)%6 == 1 or (j+1)%6 == 2:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 3 or (j+1)%6 == 5:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
elif (i+1)%n == 0 or (i+1)%n == 2:
```

International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
        if (j+1)%6 == 0 or (j+1)%6 == 5:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 2 or (j+1)%6 == 4:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
      else:
        if (j+1)%6 == 4 or (j+1)%6 == 5:
nodeVal[(i,j)] = 2
elif (j+1)%6 == 1 or (j+1)%6 == 3:
nodeVal[(i,j)] = 0
        else:
nodeVal[(i,j)] = 1
  for i in range(n):
    for j in range(n):
edgedict[((i,j),(i,(j+1)%n))] = func(nodeVal[(i,j)], nodeVal[(i,(j+1)%n)])
  for i in range(n):
edgedict[((i,0),((i+1)%n,0))] = func(nodeVal[(i,0)], nodeVal[((i+1)%n,0)])
nx.draw(C, pos, labels=nodeVal, with_labels=True, node_size=100, node_color='skyblue',
font_size=6)
nx.draw_networkx_edge_labels(graph, pos, edge_labels=edgedict, font_color='red')
corona_graph_n(6)
```

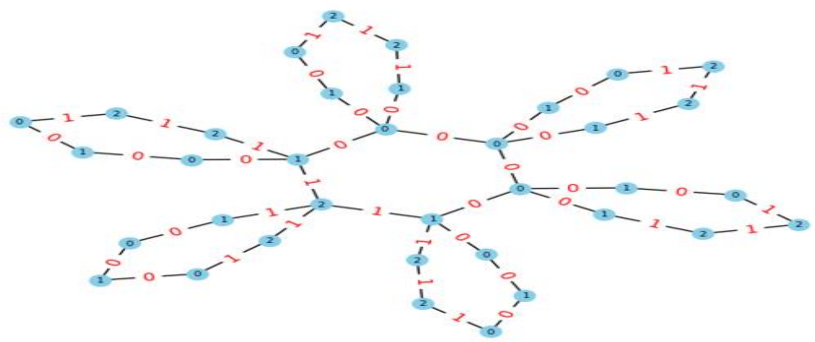

Figure 4

Received: August 06, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The preceding figure depicts the output generated by the Python program for the fifth root bi quadrate mean cordial of the coronagraph, where m and n are both multiples of 6.
Hence, the Python algorithm has been used to prove the theorem successfully.

Conclusion:

In this paper, we investigated how well the suggested fifth root biquadrate mean cordial technique works with certain types of structured graphs. The implementation of Python algorithms in labeling is extremely advantageous. We also demonstrated that complex labeled graph techniques can be effectively solved using Python algorithms.

Acknowledgment

The authors would like to thank the referees for their valuable suggestions, which greatly improved the paper.

References

[1]. Adarsh kumar Handa,Aloysius Gondinho & Tarkeshwar Singh.,On local distance antimagic labeling of graphs, AKCE International Journal of Graphs and Combinatorics.

[2]. Arumugam, S., Premalatha, K., Bača, M., Semaničová-Feňovčíková, A. (2017). Local antimagic vertex coloring of a graph. *Graphs Comb*. 33: 275–285.

[3]. Bondy J. and Murty U., Graph Theory with Applications (Macmillan, London, 1979).

[4]. Jingxiang Jin and Zhuojie Tu, Graph antimagic labeling: A survey, Discrete Mathematics, Algorithms and Applications, Vol 16, No. 01, 2330002 (2024).

[5]. Joseph A. Gallian, A Dynamic Survey of Graph Labeling, Electronic Journal of Combinatorics, Twenty-fifth edition, December 2, 2022.

[6]. R. Ponraj, S. Sathish Narayanan, and R. Kala, Difference cordial labeling of graphs, Global J. Math. Sciences: Theory and Practical, 3, (2013), 192–201.

[7]. Selvarajan, T. M., and Swapna Raveendran. "Quotient square sum cordial labeling." International Journal of Recent Technology and Engineering (IJRTE) 8. 2s3 (2019): 138-142

[8]. Selvarajan, T. M., and R. Subramoniam. "Prime graceful labeling." International Journal of Engineering and Technology 7.4.36 (2018): 750-752