Vertex Coloring and Chromatic Number of Sierpinski Complete Graph Using Scilab

Akhilesh Kumar Verma[1], Ashish Kumar[2], Chintamani Tiwari[3]

[1,2]Student, MSOS, Maharishi University of Information Technology, Lucknow, U.P., India
[3]Faculty, MSOS, Maharishi University of Information Technology, Lucknow, U.P., India

Abstract:
Graph theory is an important branch of mathematics with applications in computer science, communication networks, scheduling, and optimization problems. This research paper focuses on the vertex coloring of the Sierpinski Complete Graph S(2,K4) and the determination of its chromatic number using Scilab programming. The study explains the concepts of graph coloring, vertex coloring, and chromatic number along with the structural properties of Sierpinski graphs. A recursive backtracking algorithm was implemented in Scilab to determine the minimum number of colors required for proper coloring of the graph. The computational results show that the graph requires exactly four colors for proper vertex coloring. The study also demonstrates that the chromatic number depends strongly on the connectivity and arrangement of vertices in the graph. The obtained results are consistent with theoretical graph coloring principles and confirm that recursive backtracking is an efficient method for solving graph coloring problems.

Keywords: Graph Theory, Graph Coloring, Chromatic Number, Sierpinski Graph, Scilab, Backtracking Algorithm.

Introduction:
Graph Theory is one of the most important and rapidly developing branches of mathematics. Over the last few decades, it has gained significant importance because of its wide range of applications in computer science, communication systems, engineering, transportation networks, social sciences, biology, and many other disciplines. A graph is a mathematical structure used to represent relationships between different objects. It consists of a set of vertices (nodes) and edges that connect pairs of vertices. Graphs provide a simple and effective way to model complex systems and analyze their structural properties. Among the various areas of graph theory, graph coloring has become a major field of research due to its practical and theoretical importance. Graph coloring deals with assigning colors to the elements of a graph under certain conditions. In vertex coloring, colors are assigned to vertices in such a way that no two adjacent vertices receive the same color. The minimum number of colors required for proper coloring of a graph is known as its chromatic number. Determining the chromatic number of a graph is one of the fundamental problems in graph theory and has applications in scheduling, register allocation in compilers, map coloring, frequency assignment, and network optimization. The coloring of the Sierpinski Complete Graph (S(2,K_4)). Sierpinski graphs are recursive graphs constructed from complete graphs and possess highly connected and self-similar structures. Due to their recursive nature and complex adjacency relationships, these graphs provide an interesting area for the study of graph coloring problems. The determination of chromatic numbers for such graphs helps in understanding their structural and combinatorial properties. The chromatic number of the Sierpinski Complete Graph is determined using Scilab programming and recursive backtracking techniques. The recursive backtracking method systematically explores all possible color assignments while ensuring that adjacent vertices receive distinct colors. Whenever a conflict occurs, the algorithm rejects the current assignment and searches for another valid arrangement. This approach guarantees an exact solution for the graph

coloring problem. The main objective of this study is to analyze the vertex coloring properties of the graph ($S(2, K_4)$) and determine its chromatic number through both theoretical and computational methods. The study also demonstrates the effectiveness of recursive algorithms in solving complex graph coloring problems.

Graph Coloring: The assign of colors to the vertices of G, one color to each vertex, so that adjacent vertices are assigned different colors is called the proper coloring of G or simply vertex coloring. Then n- coloring of G is coloring of G using $n-$colors. If G has n-coloring, then G is said to be n-colorable.

There are two types of coloring.

1. Vertex coloring
2. Edge coloring

Vertex-Coloring: The assign of colors to the vertices of G, one color to each vertex, so that adjacent vertices are assign different colors is called the proper coloring of G or simply vertex coloring.

Chromatic number: The chromatic number of a graph G, denoted (G), is the least number of distinct colors with which G can be properly colored. It is denoted by $\chi(G)$.

There is no easy way of finding $\chi(G)$ of a graph G. The following rules may be helpful in finding $\chi(G)$.

1. A graph consisting of only isolated vertices is 1-chromatic.
2. $\chi(G) \le |V|$ Where $|V|$ is the number of vertices of G.
3. If a subgraph of G requires m colors then $\chi(G) \ge m$.
4. If the degree of vertex of G is d, then almost d colors are required to color vertices adjacent to it.
5. If $\delta(G)$ minimum degree of any vertex of G, then $\chi(G) \ge \dfrac{|V|}{|V| - \delta(G)}$
6. Every k-chromatic graph has at least k vertices v such that degree $(v) \ge k-1$.
7. The following statements are equivalent.

i. A graph G is two colorable

ii. G is bipartite

iii. Every cycle of G has even length

Back algorithm for vertex coloring of a graph

Input is graph G.

1. Order the vertices of G in decreasing degree (such a ordering may not be unique since some vertices may have the same degree).
2. Assign a color to paint the first vertex and to paint, in sequential order, each vertex on the list that is not adjacent to a vertex previously painted with this color.
3. Repeat 2 the process of painting previously non colored vertices using a second color.
4. Repeat 3 with the third color, then a fourth color, and so on until all vertices are colored.
5. Exit.

Sierpinski complete Graph

Sierpinski graph is a geometric figure formed by continuing a process and repeating it over and over again (*i.e.*, that occurs at different levels of iterations). We construct Sierpinski by making c copies of K_n

(*i.e.*, complete graph) which is to be connected to each other with an edge set in one-to-one correspondence with the edges of $S(c,K_n)$.

The Sierpinski graph $S(c,K_n)$, the vertex set consist of all n-tuples of the integer (for every $(n \geq 2, n \geq 3)$) that is $V(S(c,K_n)) = (1,2,3,\ldots k)^n$. There is an edge between two vertices $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$ if and only if there exists an $h \in [n]$ such that

(i) $u_j = v_j$ for $j = 1,2,3,\ldots\ldots,h-1$;

(ii) $u_h \neq v_h$; and

(iii) $u_j = v_h; v_j = u_h$; for $j = h+1,\ldots,n$.

We will assign that $\langle u_1 u_2 \ldots u_n \rangle$ for $(u_1, u_2, \ldots., u_n)$ or even shorter $u_1, u_2, \ldots., u_n$. See Fig.3.2 for $S(2, K_4)$.

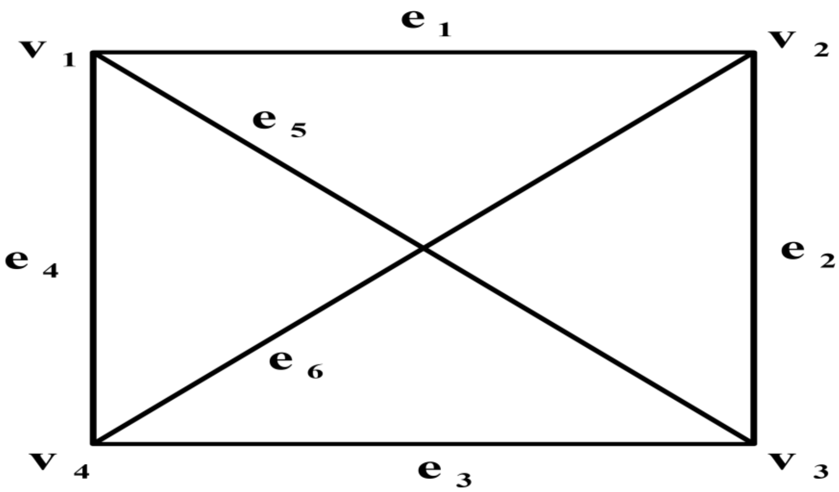

Figure 1: $S(1, K_4)$

International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

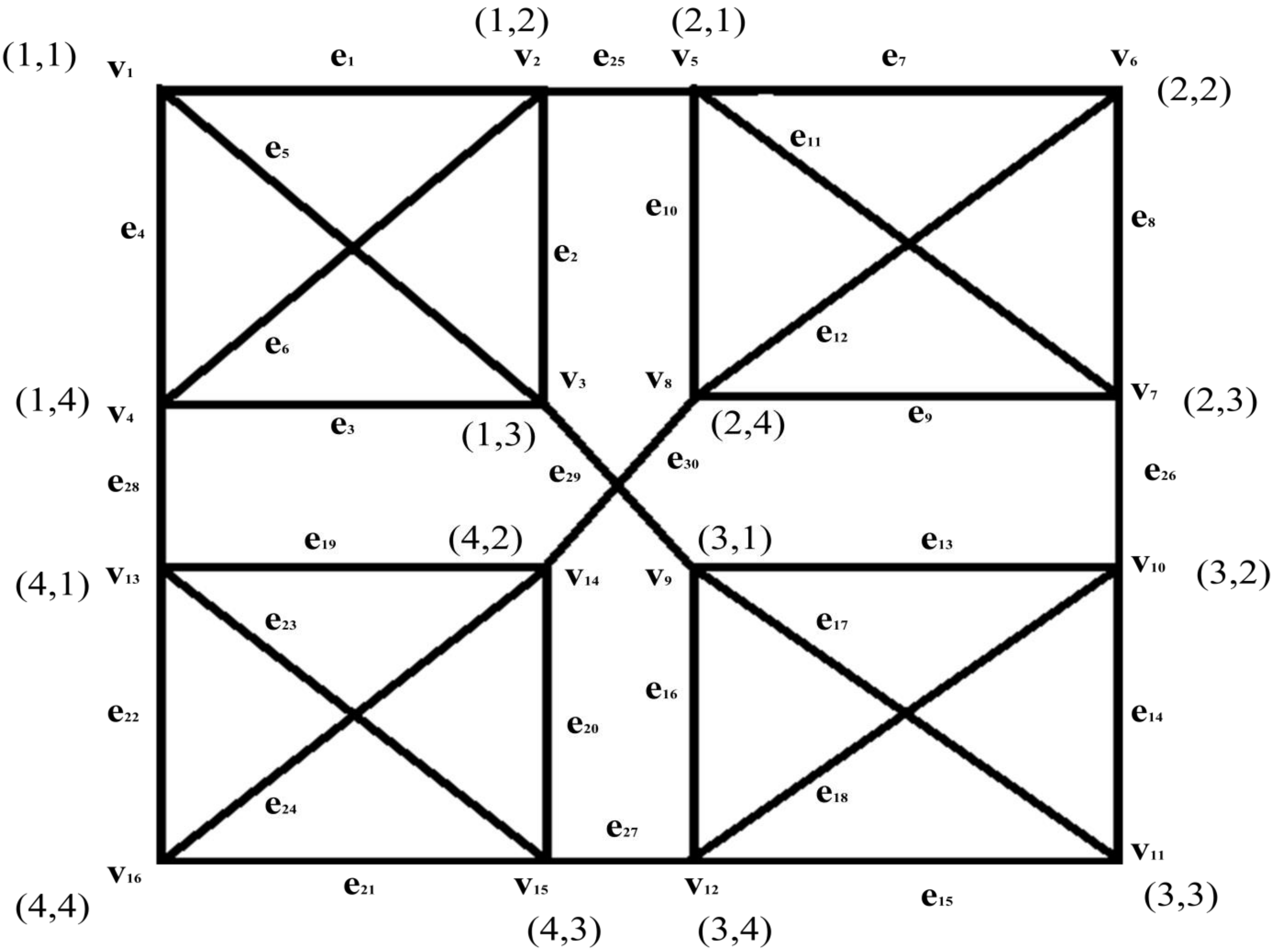

Figure2: graph $S(2, K_4)$

The vertices $\langle i.......i \rangle, i \in \{1,2,3,......,k\}$, are called extreme vertices of $S(c, K)$. For $i = 1,2,...,k$ consider $S_i(c+1, K)$ be the subgraph of $S(c+1, K)$ induced by the vertices of the form $\langle i... \rangle$. Other than extreme vertices of $S(c, K)$ called inner vertices. The extreme vertices of $S(c, K)$ are of degree $K-1$ while the degree of inner vertices is K. It is clear from above that $S_i(c+1, K)$ is isomorphic to $S(c, K)$. Consequently, $S(c+1, K), K > 2$, contains k^n copies of the graph $S(1, K) = k_K$. The edges of $S(c, K)$ that lie in no induced k_K will be called linking edges.

The Sierpinski carpet graph $S_n, n \geq 1$, is obtained from $S(2, K_4)$ by contracting all edges of $S(2, K_4)$ that lie in no square. In the given figure of Sierpinski graph $S(2, K_4)$ it is observed that the extreme vertices of $S(2, K_4)$ are (1,1), (2,2), (3,3), and (4,4).

The Chromatic numbers of some of the Complete Sierpinski Graph $S(2, K_4)$ graphs:

Chromatic number of complete graph:

$\chi(K_n) = n$ Where K_n is complete graph of n vertices.

Since any two vertices are adjacent in a complete graph, no two vertices can receive the same color. It is clear that it cannot be colored with fewer than n colors.

SciLab Code to Generate Chromatic Number of a given Complete Sierpinski Graph $S(2,K_4)$:

```
clc;
clear;
// ---------- INPUT ----------
// Number of vertices
n = 5;
// Adjacency Matrix of Graph
A = [
0 1 1 0 0;
1 0 1 1 0;
1 1 0 1 1;
0 1 1 0 1;
0 0 1 1 0
];
// ---------- INITIALIZATION ----------
global A color n;
color = zeros(1,n);
// ---------- FUNCTION TO CHECK SAFE COLOR ----------
function ok = isSafe(v,c)
    global A color n;
   ok = %t;
    for i = 1:n
        if A(v,i)==1 & color(i)==c then
            ok = %f;
            return;
        end
    end
endfunction
// ---------- BACKTRACKING FUNCTION ----------
function success = graphColoring(v,m)
    global color n;
   // All vertices colored
    if v > n then
        success = %t;
        return;
    end
    success = %f;
    // Try all colors
    for c = 1:m
      // Check whether color is safe
        if isSafe(v,c) then
               // Assign color
            color(v) = c;
            // Recur for next vertex
            if graphColoring(v+1,m) then
                success = %t;
                return;
            end
                    // Backtracking step
```

```
      color(v) = 0;
    end
  end
endfunction
// ---------- FIND CHROMATIC NUMBER ----------
for m = 1:n
  color = zeros(1,n);
  if graphColoring(1,m) then
    disp("Chromatic Number = ");
    disp(m);
    disp("Vertex Colors:");
    disp(color);
    break;
  end
end
```

OUTPUT:

Chromatic Number: 4 (Vertex Coloring)

Conclusion:

This study examined the vertex coloring of the Sierpinski complete graph using Scilab programming and recursive backtracking techniques. The computational results showed that the graph $S(2, K_4)$ requires exactly four colors for proper coloring. The study also demonstrated that graph structure and connectivity strongly influence the chromatic number. The recursive backtracking method proved to be an effective approach for determining exact chromatic numbers of complex graphs.

REFERENCES:

1. **Akyar H. [2017]** "On the incidence Chromatic Number of Sierpinski Graph", IOSR Journal of Mathematics, Vol. 13, Issue (6), pp.72-79.
2. **Bapat R. B. [2010]** "Graph and Matrices", Springer-Verlag, New York.
3. **Belmonte R., James M. and Singh B. (2024)** "Odd Chromatic Number of Graph Classes", Mathematical Review and Analysis, Vol. (22), Issue (04), pp. 112-125.
4. **Bing Z. [2013]** "Multiple circular coloring as a model for scheduling", Discrete Mathematics, Vol.3, pp.162-166.
5. **Bondy J.A. and Murty U. S. R. [1976]** "Graph theory with applications", http://www.zib.de/groestschel/teaching/WS1314/BondyMurtyGTWA.pdf
6. **Fisher Michael J and Issak Garth[2008]** "Distinguising coloring of Cartesian products of complete graphs". Discrete mathematics, Vol.308 Pp.2240-2246.
7. **Geetha J., Somasundaram K.[2015]** "Total coloring of generalized Sierpinski graphs", Australasian Journal of Combinatorics, Vol. 63, Issue (1), pp. 58-69.
8. **Kamalian Rafayel R. [2013] "**On a Number of Colors in Cyclically Interval Edge Colorings of Simple Cycles" Open journal of Discrete mathematics, Vol.3, Pp.43-48.
9. **Kikyo H., Nakaura K. and Tsuboi A. (2026)** "On Chromatic Number of Countable Graphs", Modeling and Analysis of Information Systems, Vol. (13), Issue (06), pp. 78-89.
10. **Kumar A. (2023)** "Graph coloring for Crop Sowing Optimization", Int. J. Agriworld, vol. (04), Issue (02), pp. 7-9.
11. **Kumar A., James M., and Singh B. (2019)** "Chromatic number and chromatic index of product of two isomorphic regular graphs", International of Advanced Scientific Research and Management.

12. **Kumar A. (2017)** "A study on Euler Graph and it's application", International Journal of Mathematics Trends and Technology, Vol. (43)
13. **Ling X. and Jiangliang W.[2013]** "Edge coloring of planer graphs without 6 cycles with two chords", Open journal of Discrete Mathematics, Vol.(3), pp.83-85.
14. **Wilson R.J. and Watkins J.J. (2024)** "Historical Perspectives on the Four Color Theorem", Modeling and Analysis of Systems, Vol. (09), Issue (08), pp. 17-30
15. **Xue Ling and Wu Jiangliang [2013]** "Edge coloring of planer graphs without 6 cycles with two chords", Open journal of Discrete mathematics,Vol.3, Pp.83-85
16. **Zhou Bing [2013] "**Multiple circular colouring as a model for Scheduling" Science direct discrete mathematics, Vol.3, Pp.162-166.