Approximating graphs of a class of general Sierpinski triangles and their normalized Laplacian spectra

Zhiyong Zhu

Abstract
The normalized Laplacian spectrum of a graph is an important tool that one can use to find much information about its topological and structural characteristics and also on some relevant dynamical aspects, specifically in relation to random walks. In this paper we devise an essentially algorithm to obtain the approximating graphs of a class of general Sierpinski triangles and their normalized Laplacian spectra, and illustrate such algorithm by a quasi-program of Matlab. In the meantime, our work also enriches the graphs whose spectrum is known.

Keywords
Graph, normalized Laplacian spectrum, general Sierpinski triangles, algorithm, Matlab

Received 16 May 2020; revised 9 August 2020; accepted 5 December 2020

Introduction
Spectral graph theory is an interesting topic, in a way, be regarded as the study of relation between graph properties and Laplacian matrix spectrum or adjacency matrix spectrum.1–4 The spectrum of a graph (the set of eigenvalues of a graph with their multiplicities) is important in the spectral graph theory, because almost all major graph invariants are related to it, such as connectivity, expanding property, isoperimetric number, maximum cut, independence number, genus, diameter, bandwidth-type parameters, etc (see2,5,6 and the reference therein).

Spectral graph theory has different applications in different scientific fields, such as the eigenvalues were related to the stability of molecules in chemistry, graph spectra occured naturally in minimizing energies of Hamiltonian systems in physics, the recent progress on expander graphs and eigenvalues was initiated by problems of communication networks in computer science, etc. In recent years, there has been a particular interest in the study of the eigenvalues and eigenvectors of normalized Laplacian matrix, since much important information about structural properties and dynamical processes of a graph or network is related to it. In the structural aspect, for example, F. R. Chung2 proved that the number of spanning trees in a network can be determined in terms of the product of all nonzero eigenvalues of a connected network. H. Chen and F. Zhang7 showed the resistance distance can be naturally expressed by the eigenvalues and eigenvectors of normalized Laplacian matrix. With respect to dynamical process, there has been great progress in determining the eigenvalues and eigenvectors of normalized Laplacian matrix, including the first-passage time,8 mixing time9 and Kemeny’s constant10 that can be used as a measure of the efficiency of navigation on the network. Particularly, normalized Laplacian eigenvalues and eigenvectors also are relevant in light harvesting,11 energy or exciton transport,12 chemical kinetics,13 and many other problems in chemical physics.14

Let

\[ \Delta = \{ c_1 x + c_2 \beta : c_1 + c_2 \leq 1 \text{ and } 0 \leq c_1, c_2 \leq 1 \} \]

College of Science, Northwest A&F University, Shannxi, P.R.China

Corresponding author:
Zhiyong Zhu, College of Science, Northwest A&F University, Yangling, Shannxi 712100, P.R.China.
Email: zhuzy1011@nwafu.edu.cn
be an equilateral triangle with lower-left coordinate (0, 0) and side 1, where \( x = (0, 1) \) and \( \beta = (1/2, \sqrt{3}/2) \). Given an integer \( n \geq 2 \), write

\[
D(n) = \{ l_1 x + l_2 \beta : l_1 + l_2 \leq n - 1 \text{ and } l_1, l_2 \in \mathbb{N} \cup \{0\} \}
\]

For \( E, F \subset \mathbb{R}^2 \) and \( \lambda \in \mathbb{R} \), we define

\[
\lambda E = \{ \lambda e : e \in E \}, \quad E + F = \{ e + f : e \in E, f \in F \}
\]

By, \(^{15}\) there is a unique non-empty compact subset \( K(n) \) of \( \mathbb{R}^2 \) satisfying the following set equation:

\[
K(n) = [K(n) + D(n)]/n
\]

We shall call \( K(n) \) a general Sierpinski triangle throughout this paper. A familiar example of a general Sierpinski triangle is the Sierpinski gasket, in which \( n = 2, \ D(2) = \{(0, 0), (1, 0), (1/2, \sqrt{3}/2)\} \). Recently, evolving networks are usually modeled by the general Sierpinski triangles. For example, J. Guan et al.\(^{16}\) and A. Le et al.\(^{17}\) used the classical Sierpinski triangle to construct evolving networks.

We define \( \Gamma(n, 0) = \Delta, \Gamma(n, 1) = [\Delta + D(n)]/n \), and recurrently

\[
\Gamma(n, k + 1) = [\Gamma(n, k) + D(n)]/n
\]

for \( k \geq 1 \). Then \( \Gamma(n, 0) \) is an equilateral triangle with side 1, \( \Gamma(n, k)(k \geq 1) \) is a union of smaller equilateral triangles with side \( 1/n^k \) and converges to \( K(n) \) as \( k \to \infty \) in the Hausdorff metric.\(^{15}\) We call \( \Gamma(n, k)(k \geq 0) \) the \( k \)-th approximating graph of general Sierpinski triangle \( K(n) \), in which vertex set and edge set are considered in a usually way (see Figure 1 for \( n = 2 \) and \( k = 0, 1, 2 \)).

Another interesting motivation of our work comes from the theory of analysis on fractals. J. Kigami constructed in\(^{18}\) a Laplacian on the Sierpinski gasket, and later extended their method to the p.c.f. (post-critically finite) fractals. Kigami’s idea is to approximate the Laplacian on \( K(n) \) by the normalized Laplacian on \( \Gamma(n, k) \). This coincide with the known graph Laplacian whose vertex set is \( V(n, k) \), where \( V(n, k) \) is described as in ‘Some basic structure properties and concepts on approximating graph’ section.

There has been notable progress on the study of the spectrum of normalized Laplacian matrix associated to fractal, especially in recent years. Yet much is still unknown, and in this direction the existed works concentrate mainly on the theoretical study base on a recursive method (or decimation method\(^{19}\)) as we know of. For example, A. Julaiti et al. in\(^{20}\) studied the spectra of normalized Laplacian matrices for a family of proposed fractal trees and Cayley trees. M. Dai et al. in\(^{21}\) presented a study on the spectrum of the normalized Laplacian of weighted tree-like fractals. In this paper we will obtain the approximating graphs \( \Gamma(n, k) \) and their normalized Laplacian spectra from an algorithmic point of view, which extends the work of B. Rajan et al.\(^{22}\) Our method is completely different from the decimation method. Our algorithm need to code the vertices of each approximating graph \( \Gamma(n, k) \) base on the theory of iterated function system (IFS),\(^{15}\) we call it an IFS coding algorithm throughout this paper. This is of great relevance in studying the structure and some properties of general Sierpinski triangles and related graphs, and dynamical aspects of related networks in a new perspective.

This paper is organized as follows. The next section is preliminaries. We introduce the geometrical construction, normalized Laplacian matrix and some basic structure properties of approximating graph \( \Gamma(n, k) \). In the subsequent section, we present our main result, called IFS coding algorithm, and illustrate our algorithm by a quasi-program of Matlab. Furthermore we also prove the correctness of this program and show the convergence of the algorithm in this section. The paper is concluded with conclusions.

**Some basic structure properties and concepts on approximating graph**

It is well known that a graph can be represented visually by drawing a dot for every vertex, and drawing a line segment between two vertices if they are connected.

![Figure 1. Approximation graphs of classical Sierpinski triangle \( \Gamma(2, k) \), \( k = 0, 1, 2 \).](image-url)
by an edge. According to,23 for any \( n \geq 2 \), we can obtain a drawing of \( \Gamma(n, k) \) with \( k \geq 1 \) as follows: We divide the equilateral triangle \( \triangle \) into \( n^2 \) congruent triangles by connecting \( n \) equipartition points of each pair of edges and replace it with the union of all the upward equilateral triangles. We can now apply this process to each of these upward equilateral triangles, and then repeat the procedure to the \( k \)-th step, to obtain a drawing of \( \Gamma(n, k) \) with \( k \geq 1 \). Obviously, for any \( k \geq 0 \), \( \Gamma(n, k) \) is an undirected graph without loops and multiple edges, i.e., a simple graph. Let \( V(n, k) \) and \( E(n, k) \) denote the vertex set and edge set of \( \Gamma(n, k) \), respectively. For any \( u, v \in V(n, k) \), if \( u \) and \( v \) are connected by an edge in \( E(n, k) \), then we denote it by \( u \sim v \) and say that they are adjacent. Write

\[
a_{u,v}(n,k) = \begin{cases} 
1, & \text{if } u \sim v, \\
0, & \text{others.}
\end{cases}
\]

We call \( A(n, k) = (a_{u,v}(n,k)) \) the adjacency matrix of \( \Gamma(n, k) \). For any \( v \in V(n, k) \), we use \( d_i(n,k) \) to denote the number of edges which touch \( v \), and call it the degree of \( v \). Let \( D(n, k) \) denote the diagonal matrix in which the \( (v, v) \)-th entry is \( d_i(n,k) \). We call \( L(n, k) := D(n, k) - A(n, k) \) the Laplacian matrix of \( \Gamma(n, k) \). For the matrix \( A \), let \( A^{-1} \) denote the inverse of \( A \) and \( \sqrt{A} \) raise each element of \( A \) to the power of half. For random walk on \( \Gamma(n, k) \), the Markov matrix (or probability transition matrix) is defined as

\[
M(n, k) := D(n, k)^{-1}A(n, k),
\]

which can be normalized to obtain the following real and symmetric matrix

\[
P(n, k) := (\sqrt{D(n, k)})^{-1}A(n, k)(\sqrt{D(n, k)})^{-1} = \sqrt{D(n, k)}M(n, k)(\sqrt{D(n, k)})^{-1}
\]

We see from the definition that the \( (u, v) \)-th entry of \( P(n, k) \) is

\[
p_{u,v}(n,k) = \frac{a_{u,v}(n,k)}{\sqrt{d_u(n,k)d_v(n,k)}}
\]

Furthermore, \( M(n, k) \) and \( P(n, k) \) have the same spectrum since \( M(n, k) \) and \( P(n, k) \) are similar.

**Definition 2.1.** Let \( I \) be a unit matrix with the same size as \( P(n, k) \). We define the normalized Laplacian matrix of \( \Gamma(n, k) \) by

\[
\mathcal{L}(n, k) = I - P(n, k) = I - \sqrt{D(n, k)}M(n, k)(\sqrt{D(n, k)})^{-1}
\]

**Example 2.1.** Let \( n = 2 \) and \( k = 0 \). Then

\[
A(2,0) = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}, \quad D(2,0) = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix},
\]

\[
M(2,0) = (D(2,0))^{-1}A(2,0) = \begin{pmatrix} 0 & 1/2 & 1/2 \\ 1/2 & 0 & 1/2 \\ 1/2 & 1/2 & 0 \end{pmatrix},
\]

\[
L(2,0) = D(2,0) - A(2,0) = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}
\]

and

\[
\mathcal{L}(2,0) = I - \sqrt{D(2,0)}A(2,0)(\sqrt{D(2,0)})^{-1} = \begin{pmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1 & -1/2 \\ -1/2 & -1/2 & 1 \end{pmatrix}
\]

are the adjacency matrix, diagonal matrix, Markov matrix, Laplacian matrix and normalized Laplacian matrix of \( \Gamma(2,0) \), respectively.

Let \( G = (V(G), E(G)) \) be a simple graph with vertex set \( V(G) \) and edge set \( E(G) \), and \( H = (V(\hat{G}), E(\hat{H})) \) be another simple graph. We say that \( G \) and \( H \) are isomorphic, and denote it by \( G \cong H \), if there exists a bijection \( f : V(G) \to V(H) \) such that

\[
u \sim v \text{ if and only if } f(u) \sim f(v), \quad u, v \in V(G)
\]

We say that \( H \) is a subgraph of \( G \) if \( V(H) \subseteq V(G) \) and \( E(H) \subseteq E(G) \). Let \( |V| \) denote the cardinality of a set \( A \). For given integer \( n \geq 2 \), we see from the geometrical construction of \( \Gamma(n, k) \) that for any \( k \geq 1 \), \( \Gamma(n, k) \) consists of \( D(n) \) subgraphs that are isomorphic to \( \Gamma(n, k-1) \). We denote by \( \Gamma^1(n,k), \Gamma^2(n,k) \ldots, \Gamma^{D(n)}(n,k) \) these subgraphs of \( \Gamma(n,k) \), respectively. For \( n = 2 \), \( 3 \), and let \( V^1(n,k), V^2(n,k) \ldots, V^{D(n)}(n,k) \) denote the corresponding vertex sets, respectively. Note that if \( V^i(n,k) \cap V^j(n,k) \neq \emptyset \) with \( i \neq j \), then \( V^i(n,k) \cap V^j(n,k) \) is a singleton, which is unique common vertex of small triangles \( \Gamma^i(n,1) \) and \( \Gamma^j(n,1) \). It is easy to check that \( \Gamma(n,k) \) has the following obviously structure properties.

**Proposition 2.2.** For \( n \geq 2 \),

\[
\langle D(n) \rangle = \frac{n(n+1)}{2}.
\]

**Proposition 2.3.** For \( n \geq 2 \) and \( k \geq 0 \), let \( V(n,k) \) and \( d_i(n,k) \) be defined as above. Then for any \( v \in V(n,k) \), \( d_i(n,k) \in \{2, 4, 6\} \).
Proposition 2.4. For \( n \geq 2 \), let \( d(n, m) \) denote the total number of vertices with degree \( m \) in \( \Gamma(n, 1) \), \( m = 2, 4 \) and 6. Then
\[
\begin{align*}
  d(n, 6) &= \begin{cases} 
    0, & n = 2; \\
    1, & n = 3; \\
    3, & n = 4; \\
    V(n - 3, 1), & n \geq 5.
  \end{cases}
\]
and for \( n \geq 2 \), \( d(n, 2) = 3 \) and \( d(n, 4) = V(n, 1) - d(n, 6) - 3 \).

Proposition 2.5. Let \( n \geq 2 \) and \( k \geq 1 \). Then
\[
V(n, k) = (\mathcal{D}(n))(V(n, k - 1)) - d(n, 4) - 2d(n, 6),
\]
and \( E(n, k) = 3(\mathcal{D}(n))^k \).

Example 2.2. Let \( n = 2 \). Then \( d(n, 4) = 3 \), \( d(n, 6) = 0 \), \( V(2, k) = (3^{k+1} + 3)/2 \) and \( E(2, k) = 3^{k+1} \) for \( k \geq 0 \). See Figure 3 for \( n = 2, k = 0, 1, 2, 3 \).

Proof. For \( k \geq 1 \), we have from the Propositions 2.2, 2.4 and 2.5 that
\[
V(2, k) = (\mathcal{D}(2))(V(2, k - 1)) - d(2, 4) - 2d(2, 6)
= 3V(2, k - 1) - 3
= 3(3V(2, k - 2) - 3) - 3 \cdots
= 3^k V(2, 0) - 3^k - 3^{k-1} - \cdots - 3^2 - 3
= 3^{k+1} + \frac{3(1 - 3^{k})}{2} = \frac{3^{k+1} + 3}{2}.
\]

Combing the fact that \( V(2, 0) = 3 \), we have \( V(2, k) = (3^{k+1} + 3)/2 \) for \( k \geq 0 \). Using Proposition 2.5 again and the fact \( \mathcal{D}(2) = 3 \), we get that \( E(2, k) = 3^{k+1} \) for \( k \geq 0 \). This proof is completed. \( \square \)

Similarly, we have

Example 2.3. Let \( n = 3 \). Then \( d(3, 4) = 6 \), \( d(3, 6) = 1 \), and \( V(3, k) = (7 \times 6^k + 4)/5 \) and \( E(3, k) = 3 \times 6^k \) for \( k \geq 0 \). See Figure 4 for \( n = 3, k = 1, 2 \).

IFS coding algorithm and a quasi-program of Matlab

Our aim in this paper is to describe completely the normalized Laplacian spectrum of \( \Gamma(n, k) \) for any \( k \geq 0 \). In view of Definition 2.1, we first need to find the adjacency matrix \( A(n, k) \) and the diagonal matrix \( D(n, k) \) of \( \Gamma(n, k) \), which is the most important and most central part. However writing down manually the adjacency matrix and diagonal matrix is unimaginable when \( n \) and \( k \) are large enough. So we in this section devise an algorithm base on the iterative generation process of \( \Gamma(n, k) \), called IFS coding algorithm, to code the vertices of \( \Gamma(n, k) \), and thus from the adjacency matrix \( A(n, 0) \) (resp. the diagonal matrix \( D(n, 0) \)) find the adjacency matrix \( A(n, k) \) (resp. the diagonal matrix \( D(n, k) \)) by the recursive relation between \( A(n, k - 1) \) (resp. \( D(n, k - 1) \)) and \( A(n, k) \) (resp. \( D(n, k) \)).

Let \( V^j(n, k), j = 1, \ldots, \mathcal{D}(n) \), be described as in ‘Some basic structure properties and concepts on approximating graph’ section. Denote
\[
\begin{align*}
  v_1 &= V^1(n, k) \cap V^2(n, k), \quad v_2 = V^1(n, k) \cap V^3(n, k), \\
  v_{n(n + 1)} &= V^{n(n + 1)}(n, k) \cap V^{n(n + 1)}(n, k),
\end{align*}
\]
\[
\begin{align*}
  v_i(n + 1) &= V^{i(n + 1)}(n, k) \cap V^{i(n + 1)}(n, k),
\end{align*}
\]
\[
\begin{align*}
  v_i(l + 1) &= V^{i(l + 1)}(n, k) \cap V^{i(l + 1)}(n, k),
\end{align*}
\]
\[
\begin{align*}
  v_i(l + 1) &= V^{i(l + 1)}(n, k) \cap V^{i(l + 1)}(n, k),
\end{align*}
\]

Figure 2. \( \Gamma(n, 1) \) and their subgraphs \( \Gamma^i(n, 1) \). (a) \( n = 2 \) and (b) \( n = 3 \).
\[
\frac{v_i(l+1)}{2} + i = V^{ \frac{\phi(l)}{2} + i}(n,k) \cap V^{ \frac{\phi(l)}{2} + i + 1}(n,k)
\]
where \(l = 2, \ldots, n-1\) (see Figure 2 for \(n = 2, 3\) and \(k = 1\)).

**Step 1:** Code the vertices of \(\Gamma(n,0) = \Delta\) as shown in Figure 3(a).

**Step 2:** For any \(k \geq 1\), code the vertices of \(\Gamma^i(n,k)\) as that in \(\Gamma(n,k - 1)\). By Proposition 2.5, the code of \(v_2\) in
\( \Gamma(n, k) \) is \( V(n, k - 1) \). We define \( D(1) = 1, d(1, 4) = 0 \) and \( d(1, 6) = 0 \). Note that the uncoded vertices of \( \Gamma(n, k) \) induce a graph isomorphic to \( \Gamma^1(n, k - \{1\}) \) for \( i \in \{D(m) : m = 2, \ldots, n\} \), the uncoded vertices of \( \Gamma(n, k) - \{v_{i+m}\} \) induce a graph isomorphic to \( \Gamma^1(n, k) - \{1, v_2\} \) for \( m = 2, \ldots, n - 1, i = D(m - 1) + 1, \ldots, D(m) - 1, \) and the uncoded vertices of \( \Gamma^1(n, k) - \{v_{i+m-1}\} \) induce a graph isomorphic to \( \Gamma^1(n, k) - \{1, v_2\} \) for \( i = D(n - 1) + 1, \ldots, D(n) - 1. \) Combing Propositions 2.4 and 2.5, for each vertex \( j \) in \( \Gamma^1(n, k) - \{1\} \), we code the corresponding vertex of \( \Gamma^0(n, k) - \{v_{D(m)-1}\} \) by

\[
j + (D(m) - 1)V(n, k - 1) - [d(m, 4) + 2d(m, 6)],
\]

where \( m = 2, \ldots, n \). For each vertex \( j \) in \( \Gamma^1(n, k) - \{1, v_2\} \), we code the corresponding vertex of \( \Gamma^0(n, k) - \{v_{D(m)-1}\} \) by

\[
j + (i - 1)V(n, k - 1) - [d(m - 1, 4) + 2d(m - 1, 6)] - [2(i - D(m - 1)) - 1],
\]

where \( m = 2, \ldots, n - 1, i = D(m - 1), \ldots, D(m) - 1 \). For each vertex \( j \) in \( \Gamma^1(n, k) - \{1, v_2\} \), we code the corresponding vertex of \( \Gamma^0(n, k) - \{v_{i+m-1}\} \) by \( j + (i - 1)V(n, k - 1) - [d(m - 1, 4) + 2d(m - 1, 6)] - [2(i - D(m - 1)) - 1] \), where \( i = D(n - 1) + 1, \ldots, D(n) - 1. \) We refer to Figure 3 for \( n = 2 \) and \( k = 1, 2, 3 \), and Figure 4 for \( n = 3 \) and \( k = 1, 2. \)

**Step 3:** Input the diagonal matrix and adjacency matrix of \( \Gamma(n, 0) \).

**Step 4:** Output the diagonal matrix and adjacency matrix of \( \Gamma(n, k) \) in term of coding rule of Step 2.

**Step 5:** Compute the normalized Laplacian matrix of \( \Gamma(n, k) \) by (2.1) and output eigenvalues.

For simplify, we write

\[
a(m, n, k) = (D(m) - 1)V(n, k - 1) - [d(m, 4) + 2d(m, 6)],
\]

\[
b(i, m, n, k) = (i - 1)V(n, k - 1) - [d(m - 1, 4) + 2d(m - 1, 6)] - [2(i - D(m - 1)) - 1],
\]

where \( m = 2, \ldots, n, i = D(m - 1) + 1, \ldots, D(m) - 1 \). In the following we illustrate our algorithm by a quasi-program of Matlab, which one need to normalize it according to Matlab syntax rules before using it, see Example 3.2. We denote by the percent-sign \% an explanation.

A quasi-program of Matlab base on the IFS coding algorithm

Function Sierpinskispectrum(n,m) \% 1/n is side length of each small triangle in \( \Gamma(n, 1) \) and \( m \) is iterated depth.

\[
v_1 = 0;
\]

\[
A = \begin{pmatrix}
0 & 1 & 1 \\
1 & 0 & 1 \\
1 & 1 & 0
\end{pmatrix}; D = \begin{pmatrix}
2 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 2
\end{pmatrix};
\]

\% A and D are the adjacency matrix and diagonal matrix of \( \Gamma(n, 0) = \Delta \), respectively.
\% Output the adjacency matrix and diagonal matrix of \( \Gamma(n, m - 1) \).
If \( m = 1 \)
\( A; D; \)
else
for \( i = 2 : m \)
\( A = \text{Adjmatrix}(i, A); D = \text{Diagmatrix}(i, D); \)
end
end
\% Output the eigenvalues of normalized Laplacian matrix of \( \Gamma(n, m - 1) \).
spectrum = eig(eye(size(A)) - inv(D^{1/2}) * A * inv(D^{1/2}));
Function \( A = \text{Adjmatrix}(k, A) \)
\% Determine the codes of \( v_1, \ldots, v_{(a^2 + 3n - 4)/2} \) which are described as in (1).
for $j = 1 : k - 2$

$v_1 = v_1 + \mathcal{D}(n - 1)(V(n,j - 1)) - [(d(n - 1,4) + 2d(n - 1,6)] - 1; \ 

end

for $i = 2 : n - 1$

for $j = (\mathcal{D}(i - 1) + 1) : (\mathcal{D}(i) - 1)$

$v_{j+i-1} = v_1 + b(j,i,n,k - 1), \ 

end

$v_{\mathcal{D}(i)+i-1} = v_1 + a(i,n,k - 1); \ 

end

for $j = 3 : n$

$v_{\mathcal{D}(j)-1} = v_2 + a(j - 1,n,k - 1) \ 

end

for $i = (\mathcal{D}(n - 1) + 1) : (\mathcal{D}(n) - 2)$

$v_{i+n-1} = v_1 + b(i + 1,n,n,k - 1); \ 

end

$v_{\mathcal{D}(n)+n-2} = v_1 + a(n,n,k - 1); \ 

% Output the adjacency matrix of $\Gamma(n,k - 1).$

for $i = 2 : n$

for $j = \mathcal{D}(i - 1) + 1 : (\mathcal{D}(i) - 1)$

$A = \text{blkdiag}(A, A(2 : v_2 - 1; 2 : v_2 - 1)); \ 

end

$A = \text{blkdiag}(A, A(2 : v_2; 2 : v_2)); \ 

end

$A(v_1, 2 + b(2,2,n,k - 1)) = 1; A(2 + b(2,2,n,k - 1),v_1) = 1; \ 

A(v_1, 3 + b(2,2,n,k - 1)) = 1; A(3 + b(2,2,n,k - 1),v_1) = 1; \ 

for $i = 2 : n - 1$

$j = \mathcal{D}(i - 1) + 1;$

$A\left(v_{j+i-1}, 2 + b(\mathcal{D}(i) + 1,i + 1,n,k - 1)\right) = 1;$

$A\left(2 + b(\mathcal{D}(i) + 1,i + 1,n,k - 1), v_{j+i-1}\right) = 1;$
\[ A \left( v_{i+1}, 3 + b(D(i) + 1, i + 1, n, k - 1) \right) = 1; \]
\[ A \left( 3 + b(D(i) + 1, i + 1, n, k - 1), v_{i+1} \right) = 1; \]

end for \( i = 2 \) to \( n - 1 \)
for \( j = 1 \) to \( i - 1 \):

\[ A \left( v_{i+1}, 2 + b \left( \frac{i(i+1)}{2} + j + 1, i + 1, n, k - 1 \right) \right) = 1; \]
\[ A \left( 2 + b \left( \frac{i(i+1)}{2} + j + 1, i + 1, n, k - 1 \right), v_{i+1} \right) = 1; \]
\[ A \left( v_{i+1}, 3 + b \left( \frac{i(i+1)}{2} + j + 1, i + 1, n, k - 1 \right) \right) = 1; \]
\[ A \left( 3 + b \left( \frac{i(i+1)}{2} + j + 1, i + 1, n, k - 1 \right), v_{i+1} \right) = 1; \]

if \( k = 2 \)

\[ A \left( v_{i+1}, D(i) - (i - j) \right) = 1; \]
\[ A \left( D(i) - (i - j), v_{i+1} \right) = 1; \]
\[ A \left( v_{i+1}, D(i + 1) - (i + 1 - j) \right) = 1; \]
\[ A \left( D(i + 1) - (i + 1 - j), v_{i+1} \right) = 1; \]

else

\[ A \left( v_{i+1}, V(n - 1, 1) + b \left( \frac{i(i+1)}{2} + j, i, n, k - 1 \right) \right) = 1; \]
\[ A \left( V(n - 1, 1) + b \left( \frac{i(i+1)}{2} + j, i, n, k - 1 \right), v_{i+1} \right) = 1; \]
\[ A \left( v_{i+1}, V(n, 1) - 1 + b \left( \frac{i(i+1)}{2} + j, i, n, k - 1 \right) \right) = 1; \]
\[ A \left( V(n, 1) - 1 + b \left( \frac{i(i+1)}{2} + j, i, n, k - 1 \right), v_{i+1} \right) = 1; \]

end

for \( i = 2 \) to \( n \)

\[ A \left( v_{D(i)-1}, 2 + a(i, n, k - 1) \right) = 1; \]
\[ A \left( 2 + a(i, n, k - 1), v_{D(i)-1} \right) = 1; \]
\[ A \left( v_{D(i)-1}, 3 + a(i, n, k - 1) \right) = 1; \]
\[ A \left( 3 + a(i, n, k - 1), v_{D(i)-1} \right) = 1; \]

end

for \( i = 0 \) to \( n - 2 \) % We define \( V(1, 1) = 3 \)

if \( k = 2 \)

\[ A \left( v_{D(i)+1}, D(n) - (n - i - 1) \right) = 1; \]
\[ A \left( D(n) - (n - i - 1), v_{D(i)+1} \right) = 1; \]
\[ A \left( v_{D(i)+1}, D(n + 1) - (n - i) \right) = 1; \]
\[ A \left( D(n + 1) - (n - i), v_{D(i)+1} \right) = 1; \]

else


\begin{align*}
A \left( V(n - 1, 1) + b \left( \frac{n(n - 1)}{2} + i + 1, n, n, k - 1 \right) \right) &= 1; \\
A \left( V(n - 1, 1) + b \left( \frac{n(n - 1)}{2} + i + 1, n, n, k - 1 \right), V_{\frac{n(n-1)}{2}+i+1} \right) &= 1; \\
A \left( V(n, 1) - 1 + b \left( \frac{n(n - 1)}{2} + i + 1, n, n, k - 1 \right) \right) &= 1; \\
A \left( V(n, 1) - 1 + b \left( \frac{n(n - 1)}{2} + i + 1, n, n, k - 1 \right), V_{\frac{n(n-1)}{2}+i+1} \right) &= 1;
\end{align*}

end

Function \( D = \text{Diagmatrix}(k, A) \) \% Output the diagonal matrix of \( \Gamma(n, k - 1) \).
% Using the same process as that of Function \text{Adjmatrix}(k, A)\), we can determine the codes of \( v_1, \ldots, v_{(n^2 + 3n - 4)/2} \), we omit details.

for \( i = 2 : n \)
for \( j = D(i - 1) + 1 : (D(i) - 1) \)
\( D(\text{blkdiag}(D, D(2 : v_2 - 1; 2 : v_2 - 1))) \);
end
\( D = \text{blkdiag}(D, D(2 : v_2; 2 : v_2)) \);
end

\( D(v_1, v_1) = 4 \)

for \( i = 2 : n - 1 \)
\( j = D(i - 1) + 1; \)
\( D(v_{j+1-i}, v_{j+i-1}) = 4; \)
end

for \( i = 2 : n - 1 \)
for \( j = 1 : i - 1; \)
\( D\left( V_{\frac{i(i+1)}{2}}, V_{\frac{i(i+1)}{2}+j} \right) = 6; \)
end
end
for \( i = 2 : n \)
\( D(\text{blk}(i-1), \text{blk}(i-1)) = 4; \)
end

for \( i = 0 : n - 2 \)
\( D\left( V_{\frac{(n+1)(n+2)}{2}}, V_{\frac{(n+1)(n+2)}{2}+i} \right) = 4; \)
end

Using the induction on iterated depth \( m \) for given integer \( n \geq 2 \), we can give the proof of correctness of above program as follows.
**Proof of correctness of program:** Without loss of generality, we show that the program is correct for the adjacency matrix. It is easy to see that this is true when \( m = 1, 2 \). Assume it is true for the case \( m - 1 \) with \( m \geq 3 \). Consider the case \( m \). Let \( A \) be the adjacency matrix of \( \Gamma(n,m - 1) \). Then the adjacency matrix of \( \Gamma^1(n,m) \) is \( A \) since \( \Gamma^1(n,m) \) is isomorphic to \( \Gamma(n,m - 1) \). By **Step 2**, the uncoded vertices of \( \Gamma^1(n,m) \) induce a graph isomorphic to \( \Gamma^1(n,m) - \{v_i\} \) for \( i \in \{D(l) : l = 2, \ldots, n\} \), the uncoded vertices of \( \Gamma^1(n,m) - \{v_i\} \) induce a graph isomorphic to \( \Gamma^1(n,m) - \{v_i, v_{i+1}\} \) for \( l = 2, \ldots, n - 1, i = D(l - 1) + 1, \ldots, D(l) - 1 \), and the uncoded vertices of \( \Gamma^1(n,m) - \{v_i\} \) induce a graph isomorphic to \( \Gamma^1(n,m) - \{v_i, v_{i+1}\} \) for \( i = D(n - 1) + 1, \ldots, D(n) - 1 \). We can get that the adjacency matrix \( A_i \) of \( \Gamma^1(n,m) \) by deleting the first row and column of \( A \) for \( i \in \{D(l) : l = 2, \ldots, n\} \), and the adjacency matrix \( A_i \) of \( \Gamma^1(n,m) \) by deleting the first row and column and the last row and column of \( A \) for \( l = 2, \ldots, n, i = D(l - 1) + 1, \ldots, D(l) - 1 \). Using the function ‘blkdiag’ of Matlab, we obtain a matrix with the form

\[
\begin{pmatrix}
A & * & * & * & * \\
* & A_2 & * & * & * \\
* & * & A_3 & * & * \\
* & * & * & \ddots & * \\
* & * & * & * & A_{D(n)}
\end{pmatrix}
\]

(2)

where the elements are denoted by * to be accounted for as follows(such as Example 3.1). The elements in \( A, A_2, \ldots, A_{D(n)} \) are due to all the edges except \( n^2 + 3n - 4 + 2d(n,6) \) edges (see 16 red edges shown in Figure 5 for \( n = 3 \)). The elements corresponding to these \( n^2 + 3n - 4 + 2d(n,6) \) edges in * are assigned 1 by the statements below the statement “\( A = \text{blkdiag}(A, A(2: v_2; 2: v_2)) \)” in function Adjmatrix\((k, A)\) and the others in * are 0. This completes this proof.

**Example 3.1.** Let \( n = 2 \) and \( m = 1 \). From Example 2.1, we see that

\[
A = \begin{pmatrix}
0 & 1 & 1 \\
1 & 0 & 1 \\
1 & 1 & 0
\end{pmatrix}
\]

is the adjacency matrix of \( \Gamma(2,0) \). Then the matrix (2) is

\[
\begin{pmatrix}
A & * & * \\
* & A_2 & * \\
* & * & A_3
\end{pmatrix} = \begin{pmatrix}
0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
1 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0
\end{pmatrix}
\]

which is the adjacency matrix of \( \Gamma(2,1) \).

**Example 3.2.** See Figure 6, a Matlab program for \( n = 2 \). Input Sirpinskispectrum\((3)\), we can obtain the adjacency matrix, diagonal matrix, normalized Laplacian matrix (see Figure 7), and the spectrum of normalized Laplacian matrix of \( \Gamma(2,2) \):

\[
0.0141, 0.1743, 0.2338, 0.7500, 0.7500, 0.8011, 1.0679, 1.0757, 1.2500, 1.3831, 1.5000, 1.5000, 1.5000, 1.5000, 1.5000
\]

Recall that \( V(n,k) \) and \( \mathcal{L}(n,k) \) denote the vertex set and the normalized Laplacian matrix of the approximating graph \( \Gamma(n,k) \) with \( k \geq 0 \), respectively. Fix \( n \geq 2 \). It is well known that for any \( k \geq 1 \), the normalized Laplacian matrix \( \mathcal{L}(n,k) \) can be viewed as an operator (called it normalized Laplacian) on the space of functions \( f : V(n,k) \to \mathbb{R} \), see.\(^2\) Let \( -\Delta^k \) denote the normalized Laplacian corresponding to \( \mathcal{L}(n,k) \).

We remark that our algorithm is convergent. In fact, \( \bigcup_{k=0}^{\infty} V(n,k) \) is dense in \( K(n) \) in the natural topology\(^{18} \) and we have from the [24, Section 4.5.5] coding algorithm and a quasi-program (Matlab) that there exists a normalized Laplacian \( -\Delta^k \) on \( K(n) \) such that
Figure 6. A Matlab program for $n = 2$.

```matlab
function SierpinskiSpectrum(m)
    A = [1 1; 1 1; 0 0];
    D = [1 0; 0 1];
    L = A - D;
end
```

Figure 7. The adjacency matrix, diagonal matrix and normalized Laplacian matrix of $C(2, 2)$ using Matlab. (a) The adjacency matrix $A$. (b) The diagonal matrix $D$. (c) The normalized Laplacian matrix $L$. 
\[-\Delta^n f_0(x) = \lim_{k \to \infty} -\Delta^n_k(f_0|V(n,k)), \ x \in K(n),\]

where \(f_0 \in \{f: f \text{ maps } K(n) \to \mathbb{R}\}\) whose restriction on \(V(n,k)\) is \(f_0|V(n,k)\). This implies our algorithm is convergent.

**Conclusion**

We have developed an approach to compute the normalized Laplacian spectra of approximating graphs of a class of general Sierpinski triangles in this paper. It’s easy to see that we can also get the spectra of adjacency matrix, Laplacian matrix, Markov matrix, normalized Markov matrix and signless Laplacian matrix \(D(n,k) + A(n,k)^6\) using same method. Our approach could be also used to obtain the similar approximating graphs associated with some fractals generated by iterated function system and their various spectra, such as, when one replace the \(\Delta\) in this paper by a regular tetrahedron in \(\mathbb{R}^3\). Future work should include studying the various topological properties of these approximating graphs like connectedness, toughness, etc; determining the spectra of networks associated with general Sierpinski triangle, and measuring the related structural properties and dynamical processes, specifically in relation to random walk.

**Acknowledgments**

The author would like to thank the referees for carefully reading the paper and making several helpful suggestions which led to the improvement of the manuscript. Part of this work was carried out when the author was visiting the Department of Mathematical Sciences of Georgia Southern University. He thanks the Department for its hospitality and support during his visit. The author would also like to thank Liuna Dang, Xiaojiu LU Fu Yalu, Wang Junpeng and Cui Yachao for some valuable discussions.

**Declaration of Conflicting Interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The research was supported by Shaanxi Natural Science Foundation of China (Grant No. 2017JM1018), State Foundation for Studying Abroad (CSC Scholarship No. 201706305020) and Undergraduate Training Programs for Innovation and Entrepreneurship (Grant No. S201710712015).

**ORCID iD**

Zhiyong Zhu https://orcid.org/0000-0001-6885-2478

**References**

1. Biggs NL. *Algebraic graph theory*. 2nd ed. Cambridge: Cambridge University Press, 1993.
2. Chung FRK. *Spectral graph theory*. Providence, RI: American Mathematical Society, 1997.
3. Cvetkovic DM, Doob M and Sachs H. *Spectra of graphs, theory and application*. Cambridge, MA: Academic Press, 1980.
4. Cvetkovic DM, Doob M, Gutman I, et al. *Recent results in the theory of graph spectra*. North Holland, Amsterdam: Elsevier Science Publishers B.V., 1988.
5. Merris R. Laplacian matrices of graphs: a survey. *Linear Algebra Appl* 1994; 197–198: 143–176.
6. Mohar B. The Laplacian spectrum of graphs. *Graph Theor Combin Appl* 1998; 2: 871–898.
7. Chen H and Zhang F. Resistance distance and the normalized Laplacian spectrum. *Discrete Appl Math* 2007; 155: 654–661.
8. Zhang ZZ, Shan T and Chen GR. Random walks on weighted networks. *Phys Rev E Stat Nonlin Soft Matter Phys* 2013; 87: 012112.
9. Lovász L. Random walks on graphs: a survey. In: Miklós D, Sós VT and Szőnyi T (eds) *Combinatorics, Paul Erdős is eighty*. Budapest: János Bolyai Mathematical Society, 1993, pp.1–46.
10. Xie P, Zhang Z and Comellas F. On the spectrum of the normalized Laplacian of iterated triangulations of graphs. *Appl Math Comput* 2016; 273: 1123–1129.
11. Bar-Haim A, Klafter J and Kopelman R. Dendrimers as controlled artificial energy antennae. *J Am Chem Soc* 1997; 119: 6197–6198.
12. Blumen A and Zumofen G. Energy transfer as a random walk on regular lattices. *J Chem Phys* 1981; 75: 892–907.
13. Kim SK. Mean first passage time for a random walker and its application to chemical kinetics. *J Chem Phys* 1958; 28: 1057–1067.
14. Weiss GH. First passage time problems in chemical physics. *Adv Chem Phys* 1967; 13: 1–18.
15. Hutchinson JE. Fractals and self-similarity. *Indiana Univ Math J* 1981; 30: 713–747.
16. Guan J, Wu Y, Zhang Z, et al. A unified model for Sierpinski networks with scale-free scaling and small-world effect. *Phys A* 2009; 388: 2571–2578.
17. Le A, Gao F, Xi L, et al. Complex networks modelled on the Sierpinski gasket. *Phys A* 2015; 436: 646–657.
18. Kigami J. *Analysis on fractals, Cambridge tracts in mathematics*. Vol. 143. Cambridge: Cambridge University Press, 2001.
19. Domany E, Alexander S, Bensimon D, et al. Solutions to the Schrödinger equation on some fractal lattices. *Phys Rev B* 1983; 28: 3110–3123.

20. Julaiti A, Wu B and Zhang ZZ. Eigenvalues of normalized Laplacian matrices of fractal trees and dendrimers: analytical results and applications. *J Chem Phys* 2013; 138: 204116.

21. Dai M, Chen Y, Wang X, et al. Spectral analysis for weighted tree-like fractals. *Phys A* 2018; 492: 1892–1900.

22. Rajan B, Rajasingh I, Stephen S, et al. Spectrum of Sierpinski triangles using MATLAB. In: *2nd international conference on digital information and communication technology and it’s applications (DICTAP)*, Bangkok, Thailand, 16-18 May 2012, pp. 400–403. Piscataway NJ: IEEE.

23. David D and Semmes S. *Fractured fractals and broken dreams: self-similar geometry through metric and measure*. Oxford: Oxford University Press, 1997.

24. Aaron S, Conn Z, Strichartz R, et al. Hodge-deRham theory on fractal graphs and fractals. *CPAA* 2013; 13: 903–928.