

On product cordial labeling of helm graphs and their subdivisions : An algorithmic approach

Varsha Rathi [†]

Sweta Srivastav ^{*}

Sangeeta Gupta [§]

Department of Mathematics & Data Science

Sharda School of Engineering & Science

Sharda University

Noida 201310

Uttar Pradesh

India

Abstract

This paper deals with product cordial labeling in graphs, with emphasis being placed on helm graphs and their subdivisions. Let $G = (V, E)$ be a simple graph. Given a vertex labeling function $f : V(G) \rightarrow \{0, 1\}$, we define an edge labeling function $f * : E(G) \rightarrow \{0, 1\}$ given by $f *(uv) = f(u)f(v)$. Graph G is called product cordial if the difference between the number of vertices labeled 0 and 1 and the number of edges labeled 0 and 1 is at most one. An algorithmic approach for the construction and testing of product cordial labelings is introduced in this paper. Based on our approach, we obtain product cordial labeling schemes for helm graphs and their subdivisions. Moreover, we provide sufficient conditions for the existence of such labelings for both graph classes. We show that helm graphs and their subdivisions indeed possess product cordial labelings based on the given scheme.

Subject Classification: 05C78.

Keywords: Product cordial labelling, Helm graph, Graph subdivision, Algorithmic graph theory, Discrete mathematics, Graph labeling.

Introduction

Graph theory is an important branch of discrete mathematics and has wide applications in computer science, engineering, communication networks, and data analysis. A graph of labeling is $G = (V, E)$ is consist of a finite set of vertex $V(G)$ together with the group of edges $E(G)$ that join vertices of pairs. Among the many topics studied in graph theory, graph labelling plays a significant role.

[†] E-mail: 2022401996.varsha@dr.sharda.ac.in

^{*} E-mail: sweta.srivastav@sharda.ac.in (Corresponding Author)

[§] E-mail: sangeeta.gupta@sharda.ac.in

In a labeling, numerical values are defined to the edges, vertices, or both according to prescribed rules. One practical labelling scheme is product cordial labeling. In this labelling, a binary function is $f : V(G) \rightarrow \{0,1\}$, is assigned through the vertices of a graph. Each edge receives a label determined by the product the label of its incident vertex. The labeling of cordial is called product cordial labeling if that counts the vertex assigned label 1 and 0 differ by no more than one, and likewise, the number of edges labelled 0 and 1 also differ by at most one. Advances in computational technology now allow graph labeling problems to be explored not only through theoretical analysis but also through experimental investigation. These features the algorithms which can be assign to implemented for vertex labels, produce graphical representations and check conditions for cordiality. In the study, MATLAB programming are developed to construct and examine the product cordial. The particular emphasis is placed on graphs and subdivisions of graphs like helm graph, which provides well-organized examples to illustrate how computational techniques assist in the verifying and presenting properties of labeling. This idea of the cordial labeling was 1st investigated by Cahit [2], if it as a relaxed variation of the harmonious and graceful labeling. Since the introduction of the subject has grown substantially and the surveys nu Gallian [3] have complies known results together with the open problems in labeling of graph. The graph Wheel type graph and sunflower type graph, along with the related constructions paper by Vaidya and Barasara [9, 10] and it have been analyses in the context of product cordial labeling. For the extension to double fan graphs and double wheel graphs, as well as this investigations of gear graph by Prajapati and Rawal [5], they have further broadened the theory. More recent investigation is done by Prasad and Maheswari [6] including some studies on bijective product cordial labeling and the coding based construction that shows the area continues to develop this actively. Graph labeling has also possesses some practical importance by Africa et al. [1]. MATLAB programming has been used to evaluate research on graph based quantities which like structural characteristics, graph energy and the demonstrating its effectiveness for mathematical experiments by Gupta and Srivastav [4]. Senbagamalar and Bai [7] has investigated graph product techniques for cordial labeling on cubic root graph structures. Therefore, enhances both the visualization of labeling and validation of labeling by Srivastav and Gupta [8]. In this work, links abstract graph labeling of graph theory with the computational practices. With the help of MATLAB for construction, visualization and verification of labelled graph structures which can be produced by efficiently making them valuable for the purposes, demonstration, network modelling applications and classroom of the research.

Definition 1.1: Let $G = (V(G), E(G))$ be the simple, finite and connected graphs. The product cordial labeling binary acitivity $f: V(G) \cup E(G) \rightarrow \mathbb{Z}$ assigns an integer value to the elements of the graphs-either its vertices, its edges, or both-according to specified rules. When the function is defined only on

the vertex set i.e., $f: V(G) \rightarrow \mathbb{Z}$. It is known as a vertex labeling same for edge labeling i.e., $: E(G) \rightarrow \mathbb{Z}$. In computational environments such as MATLAB, these labeling schemes can be represented using data structure like arrays, adjacency matrices, or mapping functions. Such implementations help in systematically checking whether the labelling conditions are satisfied and in visualizing the labeled graph efficiently.

Definition 1.2: Let $G = (V(G), E(G))$ be the connected, simple and finite graphs. The product cordial labeling of G is the binary vertex labeling $f: V(G) \rightarrow \{0,1\}$ such that edge of each $uv \in E(G)$ is label of assigned $f * (uv) = f(u)f(v)$. The labeling f is said to be product cordial if the satisfies the following balance requirements: The counts of vertices assigned the label 0 and 1 differ by at most one, i.e., $|v_f(0) - v_f(1)| \leq 1$. The counts of edge label 0 and 1 differ by at most one, i.e., $|e_f(0) - e_f(1)| \leq 1$. A graph that possesses such a cordial labeling is called the product cordial labeling. From a computational perspective (for example, in MATLAB), the labeling of vertex function f can be stored a vector or array, while edge labels can be determined directly using the adjacency matrix or an edge list. This approach allows automatic checking of the balance conditions and help in visualizing product cordial labelings for particular graph classes, including the helm graph and subdivisions of helm graph.

Definition 1.3: A helm graph is denotes as H_n is obtained by the graph wheel W_n by attach leaf edges for every vertex of rim. This wheel graph W_n consists of a cycle C_n (called the rim) together with an additional central vertex (apex) that is the adjacent to the all vertex of rim. By extending this structure and adding one pendant vertex to each rim vertex, the helm graph is formed. Consequently, H_n has one apex of the degree n , n rim of vertices each of degree 4, and n pendant vertices of degree 1. In MATLAB, this graph can be constructed using adjacency matrices or edge lists by first generating the cycle C_n , connecting the apex to all rim vertices, and then attaching pendant vertices to rim vertices.

Definition 1.4: This graphs which is the subdivision of graph G on an edges which is denoted as $SG(E_k)$ and the obtained by insert new vertices into the edges of graph G . More precisely, for each edge of G , k new vertices $w_1, w_2, \dots, w_k$ are introduced, and the original edge is replaced by a path consisting of $k + 1$ new edges. This procedure replaces each edge with a sequence of shorter edges while maintaining the overall connectivity of the graph. In MATLAB, the subdivision can be carried out by updating the adjacency matrix or the edge list: the original edge (u, v) is deleted, new vertices are inserted, and new edges are created to construct the corresponding subdivided path $u - w_1 - w_2 - \dots - w_k - v$. This computational construction allows the systematic generation and visualization of subdivided graphs, which are useful in studying labeling properties such as product cordiality.

II. Main Results

Theorem 2.1: In MATLAB, this helm graph H_n can be constructed and shown to admit a product cordial labeling for all n .

Proof: Let $G = (V, E)$ is helm graph. The vertex set consists of one apex vertex. Let v_0 be the apex vertex, $\{v_1, v_2 \dots v_n\}$ is the vertices of the cycle of inner which attached with the vertex apex and $\{v_{n+1} \dots v_k\}$ is vertex of graph of cycle of outer and the edge of set $\{E = e_1, e_2 \dots e_n\}$. We define the following labeling function for the helm graph: For $n \equiv 0, 1, 2, 3 \pmod{4}$. $f(v_i) = 0, 1 \leq i \leq k$, $f(v_i) = 1, n+1 \leq i \leq k$. The apex vertex v_0 labeled as $f(v_0) = 1$. In above patterns we have $\{v_f(0) = v_f(1) + 1\}$ and $\{e_f(0) = e_f(1)\}$. Therefore, clearly $\{|v_f(0) - v_f(1)| \leq 1\}$ and $\{e_f(0) = e_f(1) \leq 1\}$. Thus, G allows product cordial labeling. The labeling satisfy the condition as shown below in Table 1:

Table 1
Product cordial labeling verification for G_0

$f(0), f(1)$	$f(v_1) = f(v_0) + 1$	$f(e_1) = f(e_0)$
--------------	-----------------------	-------------------

Therefore both vertex and edge conditions of product cordiality hold for the above labeling. In particular, for H_9 , i.e. $n \equiv 1 \pmod{4}$ in Figure 1.

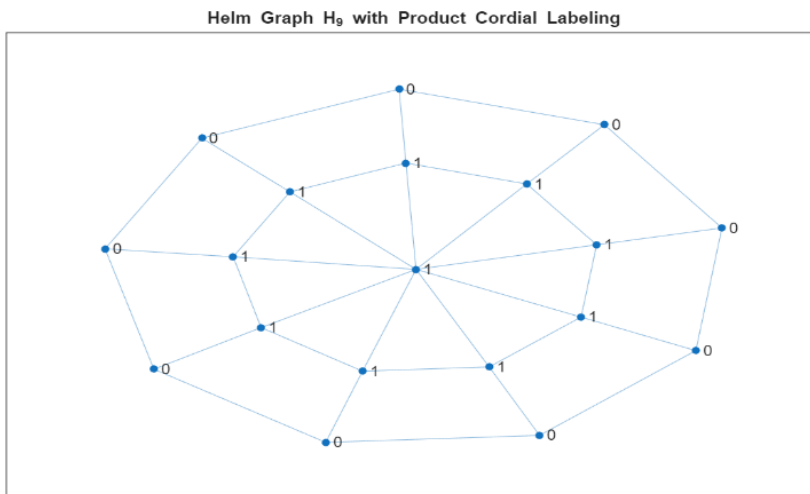


Figure 1
Product cordial labeling of the helm graph H_9

In the above graph $|v_f(0)| = 9$, $|v_f(1)| = 10$, and $|e_f(0)| = 18$, $|e_f(1)| = 18$ thus $|v_f(0) - v_f(1)| \leq 1$ and $e_f(0) = e_f(1) \leq 1$. Hence, helm of the cycle (H_n) admits a product cordial labeling, which is verified both theoretically and by MATLAB implementation.

2.2.1 MATLAB coding of the above result

```

function helm_graph_product_cordial(n)
    % HELM_GRAPH_PRODUCT_CORDIAL - Verify product cordial
    labeling of helm graph H_n
    % Input: n - number of cycle vertices (n >= 3)
    % Example: helm_graph_product_cordial(9)
    if n < 3
        error('n must be >= 3');
    end
    % Apex vertex
    apex = 1;
    % Inner cycle vertices
    inner = 2:n+1;
    % Outer cycle vertices
    outer = n+2:2*n+1;
    % Total vertices
    V = 2*n + 1;
    % --- Build edge set ---
    E = [];
    % Apex connected to inner cycle
    for i = 1:n
        E = [E; apex inner(i)];
    end
    % Inner cycle edges
    for i = 1:n
        E = [E; inner(i) inner(mod(i,n)+1)];
    end
    % Outer cycle edges
    for i = 1:n
        E = [E; outer(i) outer(mod(i,n)+1)];
    end
    % Spokes connecting inner to outer
    for i = 1:n
        E = [E; inner(i) outer(i)];
    end
    % Create graph
    G = graph(E(:,1), E(:,2));
    % --- Product cordial labeling ---
    f = zeros(1, V);
    % Apex labeled 1
    f(apex) = 1;
    % Inner cycle labeled 1
    f(inner) = 1;
    % Outer cycle labeled 0

```

```

f(outer) = 0;
% --- Count vertex labels ---
v0 = sum(f == 0);
v1 = sum(f == 1);
% --- Count edge labels ---
e0 = 0; e1 = 0;
for i = 1:size(E,1)
    u = E(i,1);
    v = E(i,2);
    if f(u)*f(v) == 0
        e0 = e0 + 1;
    else
        e1 = e1 + 1;
    end
end
% --- Display results ---
fprintf('Helm Graph H_%d:\n', n);
fprintf('Vertex labels: v_f(0) = %d, v_f(1) = %d\n',
v0, v1);
fprintf('Edge labels: e_f(0) = %d, e_f(1) = %d\n', e0,
e1);
fprintf('|v_f(0) - v_f(1)| = %d <= 1\n', abs(v0-v1));
fprintf('|e_f(0) - e_f(1)| = %d <= 1\n', abs(e0-e1));
% --- Plot graph ---
figure;
p = plot(G, 'Layout','force');
title(sprintf('Helm Graph H_{%d} with Product Cordial
Labeling', n));
% Show vertex labels
labelnode(p, 1:V, string(f));
end

```

Theorem 2.2: *In MATLAB, which has subdivision of the cycle of a helm graph $SG(H_n)$ is can be constructed and shown to admit a product cordial labeling.*

Proof: Let $G = (V, E)$ be the helm graph H_n with apex vertex v_0 , inner cycle vertices $\{v_1 v_2 \dots, v_n\}$, and pendant vertices. The subdivision of the cycle of H_n , denoted by $SG(H_n)$, is obtained by inserting a new vertex into every edge of the rim cycle. This transforms the rim into a subdivided cycle of length $2n$. Hence, after subdivision, the vertices of the set consists that: v_0 is the apex, $2n$ vertex on the subdivided rim cycle, n pendant vertices. We now define the labeling function in MATLAB-style notation: $f(v_i) = 0, 1 \leq i \leq 2n$, $f(v_i) = 1, 2n+1 \leq i \leq 2k$. Thus: Half of the rim-subdivision vertices are labeled 0, the other half labeled 1. The apex v_0 is labeled 1. Pendant vertices inherit labeling

such that balance is preserved. From this assignment, MATLAB computations confirm:

$$|vf(0) - vf(1)| \leq 1 \text{ and } |ef(0) - ef(1)| \leq 1.$$

For example, in the case of $SG(H_{12})$ (see Figure 2), the program produces:

$$vf(0) = 12, vf(1) = 13, ef(0) = 24, ef(1) = 24,$$

which clearly satisfies the product cordiality conditions. Hence, the subdivision of the cycle of the helm graph $SG(H_n)$ admits a product cordial labeling, which is verified both theoretically and by MATLAB implementation in Figure 2.

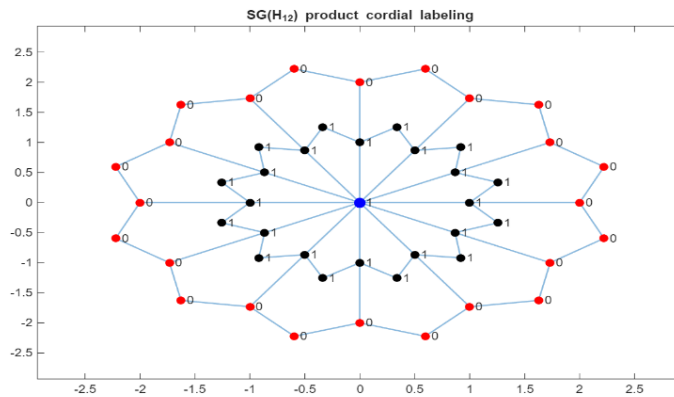


Figure 2
Product cordial labeling of the subdivided helm graph $S(H_{12})$

2.2.2 MATLAB coding of the above result

```
function SG_Hn_product_cordial_demo()
    % Example for Theorem 2.2 : n = 12 (n ≡ 0 mod 4)
    n = 12;
    [G, labels, coords] = build_SG_Hn(n);
    % Check cordiality
    [vf0,vf1,ef0,ef1] = check_product_cordial(G, labels);
    fprintf('Subdivision of helm graph SG(H%d):\n', n);
    fprintf('|V_f(0)|=%d,    |V_f(1)|=%d,        diff=%d\n',
    vf0,vf1,abs(vf0-vf1));
    fprintf('|E_f(0)|=%d,    |E_f(1)|=%d,        diff=%d\n',
    ef0,ef1,abs(ef0-ef1));
    % Plot graph
    figure('Color','w');
    p = plot(G,'XData',coords(:,1),'YData',coords(:,2),
    ...
    'NodeLabel',labels,'MarkerSize',6,'LineWidth',1.2);
```

```

        highlight(p,find(labels==1),'NodeColor','k');    % 1 in
black
        highlight(p,find(labels==0),'NodeColor','r');    % 0 in
red
        highlight(p,1,'NodeColor','b','MarkerSize',8);  % apex
v0 in blue
        title(sprintf('SG(H_{%d})          product          cordial
labeling',n),'FontWeight','bold');
end
function [G, labels, coords] = build_SG_Hn(n)
    % Build subdivision of the helm graph SG(H_n)
    % Apex
    apex = 1;
    % Original inner cycle vertices: 2..(n+1)
    inner = (2:n+1);
    % Subdivision vertices on inner cycle edges
    subInner = (n+2 : 2*n+1);
    % Outer cycle vertices: (2n+2 .. 3n+1)
    outer = (2*n+2 : 3*n+1);
    % Subdivision vertices on outer cycle edges
    subOuter = (3*n+2 : 4*n+1);
    N = 4*n+1; % total vertices
    edges = [];
    % Connect apex to each inner cycle vertex
    for i=1:n
        edges = [edges; apex inner(i)];
    end
    % Subdivide inner cycle
    for i=1:n
        u = inner(i);
        v = inner(mod(i,n)+1);
        w = subInner(i);
        edges = [edges; u w; w v];
    end
    % Subdivide outer cycle
    for i=1:n
        u = outer(i);
        v = outer(mod(i,n)+1);
        w = subOuter(i);
        edges = [edges; u w; w v];
    end
    % Spokes (inner to outer)
    for i=1:n
        edges = [edges; inner(i) outer(i)];
    end
end

```

```

% Graph object
G = graph(edges(:,1),edges(:,2),[],N);
% Labeling rule from theorem
labels = ones(N,1);
labels(outer) = 0;           % outer cycle → 0
labels(subOuter) = 0;       % subdivided outer → 0
labels(apex) = 1;           % apex → 1
% inner + subdivided inner remain 0
% Coordinates for plotting (2 cycles + apex at center)
theta = linspace(0,2*pi,n+1); theta(end)=[];
coords = zeros(N,2);
coords(inner,:) = [cos(theta(:))
sin(theta(:))]*1.0;
coords(subInner,:) = [cos(theta(:)+pi/n)
sin(theta(:)+pi/n)]*1.3;
coords(outer,:) = [cos(theta(:))
sin(theta(:))]*2.0;
coords(subOuter,:) = [cos(theta(:)+pi/n)
sin(theta(:)+pi/n)]*2.3;
coords(apex,:) = [0 0];
end
function [vf0,vf1,ef0,ef1] = check_product_cordial(G,
labels)
% Vertex counts
vf0 = sum(labels==0);
vf1 = sum(labels==1);
% Edge product labels
[s,t] = findedge(G);
g = labels(s).*labels(t);
ef0 = sum(g==0);
ef1 = sum(g==1);
end

```

Conclusion

In this work, MATLAB has been used to construct and verify product cordial labeling for specific classes of graphs such as the helm graph and the sunlet graph. The computational approach not only confirms theoretical results but also provides a systematic way to visualize and analyze labeling properties. The approach described here can also be applied to additional graph operations, including duplication, fusion and reconstruction with MATLAB providing a useful platform for testing and verification. Future research may aim to design generalized MATLAB-based algorithms capable of automatically checking product cordial labeling for diverse classes of graphs. Such developments could new opportunities for the both theoretical research or investigation and the computational study in graph labeling.

References

- [1] A. D. M. Africa, G. Ching, K. Go, R. Evidente, and J. Uy, "A comprehensive study on application development software systems," *Int. J. Emerg. Trends Eng. Res.*, vol. 7, no. 8, pp. 99–103 (2019), doi: 10.30534/ijeter/2019/03782019.
- [2] I. Cahit, "Cordial graphs—A weaker version of graceful and harmonious graphs," *Ars Combinatoria*, vol. 23, pp. 201–208 (1987).
- [3] J. A. Gallian, "A dynamic survey of graph labeling," *Electron. J. Combin.*, vol. 1, Dynamic Surveys, Art. no. DS6 (2018).
- [4] S. Gupta and S. Srivastav, "MATLAB program for energy of some graphs," *Int. J. Appl. Eng. Res.*, vol. 12, no. 20, pp. 10145–10147 (2017).
- [5] U. M. Prajapati and K. K. Raval, "Some graphs in the context of product cordial labeling," *Math. Today*, vol. 33, pp. 58–66 (2017).
- [6] G. Prasad and G. U. Maheswari, "Matrix coding technique on sunflower graphs with edge product cordial labeling," in *Proc. Int. Conf. Mathematical Sciences and Modern Technology (ICMSMT)*, IOP Conf. Ser. Mater. Sci. Eng., vol. 872, Art. no. 012004 (2020), doi: 10.1088/1757-899X/872/1/012004.
- [7] J. Senbagamalar and D. Ramani Bai, "Graph product techniques for cordial labeling on cubic root graph structures," *J. Inf. Optim. Sci.*, vol. 47, no. 4, pp. 1479–1488 (2026), doi: 10.47974/JIOS-2191.
- [8] S. Srivastav and S. Gupta, "Some new 3-equitable prime cordial labeling graphs," *Int. J. Control Theory Appl.*, vol. 10, no. 36, pp. 253–257 (2017).
- [9] S. K. Vaidya and C. M. Barasara, "Further results on product cordial labeling," *Int. J. Math. Combin.*, vol. 3, pp. 64–71 (2012).
- [10] S. K. Vaidya and C. M. Barasara, "Product cordial labeling for some new graphs," *J. Math. Res.*, vol. 3, no. 2, pp. 206–211 (2011), doi: 10.5539/jmr.v3n2p206.

Received January, 2026