

## Research Article

Jing Su, Xiaomin Wang, Mingjun Zhang\*, and Bing Yao

# Eigenvalues of transition weight matrix for a family of weighted networks

<https://doi.org/10.1515/math-2022-0464>

received November 12, 2021; accepted May 16, 2022

**Abstract:** In this article, we design a family of scale-free networks and study its random target access time and weighted spanning trees through the eigenvalues of transition weight matrix. First, we build a type of fractal network with a weight factor  $r$  and a parameter  $m$ . Then, we obtain all the eigenvalues of its transition weight matrix by revealing the recursive relationship between eigenvalues in every two consecutive time steps and obtain the multiplicities corresponding to these eigenvalues. Furthermore, we provide a closed-form expression of the random target access time for the network studied. The obtained results show that the random target access is not affected by the weight; it is only affected by parameters  $m$  and  $t$ . Finally, we also enumerate the weighted spanning trees of the studied networks through the obtained eigenvalues.

**Keywords:** Laplacian eigenvalue, weighted network, random walk, random target access time, weighted spanning tree

**MSC 2020:** 05C81

## 1 Introduction

In the research related to complex networks, in addition to the topological properties, it is also necessary to describe their dynamic processes. Random walk is one of the powerful research tools that can describe various dynamical behaviors in complex systems, such as remote sensing image segmentation [1], epidemic spreading [2], community detecting [3], cell sampling [4], just to name but a few. Random walks have been studied for many decades on both regular lattices and networks with a variety of structures [5]. For undirected weighted networks, the weight matrices of several classes of local random walk dynamics have been extensively studied, including normal random walks, biased random walks and preferential navigation, random walks in the context of digital image processing and maximum entropy random walks, and non-local random walks, including applications in the context of human mobility [6].

In the research methods of random walks, the random target access time [7], mean first passage time [8], mixing time, and other indicators [9] can be used to measure the efficiency of propagation [10,11]. The random target access time is a manifestation of the global characteristics of the network; it refers to the average of mean first passage time from one node to another node over all target nodes according to the stationary state; it is formulated as the sum of reciprocals of every nonzero eigenvalue of the normalized

---

\* Corresponding author: Mingjun Zhang, China Northwest Center of Financial Research, Lanzhou University of Finance and Economics, Lanzhou 730020, China; School of Information Engineering, Lanzhou University of Finance and Economics, Lanzhou 730020, China; Key Laboratory of E-Business Technology and Application, Lanzhou 730020, China, e-mail: zhangmjz@163.com

Jing Su, Xiaomin Wang: School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China; Key Laboratory of High Confidence Software Technologies, Peking University, Beijing 100871, China

Bing Yao: College of Mathematics and Statistics, Northwest Normal University, Lanzhou, 730070, China

Laplacian matrix for the researched network. For example, some related studies on biased random walks of non-fractal and fractal structure [12].

The eigenvalues and eigenvectors of the transition matrix play a very vital role [13], as they are closely related to determining the aforementioned measures. There has been some research related to the eigenvalues of transition matrices on common networks in this work, such as Sierpinski gaskets and extended T-fractals, polymer networks, and it is worth emphasizing that these studies are based on networks without weight [14–16]. However, the weight of the network has important research significance in biological neural network, railway and air transportation [17,18]; therefore, finding the relationship between random walks and weight factor on different weighted networks is meaningful. Zhang et al. [19] studied the spectra of a weighted scale-free network and proved that random target access time  $H_t$  is closely related to a weight factor  $r$ . Dai et al. [20] obtained  $H_t = N_t^{\ln(4r+4)/\ln 4}$  for a class of weighted scale-free triangulation networks. Zou et al. [21] proved the  $H_t \sim N_t^{\log_2(1+4r)}$  for  $r > 1$  of weighted network with two hub nodes. These studies show that weight has a great influence on random target access time of the networks.

The main research content of this article is arranged as follows. In the next section, we propose a graphic operation called the weighted edge multi-division operation and give the generation algorithm about a family of weighted networks. In Section 3, we find all eigenvalues of the transition weight matrix and determine their multiplicities. Then, we deduce all the eigenvalues of the normalized Laplacian matrix of our networks, and then derive the closed-form expressions for random target access time and enumerate the weighted spanning trees in Section 4. In Section 5, we summarize the work of this article.

## 2 Design-weighted network models

Many phenomena can be reduced to graphs for research [22–24], so we are about to show the corresponding model of a family of weighted scale-free networks. But, before that, we define a graphic operation called the weighted edge multi-division operation and obtain our weighted networks through the recursion of this graphic operation.

**Weighted edge multi-division operation:** For a given edge  $ii'$  with two end-nodes  $i$  and  $i'$  with weight  $w$ , replace the original edge  $ii'$  with  $m$  roads  $i_q j_q i'_q$  ( $q = 1, 2, \dots, m$ ); the end-nodes  $i_q$  and  $i'_q$  of these roads coincide with  $i$  and  $i'$ , respectively. The weight of the new edges  $i_q j_q$  and  $j_q i'_q$  is  $r$  times the original edge  $ii'$ , and the diagram of a weighted edge multi-division operation is shown in Figure 1.

With the preparation of the weighted edge multi-division operation, the weighted scale-free networks are constructed in an iterative way as depicted. Let  $G(m, r, t)$  ( $m \geq 2, r > 0, t \geq 0$ ) denote the weighted scale-free networks after  $t$  time steps. Initially,  $t = 0, G(m, r, 0)$  is an edge with unit weight connecting two nodes. For  $t \geq 1$ ,  $G(m, r, t)$  can be obtained from  $G(m, r, t - 1)$  by performing weighted edge multi-division operation on each existing edge in  $G(m, r, t - 1)$ . If the weight of edge in  $G(m, r, t - 1)$  is  $w(t - 1)$ , then the weight of edge in  $G(m, r, t)$  becomes  $rw(t - 1)$  according to the above graphic operation. Figure 2 illustrates the two particular examples  $G(m, r, t)$  when  $m = 3, r = 1, t = 3$  and  $m = 2, r = 1, t = 4$ , the weight of each edge in these two network examples is 1, which is omitted in Figure 2.

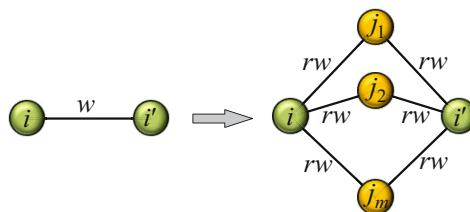


Figure 1: A weighted edge multi-division operation.

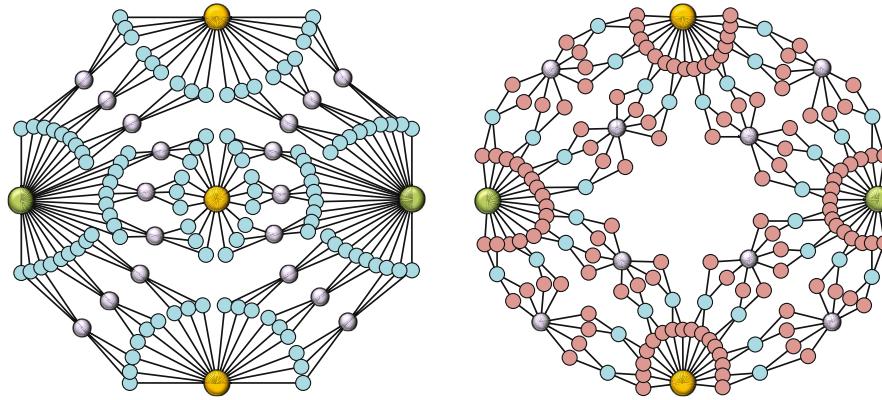


Figure 2: Illustration for two networks  $G(3, 1, 3)$  and  $G(2, 1, 4)$ .

The iterative growth of the network allows us to precisely analyze its topological characteristics, such as the order and the size, the sum of weights of all edges, and so on. At each time step  $t_i$  ( $t_i \geq 1$ ), the number of newly added nodes is recorded as  $V_{t_i} = m(2m)^{t_i-1}$ . Suppose the network order  $N_t$  represents the total number of nodes in  $G(m, r, t)$ , we have

$$N_t = V_0 + \sum_{t_i=1}^t V_{t_i} = \frac{m(2m)^t + 3m - 2}{2m - 1}. \quad (1)$$

The network size  $E_t$  is the number of total edges in  $G(m, r, t)$ , then

$$E_t = 2mE_{t-1} = (2m)^t. \quad (2)$$

Let symbol  $d_i(t)$  be the degree of node  $i$  in  $G(m, r, t)$  that was generated at time step  $t_i$ , then we can obtain a recursive formula for the degree of node as  $d_i(t+1) = md_i(t) = 2m^{t-t_i}$ .

The networks studied in this article display some representative topological characteristics observed in different real-life systems. They obey a significant power-law degree distribution with exponent  $\gamma = 2 + \ln 2 / \ln m$ , and have a fractal dimension  $f_B = \ln(2m) / \ln 2$ , both of which show that the networks are fractal and scale-free [25,26].

Let  $Q_t$  represent the total weight of all edges in  $G(m, r, t)$ , by construction, we have

$$Q_t = 2mrQ_{t-1}, \quad (3)$$

then by considering the initial condition  $Q_0 = 1$ , we can solve

$$Q_t = (2mr)^t. \quad (4)$$

For an edge  $e = ij$  connecting two nodes  $i$  and  $j$  in  $G(m, r, t)$ ,  $w_{ij}(t)$  denote the weight of edge  $ij$ , let  $s_i(t)$  be the strength of node  $i$  in time step  $t$ , which represents the sum of the weights of all adjacent edges of node  $i$ , it can be formulated as

$$s_i(t) = \sum_{j \in N(i)} w_{ij}(t) = mrs_i(t-1) = (mr)^{t-t_i} s_i(t_i), \quad (5)$$

where  $N(i)$  is the set of all neighbors of node  $i$  in the network  $G(m, r, t)$ .

### 3 Eigenvalues and their multiplicities of transition weight matrix

In this section, we determine all eigenvalues and multiplicities of transition weight matrix in our networks, according to the relationship between the eigenvalues of the transition weight matrix of two consecutive generations.

### 3.1 Relationship between eigenvalues

Let  $W_t$  be the weighted adjacency matrix of network  $G(m, r, t)$  and its entries  $W_t(i, j)$  are defined as  $W_t(i, j) = w_{ij}(t)$  if nodes  $i$  and  $j$  are connected by an edge with weight  $w_{ij}(t)$  in  $G(m, r, t)$ , or  $W_t(i, j) = 0$  if there is no edge between node  $i$  and node  $j$ . Another important matrix  $T_t$  involved in describing weighted random walks is called transition weight matrix, which is defined as  $T_t = S_t^{-1}W_t$ , where  $S_t$  is the diagonal strength matrix of network  $G(m, r, t)$  with its  $i$ th diagonal entry being the strength  $s_i(t)$  of  $i$ . Thus, the  $(i, j)$ th element of  $T_t$  is written as  $T_t(i, j) = w_{ij}(t)/s_i(t)$ , which represents the corresponding transition probability for a particle going from starting node  $i$  to ending node  $j$ . Since  $T_t$  is asymmetric, we introduce a real and symmetric matrix  $P_t$  to assist our research,

$$P_t = S_t^{-\frac{1}{2}}W_tS_t^{-\frac{1}{2}} = S_t^{\frac{1}{2}}T_tS_t^{-\frac{1}{2}}, \quad (6)$$

which shows that matrix  $P_t$  is similar to matrix  $T_t$ , and they have the same eigenvalue set. In addition, the entry in the  $i$ th row and  $j$ th column of  $P_t$  is essentially  $P_t(i, j) = w_{ij}(t)/\sqrt{s_i(t)}\sqrt{s_j(t)}$  according to equation (6). Furthermore, we set  $\phi$  as an eigenvector of matrix  $P_t$  associated with eigenvalue  $\lambda$ , then the eigenvector corresponding to eigenvalue  $\lambda$  in  $T_t$  can be written as  $S^{-\frac{1}{2}}\phi$ . Therefore, we only need to find all the eigenvalues of matrix  $P_t$ , and the eigenvalues of the another matrix  $T_t$  can be obtained through transformation. Furthermore, we introduce the normalized Laplacian matrix  $L_t = I_t - P_t$ , where  $I_t$  is the  $N_t \times N_t$  identity matrix. In the following, we use the decimation approach [27] to enumerate all eigenvalues of normalized Laplacian matrix in our network.

For  $G(m, r, t)$ , we suppose that  $\lambda_i(t)$  is an eigenvalue of  $P_t$ ,  $\phi = (\phi_1, \phi_2, \dots, \phi_{N_t})^T$ , which denote the eigenvector corresponding to the eigenvalue  $\lambda_i(t)$ , where  $\phi_i$  is the element corresponding to node  $i$ . In order to distinguish between the newly generated nodes at time step  $t$  and the existing old nodes (they were generated before time step  $t$ ), we let  $\phi^*$  represent a vector with  $N_{t-1}$  dimension that is obtained from  $\phi$  by restricting its components to the old nodes at time step  $t$ , with the set of all old nodes in  $G(m, r, t)$  is denoted as  $V(t-1)$ , then

$$\lambda_i(t)\phi = P_t\phi. \quad (7)$$

$\phi^*$  as an eigenvector of matrix  $P_{t-1}$ , it associated with the eigenvalue  $\lambda_i(t-1)$ , similar to equation (7), we can obtain

$$\lambda_i(t-1)\phi^* = P_{t-1}\phi^*. \quad (8)$$

Let  $i \in V(t-1)$  be an old node in  $G(m, r, t)$ , we also have the following equation according to equation (7),

$$\lambda_i(t)\phi_i = \sum_{j \in N(i)} P_t(i, j)\phi_j. \quad (9)$$

Given two old nodes  $i, i' \in V(t-1)$  in network  $G(m, r, t)$ , and the weight of the connecting edge between these two nodes in  $G(m, r, t-1)$  is equal to  $w_{ii'}(t-1)$ . According to the way the network grows in two consecutive time steps, let  $j_1, j_2, \dots, j_m$  be the  $m$  neighbors of nodes  $i$  and  $i'$ , then expand equation (9) in detail to obtain the following equation, let  $N_i(t)$  be the set of neighbors of node  $i$ ,

$$\lambda_i(t)\phi_i = \sum_{j_1, \dots, j_m \in N_i(t)} \left[ \frac{w_{ij_1}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_1}(t)}}\phi_{j_1} + \frac{w_{ij_2}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_2}(t)}}\phi_{j_2} + \dots + \frac{w_{ij_m}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_m}(t)}}\phi_{j_m} \right]. \quad (10)$$

For each element  $\phi_{j_q}$  ( $q = 1, 2, \dots, m$ ) corresponding to a new node, for every new neighbor  $j_q$ , we can obtain

$$\begin{cases} \lambda_i(t)\phi_{j_1} = \frac{w_{ij_1}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_1}(t)}}\phi_i + \frac{w_{i'j_1}(t)}{\sqrt{s_{i'}(t)}\sqrt{s_{j_1}(t)}}\phi_{i'} \\ \lambda_i(t)\phi_{j_2} = \frac{w_{ij_2}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_2}(t)}}\phi_i + \frac{w_{i'j_2}(t)}{\sqrt{s_{i'}(t)}\sqrt{s_{j_2}(t)}}\phi_{i'} \\ \dots \\ \lambda_i(t)\phi_{j_m} = \frac{w_{ij_m}(t)}{\sqrt{s_i(t)}\sqrt{s_{j_m}(t)}}\phi_i + \frac{w_{i'j_m}(t)}{\sqrt{s_{i'}(t)}\sqrt{s_{j_m}(t)}}\phi_{i'} \end{cases} \quad (11)$$

We know that  $w_{ij_q}(t) = w_{i'j_q}(t) = rw_{ii'}(t-1)$  for  $q = 1, 2, \dots, m$ ,  $s_i(t) = mrs_i(t-1)$ ,  $s_{i'}(t) = mrs_{i'}(t-1)$ , and  $s_{j_q}(t) = mrw_{ii'}(t-1)$ , and substituting equation (11) into equation (10), equation (10) becomes

$$\begin{aligned} \lambda_i(t)\phi_i &= \sum_{i' \in N_i(t-1)} \left[ \frac{w_{ii'}(t-1)}{2\lambda_i(t)s_i(t-1)}\phi_i + \frac{w_{ii'}(t-1)}{2\lambda_i(t)\sqrt{s_i(t-1)}\sqrt{s_{i'}(t-1)}}\phi_{i'} \right] \\ &= \frac{1}{2\lambda_i(t)} \left[ \phi_i + \sum_{i' \in N_i(t-1)} \frac{w_{ii'}(t-1)}{\sqrt{s_i(t-1)}\sqrt{s_{i'}(t-1)}}\phi_{i'} \right] \\ &= \frac{1}{2\lambda_i(t)}\phi_i + \frac{1}{2\lambda_i(t)} \sum_{i' \in N_i(t-1)} P_{t-1}(i, i')\phi_{i'}. \end{aligned} \quad (12)$$

For an old node  $i \in V(t-1)$ , we can further simplify equation (12) to obtain

$$\left\{ 2\lambda_i(t) \left[ \lambda_i(t) - \frac{1}{2\lambda_i(t)} \right] \right\} \phi_i = \sum_{i' \in N_i(t-1)} P_{t-1}(i, i')\phi_{i'}. \quad (13)$$

Comparing equations (13) and (8) corresponding to any old node  $i$ , we obtain the following equation:

$$2\lambda_i(t) \left[ \lambda_i(t) - \frac{1}{2\lambda_i(t)} \right] = \lambda_i(t-1); \quad (14)$$

furthermore, the solution of  $\lambda_i(t)$  is expressed by  $\lambda_i(t-1)$  as

$$\lambda_i(t) = \pm \sqrt{\frac{\lambda_i(t-1) + 1}{2}}. \quad (15)$$

The above result shows that for an eigenvalue  $\lambda_i(t-1)$  of matrix  $P_{t-1}$ , we can obtain two eigenvalues  $\lambda_{i,1}(t)$  and  $\lambda_{i,2}(t)$  of the matrix  $P_t$  through the recursive relationship as shown in equation (15). Therefore, if all the eigenvalues of  $P_{t-1}$  are known, then we can calculate all the eigenvalues of  $P_t$ . If there are also eigenvalues  $\lambda_i(t)$  that cannot be derived from equation (15), then they are zero eigenvalues.

### 3.2 Multiplicities of eigenvalues

On the basis of the eigenvalues obtained above, we can determine their corresponding multiplicity for matrix  $P_t$ . For convenience, let  $D_t^{\text{mul}}(\lambda_i(t))$  be the degeneracy of eigenvalue  $\lambda_i(t)$  for matrix  $P_t$ , because  $P_{t-1}$  is a real and symmetrical matrix; hence, every eigenvalue  $\lambda_i(t-1)$  of  $P_{t-1}$  has  $D_{t-1}^{\text{mul}}(\lambda_i(t-1))$  linearly independent eigenvectors.

For small networks, we can calculate their eigenvalues and corresponding multiplicities directly, for instance, the set of eigenvalues of  $P_0$  is  $\{-1, 1\}$ . For  $P_1$ , its eigenvalues are  $-1, 0, 1$ , where two eigenvalues  $-1$  and  $1$  are generated by eigenvalue  $1$  of  $P_0$ , and eigenvalue  $0$  is generated by eigenvalue  $-1$  of  $P_0$ . Similarly, we can obtain five different eigenvalues  $-1, -\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 1$  of  $P_2$  according to the eigenvalues of  $P_1$ .

We can observe the fact that in addition to the eigenvalues  $\lambda_i(t) = 0$ , all eigenvalues of a given time step  $t_i$  must exist in the subsequent time step  $t_i + 1$ , and each eigenvalue keeps the degeneracy of the previous

time step. So, for  $t \geq 2$ , each eigenvalue in network  $G(m, r, t - 1)$  will bring about two eigenvalues in  $G(m, r, t - 1)$  in the light of equation (15), except for eigenvalue  $-1$  and  $1$ , because  $-1$  can only produce eigenvalue  $0$ , and  $1$  of  $P_{t-1}$  produce  $-1$  and  $1$  in the next generation. Therefore, we need to individually consider the multiplicity of eigenvalue  $0$ , as well as the multiplicities of its descendants.

In general, we usually use  $r(M)$  to represent the rank of a matrix  $M$ , and the multiplicity of the zero eigenvalues for matrix  $P_t$  is expressed as

$$D_t^{\text{mul}}(0) = N_t - r(P_t). \quad (16)$$

For the set of all nodes in  $G(m, r, t)$ , let  $\alpha$  be the set of all nodes in  $G(m, r, t - 1)$ , and  $\beta$  denote the set of nodes newly added at time step  $t$ . Now, we identify that the rank of  $P_t$ ,  $P_t$  can be written in a block form due to the topology of the network,

$$P_t = \begin{pmatrix} P_{\alpha, \alpha} & P_{\alpha, \beta} \\ P_{\beta, \alpha} & P_{\beta, \beta} \end{pmatrix} = \begin{pmatrix} 0 & P_{\alpha, \beta} \\ P_{\beta, \alpha} & 0 \end{pmatrix}, \quad (17)$$

where the fact that  $P_{\alpha, \alpha}$  is the  $N_{t-1} \times N_{t-1}$  zero matrix and  $P_{\beta, \beta}$  is the  $(N_t - N_{t-1}) \times (N_t - N_{t-1})$  zero matrix are substituted into equation (17).

Because the matrix  $P_{\beta, \alpha}$  is the transpose of the matrix  $P_{\alpha, \beta}$ , then we only analyze the rank of the matrix  $P_{\alpha, \beta}$ ,  $P_{\alpha, \beta}$ , which is an  $N_{t-1} \times (N_t - N_{t-1})$ -order matrix, taking the form

$$P_{\alpha, \beta} = \begin{pmatrix} p_1 & \cdots & \cdots & \cdots & p_1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & p_1 & \cdots & \cdots & \cdots & p_1 \\ p_2 & \cdots & p_2 & & & & p_2 & \cdots & p_2 & \\ \vdots & & & & & & \vdots & & & \\ p_2 & \cdots & p_2 & p_2 & \cdots & p_2 & & & & \end{pmatrix}, \quad (18)$$

where the first  $(N_t - N_{t-1})/2$  entries in the first row of the matrix  $P_{\alpha, \beta}$  are  $p_1$ ; the last  $(N_t - N_{t-1})/2$  entries of the second row are  $p_1$ ; there are  $2m$  elements  $p_2$  in each row from the third row to the  $N_{t-1}$ th row; the “ $p_2 \cdots p_2$ ” in equation (18) means that it contains  $m$  entries  $p_2$ . The unmarked entries in matrix equation (18) are zeros. Performing some elementary row operations on matrix  $P_{\alpha, \beta}$  consists of adding the entries of the second row to the first row and multiplying the entries of the third row to the  $N_{t-1}$ th rows by  $-1$ , and then adding it to the first row. The result of the elementary row operations shows that the entries in the first row are all zeros, and the matrix  $P_{\alpha, \beta}$  has following form:

$$P_{\alpha, \beta} = \begin{pmatrix} 0 & \cdots & \cdots & \cdots & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & p_1 & \cdots & \cdots & \cdots & p_1 \\ p_2 & \cdots & p_2 & & & & p_2 & \cdots & p_2 & \\ \vdots & & & & & & \vdots & & & \\ p_2 & \cdots & p_2 & p_2 & \cdots & p_2 & & & & \end{pmatrix}, \quad (19)$$

from which we obtain  $r(P_{\alpha, \beta}) = N_{t-1} - 1$ , and consider  $P_{\beta, \alpha} = P_{\alpha, \beta}^T$ , then,

$$r(P_{\alpha, \beta}) = r(P_{\beta, \alpha}) = N_{t-1} - 1 = \frac{m(2m)^{t-1} + 3m - 2}{2m - 1} - 1. \quad (20)$$

In addition, we have

$$r(P_t) = r\left(\begin{bmatrix} 0 & P_{\alpha, \beta} \\ P_{\beta, \alpha} & 0 \end{bmatrix}\right) = 2N_{t-1} - 2 = \frac{2m(2m)^{t-1} + 6m - 4}{2m - 1} - 2. \quad (21)$$

Therefore, we obtain  $D_t^{\text{mul}}(0) = N_t - 2N_{t-1} + 2$ , which indicates that the multiplicity of eigenvalues zero of  $P_t$  is

$$D_t^{\text{mul}}(0) = \begin{cases} 0, & t = 0, \\ \frac{(2m)^t(m-1) + m}{2m-1}, & t \geq 1. \end{cases} \quad (22)$$

The total number of eigenvalue 0 and its descendants in  $P_t(t \geq 1)$  is denoted as  $N_t^{\text{seed}}(0)$ , then

$$N_t^{\text{seed}}(0) = \sum_{i=1}^t D_i^{\text{mul}}(0) \cdot 2^{t-i} = \frac{m-1}{2m-1} \sum_{i=1}^t (2m)^i 2^{t-i} + \frac{m}{2m-1} \sum_{i=1}^t 2^{t-i} = \frac{m(2m)^t - m}{2m-1}. \quad (23)$$

For eigenvalues  $-1$  and  $1$ , we found that the eigenvalue  $-1$  in  $P_{t-1}$  produces eigenvalue  $0$  according to equation (15); hence, for eigenvalue  $-1$ , we have included the number of its descendants in the number of eigenvalue  $0$ , which has been calculated in equation (23). Next, considering the multiplicity of eigenvalues  $1$  and  $-1$  in  $P_t(t \geq 0)$ , we have

$$N_t^{\text{mul}}(-1) = N_t^{\text{mul}}(1) = 1. \quad (24)$$

Apparently, eigenvalues  $-1$  and  $1$  in  $P_t$  are generated by the eigenvalue  $1$  in  $P_{t-1}$ . Adding the number of all the eigenvalues obtained above, we obtain the following result:

$$N_t^{\text{seed}}(0) + N_t^{\text{mul}}\left(-\frac{1}{2}\right) + N_t^{\text{mul}}(1) = \frac{m(2m)^t + 3m - 2}{2m-1} = N_t, \quad (25)$$

which proves that we have found all the eigenvalues, and the corresponding multiplicities of the matrix  $P_t$  are determined. In addition, then we also obtain all the eigenvalues and their multiplicities of transition weight matrix  $T_t$  by simple transformation.

## 4 Applications of eigenvalues

The eigenvalues of the normalized Laplacian matrix of  $G(m, r, t)$  can be determined from the eigenvalues obtained in Section 3.2, and it can be used to calculate some quantities for the weighted scale-free networks, such as the random target access time for random walks, and also to enumerate weighted spanning trees [27].

### 4.1 Random target access time

We set  $H_{ij}(t)$  as the expected time for a particle starting from starting node  $i$  to visit ending node  $j$  for the first time in  $G(m, r, t)$ . Let  $\pi = (\pi_1, \pi_2, \dots, \pi_{N_t})^T$  be the steady-state distribution for random walks on  $G(m, r, t)$ , where  $\pi_i = s_i(t)/(2Q_t)$  satisfying  $\sum_{i=1}^{N_t} \pi_i = 1$  and  $\pi^T T_t = \pi^T$ . The random target access time  $H_t$  for random walks on  $G(m, r, t)$  is defined as the expected time needed by a particle from a node  $i$  to another target node  $j$ , chosen randomly from all nodes according to the steady-state distribution [9], then it can be formulated as

$$H_t = \sum_{j=1}^{N_t} \pi_j H_{ij}(t). \quad (26)$$

The quantity  $H_t$  does not change due to different starting nodes, so it can be reexpressed as

$$H_t = \sum_{i=1}^{N_t} \pi_i \sum_{j=1}^{N_t} \pi_j H_{ij}(t) = \sum_{j=1}^{N_t} \pi_j \sum_{i=1}^{N_t} \pi_i H_{ij}(t). \quad (27)$$

Equation (27) implies that the random target access time  $H_t$  can be looked upon as the average trapping time of a particular trapping problem; it contains much valuable information about trapping in network  $G(m, r, t)$ . The random target access time in network  $G(m, r, t)$  can be obtained by the sum of the reciprocal of 1 minus each eigenvalue of  $T_t$ , and the eigenvalue 1 is not included here [7].

The normalized Laplacian matrix of  $G(m, r, t)$  is defined as  $L_t = I_t - P_t$ , and  $I_t$  denotes the identity matrix with order  $N_t \times N_t$ . Let  $\{\lambda_i(t) : 1 \leq i \leq N_t\}$  be the  $N_t$  eigenvalues of matrix  $P_t$ . By definition,

let  $\sigma_i(t) = 1 - \lambda_i(t)$  be the eigenvalue of  $L_t$  for  $1 \leq i \leq N_t$ , then there is a one-to-one relationship between  $\sigma_i(t)$  and  $\lambda_i(t)$ . It can be proved that  $H_t$  can be represented in terms of the nonzero eigenvalues of  $L_t$  [28] as follows:

$$H_t = \sum_{i=2}^{N_t} \frac{1}{\sigma_i(t)} = \sum_{i=2}^{N_t} \frac{1}{1 - \lambda_i(t)}. \quad (28)$$

**Theorem 1.** For  $t > 0$ , the random target access time of weighted network  $G(m, r, t)$  is

$$H_t = \frac{2(2m)^t(m-1) + 16m^2 - 4m - 1}{2(2m-1)}$$

when  $t \rightarrow \infty$ , and the relationship between  $H_t$  and the network order  $N_t$  is

$$H_t \sim \frac{m-1}{m} N_t.$$

**Proof.** According to equation (15) and  $\sigma_i(t) = 1 - \lambda_i(t)$ , we can easily obtain the following recursive relation commanding the eigenvalues of two normalized Laplacian matrices  $L_{t-1}$  and  $L_t$ ,

$$\sigma_i(t) = 1 \pm \sqrt{\frac{2 - \sigma_i(t-1)}{2}}. \quad (29)$$

We suppose that  $\Omega_t$  contains all eigenvalues of matrix  $L_t$  at time step  $t$ , then equation (29) means that each eigenvalue  $\sigma_i(t-1)$  in  $\Omega_{t-1}$  gives rise to two eigenvalues  $\sigma_{i,1}(t)$  and  $\sigma_{i,2}(t) \in \Omega_t$ .

It is easy to obtain the following result due to equation (29):  $\Omega_0 = \{0, 2\}$ ,  $0 \in \Omega_0$  generates eigenvalue  $0, 2 \in \Omega_1$ ;  $2 \in \Omega_0$  generates two eigenvalues  $1 \in \Omega_1$ , so  $\Omega_1 = \{0, 1, 1, 2\}$ ;  $0 \in \Omega_1$  generates eigenvalue  $0, 2 \in \Omega_2$ ;  $2 \in \Omega_1$  generates eigenvalue  $1 \in \Omega_2$ ; and  $1 \in \Omega_1$  generates two eigenvalues  $1 + \frac{\sqrt{2}}{2}$  and  $1 - \frac{\sqrt{2}}{2}$ , thus we can obtain  $\Omega_2 = \{0, 1 - \frac{\sqrt{2}}{2}, 1 - \frac{\sqrt{2}}{2}, 1, 1, 1, 1, 1 + \frac{\sqrt{2}}{2}, 1 + \frac{\sqrt{2}}{2}, 2\}$ . Similarly,  $\Omega_3 = \{0, 2, \Omega_3^{(1)}, \Omega_3^{(2)}\}$ , where  $\Omega_3^{(1)} = \{1, 1, \dots, 1\}$  contains all eigenvalues 1 and  $\Omega_3^{(2)}$  is generated by equation (29) from  $\Omega_2 \setminus \{0, 2\}$ .

In order to facilitate calculations, we divide the set  $\Omega_t$  into three disjoint subsets represented by  $\Omega_t^{(1)}, \Omega_t^{(2)}$  and the set  $\Omega_t^{(3)} = \{0, 2\}$ , which can be decomposed into  $\Omega_t = \Omega_t^{(1)} \cup \Omega_t^{(2)} \cup \{0, 2\}$ , where  $\Omega_t^{(1)}$  consists of eigenvalue 1 with multiplicity  $\frac{(2m)^t(m-1)+m}{2m-1}$ , and these eigenvalues generated by equation (29) from  $\sigma_i(t-1) \in \Omega_{t-1} \setminus \{0, 2\}$  are involved in  $\Omega_t^{(2)}$ . First, we consider the subset  $\Omega_t^{(1)}$ , then

$$\sum_{\sigma_i(t) \in \Omega_t^{(1)}} \frac{1}{\sigma_i(t)} = \frac{(2m)^t(m-1) + m}{2m-1}. \quad (30)$$

Hence, we still need to evaluate  $\sum_{\sigma_i(t) \in \Omega_t^{(2)}} \frac{1}{\sigma_i(t)}$  in order to determine  $H_t$ . According to Vieta's formulas, we have  $\sigma_{i,1}(t) + \sigma_{i,2}(t) = 2$  and  $\sigma_{i,1}(t) \cdot \sigma_{i,2}(t) = \sigma_i(t-1)/2$ . In addition, we have

$$\frac{1}{\sigma_{i,1}(t)} + \frac{1}{\sigma_{i,2}(t)} = \frac{4}{\sigma_i(t-1)}, \quad (31)$$

which indicates that

$$\sum_{\sigma_i(t) \in \Omega_t^{(2)}} \frac{1}{\sigma_i(t)} = 4 \sum_{\sigma_i(t-1) \in \Omega_{t-1} \setminus \{0, 2\}} \frac{1}{\sigma_i(t-1)}. \quad (32)$$

Combining the above results in equations (30) and (32), the relationship between  $H_t$  and  $H_{t-1}$  can be obtained as follows:

$$H_t = \sum_{\sigma_i \in \Omega_t^{(1)}} \frac{1}{\sigma_i(t)} + \sum_{\sigma_i \in \Omega_t^{(2)}} \frac{1}{\sigma_i(t)} + \frac{1}{2} = 4H_{t-1} + \frac{(2m)^t(m-1) + m}{2m-1} + \frac{1}{2}. \quad (33)$$

Furthermore, using the initial condition  $H_0 = \frac{1}{2}$ , we can obtain

$$H_t = 4m + \frac{(2m)^t(m-1) + m}{2m-1} + \frac{1}{2} = \frac{2(2m)^t(m-1) + 16m^2 - 4m - 1}{2(2m-1)}. \quad (34)$$

Finally, for large weighted fractal networks, that is  $t \rightarrow \infty$ , we explore how the random target access time of  $G(m, r, t)$  changes with the increase of the network order, we know  $N_t = \frac{m(2m)^t + 3m - 2}{2m-1}$ , then we can obtain

$$H_t \approx \frac{m-1}{m}N_t + \frac{16m^3 + 2m^2 - 11m + 4}{2m(2m-1)}. \quad (35)$$

Equation (35) implies that in the large  $t$  limit, the random target access time increases as a linear function of the network order;  $H_t$  is only affected by the topological parameter  $m$ ; and it has no relationship with the weight factor  $r$ . In Figure 3, we give the schematic diagram to reveal the relationships between  $H_t$  and two parameters  $t$  and  $m$ . In more detail, when the value of  $m$  is determined,  $H_t$  increases monotonically with  $t$  for  $t \geq 0$  as shown in Figure 4. Similarly, for a given  $t$ ,  $H_t$  increases monotonically with  $m$  for  $m \geq 2$  as shown in Figure 5.  $\square$

## 4.2 Weighted counting of spanning trees

The spanning tree of a connect network  $G(m, r, t)$  is a subgraph of it, that is, a tree  $T$ , and includes all the nodes of  $G(m, r, t)$  [29]. For a weighted network  $G(m, r, t)$ , let  $\Lambda(G(m, r, t))$  be its spanning trees set, and  $w(T) = \prod_{e \in T} w_e$ , which is defined to be the product of weight of all edges in  $T$ , where  $w_e$  is the weight of edge  $e$ . Let  $N_{st}^w(G(m, r, t))$  denote the weight counting of spanning trees of weighted fractal network  $G(m, r, t)$ , that is,  $N_{st}^w(G(m, r, t)) = \sum_{T \in \Lambda(G(m, r, t))} w(T)$ .

Different from using the electrical networks theory to obtain the closed-form formula for the spanning tree enumeration [30,31] and calculating the weighted spanning trees through the special structure of the networks [32], in this subsection, we count  $N_{st}^w(G(m, r, t))$  by using the eigenvalues of normalized Laplacian matrix  $L_t$  of  $G(m, r, t)$ , that is,

$$N_{st}^w(G(m, r, t)) = \frac{\prod_{i=1}^{N_t} s_i(t) \prod_{i=2}^{N_t} \sigma_i(t)}{\sum_{i=1}^{N_t} s_i(t)}. \quad (36)$$

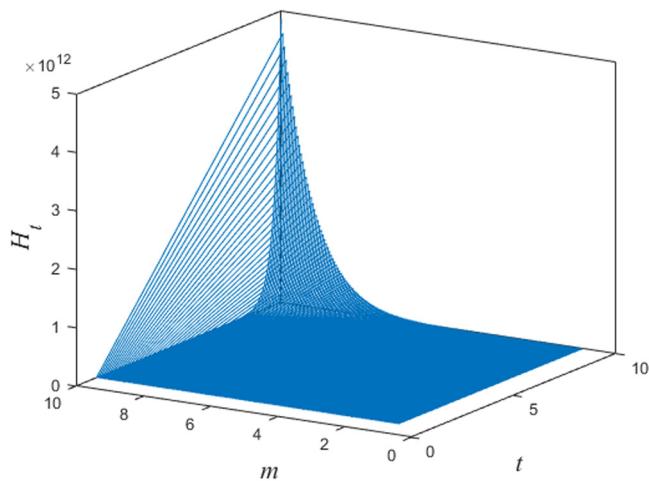


Figure 3: Random target access time  $H_t$  of  $G(m, r, t)$ .

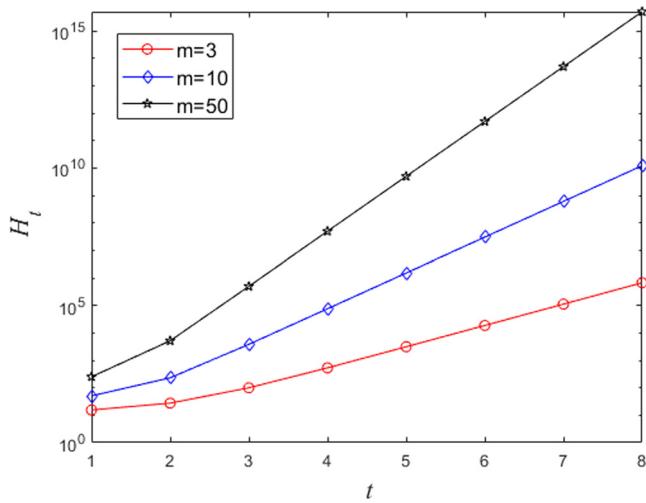


Figure 4: When  $m$  is determined,  $H_t$  increases monotonically with  $t$  for  $t \geq 0$ .

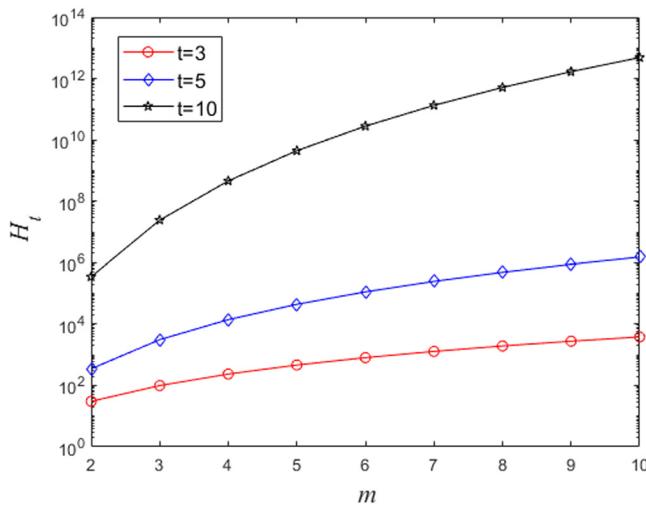


Figure 5: When  $t$  is fixed,  $H_t$  increases monotonically with  $m$  for  $m \geq 2$ .

**Theorem 2.** For  $t > 0$ , the weighted spanning tree of weighted network  $G(m, r, t)$  is

$$N_{st}^w(G(m, r, t)) = \left(\frac{2}{mr}\right)^t (mr)^{\sum_{i=1}^t N_i} \left(\frac{1}{2}\right)^{\sum_{i=0}^{t-1} N_i}.$$

**Proof.** We first need to determine the three terms in equation (36). For the sum term in the denominator, we have

$$\sum_{i=1}^{N_t} s_i(t) = 2Q_t = 2(2mr)^t. \quad (37)$$

Consider the two product terms in the numerator of equation (36), let  $\Phi_t$  represent the product  $\prod_{i=1}^{N_t} s_i(t)$ , and  $\Theta_t$  represent the product term  $\prod_{i=2}^{N_g} \sigma_i(t)$ , respectively. According to the calculation formula of the strength of the node, the quantity  $\Phi_t$  obeys the following recursive relations:

$$\Phi_t = (mr)^{N_t} \times \Phi_{t-1} \quad (38)$$

Based on the results of the eigenvalues obtained above, the following equation holds:

$$\Theta_t = 2 \times \left(\frac{1}{2}\right)^{N_{t-1}-1} \times \Theta_{t-1}. \quad (39)$$

Then, multiplying equations (38) by (39) results in

$$\Phi_t \Theta_t = 4(mr)^{N_t} \left(\frac{1}{2}\right)^{N_{t-1}} \Phi_{t-1} \Theta_{t-1}. \quad (40)$$

For the simple case of  $t = 0$ , we can easily obtain  $\Phi_0 = 1$  and  $\Theta_0 = 2$ , that is,  $\Phi_0 \Theta_0 = 2$ , then equation (40) is solved to give a solution,

$$\Phi_t \Theta_t = 2^{2t+1} (mr)^{\sum_{i=1}^t N_i} \left(\frac{1}{2}\right)^{\sum_{i=0}^{t-1} N_i}. \quad (41)$$

Inserting the results of two equations (37) and (41) into equation (36), we can obtain the following expression of  $N_{st}^w(G(m, r, t))$  in the studied weighted network  $G(m, r, t)$ :

$$N_{st}^w(G(m, r, t)) = \left(\frac{2}{mr}\right)^t (mr)^{\sum_{i=1}^t N_i} \left(\frac{1}{2}\right)^{\sum_{i=0}^{t-1} N_i}. \quad (42)$$

The result of equation (42) is consistent with the result of direct enumeration, which verifies that our calculation for the eigenvalues of the transition weight matrix of the weighted network  $G(m, r, t)$  is correct.  $\square$

## 5 Conclusion

There are many documents that have verified that the weights of some networks have a serious impact on the random target access time. Unlike the existing weighted networks, we have found a family of weighted networks whose random target access time is not controlled by its weight factor, and these networks have been proven to exhibit the remarkable scale-free properties observed in various real-life complex systems. We have listed all eigenvalues for the transition weight matrix of  $G(m, r, t)$  by giving the explicit recursive expression governing the eigenvalues of networks of two consecutive generations, it means that two eigenvalues of  $P_t$  can be derived from the  $\lambda_i(t-1)$  of  $P_{t-1}$ . On this basis, we have harvested all the eigenvalues and their corresponding multiplicities for transition weight matrix  $T_t$  of the network  $G(m, r, t)$  and prove that  $H_t$  is only controlled by the parameter  $m$ . Finally, we also use the obtained eigenvalues of the normalized Laplacian matrix to further enumerate the weighted spanning tree of the network. In the future, we will explore the influence of the weight on the efficiency of random walks in the network with other properties besides scale-free.

**Funding information:** This research was supported by the National Key Research and Development Plan under Grant No. 2019YFA0706401 and the National Natural Science Foundation of China under Grant Nos. 61872166 and 61662066, the Technological Innovation Guidance Program of Gansu Province: Soft Science Special Project (21CX1ZA285), and the Northwest China Financial Research Center Project of Lanzhou University of Finance and Economics (JYYZ201905).

**Author contributions:** B.Y. and J.S. created and conceptualized the idea. J.S. wrote the original draft. X.W. and M.Z. reviewed and edited the draft. All authors have accepted responsibility for the entire content of this manuscript and approved its submission. The authors applied the SDC approach for the sequence of authors.

**Conflict of interest:** The authors state no conflict of interest.

**Data availability statement:** Data sharing is not applicable to this article as no datasets were generated or analyzed during this study.

## References

- [1] X. D. Zhao, R. Tao, X. J. Kang, and W. Li, *Hierarchical-biased random walk for urban remote sensing image segmentation*, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **12** (2019), no. 5, 1–13, DOI: <https://doi.org/10.1109/JSTARS.2019.2905352>.
- [2] M. Bestehorn, A. P. Riascos, T. M. Michelitsch, and B. A. Collet, *A Markovian random walk model of epidemic spreading*, *Continuum Mech. Thermodyn.* **33** (2021), no. 10, 1207–1221, DOI: <https://doi.org/10.1007/s00161-021-00970-z>.
- [3] B. F. Hu, H. Wang, and Y. J. Zheng, *Sign prediction and community detection in directed signed networks based on random walk theory*, *Int. J. Emb. Syst.* **11** (2019), no. 2, 200, DOI: <https://doi.org/10.1504/IJES.2019.10019713>.
- [4] H. Zhang, H. Q. Zhu, and X. F. Ling, *Polar coordinate sampling-based segmentation of overlapping cervical cells using attention U-Net and random walk*, *Neurocomputing* **383** (2020), no. 3, 212–223, DOI: <https://doi.org/10.1016/j.neucom.2019.12.036>.
- [5] N. Masuda, M. A. Porter, and R. Lambiotte, *Random walks and diffusion on networks*, *Phys. Rep.* **716–717** (2017), 1–58, DOI: <https://doi.org/10.1016/j.physrep.2017.07.007>.
- [6] A. P. Riascos and J. L. Mateos, *Random walks on weighted networks: a survey of local and non-local dynamics*, *J. Complex Netw.* **9** (2021), 1–39, DOI: <https://doi.org/10.1093/comnet/cnab032>.
- [7] M. Levine and G. Loizou, *Kemeny's constant and the random surfer*, *Amer. Math. Monthly* **109** (2002), no. 8, 741–745, DOI: <https://doi.org/10.1080/00029890.2002.11919905>.
- [8] Z. Z. Zhang, A. Julaiti, B. Y. Hou, H. J. Zhang, and G. R. Chen, *Mean first-passage time for random walks on undirected network*, *Eur. Phys. J. B.* **84** (2011), no. 4, 691–697, DOI: <https://doi.org/10.1140/epjb/e2011-20834-1>.
- [9] E. I. Milovanovi, M. M. Mateji, and I. Z. Milovanovi, *On the normalized Laplacian spectral radius, Laplacian incidence energy and Kemeny's constant*, *Linear Algebra Appl.* **582** (2019), no. 1, 181–196, DOI: <https://doi.org/10.1016/j.laa.2019.08.004>.
- [10] P. C. Xie, Z. Z. Zhang, and F. Comellas, *On the spectrum of the normalized Laplacian of iterated triangulations of graphs*, *Appl. Math. Comput.* **273** (2016), no. 15, 1123–1129, DOI: <https://doi.org/10.1016/j.amc.2015.09.057>.
- [11] Y. F. Chen, M. F. Dai, X. Q. Wang, Y. Sun, and W. Y. Su, *Spectral analysis for weighted iterated triangulations of graphs*, *Fractals* **26** (2018), no. 1, 1850017, DOI: <https://doi.org/10.1142/S0218348X18500172>.
- [12] L. Gao, J. Peng, C. Tang, and A. P. Riascos, *Trapping efficiency of random walks in weighted scale-free trees*, *J. Stat. Mech. Theory Exp.* **2021** (2021), 063405, DOI: <https://doi.org/10.1088/1742-5468/ac02cb>.
- [13] M. Liu, Z. Xiong, Y. Ma, P. Zhang, J. Wu, and X. Qi, *DPRank centrality: Finding important vertices based on random walks with a new defined transition matrix*, *Future Gener. Comput. Syst.* **83** (2017), 376–389, DOI: <https://doi.org/10.1016/j.future.2017.10.036>.
- [14] N. Bajorin, T. Chen, A. Dagan, C. Emmons, M. Hussein, M. Khalil, et al., *Vibration modes of 3n-gaskets and other fractals*, *J. Phys. A Math. Theor.* **41** (2008), no. 1, 015101, DOI: <https://doi.org/10.1088/1751-8113/41/1/015101>.
- [15] A. Julaiti, B. Wu, and Z. Z. Zhang, *Eigenvalues of normalized Laplacian matrices of fractal trees and dendrimers: Analytical results and applications*, *J. Chem. Phys.* **138** (2013), no. 20, 204116–204116, DOI: <https://doi.org/10.1063/1.4807589>.
- [16] M. F. Dai, X. Q. Wang, Y. Q. Sun, Y. Sun, and W. Su, *Eigentime identities for random walks on a family of treelike networks and polymer networks*, *Phys. A.* **484** (2017), no. 15, 132–140, DOI: <https://doi.org/10.1016/j.physa.2017.04.172>.
- [17] A. Irmanova, I. Dolzhikova, and A. P. James, *Selftuning stochastic weighted neural networks*, *IEEE Int. Symp. Circuits Syst.* **2020** (2020), 1–5, DOI: <https://doi.org/10.1109/ISCAS45731.2020.9180809>.
- [18] A. D. Bona, D. Marcelo, K. O. Fonseca, and R. Luders, *A reduced model for complex network analysis of public transportation systems*, *Phys. A.* **567** (2021), 125715, DOI: <https://doi.org/10.1016/j.physa.2020.125715>.
- [19] Z. Z. Zhang, X. Y. Guo, and Y. H. Yi, *Spectra of weighted scale-free networks*, *Sci. Rep.* **5** (2015), no. 1, 17469, DOI: <https://doi.org/10.1038/srep17469>.
- [20] M. F. Dai, J. Y. Liu, J. W. Chang, D. L. Tang, T. T. Ju, Y. Sun, et al., *Eigentime identity of the weighted scale-free triangulation networks for weight-dependent walk*, *Phys. A.* **513** (2019), 202–209, DOI: <https://doi.org/10.1016/j.physa.2018.08.172>.
- [21] J. H. Zou, M. F. Dai, X. Q. Wang, H. L. Tang, D. He, Y. Sun, et al., *Eigenvalues of transition weight matrix and eigentime identity of weighted network with two hub nodes*, *Canad. J. Phys.* **96** (2017), no. 3, 255–261, DOI: <https://doi.org/10.1139/cjp-2017-0274>.
- [22] E. Q. Zhu, F. Jiang, C. J. Liu, and J. Xu, *Partition independent set and reduction-based approach for partition coloring problem*, *IEEE Trans. Cybern.* **52** (2022), no. 6, 1–10, DOI: <https://doi.org/10.1109/TCYB.2020.3025819>.

- [23] C. J. Liu, *A note on domination number in maximal outerplanar graphs*, Discrete Appl. Math. **293** (2021), no. 1, 90–94, DOI: <https://doi.org/10.1016/j.dam.2021.01.021>.
- [24] C. J. Liu, E. Q. Zhu, Y. K. Zhang, Q. Zhang, and X. P. Wei, *Characterization, verification and generation of strategies in games with resource constraints*, Automatica **140** (2022), 110254, DOI: <https://doi.org/10.1016/j.automatica.2022.110254>.
- [25] Z. Z. Zhang, S. G. Zhou, and T. Zou, *Self-similarity, small-world, scale-free scaling, disassortativity, and robustness in hierarchical lattices*, Eur. Phys. J. B. **56** (2007), no. 3, 259–271, DOI: <https://doi.org/10.1140/epjb/e2007-00107-6>.
- [26] H. D. Rozenfeld, S. Havlin, and D. Ben-Avraham, *Fractal and transfractal recursive scale-free nets*, New J. Phys. **9** (2007), 175, DOI: <https://doi.org/10.1088/1367-2630/9/6/175>.
- [27] A. Julaiti, B. Wu, and Z. Z. Zhang, *Eigenvalues of normalized Laplacian matrices of fractal trees and dendrimers: Analytical results and applications*, J. Chem. Phys. **138** (2013), no. 20, 204116–204116, DOI: <https://doi.org/10.1063/1.4807589>.
- [28] J. G. Kemeny, J. L. Snell, J. G. Kemeny, H. Mirkill, J. L. Snell, G. L. Thompson, et al., *Finite Markov chains*, Amer. Math. Monthly. **31** (1961), no. 67, 2789587, DOI: <https://doi.org/10.2307/2309264>.
- [29] Z. Z. Zhang, S. Q. Wu, M. Y. Li, and F. Comellas, *The number and degree distribution of spanning trees in the Tower of Hanoi graph*, Theoret. Comput. Sci. **609** (2016), no. 2, 443–455, DOI: <https://doi.org/10.1016/j.tcs.2015.10.032>.
- [30] W. G. Sun, S. Wang, and J. Y. Zhang, *Counting spanning trees in prism and anti-prism graphs*, Appl. Anal. Comput. **6** (2016), no. 1, 65–75, DOI: <https://doi.org/10.11948/2016006>.
- [31] Y. L. Shang, *On the number of spanning trees, the Laplacian eigenvalues, and the Laplacian Estrada index of subdivided-line graphs*, Open Math. **14** (2016), no. 1, 641–648, DOI: <https://doi.org/10.1515/math-2016-0055>.
- [32] F. Ma and B. Yao, *The number of spanning trees of a class of self-similar fractal models*, Inform. Process. Lett. **136** (2018), 64–69, DOI: <https://doi.org/10.1016/j.ipl.2018.04.004>.