Explorations on the Sierpinski Gasket Graph

ANUSHKA TONAPI

In [6] the author describes the general properties of fractals and elaborates on the construction of a specific well-known fractal, the *Sierpinski Triangle*. The article delves into how pre-service teachers can explore various attributes of the fractal, such as the number of shaded triangles and shaded area at each stage. In this article we link the Sierpinski triangle (also known as the *Sierpinski Gasket*) to graph theory. We illustrate how the Sierpinski Gasket is related to the Sierpinski Gasket graph and explore Eulerian and Hamiltonian cycles in the graph using the CAS *Mathematica*.

Some mathematical preliminaries

In this section we shall elaborate on the mathematical concepts and definitions which are required for understanding the properties and characteristics of the Sierpinski Gasket graph.

The Sierpinski Gasket is a strictly self-similar fractal. This means that any arbitrary portion of the fractal at any given stage is a copy (at a reduced scale) of some previous stage of the fractal. It is initiated by considering a shaded or coloured equilateral triangular region whose sides are of unit length, and joining the midpoints of the three sides to create four smaller equilateral triangular regions. Once this is done, the central triangle is removed, leaving three smaller equilateral triangles. The initial equilateral triangular region is referred to as stage 0, and the next stage which comprises three scaled down smaller triangles (and a triangular hole) is stage 1 of the fractal. To continue the construction, the same process

Keywords: Fractals, Sierpinski, graph theory, Hamiltonian graphs, Mathematica.

Figure 1. The first five stages of the Sierpinski gasket fractal. Image Credit: Fractal Foundation

is repeated on the three equilateral triangles of stage 1 to obtain stage 2 as shown in Figure 1. This process of replication continues wherein each smaller triangle is split into three still smaller triangles by joining the midpoints of these triangles and removing the central triangle. The reader may find the construction process in the article mentioned at the beginning of this article.

The next few definitions are related to graph theory, which is the study of *graphs*. In the context of graph theory, graphs are mathematical representations of pairwise relationships between objects.

A *graph* is a structure that consists of a set of *vertices* or *nodes*, denoted by *V*, and a set of edges, denoted by *E*, which consists of two-element subsets of *V*.

The number of vertices connected to a given vertex *v* through an edge is the *degree of that vertex* and is denoted by "deg" *(v)*.

A *walk* in a graph *G* is a finite sequence of consecutive edges and vertices that are all connected. A walk in which all the edges are distinct, that is, no edge is repeated, is called a *trail*.

A trail in which all vertices are distinct is called a *path*. If the walk ends at the same vertex where it began, then it is said to be closed. A closed path containing at least one edge is called a *cycle*.

A graph G is *connected* if there is a path between every pair of vertices in G.

A graph is *bipartite* if its vertices can be separated into two disjoint sets such that every edge in the graph connects a vertex in one set to a vertex in the other. However, there are no edges connecting pairs of vertices from the same set. An example of a bipartite graph is shown in Figure 2 where the blue and red vertices are disjoint sets and all the edges (the black lines) connect a blue vertex to a red one. However, there are no edges among the blue vertices nor the red ones.

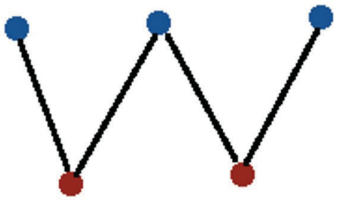


Figure 2. An example of a bipartite graph.

A connected graph G is *Eulerian* if there exists a closed trail containing every edge of G. Such a trail is called a *Eulerian trail*.

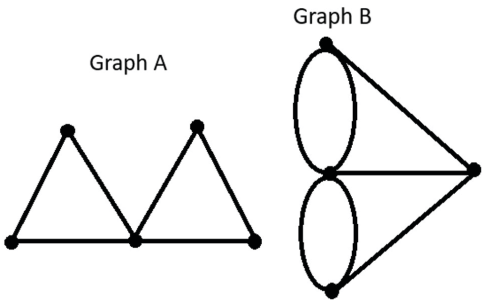


Figure 3. Graph A is Eulerian, while Graph B is non-Eulerian

A connected graph G is *Hamiltonian* if it contains a closed walk which passes through every vertex exactly once and in which no edge is repeated. Subsequently, a graph G is Hamiltonian if it contains a Hamiltonian cycle. Figure 4 is an example of a Hamiltonian cycle in a dodecagon graph.

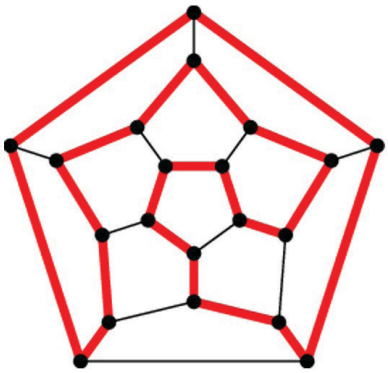


Figure 4. Hamiltonian Dodecagon graph.
Image credit: ResearchGate

The Sierpinski gasket graph and its properties

The *Sierpinski Gasket Graph* is the graph formed when the fractal of the same name is recognized as a system of edges and vertices rather than as a system of triangles. This is illustrated in Figure 5. It has the following properties: it is Eulerian, Hamiltonian, non-bipartite and connected. This means that it is possible to find a cycle which starts at a vertex, traverses each vertex and each edge exactly once. Further, it cannot be separated into two distinct sets of vertices A and B, such that all edges connect vertices in A with those in B.

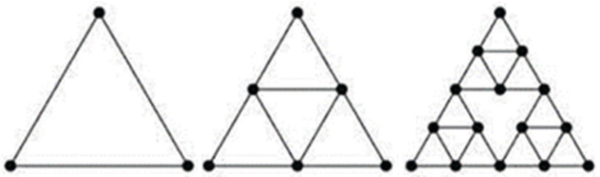


Figure 5. Stages 0, 1 and 2 of the Sierpinski gasket graph

Hamiltonian paths in the Sierpinski gasket graph

In this section we shall illustrate that the Sierpinski gasket graph is Hamiltonian, that is, it has a Hamiltonian path. A Hamiltonian path is a non-empty consecutive succession of edges that covers all the vertices in a graph exactly once. A Hamiltonian cycle is a closed walk in which all vertices in the graph appear exactly once and it ends at the same point at which it started. To determine whether a graph is Hamiltonian or not is a non-trivial problem.

Due to the Sierpinski graph's fractal nature and connectedness, there exists an inductive method of constructing Hamiltonian cycles in any given stage of the graph. For any stage n, we identify a Hamiltonian path and replicate this path in the three copies of the nth iteration that constitute the $(n + 1)$th iteration. This has been illustrated in Table 1 for $n = 0$, 1, 2 and 3, where all Hamiltonian paths (in the nth iteration) are indicated in blue and corresponding Hamiltonian cycles (in the $(n + 1)$th iteration) are represented in red.

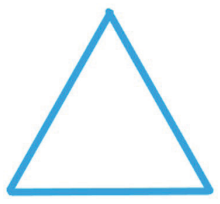
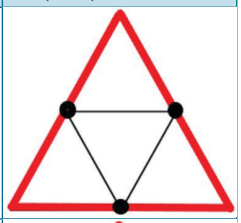
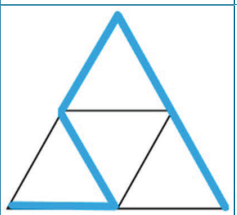
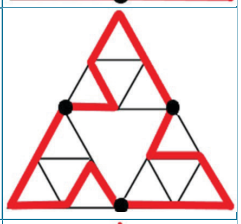
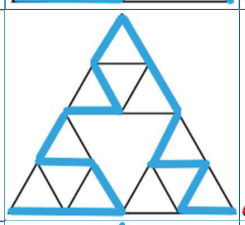
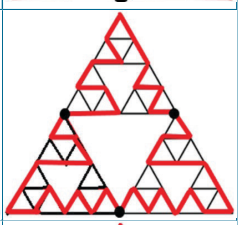
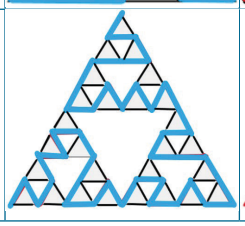
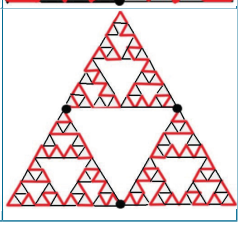
Iteration	Hamiltonian path in nth iteration	Hamiltonian cycle in $(n+1)$th iteration
$n = 0$		
$n = 1$		
$n = 2$		
$n = 3$		

Table 1.

We proceed by mathematical induction on n, or the nth iteration of the fractal graph. Consider the base case $n = 0$, the generator of the fractal graph, or the equilateral triangle from which the fractal is constructed. The generator itself is trivially Hamiltonian, since it is a closed loop where each of its three vertices is visited exactly once.

For our induction hypothesis, we shall assume that the Sierpinski Gasket graph of stage k is Hamiltonian, i.e., it contains a Hamiltonian path. We will prove that the Sierpinski Gasket graph of the $(k + 1)$th iteration is also Hamiltonian. Note that the $(k + 1)$th iteration of the graph is constructed using three replicas of the kth iteration. Before proceeding further, we need to introduce a nomenclature with regard to the vertices of the graph. The red vertices in figure 6 are referred to as *apex* vertices and the blue ones are referred to as *midpoint* vertices.

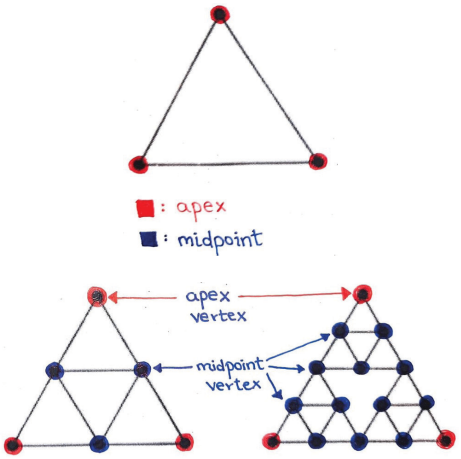


Figure 6. Apex and midpoint vertices of the Sierpinski Gasket graph

From our induction hypothesis, where we assumed that the kth iteration of the fractal graph is Hamiltonian, we can replicate the Hamiltonian path of the kth stage to traverse all three replicas of the (k + 1)th stage. Since the apex vertices of the three replicas in the (k + 1)th stage are connected through three edges (shown in black), we can traverse these edges to create a cycle which passes through all the vertices of the (k + 1)th stage exactly once. Therefore, this cycle is Hamiltonian by definition.

For example, the path in stage $n = 0$ can be reproduced using a similar method in its three replicas in stage $n = 1$ to produce a Hamiltonian cycle. This can be done by travelling the paths in the three replicas via the three connecting edges, as illustrated in the first row of Table 1.

Similarly, the path in stage $n = 1$ can be produced in the three replicas present in stage $n = 2$. These paths can be connected to make a Hamiltonian cycle via the three connecting midpoint vertices (marked as black dots) of these replicas.

However, owing to the graph's fractal nature, as n increases, the transition from the nth to the $(n + 1)$ th stage becomes harder to construct. We will now discuss a method that can be used to construct Hamiltonian cycles on the Sierpinski gasket graph using Mathematica which can be accessed via the website wolframcloud.com.

Introduction to Mathematica

Wolfram Mathematica is a versatile software system based on the Wolfram language. It has powerful reach across various fields including mathematics, computer science, economics, statistics, machine learning, geometry and data science. It has numerous advantages which include a user-friendly programming interface, with a programming language built to resemble English-like function names and a coherent design. Its algorithms are very efficient, capable of tackling large-scale problems, due to its large database based on the Wolfram Knowledgebase.

In this section we will explore the graph theoretic features of the Sierpinski gasket graph with respect to programming components in the *Combinatorica* package of Wolfram Mathematica.

Constructing the Sierpinski Gasket Graph using Mathematica

In order to construct the graph, we need to use the built-in function in Mathematica called **`GraphData`**, which, depending on the type of command in the suffix, returns the required graph based on its conventional name, usually accompanied by its properties or its class. The Sierpinski gasket graph is obtained using the command **`GraphData [{"Sierpinski, n"}]`**. Here, n stands for the iteration number, and can range only between 0 and 5.

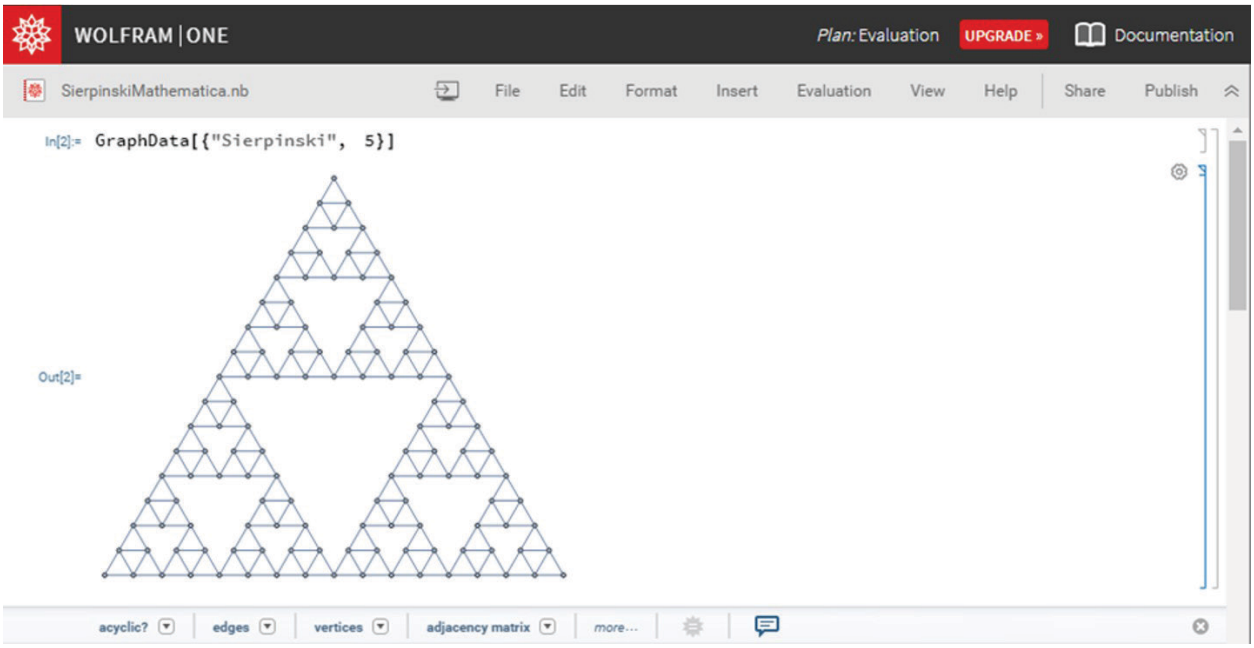


Figure 6. The fifth iteration of the Sierpinski gasket graph using Mathematica

For example, the 5th iteration of the Sierpinski gasket graph is generated in Mathematica as shown in Figure 6.

Constructing Hamiltonian Cycles using Mathematica

The **`FindHamiltonianCycle[g]`** function [4] helps us find a Hamiltonian cycle in the graph g, by giving us vertex directions with respect to labelled vertices in the graph. Mathematica chooses to label vertices using numbers. The **`HighlightGraph`** function helps us highlight the said cycle through a specific iteration of the graph. Figures 7 and 8 illustrate the Hamiltonian cycles in stages 1 and 2 of the Sierpinski gasket graph respectively.

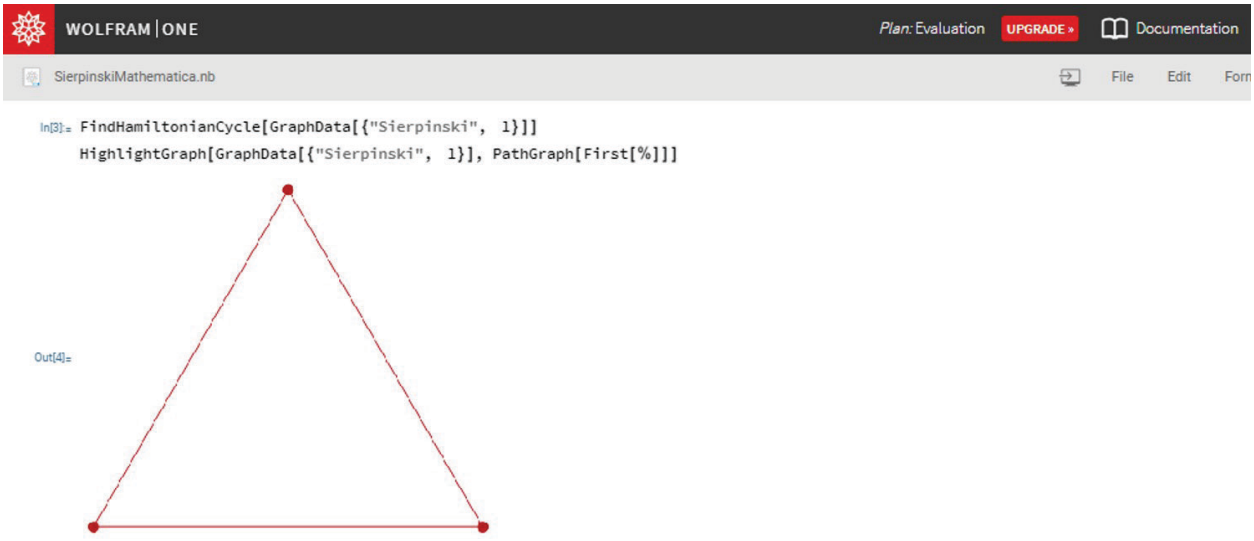


Figure 7. Hamiltonian cycle of stage 1 of the Sierpinski gasket graph

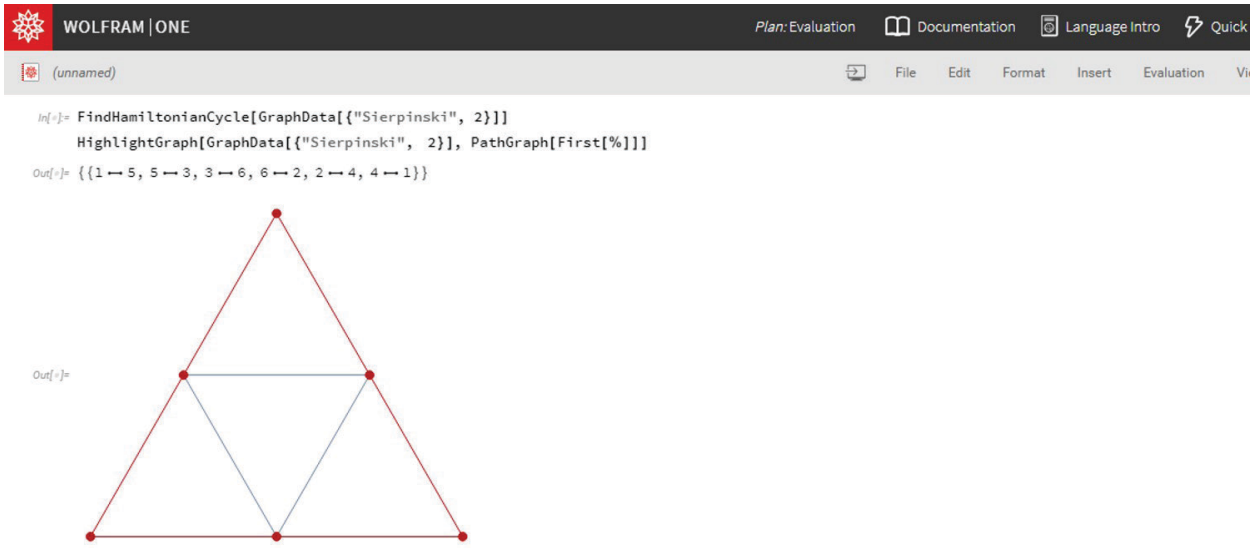


Figure 8. Hamiltonian cycle of stage 2 of the Sierpinski gasket graph

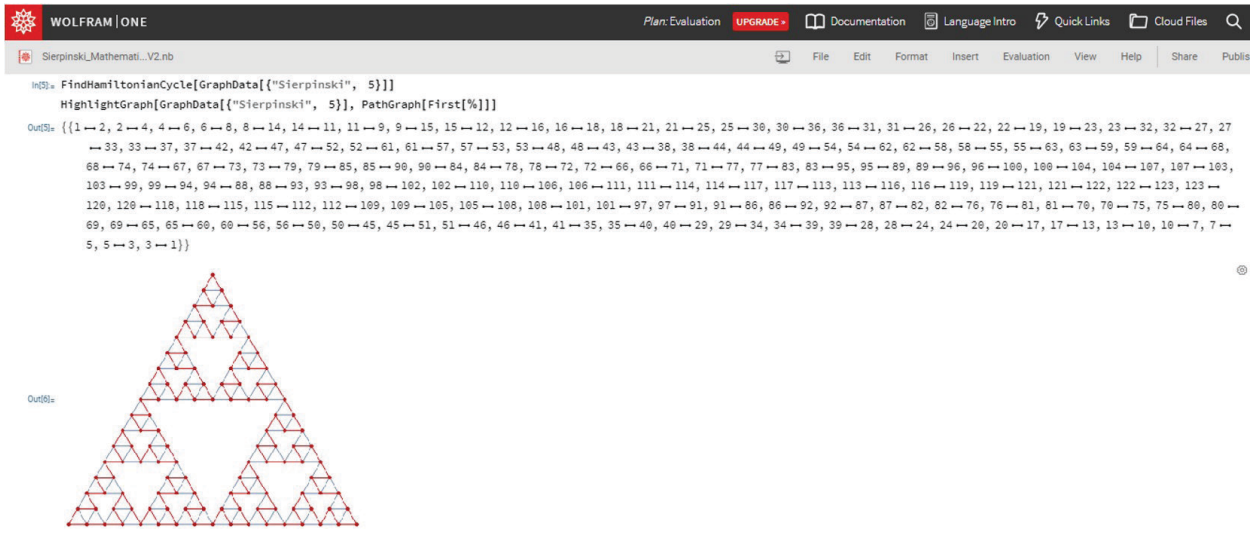


Figure 9. Hamiltonian cycle of stage 1 of the Sierpinski gasket graph

Figure 9 shows the output of the Mathematica codes when the functions are applied on the fifth iteration of the Sierpinski Gasket graph.

Eulerian Paths in the Sierpinski gasket graph

An Eulerian circuit is a closed walk in which each edge appears exactly once. It is derived from the Seven Bridges of Konigsberg Problem (which can be studied in detail in the article *Leonard Euler's Solution to the Konigsberg Bridge Problem*, by Teo Paoletti), wherein one must cross the seven bridges in a town without repeating a bridge. The bridges create a graph structure, and this problem was first solved by the mathematician Leonhard Euler.

Before exploring Eulerian paths, we need to mention an important result here.

A graph G is Eulerian if and only if every vertex in G is of even degree. An example is illustrated in the below graph.

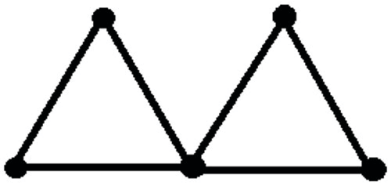


Figure 10. Example of an Eulerian graph in which every vertex is of even degree, that is, there are an even number of edges attached to every vertex.

An intuitive explanation for the even degree argument is that while travelling a graph, if it has an Eulerian cycle, we must enter every vertex via one edge and exit via another, i.e., there must be a receiving edge and a leaving edge. It can be shown that every vertex in the Sierpinski Gasket graph is of even degree, and hence the graph is Eulerian. Referring to figure 6, each apex vertex is of degree 2 since it is incident on only two other vertices. The midpoint vertices have degree 4, since they are incident on two vertices located in front of the vertex and two towards the right and left directions of that particular vertex. Therefore, all vertices in the Sierpinski Gasket graph have even degree.

Constructing Eulerian cycles using Mathematica

The `FindEulerianCycle[g]` function [5] helps us find an Eulerian cycle in the graph g, by giving us vertex directions with respect to labelled vertices in the graph. The `HighlightGraph` function helps us highlight the cycle through a specific iteration of the graph. The code (as shown in Figure 10) also displays the vertex progression of the cycle. The output figure of this code, unlike the Hamiltonian cycle code, does not highlight the edges since all the edges are covered in an Eulerian cycle. Thus, all the edges appear as one colour.

Real-world Applications

The Sierpinski Gasket graph has several applications. It is used in modelling quantum transport and quantum walks as discussed in [2]. It is used to investigate electronic properties and molecular chains to simulate experimental synthesised fractal nanostructures [3]. It also has many applications in the area of cellular automata.

It is modelled as a planar superconducting fractal lattice and exposed to a perpendicular magnetic field. The self-similarity of the fractal plays a role in addressing two central issues, namely, flux-quantization phenomena in loops and the low-field scaling behaviour of the magneto-inductance.

The Sierpinski triangle is used as a model for a bowtie antenna, showing advantages such as an efficient SERS substrate, 14% shrinkage (more compact) and higher resonance. The transducer of Sierpinski curve geometry was utilised for the miniaturisation of a microstrip patch strain sensor. The results showed the possibility of a dimension reduction due to the fractal structure.

Conclusion

In this article we have made connections between the Sierpinski gasket fractal and graph theory by exploring some properties of the Sierpinski Gasket graph. These properties elicit

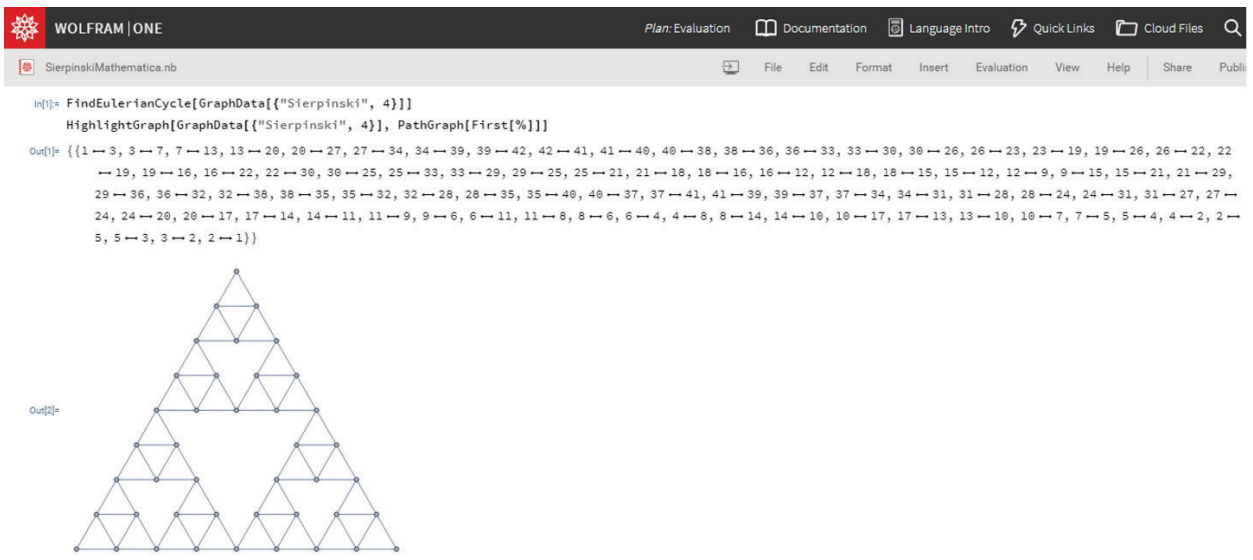


Figure 10: Generating an Eulerian cycle in the 4th iteration of the Sierpinski gasket graph.

the intricate nature of these fractal graphs. In particular, we have explored Eulerian and Hamiltonian cycles on the Sierpinski Gasket graph. However, such cycles tend to become intricate and complex as the number of stages of the fractal increases and it becomes difficult to construct these manually. Computer Algebra Systems such as Mathematica can be effectively used to identify such cycles. The results are faster and more accurate. Thus, this article illustrates the importance of computer algorithms with respect to cycle construction in graphs.

Bibliography

[1] Teguia, Alberto M. and Godbole, Anant P. (2006): "Sierpinski Gasket Graph and some of their Properties", https://arxiv.org/pdf/math/0509259.pdf

[2] Darázs Z, Anishchenko A, Kiss T, Blumen A, Mülken O. "Transport properties of continuous-time quantum walks on Sierpinski fractals." Phys Rev E Stat Nonlin Soft Matter Phys. 2014 September

[3] L Lage, L Latgé A. "Electronic fractal patterns in building Sierpinski-triangle molecular systems." Chem Phys. 2022 August

[4] FindHamiltonianCycle - Wolfram Mathematica Reference, https://reference.wolfram.com/language/ref/FindHamiltonianCycle.html

[5] FindEulerianCycle - Wolfram Mathematica Reference, https://reference.wolfram.com/language/ref/FindEulerianCycle.html

[6] Jonaki Ghosh, "Fractal Constructions Leading to Algebraic Thinking", At Right Angles, November 2016, http://publications.azimpremjifoundation.org/3135/1/13-jonaki_algebraic-thinking.pdf

ANUSHKA TONAPI is a fourteen-year-old student of Sri Kumaran Children's Home in Bangalore. She is multifaceted, and loves to learn and discuss new things, especially about Mathematics, Science, app development, Carnatic music, and writing. She has published short stories and poems in various Children's magazines and newspapers. Anushka can be reached at anushka.tonapi@gmail.com