doi: 10.1093/comjnl/bxab070

Modeling Higher-Order Interactions in Complex Networks by Edge Product of Graphs

YUCHENG WANG, YUHAO YI, WANYUE XU AND ZHONGZHI ZHANG*

Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 200433, China; Shanghai Engineering Research Institute of Blockchain, Shanghai 200433, China; Research Institute of Intelligent Complex Systems, Fudan University, Shanghai 200433, China

**Corresponding author: zhangzz@fudan.edu.cn*

Many graph products have been applied to generate complex networks with striking properties observed in real-world systems. In this paper, we propose a simple generative model for simplicial networks by iteratively using edge corona product. We present a comprehensive analysis of the structural properties of the network model, including degree distribution, diameter, clustering coefficient, as well as distribution of clique sizes, obtaining explicit expressions for these relevant quantities, which agree with the behaviors found in diverse real networks. Moreover, we obtain exact expressions for all the eigenvalues and their associated multiplicities of the normalized Laplacian matrix, based on which we derive explicit formulas for mixing time, mean hitting time and the number of spanning trees. Thus, as previous models generated by other graph products, our model is also an exactly solvable one, whose structural properties can be analytically treated. More interestingly, the expressions for the spectra of our model are also exactly determined, which is sharp contrast to previous models whose spectra can only be given recursively at most. This advantage makes our model a good test bed and an ideal substrate network for studying dynamical processes, especially those closely related to the spectra of normalized Laplacian matrix, in order to uncover the influences of simplicial structure on these processes.

Keywords: graph product; edge corona product; complex network; random walk; graph spectrum; hitting time; mixing time.

Received 8 October 2020; Revised 7 April 2021; Accepted 27 April 2021
Handling editor: Professor Daniel Paulusma

1. INTRODUCTION

Complex networks are a powerful tool for describing and studying the behavior of structural and dynamical aspects of complex systems [50]. An important achievement in the study of complex networks is the discovery that various real-world systems from biology to social networks display some universal topological features, such as scale-free behavior [2] and small-world effect [67]. The former implies that the fraction of vertices with degree d obeys a distribution of power-law form $P(d) \sim d^{-\gamma}$ with $2 < \gamma \leq 3$. The latter is characterized by small average distance (or diameter) and high clustering coefficient [67]. In addition to these two topological aspects, a lot of real networks are abundant in nontrivial patterns, such as q-cliques [66] and many cycles at different scales [35, 62]. For example, spiking neuron populations form cliques in neural networks [22, 61], while coauthors of a paper constitute a clique in scientific collaboration networks [54]. These remarkable structural properties or patterns greatly affect combinatorial [32, 72], structural [13] and dynamical [11, 70] properties of networks and lead to algorithmic efforts on finding nontrivial subgraphs, e.g. q-cliques [31, 48].

In order to capture or account for universal properties observed in practical networks, a lot of mechanisms, approaches and models were developed in the community of network science [50]. In recent years, cliques, also called simplices, have become very popular to model complex networks [48, 56]. Since large real-world networks are usually made up of small pieces, for example, cliques [66], motifs [47] and communities [21], graph products are an important and natural way for modeling real networks, which generate a large graph out of two or more smaller ones. An obvious advantage of graph operations is the allowance of tractable analysis on various properties of the resultant composite graphs. In the past years, various graph products have been exploited to mimic real complex networks, including Cartesian product [30], corona product [42, 57], hierarchical product [4–6, 58] and Kronecker product [37, 38, 43, 44, 68] and many more [53].

Most current models based on graph operations either fail to reproduce serval properties of real networks or are hard to exactly analyze their spectral properties. For example, iterated corona product on complete graphs only yields small cycles [42, 57], while for most networks created by graph products, their spectra can be determined recursively at most. On the other hand, in many real networks [7, 63], such as brain networks [22, 61] and protein–protein interaction networks [69], there exist higher-order nonpairwise relations between more than two nodes at a time. These higher-order interactions, also called simplicial interactions, play an important role in other structural and dynamical properties of networks, including percolation [10], synchronization [46, 65], disease spreading [36, 45] and voter [27]. Unfortunately, most models generated by graph products and generators cannot capture higher-order interactions, and how simplicial interactions affect random walk dynamics, i.e. mixing time [40], is still unknown.

From a network perspective, higher-order interactions can be described and modeled by hypergraphs [20, 34]. Here we model the higher-order interactions by simplicial complexes [23] generated by a graph product. Although both simplicial complexes and hypergraphs can be applied for the modeling and analysis of realistic systems with higher-order interactions, they differ in some aspects. First, simplicial complexes have a geometric interpretation [17]. For example, they can be explained as the result of gluing nodes, edges, triangles, tetrahedra, etc. along their faces. This interpretation for simplicial complexes can be exploited to characterize the resulting network geometry, such as network curvatures [52]. Moreover, a higher-order interaction described by hypergraphs does not require the presence of all low-order interactions.

In this paper, by iteratively applying edge corona product [28] first proposed by Haynes and Lawson [24, 25] to complete graphs or q-cliques $\mathcal{K}_q$ with $q \geq 1$, we propose a mathematically tractable model for complex networks with various cycles at different scales. Since the resultant networks are composed of cliques of different sizes, we call these networks as *simplicial networks*. The networks can describe simplicial interactions, which have rich structural, spectral and dynamical properties depending on the parameter q. Thus, they can be used to study the influence of simplicial interactions on various dynamics.

Specifically, we present an extensive and exact analysis of relevant topological properties for the simplicial networks, including degree distribution, diameter, clustering coefficient and distribution of clique sizes, which reproduce the common properties observed for real-life networks. We also determine exact expressions for all the eigenvalues and their multiplicities of the transition probability matrix and normalized Laplacian matrix. As applications, we further exploit the obtained eigenvalues to derive leading scaling for mixing time, as well as explicit expressions for average hitting time and the number of spanning trees. The proposed model allows for rigorous analysis of structural properties, as previous models generated by graph products. In contrast to existing models for which the eigenvalues for related matrices are given recursively at most, the eigenvalues of transition probability matrix for our model can be exactly determined. This advantage allows to study analytically even exactly related dynamical processes determined by one or several eigenvalues, for example, mixing time of random walks, which gives deep insight into behavior for mixing time in real-life networks.

2. NETWORK CONSTRUCTION

The network family proposed and studied here is constructed based on the edge corona product of graphs defined as follows [24, 25, 28], which is a variant of the corona product first introduced by Frucht and Harary [19] of two graphs. Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be two graphs with disjoint vertex sets, with the former $\mathcal{G}_1$ having n_1 vertices and m_1 edges. The edge corona $\mathcal{G}_1 \circledcirc \mathcal{G}_2$ of $\mathcal{G}_1$ and $\mathcal{G}_2$ is a graph obtained by taking one copy of $\mathcal{G}_1$ and m_1 copies of $\mathcal{G}_2$ and then connecting both end vertices of the ith edge of $\mathcal{G}_1$ to each vertex in the ith copy of $\mathcal{G}_2$ for $i = 1, 2, \ldots, m_1$.

Let $\mathcal{K}_q$, $q \geq 1$, be the complete graph with q vertices. When $q = 1$, we define $\mathcal{K}_q$ as a graph with an isolate vertex. Based on the edge corona product and the complete graphs, we can iteratively build a set of graphs, which display the striking properties of real-world networks. Let $\mathcal{G}_q(g)$, $q \geq 1$ and $g \geq 0$, be the network after g iterations. Then, $\mathcal{G}_q(g)$ is constructed in the following way.

DEFINITION 2.1. *For $g = 0$, $\mathcal{G}_q(0)$ is the complete graph $\mathcal{K}_{q+2}$. For $g \geq 1$, $\mathcal{G}_q(g+1)$ is obtained from $\mathcal{G}_q(g)$ and $\mathcal{K}_q$ by performing edge corona product on them: for every existing edge of $\mathcal{G}_q(g)$, we introduce a copy of the complete graph $\mathcal{K}_q$ and connect all its vertices to both end vertices of the edge. That is, $\mathcal{G}_q(g+1) = \mathcal{G}_q(g) \circledcirc \mathcal{K}_q$.*

Figure 1 shows the operation for the network construction given in Definition 2.1. Moreover, Fig. 2 illustrates the construction process of $\mathcal{G}_q(g)$ for two particular cases of $q = 1$ and $q = 2$. Note that for $q = 1$, $\mathcal{G}_q(g)$ is reduced to the pseudofractal scale-free web [18], which only contains triangles but excludes other complete graphs with more than three vertices. We also note that our model is different from that created by copying mechanisms [8].

Let $N_q(g)$ and $M_q(g)$ be the number of vertices and number of edges in graph $\mathcal{G}_q(g)$, respectively. Suppose $L_v(g)$ and $L_e(g)$

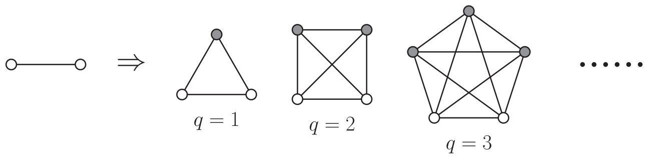

FIGURE 1. Iterative construction approach for the network; for each existing edge in network $\mathcal{G}_q(g)$, performing the operation on the right-hand side of the arrow generates network $\mathcal{G}_q(g+1)$.

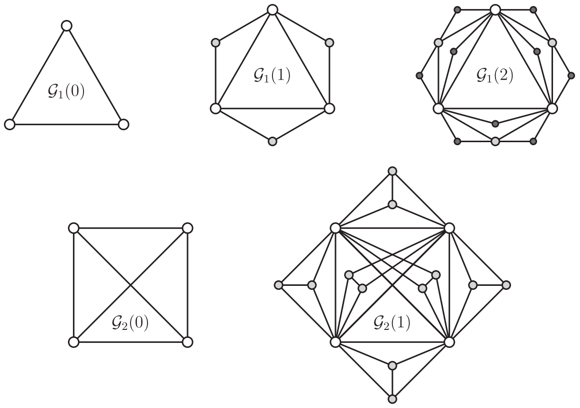

FIGURE 2. The first several iterations of $\mathcal{G}_q(g)$ for $q = 1$ and $q = 2$.

be the number of vertices and the number of edges generated at iteration g. Then for $g = 0$, $L_v(0) = N_q(0) = q + 2$ and $L_e(0) = M_q(0) = \frac{(q+1)(q+2)}{2}$. For all $g \geq 1$, by Definition 2.1, we obtain the following two relations:

$$L_v(g+1) = qM_q(g) \tag{1}$$

and

$$L_e(g+1) = \left[\frac{(q+1)(q+2)}{2} - 1\right] M_q(g), \tag{2}$$

which lead to recursive relationships for $N_q(g)$ and $M_q(g)$ as

$$M_q(g+1) = \frac{(q+1)(q+2)}{2} M_q(g) \tag{3}$$

and

$$N_q(g+1) = qM_q(g) + N_q(g). \tag{4}$$

Considering the initial conditions $N_q(0) = q + 2$ and $M_q(0) = \frac{(q+1)(q+2)}{2}$, the above two equations are solved to obtain

$$M_q(g) = \left[\frac{(q+1)(q+2)}{2}\right]^{g+1} \tag{5}$$

and

$$N_q(g) = \frac{2}{q+3}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} + \frac{2(q+2)}{q+3}. \tag{6}$$

Then, the average degree of vertices in graph $\mathcal{G}_q(g)$ is $2M_q(g)/N_q(g)$, which tends to $q+3$ when g is large. Therefore, the graph family $\mathcal{G}_q(g)$ is sparse.

In addition, inserting Equations (5) and (6) into Equations (1) and (2) gives $L_v(g) = q\left[\frac{(q+1)(q+2)}{2}\right]^g$ and $L_e(g) = \left[\frac{(q+1)(q+2)}{2} - 1\right]\left[\frac{(q+1)(q+2)}{2}\right]^g$ for $g \geq 1$, which are helpful for the computation in the sequel.

3. STRUCTURAL PROPERTIES

In this section, we study some relevant structural characteristics of $\mathcal{G}_q(g)$, focusing on degree distribution, diameter, clustering coefficient and distribution of clique sizes.

3.1. Degree distribution

The degree distribution $P(d)$ for a network is the probability of a randomly selected vertex v has exactly d neighbors. When a network has a discrete sequence of vertex degrees, one can also use cumulative degree distribution $P_{\text{cum}}(d)$ instead of ordinary degree distribution [50], which is the probability that a vertex has degree greater than or equal to d:

$$P_{\text{cum}}(d) = \sum_{d'=d}^{\infty} P(d'). \tag{7}$$

For a graph with degree distribution of power-law form $P(d) \sim d^{-\gamma}$, its cumulative degree distribution is also power-law satisfying $P_{\text{cum}}(d) \sim d^{-(\gamma-1)}$.

For every vertex in graph $\mathcal{G}_q(g)$, its degree can be explicitly determined. Let $d_v(g)$ be the degree of vertex v in graph $\mathcal{G}_q(g)$. When v was generated at iteration g_v, it has a degree of $q+1$. By construction, for any edge incident with v at current iteration, it will lead to q additional new edges adjacent to v at the following iteration. Therefore,

$$d_v(g) = (q+1)^{g-g_v+1}. \tag{8}$$

On the other hand, in graph $\mathcal{G}_q(g)$ the degree of all simultaneously emerging vertices is the same. Then, the number of vertices with the degree $(q+1)^{g-g_v+1}$ is $q+2$ and $q\left[\frac{(q+1)(q+2)}{2}\right]^{g_v}$ for $g_v = 0$ and $g_v > 0$, respectively.

Proposition 3.1. *The degree distribution of graph $\mathcal{G}_q(g)$ follows a power-law form $P(d) \sim d^{-\gamma}$ with the power exponent $\gamma = 2 + \frac{\ln(q+2)}{\ln(q+1)} - \frac{\ln 2}{\ln(q+1)}$.*

Proof. As shown above, the degree sequence of vertices in $\mathcal{G}_q(g)$ is discrete. Thus, we can get the degree distribution $P(d)$ for $d = (q+1)^{g-g_v+1}$ via the cumulative degree distribution given by

$$\begin{aligned} P_{\text{cum}}(d) &= \frac{1}{N_q(g)} \sum_{\tau \leqslant g_v} L_v(\tau) \\ &= \frac{\left[\frac{1}{2}(q+1)(q+2)\right]^{g_v+1} + q + 2}{\left[\frac{1}{2}(q+1)(q+2)\right]^{g+1} + q + 2}. \end{aligned} \tag{9}$$

From Equation (8), we derive $g_v = g + 1 - \frac{\ln d}{\ln(q+1)}$. Plugging this expression for g_v into the above equation leads to

$$\begin{aligned} P_{\text{cum}}(d) &= \frac{2^{\frac{\ln d}{\ln(q+1)}-g-2}\left[(q+1)(q+2)\right]^{-\frac{\ln d}{\ln(q+1)}+g+2} + q + 2}{2^{-g-1}\left[(q+1)(q+2)\right]^{g+1} + q + 2} \\ &= \frac{d^{-\left(\frac{\ln(q+2)}{\ln(q+1)}+1-\frac{\ln 2}{\ln(q+1)}\right)} 2^{-g-2}\left[(q+1)(q+2)\right]^{g+2} + q + 2}{2^{-g-1}\left[(q+1)(q+2)\right]^{g+1} + q + 2}. \end{aligned} \tag{10}$$

When $g \to \infty$, we obtain

$$P_{\text{cum}}(d) = \frac{(q+1)(q+2)}{2} d^{-\left(\frac{\ln(q+2)}{\ln(q+1)}+1-\frac{\ln 2}{\ln(q+1)}\right)}. \tag{11}$$

So the degree distribution follows a power-law form $P(d) \sim d^{-\gamma}$ with the exponent $\gamma = 2 + \frac{\ln(q+2)}{\ln(q+1)} - \frac{\ln 2}{\ln(q+1)}$. ■

It is not difficult to see that the power exponent γ lies in the interval $[\frac{\ln 2}{\ln 3} + 2, 3]$. Moreover, it is a monotonically increasing function of q: when q increases from 2 to infinite, γ increases from $\frac{\ln 2}{\ln 3} + 2$ to 3. Note that for most real scale-free networks [50], their power exponent γ is in the range between 2 and 3.

3.2. Diameter

In a graph $\mathcal{G}$, where every edge having unit length, a shortest path between a pair of vertices u and v is a path connecting u and v with least edges. The distance $d(u, v)$ between u and v is defined as the number of edges in such a shortest path. The diameter of graph $\mathcal{G}$, denoted by $D(\mathcal{G})$, is the maximum of the distances among all pairs of vertices.

Proposition 3.2. *The diameter $D(\mathcal{G}_q(g))$ of graph $\mathcal{G}_q(g)$, is $D(\mathcal{G}_1(g)) = g + 1$ for $q = 1$ and $D(\mathcal{G}_q(g)) = 2g + 1$ for $q \geq 2$.*

Proof. For the case of $q = 1$, $D(\mathcal{G}_1(g)) = g + 1$ was proved in [71]. Below we only prove the case of $q \geq 2$.

For $g = 0$, $D(\mathcal{G}_q(g)) = 1$, the statement holds. By Definition 2.1, it is obvious that the diameter of graph $\mathcal{G}_q(g)$ increases at most 2 after each iteration, which means $D(\mathcal{G}_q(g)) \leq 2g + 1$. In order to prove $D(\mathcal{G}_q(g)) = 2g + 1$, we only need to show that for $q \geq 2$ there exist two vertices in $\mathcal{G}_q(g)$, whose distance $2g + 1$. To this end, we alternatively prove an extended proposition that in $\mathcal{G}_q(g)$ there exist two pairs of adjacent vertices: u_1 and u_3, u_2 and u_4, such that $d(u_1, u_2) = d(u_1, u_4) = d(u_3, u_2) = d(u_3, u_4) = 2g + 1$. We next prove this extended proposition by induction on g.

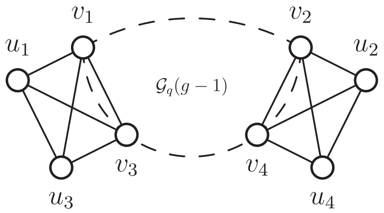

FIGURE 3. Illustrative proof of the extended proposition.

For $g = 0$, $\mathcal{G}_q(0)$, $q \geq 2$, is the complete graph $\mathcal{K}_{q+2}$. We can arbitrarily choose four vertices as u_1, u_2, u_3, u_4 to meet the condition. For $g \geq 1$, suppose that the statement holds for $\mathcal{G}_q(g-1)$, see Fig. 3. In other words, there exist two pairs of adjacent vertices: v_1 and v_3, v_2 and v_4 in $\mathcal{G}_q(g-1)$, with their distances in $\mathcal{G}_q(g-1)$ satisfying $d(v_1, v_2) = d(v_1, v_4) = d(v_3, v_2) = d(v_3, v_4) = 2g - 1$. For $\mathcal{G}_q(g-1)$, let u_1 and u_3 be two adjacent vertices generated by the edge connecting v_1 and v_3 at iteration g, and let u_2 and u_4 be two adjacent vertices generated by the edge connecting v_2 and v_4 at iteration g. Then, by assumption, for the vertex pair u_1 and u_2 in graph $\mathcal{G}_q(g-1)$, their distance obeys $d(u_1, u_2) = \min\{d(v_1, v_2), d(v_1, v_4), d(v_3, v_2), d(v_3, v_4)\} + 2 = 2g + 1$. Similarly, we can prove that in $\mathcal{G}_q(g-1)$, the distances of related vertex pairs satisfy $d(u_1, u_4) = d(u_3, u_2) = d(u_3, u_4) = 2g + 1$. ■

From Equation (6), the number of vertices $N_q(g) \sim \left[\frac{(q+1)(q+2)}{2}\right]^{g+1}$. Thus, the diameter $D(\mathcal{G}_q(g))$ of $\mathcal{G}_q(g)$ scales logarithmically with $N_q(g)$, which means that the graph family $\mathcal{G}_q(g)$ is small-world.

3.3. Clustering coefficient

Clustering coefficient [67] is another crucial quantity characterizing network structure. In a graph $\mathcal{G} = \mathcal{G}(\mathcal{V}, \mathcal{E})$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$, the clustering coefficient $C_v(\mathcal{G})$ of a vertex v with degree d_v is defined [67] as the ratio of the number ϵ_v of edges between the neighbors of v to the possible maximum value $d_v(d_v - 1)/2$, that is, $C_v(\mathcal{G}) = \frac{2\epsilon_v}{d_v(d_v-1)}$. The clustering coefficient $C(\mathcal{G})$ of the whole network $\mathcal{G}$ is defined as the average of $C_v(\mathcal{G})$ over all vertices: $C(\mathcal{G}) = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} C_v(\mathcal{G})$.

For graph $\mathcal{G}_q(g)$, the clustering coefficient for all vertices and their average value can be determined explicitly.

Proposition 3.3. *In graph $\mathcal{G}_q(g)$, the clustering coefficient $C_v(\mathcal{G}_q(g))$ of any vertex with degree $d_v(g)$ is*

$$C_v(\mathcal{G}_q(g)) = \frac{q+1}{d_v(g)}\,. \tag{12}$$

Proof. By Definition 2.1, when a vertex v was created at iteration g_v, its degree and clustering coefficient are $q+1$ and 1, respectively. In any two successive iterations t and $t-1$ ($t \leq g$), its degrees increases by a factor of q as $d_v(t) = (q+1)d_v(t-1)$. Moreover, once its degree increases by q, then the number of edges between its neighbors increases by $q(q+1)/2$. Then, in network $\mathcal{G}_q(g)$, the clustering coefficient $C_v(\mathcal{G}_q(g))$ of vertex v with degree $d_v(g)$ is

$$C_v(\mathcal{G}_q(g)) = \frac{\frac{q(q+1)}{2} + \frac{d_v(g)-q-1}{q}\frac{q(q+1)}{2}}{\frac{d_v(g)(d_v(g)-1)}{2}} = \frac{q+1}{d_v(g)}, \tag{13}$$

as claimed by the proposition. ■

Thus, in graph $\mathcal{G}_q(g)$, the clustering coefficient of any vertex is inversely proportional to its degree, a scaling observed in various real-world networked systems [59].

Proposition 3.4. *For all $g \geq 0$, the clustering coefficient of $\mathcal{G}_q(g)$ is*

$$C(\mathcal{G}_q(g)) = \frac{\left[\frac{(q+1)^2(q+2)}{2}\right]^{g+1} + q^2+4q+4}{\frac{q^2+4q+5}{(q+1)(q+3)}\left[\frac{(q+1)^2(q+2)}{2}\right]^{g+1} + \frac{(q+2)(q^2+4q+5)}{q+3}(q+1)^g}\,. \tag{14}$$

Proof. By using Proposition 3.3, the quantity $C(\mathcal{G}_q(g))$ can be calculated by

$$\begin{aligned} C(\mathcal{G}_q(g)) &= \frac{1}{N_q(g)}\left(\sum_{g_v=0}^{g} L_v(g_v)\cdot\frac{q+1}{d_v(g)}\right) \\ &= \frac{1}{N_q(g)}\left\{\frac{(q+2)}{(q+1)^g} + \sum_{g_v=1}^{g} q\left[\frac{(q+1)(q+2)}{2}\right]^{g_v}\frac{q+1}{(q+1)^{g-g_v+1}}\right\} \\ &= \frac{\left[\frac{(q+1)^2(q+2)}{2}\right]^{g+1} + q^2+4q+4}{\frac{q^2+4q+5}{(q+1)(q+3)}\left[\frac{(q+1)^2(q+2)}{2}\right]^{g+1} + \frac{(q+2)(q^2+4q+5)}{q+3}(q+1)^g}. \end{aligned} \tag{15}$$

This finishes the proof. ■

From Proposition 3.4, we can see that the clustering coefficient of graph $\mathcal{G}_q(g)$ is very high. For large g, the clustering coefficient $\mathcal{G}_q(g)$ converges to a large constant as

$$\lim_{g\to\infty} C(\mathcal{G}_q(g)) = \frac{q^2+4q+3}{q^2+4q+5}\,. \tag{16}$$

Thus, similarly to the degree exponent γ, clustering coefficient $C(\mathcal{G}_q(g))$ is also dependent on q, with large q corresponding to large $C(\mathcal{G}_q(g))$. When $q \to \infty$, the clustering coefficient of the graph tends to 1.

3.4. Distribution of clique sizes

It is apparent that graph $\mathcal{G}_q(g)$ contains many cliques as subgraphs. Let $N_k(G_q(g))$ denote the number of k-cliques in graph $\mathcal{G}_q(g)$. Since graph $\mathcal{G}_q(0)$ is a $q+2$ complete graph, the maximum clique size in it is $q+2$. Then in $\mathcal{G}_q(0)$ the number $N_k(\mathcal{G}_q(0))$ of k-cliques is the combinatorial number $C_{q+2}^k = \frac{(q+2)!}{k!(q+2-k)!}$ for $k = 2, 3, \ldots, q+2$, and is 0 for $k > q+2$. For graph $\mathcal{G}_q(g)$ with $g \geq 1$, the number of 2-cliques equals the number of edges, while for cliques with size more than 2, we have the following proposition.

Proposition 3.5. *For $g \geq 0$, we have*

$$N_k(\mathcal{G}_q(g)) = \frac{\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} - 1}{\frac{(q+1)(q+2)}{2} - 1}\,\frac{(q+2)!}{k!\,(q+2-k)!}, \tag{17}$$

for $k = 3, 4, \ldots, q+2$. Moreover, $N_k(\mathcal{G}_q(g)) = 0$, for $k > q+2$.

Proof. The proposition is naturally satisfied in graph $\mathcal{G}_q(0)$. Thus, we only need to prove the proposition for $g \geq 1$. By definition, when $g \geq 1$, $\mathcal{G}_q(g)$ is obtained from $\mathcal{G}_q(g-1)$ by introducing a new q-complete graph for every edge. Then, all the k-cliques in $\mathcal{G}_q(g)$ can be partitioned into two parts: (i) the k-cliques in $\mathcal{G}_q(g-1)$ and (ii) the k-cliques that contain at least one newly introduced vertex.

For part (i), the number of k-cliques is $N_k(\mathcal{G}_q(g-1))$. For part (ii), every newly introduced vertex is only connected to other vertices in a newly created $q+2$ complete graph generated by an edge of $\mathcal{G}_q(g-1)$. Any k-clique containing this newly introduced vertex must be a subgraph of the same $q+2$ complete graph containing the vertex. The number of new $q+2$ complete graphs equals the number $M_q(g-1)$ of edges in $\mathcal{G}_q(g-1)$, and in every new $q+2$ complete graph, the number of k-cliques is the combinatorial number C_{q+2}^k for $k \leq q+2$. Since in every new $q+2$ complete graph, there are only two old vertices, each of its k-clique subgraph with $k \geq 3$ includes at least one new vertex. Thus, for part (ii) the number of k-cliques can be calculated by $M_q(g-1)C_{q+2}^k$ for $3 \leq k \leq q+2$, and is obviously 0 for $k > q+2$.

Combining the above results, we have that for $g \geq 1$,

$$N_k(\mathcal{G}_q(g)) = N_k(\mathcal{G}_q(g-1)) + M_q(g-1)C_{q+2}^k, \tag{18}$$

for $3 \leq k \leq q{+}2$, and $N_k(\mathcal{G}_q(g)) = N_k(\mathcal{G}_q(g{-}1))$ for $k > q{+}2$. Together with $M_q(g-1) = \left[\frac{(q+1)(q+2)}{2}\right]^g$, $C_{q+2}^k = \frac{(q+2)!}{k!(q+2-k)!}$, and the initial values for $\mathcal{G}_q(0)$, the above recursive relation is solved to obtain the proposition. ■

4. SPECTRA OF PROBABILITY TRANSITION MATRIX AND NORMALIZED LAPLACIAN MATRIX

Let $\mathbf{A}_g = \mathbf{A}(\mathcal{G}_q(g))$ denote the adjacency matrix of graph $\mathcal{G}_q(g)$, the entries $A_g(i,j)$ of which are defined as follows: $A_g(i,j) = 1$ if the vertex pair of i and j is adjacent in $\mathcal{G}_q(g)$ by an edge denoted by $i \sim j$, or $A_g(i,j) = 0$ otherwise. The vertex-edge incident matrix $\mathbf{R}_g = \mathbf{R}(\mathcal{G}_q(g))$ of graph $\mathcal{G}_q(g)$ is an $N_q(g) \times M_q(g)$ matrix, the entries $R_g(v,e)$ of which are defined in the following way: $R_g(v,e) = 1$ if vertex v is incident to edge e, and $R_g(v,e) = 0$ otherwise. The diagonal degree matrix of $\mathcal{G}_q(g)$ is $\mathbf{D}_g = \mathbf{D}(\mathcal{G}_q(g)) = \mathrm{diag}\{d_1(g), d_2(g), \ldots, d_{N_q(g)}(g)\}$, where the ith nonzero entry is the degree $d_i(g)$ of vertex i in graph $\mathcal{G}_q(g)$. The Laplacian matrix $\mathbf{L}_g = \mathbf{L}(\mathcal{G}_q(g))$ of graph $\mathcal{G}_q(g)$ is $\mathbf{L}_g = \mathbf{D}_g - \mathbf{A}_g$. The transition probability matrix of $\mathcal{G}_q(g)$, denoted by $\mathbf{P}_g = \mathbf{P}(\mathcal{G}_q(g))$, is defined by $\mathbf{P}_g = \mathbf{D}_g^{-1}\mathbf{A}_g$, with the (i,j)th element $P_g(i,j) = 1/d_i(g)$ representing the transition probability for a walker going from vertex i to vertex j in graph $\mathcal{G}_q(g)$. Matrix $\mathbf{P}_g$ is asymmetric but is similar to the normalized adjacency matrix $\tilde{\mathbf{A}}_g(\mathcal{G}_q(g)) = \tilde{\mathbf{A}}_g$ of graph $\mathcal{G}_q(g)$ defined by $\tilde{\mathbf{A}}_g = \mathbf{D}_g^{-\frac{1}{2}}\mathbf{A}_g\mathbf{D}_g^{-\frac{1}{2}}$, since $\tilde{\mathbf{A}}_g = \mathbf{D}_g^{-\frac{1}{2}}\mathbf{P}_g\mathbf{D}_g^{\frac{1}{2}}$. By definition, the (i,j)th entry of matrix $\tilde{\mathbf{A}}_g$ is $\tilde{A}_g(i,j) = \frac{A_g(i,j)}{\sqrt{d_i(g)}\sqrt{d_j(g)}}$. Thus, matrix $\tilde{\mathbf{A}}_g$ is real and symmetric and has the same set of eigenvalues as the transition probability matrix $\mathbf{P}_g$. For graph $\mathcal{G}_q(g)$, its normalized Laplacian matrix $\tilde{\mathbf{L}}_g(\mathcal{G}_q(g)) = \tilde{\mathbf{L}}_g$ is defined by $\tilde{\mathbf{L}}_g = \mathbf{I}_g - \tilde{\mathbf{A}}_g$, where $\mathbf{I}_g$ is the $N_q(g) \times N_q(g)$ identity matrix.

In the remainder of this section, we will study the full spectrum of transition probability matrix $\mathbf{P}_g$ and normalized Laplacian matrix $\tilde{\mathbf{L}}_g$ for graph $\mathcal{G}_q(g)$. For $i = 1, 2, \cdots, N_q(g)$, let $\lambda_i(g) = \lambda_i(G_q(g))$ and $\sigma_i(g) = \sigma_i(G_q(g))$ denote the $N_q(g)$ eigenvalues of matrices $\mathbf{P}_g$ and $\tilde{\mathbf{L}}_g$, respectively. Let Λ_g and Σ_g denote the set of eigenvalues of matrices $\mathbf{P}_g$ and $\tilde{\mathbf{L}}_g$, respectively, that is $\Lambda_g = \{\lambda_1(g), \lambda_2(g), \ldots, \lambda_{N_q(g)}(g)\}$ and $\Sigma_g = \{\sigma_1(g), \sigma_2(g), \ldots, \sigma_{N_q(g)}(g)\}$. It is obvious that for all $i = 1, 2, \cdots, N_q(g)$, the relation $\lambda_i(g) = 1 - \sigma_i(g)$ holds. Moreover, the eigenvalues of matrices $\mathbf{P}_g$ and $\tilde{\mathbf{L}}_g$ can be listed in a nonincreasing (or nondecreasing) order as: $1 = \lambda_1(g) \geq \lambda_2(g) \geq \ldots \geq \lambda_{N_q(g)}(g) \geq -1$ and $0 = \sigma_1(g) \leq \sigma_2(g) \leq \cdots \leq \sigma_{N_q(g)}(g) \leq 2$.

The one-to-one correspondence $\lambda_i(g) = 1 - \sigma_i(g)$ between $\lambda_i(g)$ and $\sigma_i(g)$, for all $i = 1, 2, \cdots, N_q(g)$, indicates that if one determines the eigenvalues of matrix $\mathbf{P}_g$, then the eigenvalues of matrix $\tilde{\mathbf{L}}_g$ are easily found.

LEMMA 4.1. *For $\lambda \neq -\frac{1}{q+1}$ and $\lambda \neq \frac{q-1}{q+1}$, λ is an eigenvalue of $\mathbf{P}_{g+1}$ if and only if $(q+1)\lambda - q$ is an eigenvalue of $\mathbf{P}_g$, and the multiplicity of λ of $\mathbf{P}_{g+1}$, denoted by $m_{g+1}(\lambda)$, is the same as the multiplicity of eigenvalue $(q+1)\lambda - q$ of $\mathbf{P}_g$, denoted by $m_g((q+1)\lambda - q)$, i.e. $m_{g+1}(\lambda) = m_g((q+1)\lambda - q)$.*

Proof. Let $\mathcal{V}_{g+1}$ be the set of vertices in graph $\mathcal{G}_q(g+1)$. It can be looked upon the union of two disjoint sets $\mathcal{V}_g$ and $\mathcal{V}'_{g+1} = \mathcal{V}_{g+1} \backslash \mathcal{V}_g$, where $\mathcal{V}'_{g+1}$ includes all the newly introduced vertices by the edges in $\mathcal{G}_q(g)$. For all vertices in $\mathcal{V}_{g+1}$, we label those in $\mathcal{V}_g$ from 1 to $N_q(g)$, while label the vertices $\mathcal{V}'_{g+1}$ from $N_q(g)+1$ to $N_q(g+1)$. In the following statement, we represent all the vertices by their labels.

Let $\mathbf{y} = (y_1, y_2, \ldots, y_{N_q(g+1)})^\top$ denote the eigenvector of eigenvalue λ of matrix $\mathbf{P}_{g+1}$, where the component y_i corresponds to vertex i in $\mathcal{G}_q(g+1)$. Then,

$$\lambda\,\mathbf{y} = \mathbf{P}_{g+1}\,\mathbf{y}. \tag{19}$$

By construction, for any two adjacent old vertices u and v in $\mathcal{V}_g$, there are q vertices newly introduced by the edge connecting u and v, which are denoted by $h_1, h_2, \ldots, h_q$. These q vertices, together with u and v, form a complete graph of $q{+}2$ vertices. Moreover, each vertex h_i in set $\{h_1, h_2, \ldots, h_q\}$ is exactly connected to u, v, and other vertices in $\{h_1, h_2, \ldots, h_q\}$ excluding h_i itself. Then the row in Equation (19) corresponding to vertex h_i, $i = 1, 2, \ldots, q$, can be written as

$$\begin{aligned}\lambda\,y_{h_i} &= \sum_{j=1}^{N_q(g+1)} P_{g+1}(h_i,j)y_j \\ &= \frac{1}{d_{h_i}(g+1)}\sum_{j\sim h_i} y_j \\ &= \frac{1}{q+1}(y_u + y_v + y_{h_1} + \ldots + y_{h_{i-1}} \\ &\quad + y_{h_{i+1}} + \ldots + y_{h_q}),\end{aligned} \tag{20}$$

Adding $\frac{1}{q+1}y_{h_i}$ to both sides of the above equation yields

$$\left(\lambda + \frac{1}{q+1}\right)y_{h_i} = \frac{1}{q+1}\left(y_u + y_v + \sum_{j=1}^{q} y_{h_j}\right), \tag{21}$$

for all $i = 1, 2, \ldots, q$. Therefore, for $\lambda \neq -\frac{1}{q+1}$,

$$y_{h_1} = y_{h_2} = \ldots = y_{h_q}. \tag{22}$$

Combining Equations (21) and (22), we can derive that for $\lambda \neq \frac{q-1}{q+1}$

$$y_{h_i} = \frac{1}{(q+1)\lambda - q - 1}(y_u + y_v) \tag{23}$$

holds for $i = 1, 2, \ldots, q$. According to Equation (19), we can also express the rows corresponding to components y_u and y_v. For the row associated with component y_u, we have

$$\begin{aligned}\lambda\, y_u &= \sum_{j=1}^{N_q(g+1)} P_{g+1}(u,j) y_j \\ &= \frac{1}{d_u(g+1)} \left(\sum_{\substack{j \le N_q(g) \\ j \sim u}} y_j + \sum_{\substack{j > N_q(g) \\ j \sim u}} y_j \right). \end{aligned} \tag{24}$$

By Definition 2.1, for an old vertex u, all its adjacent vertices in $\mathcal{V}'_{g+1}$ are introduced by the edges between u and its neighboring vertices in $\mathcal{V}_g$. Thus, combining Equations (23) and (24), we derive

$$\lambda\, y_u = \frac{1}{d_u(g+1)} \left(\sum_{\substack{j \le N_q(g) \\ j \sim u}} y_j + \sum_{\substack{j \le N_q(g) \\ j \sim u}} \frac{q(y_u + y_j)}{(q+1)\lambda - q - 1} \right). \tag{25}$$

Considering $d_u(g+1) = (q+1)d_u(g)$, Equation (25) can be recast as

$$\begin{aligned} &\left((q+1)\lambda - \frac{q}{(q+1)\lambda - q - 1} \right) y_u \\ &= \frac{1}{d_u(g)} \sum_{\substack{j \le N_q(g) \\ j \sim u}} \left(1 + \frac{q}{(q+1)\lambda - q - 1} \right) y_j. \end{aligned} \tag{26}$$

When $\lambda \neq -\frac{1}{q+1}$ and $\lambda \neq \frac{q-1}{q+1}$, the above equation is simplified as

$$\begin{aligned} [(q+1)\lambda - q]\, y_u &= \frac{1}{d_u(g)} \sum_{\substack{j \le N_q(g) \\ j \sim u}} y_j \\ &= \sum_{j=1}^{N_q(g)} P_g(u,j) y_j, \end{aligned} \tag{27}$$

which implies if $\mathbf{y} = (y_1, y_2, \ldots, y_{N_q(g)}, \ldots, y_{N_q(g+1)})^\top$ is an eigenvector of matrix $\mathbf{P}_{g+1}$ associated with eigenvalue λ, then $\tilde{\mathbf{y}} = (y_1, y_2, \ldots, y_{N_q(g)})^\top$ is an eigenvector of matrix $\mathbf{P}_g$ associated with eigenvalue $(q+1)\lambda - q$.

On the other hand, suppose that $\tilde{\mathbf{y}} = (y_1, y_2, \ldots, y_{N_q(g)})^\top$ is an eigenvector of matrix $\mathbf{P}_g$ associated with eigenvalue $(q+1)\lambda - q$, then $\mathbf{y} = (y_1, y_2, \ldots, y, \ldots, y_{N_q(g+1)})^\top$ is an eigenvector of matrix $\mathbf{P}_{g+1}$ associated with eigenvalue λ if and only if its components y_i, $i = N_q(g)+1, N_q(g)+2, \ldots, N_q(g+1)$, can be expressed by Equation (23). Thus, the number of linearly independent eigenvectors of λ is the same as that of $(q+1)\lambda - q$. Since both $\mathbf{P}_g$ and $\mathbf{P}_{g+1}$ are normal matrices, which are diagonalizable, the multiplicity of λ (or $(q+1)\lambda - q$) is equal to the number of its linearly independent eigenvectors. Hence, $m_{g+1}(\lambda) = m_g((q+1)\lambda - q)$. ■

Lemma 4.1 indicates that except $\lambda \neq -\frac{1}{q+1}$ and $\frac{q-1}{q+1}$, all eigenvalues λ of matrix $\mathbf{P}_{g+1}$ can be derived from those of matrix $\mathbf{P}_g$. However, it is easy to check that both $-\frac{1}{q+1}$ and $\frac{q-1}{q+1}$ are eigenvalues of matrix $\mathbf{P}_{g+1}$. Moreover, their multiplicities can be determined explicitly. The following lemma gives the multiplicity of $-\frac{1}{q+1}$, while the multiplicity of $\frac{q-1}{q+1}$ will be provided later.

Lemma 4.2. *The multiplicity of $-\frac{1}{q+1}$ as an eigenvalue of matrix $\mathbf{P}_{g+1}$ is $(q-1)M_q(g) + N_q(g)$, i.e. $m_{g+1}(-\frac{1}{q+1}) = (q-1)M_q(g) + N_q(g)$.*

Proof. Let $\mathbf{y} = (y_1, y_2, \ldots\ldots, y_{N_q(g+1)})^\top$ be an eigenvector associated with eigenvalue $-\frac{1}{q+1}$ of matrix $\mathbf{P}_{g+1}$. Then,

$$-\frac{1}{q+1}\mathbf{y} = \mathbf{P}_{g+1}\,\mathbf{y}. \tag{28}$$

For an edge e_x, $x = 1, 2 \ldots, M_q(g)$, in graph $\mathcal{G}_q(g)$ with end vertices u and v, at iteration $g+1$, it will generate q vertices $h_1, h_2, \ldots, h_q$ in $\mathcal{V}'_{g+1}$. Then, the row in Equation (28) corresponding to vertex h_i, $i = 1, 2, \ldots, q$, can be expressed by

$$\begin{aligned} -\frac{1}{q+1} y_{h_i} &= \sum_{j=1}^{N_q(g+1)} P_{g+1}(h_i, j)\, y_j \\ &= \frac{1}{q+1}(y_u + y_v + y_{h_1} + \ldots + y_{h_{i-1}} \\ &\quad + y_{h_{i+1}} + \ldots + y_{h_q}), \end{aligned} \tag{29}$$

which is equivalent to

$$\sum_{i=1}^{q} y_{h_i} = -(y_u + y_v). \tag{30}$$

On the other hand, the row in Equation (28) corresponding to vertex u can be expressed as

$$-\frac{1}{q+1}y_u = \frac{1}{d_u(g+1)}\left(\sum_{\substack{j\le N_q(g)\\ j\sim u}} y_j + \sum_{\substack{j>N_q(g)\\ j\sim u}} y_j\right). \tag{31}$$

Note that Equation (30) holds for every pair of adjacent vertices in graph $\mathcal{G}_q(g)$ and the q new vertices it generates at iteration $g+1$. Plugging Equation (30) into the right-hand side of Equation (31) leads to

$$\begin{aligned}
&\frac{1}{d_u(g+1)}\left(\sum_{\substack{j\le N_q(g)\\ j\sim u}} y_j + \sum_{\substack{j>N_q(g)\\ j\sim u}} y_j\right)\\
&= \frac{1}{d_u(g+1)}\left(\sum_{\substack{j\le N_q(g)\\ j\sim u}} y_j + \sum_{\substack{j\le N_q(g)\\ j\sim u}} -(y_u+y_j)\right)\\
&= \frac{1}{d_u(g+1)}\left(\sum_{\substack{j\le N_q(g)\\ j\sim u}} -y_u\right)\\
&= -\frac{1}{(q+1)}y_u.
\end{aligned} \tag{32}$$

Therefore, the constraint on $\mathbf{y}$ in Equation (28) is equivalent to the constraint provided by $M_q(g)$ equations in Equation (30). The matrix form of these $M_q(g)$ equations can be written as

$$\left[\begin{array}{c|ccccc}
 & 1\,1\cdots 1 & & & & \\
 & & 1\,1\cdots 1 & & & \\
-\mathbf{R}_g^\top & & & \ddots & & \\
 & & & & \cdots & \\
 & & & & & 1\,1\cdots 1
\end{array}\right]\mathbf{y} = \mathbf{0}, \tag{33}$$

where $\mathbf{R}_g^\top$ is the transpose of $\mathbf{R}_g$ and the unmarked entries are vanishing. It is straightforward that the right partition of the matrix in Equation (33) is an $M_q(g) \times qM_q(g)$ matrix, with each row corresponding to an edge $e_x, x = 1, 2, \ldots, M_q(g)$, in graph $\mathcal{G}_q(g)$. Moreover, in each row associated with e_x, 1 repeats q times, corresponding to the q vertices newly created by edge e_x.

Since the row vectors of the matrix in Equation (33) are linearly independent, the dimension of the solution space of Equation (33) is $N_q(g+1) - M_q(g) = (q-1)M_q(g) + N_q(g)$. Therefore, the multiplicity of eigenvalue $-\frac{1}{q+1}$ for matrix $\mathbf{P}_{g+1}$ is $(q-1)M_q(g) + N_q(g)$. ■

THEOREM 4.3. *Let Λ_g, $g \ge 0$, be the set of the $N_q(g)$ eigenvalues $\lambda_1(g)$, $\lambda_2(g)$, …, $\lambda_{N_q(g)}(g)$ for matrix $\mathbf{P}_g$, satisfying $1 = \lambda_1(g) \ge \lambda_2(g) \ge \ldots \ge \lambda_{N_q(g)}(g) \ge -1$. Then the $N_q(g+1)$ eigenvalues for $\mathbf{P}_{g+1}$ forming the set Λ_{g+1} can be listed in a descending order as*

$$\begin{aligned}
\Lambda_{g+1} = \Bigg\{&\frac{\lambda_1(g)+q}{q+1}, \frac{\lambda_2(g)+q}{q+1}, \ldots, \frac{\lambda_{N_q(g)}(g)+q}{q+1},\\
&\underbrace{\frac{q-1}{q+1}, \frac{q-1}{q+1}, \ldots, \frac{q-1}{q+1},}_{M_q(g)-N_q(g)}\\
&\underbrace{-\frac{1}{q+1}, -\frac{1}{q+1}, \ldots, -\frac{1}{q+1}}_{(q-1)M_q(g)+N_q(g)}\Bigg\}.
\end{aligned} \tag{34}$$

Proof. We prove this theorem by induction on g. First, for $g = 0$, it is easy to verify that the statement holds. For graph $\mathcal{G}_q(g)$, $g \ge 1$, assume that the relation between Λ_{g-1} and Λ_g is valid. We now prove that the result is true for graph $\mathcal{G}_q(g+1)$.

For each eigenvalue $\lambda_i(g) \in \Lambda_g$, $i = 1, 2, \ldots, N_q(g)$, we have $\lambda_i(g) > -1$ by the assumption. Therefore, for $i = 1, 2, \ldots, N_q(g)$,

$$\frac{\lambda_i(g)+q}{q+1} > \frac{q-1}{q+1}, \tag{35}$$

which implies $\frac{\lambda_i(g)+q}{q+1} \ne \frac{q-1}{q+1}$ and $\frac{\lambda_i(g)+q}{q+1} \ne -\frac{1}{q+1}$. By Lemma 4.1, $\frac{\lambda_i(g)+q}{q+1}$ is an eigenvalue of $\mathbf{P}_{g+1}$ with the same multiplicity of $\lambda_i(g)$ as an eigenvalue of $\mathbf{P}_g$, namely,

$$m_{g+1}\left(\frac{\lambda_i(g)+q}{q+1}\right) = m_g\left(\lambda_i(g)\right). \tag{36}$$

Moreover, by Lemma 4.1, for each eigenvalue λ of $\mathbf{P}_{g+1}$ satisfying $\lambda \ne -\frac{1}{q+1}$ and $\lambda \ne \frac{q-1}{q+1}$, $(q+1)\lambda - q$ must be an eigenvalue of $\mathbf{P}_g$, which means λ can be expressed by as $\lambda = \frac{\lambda_i(g)+q}{q+1}$ with $i \in \{1, 2, \ldots, N_q(g)\}$. Therefore, the sum of multiplicity of all eigenvalues of $\mathbf{P}_{g+1}$ excluding $-\frac{1}{q+1}$ and $\frac{q-1}{q+1}$ is $N_q(g)$, that is,

$$m_{g+1}\left(\lambda \notin \left\{-\frac{1}{q+1}, \frac{q-1}{q+1}\right\}\right) = N_q(g). \tag{37}$$

We proceed to compute the multiplicity $m_{g+1}\left(\frac{q-1}{q+1}\right)$ of eigenvalue $\frac{q-1}{q+1}$ for matrix $\mathbf{P}_{g+1}$, which obeys

$$
\begin{aligned}
& m_{g+1}\left(-\frac{1}{q+1}\right)+m_{g+1}\left(\frac{q-1}{q+1}\right) \\
& \quad + m_{g+1}\left(\lambda \notin \left\{-\frac{1}{q+1}, \frac{q-1}{q+1}\right\}\right)=N_q(g+1). \qquad (38)
\end{aligned}
$$

Using Equation (37) and Lemma 4.2, one obtains

$$
m_{g+1}\left(\frac{q-1}{q+1}\right)=M_q(g)-N_q(g). \qquad (39)
$$

Combining Equations (35), (36) and (39) and Lemma 4.2 yields (34). ■

For $g=0$, $\mathcal{G}_q(0)$ is a complete graph with $q+2$ vertices. The set of the eigenvalues of matrix $\mathbf{P}_0$ is

$$
\Lambda_0=\left\{1,-\frac{1}{q+1},-\frac{1}{q+1},\ldots,-\frac{1}{q+1}\right\}. \qquad (40)
$$

By recursively applying Theorem 4.3, we can obtain all the eigenvalues of matrix $\mathbf{P}_g$ for $g \geq 1$.

Using Theorem 4.3 and the one-to-one correspondence between matrices $\tilde{\mathbf{L}}_g$ and $\tilde{\mathbf{P}}_g$, we can also obtain relation for the set of eigenvalues for $\tilde{\mathbf{L}}_g$ and $\tilde{\mathbf{L}}_{g+1}$.

Theorem 4.4. *Let Σ_g, $g \geq 0$, be the set of the $N_q(g)$ eigenvalues $\sigma_1(g)$, $\sigma_2(g)$, …, $\sigma_{N_q(g)}(g)$ for matrix $\tilde{\mathbf{L}}_g$, satisfying $0=\sigma_1(g) \leq \sigma_2(g) \leq \ldots \leq \sigma_{N_q(g)}(g) \leq 2$. Then the $N_q(g+1)$ eigenvalues for $\tilde{\mathbf{L}}_{g+1}$ forming the set Σ_{g+1} can be listed in an increasing order as*

$$
\begin{aligned}
\Sigma_{g+1}=\Bigg\{ & \frac{\sigma_1(g)}{q+1}, \frac{\sigma_2(g)}{q+1}, \ldots, \frac{\sigma_{N_q(g)}(g)}{q+1}, \\
& \underbrace{\frac{2}{q+1}, \frac{2}{q+1}, \ldots, \frac{2}{q+1}}_{M_q(g)-N_q(g)}, \underbrace{\frac{q+2}{q+1}, \frac{q+2}{q+1}, \ldots, \frac{q+2}{q+1}}_{(q-1)M_q(g)+N_q(g)}\Bigg\}. \qquad (41)
\end{aligned}
$$

Proof. The proof is easily obtained by combining the relation $\lambda_i(g)=1-\sigma_i(g)$ and Theorem 4.3. ■

The set Σ_0 of eigenvalues for matrix $\tilde{\mathbf{L}}_0$ is $\Sigma_0 = \left\{0, \frac{q+2}{q+1}, \frac{q+2}{q+1}, \ldots, \frac{q+2}{q+1}\right\}$. For $g \geq 1$, by recursively applying Theorem 4.4, we can obtain the exact expressions for all eigenvalues for matrix $\tilde{\mathbf{L}}_g$ for any q and g, given by

$$
\begin{aligned}
\Sigma_g=\Bigg\{0, & \underbrace{\frac{q+2}{(q+1)^{g+1}}, \frac{q+2}{(q+1)^{g+1}}, \ldots, \frac{q+2}{(q+1)^{g+1}}}_{q+1}, \\
& \underbrace{\frac{2}{(q+1)^g}, \frac{2}{(q+1)^g}, \ldots, \frac{2}{(q+1)^g}}_{M_q(0)-N_q(0)}, \\
& \underbrace{\frac{q+2}{(q+1)^g}, \frac{q+2}{(q+1)^g}, \ldots, \frac{q+2}{(q+1)^g}}_{(q-1)M_q(0)+N_q(0)}, \\
& \underbrace{\frac{2}{(q+1)^{g-1}}, \frac{2}{(q+1)^{g-1}}, \ldots, \frac{2}{(q+1)^{g-1}}}_{M_q(1)-N_q(1)}, \\
& \underbrace{\frac{q+2}{(q+1)^{g-1}}, \frac{q+2}{(q+1)^{g-1}}, \ldots, \frac{q+2}{(q+1)^{g-1}}}_{(q-1)M_q(1)+N_q(1)}, \\
& \ldots\ldots, \\
& \underbrace{\frac{2}{q+1}, \frac{2}{q+1}, \ldots, \frac{2}{q+1}}_{M_q(g-1)-N_q(g-1)}, \underbrace{\frac{q+2}{q+1}, \frac{q+2}{q+1}, \ldots, \frac{q+2}{q+1}}_{(q-1)M_q(g-1)+N_q(g-1)}\Bigg\}.
\end{aligned}
\qquad (42)
$$

5. APPLICATIONS OF THE SPECTRA

In this section, we apply the above-obtained eigenvalues and their multiplicities of related matrices to evaluate some relevant quantities for graph $\mathcal{G}_q(g)$, including mixing time, mean hitting time also called Kemeny constant and the number of spanning trees.

5.1. Mixing time

As is well known, the probability transition matrix $\mathbf{P}(\mathcal{G})$ of a graph $\mathcal{G}$ characterizes the process of random walks on the graph. As a classical Markov chain, random walks describe various phenomena or other dynamical processes in graphs. Many interesting quantities about random walks can be extracted from the eigenvalues of the probability transition matrix. In this paper, we only consider mixing time and mean hitting time.

For an ergodic random walk on an un-bipartite graph $\mathcal{G}$ with N vertices, it has a unique stationary distribution $\pi = (\pi_1, \pi_2, \ldots, \pi_N)^\top$ with $\sum_{i=1}^N \pi_i = 1$, where π_i represents the probability that the walker is at vertex i when the random walk converges to equilibrium state [33]. The mixing time is defined as the expected time that the walker needs to approach the stationary distribution. Let $1=\lambda_1 > \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_N >$

-1 be the N eigenvalues for matrix $\mathbf{P}(\mathcal{G})$. Then the speed of convergence to the stationary distribution [64] approximately equals the reciprocal of $1 - \lambda_{\max}$, where $\lambda_{\max}$ is the second largest eigenvalue modulus defined by $\lambda_{\max} = \max(\lambda_2, |\lambda_N|)$. Mixing time has found numerous applications in man different aspects [40].

As our first application of eigenvalues for matrix $\mathbf{P}_g$, we use them to evaluate the mixing time for random walks on $\mathcal{G}_q(g)$, for which the component of stationary distribution π corresponding to vertex i is $\pi_i = d_i(g)/(2M_q(g))$. According to the above arguments, the second largest eigenvalue modulus $\lambda_{\max}(g)$ of $\mathbf{P}_g$ is $\lambda_{\max}(g) = 1 - \frac{q+2}{(q+1)^{g+1}}$. Since the mixing time is characterized by a parameter, it cannot be exactly determined [64], but one can evaluate it by using the reciprocal of $\lambda_{\max}(g)$. Then, the dominating term of the mixing time for random walks on $\mathcal{G}_q(g)$ is $(q+1)^{g+1}/(q+2)$, which scales sublinearly with the vertex number $N_q(g)$ as $(N_q(g))^{2/\theta(q)}$, where $\theta(q) = 2/\log_{(q+1)(q+2)/2}(q+1)$ is the spectral dimension [46] of graph $\mathcal{G}_q(g)$ that is a function of q. Note that for $q = 1$, the spectral dimension $\theta(2) = 2\ln 3/\ln 2$ reduces to the result obtained in [9].

Note that it is believed that real-world networks are often fast mixing with their mixing time at most $O(\log N)$, where N is the number of vertices. However, it was experimentally reported that the mixing time of some real-world social networks is much higher than anticipated [49]. Our obtained sublinear scaling of mixing time on graph $\mathcal{G}$ supports this recent study and sheds lights on understanding the scalings of mixing time.

5.2. Mean hitting time

Our second application for our obtained eigenvalues is the mean hitting time. For a random walk on graph $\mathcal{G}$, the hitting time H_{ij}, also called first-passage time [15, 51, 60], from vertex i to vertex j, is defined as the expected time taken by a walker starting from vertex i to reach vertex j for the first time. The mean hitting time H, also known as the Kemeny constant, is defined as the expected time for a random walker going from a vertex i to another vertex j that is chosen randomly from all vertices in $\mathcal{G}$ according to the stationary distribution [1, 41]:

$$H = \sum_{j=2}^{n} \pi_j H_{ij}. \tag{43}$$

Interestingly, the quantity H is independent of the starting vertex i and can be expressed in terms of the $N-1$ nonzero eigenvalues σ_i, $i = 2, 3, \cdots, N$, of the normalized Laplacian matrix $\tilde{\mathbf{L}}(\mathcal{G})$ for graph $\mathcal{G}$, given by [1, 41]

$$H = \sum_{i=2}^{N} \frac{1}{\sigma_i}. \tag{44}$$

Mean hitting time can be applied to measure the efficiency of user navigation through the World Wide Web [39] and the efficiency of robotic surveillance in network environments [55]. We refer to the reader to [29] for many other applications of mean hitting time.

In this subsection, we use the eigenvalues of the normalized Laplacian matrix for graph $\mathcal{G}_q(g)$ to compute the mean hitting time of $\mathcal{G}_q(g)$.

THEOREM 5.1. *Let $H_q(g)$ be the mean hitting time for random walk in $\mathcal{G}_q(g)$. Then, for all $g \geq 0$,*

$$\begin{aligned} H_q(g) = {} & \left[\frac{(q+1)^2}{q+2} - \frac{3(q+1)}{2}\right](q+1)^g \\ & + \frac{(q+1)(3q+7)}{2(q+3)}\left[\frac{(q+1)(q+2)}{2}\right]^g + \frac{q+1}{q+3}. \end{aligned} \tag{45}$$

Proof. For $g = 0$, by using (40), one obtains $H_q(0) = \frac{(q+1)^2}{(q+2)}$. For $g \geq 1$, according to Theorem 4.4 and Equation (44), we have

$$\begin{aligned} & H_q(g+1) \\ & = \frac{q+1}{2}\big(M_q(g) - N_q(g)\big) \\ & \quad + \frac{q+1}{q+2}\big((q-1)M_q(g) + N_q(g)\big) + \sum_{i=2}^{N_q(g)} \frac{q+1}{\sigma_i(g)} \\ & = \frac{3q(q+1)}{2(q+2)}M_q(g) - \frac{q(q+1)}{2(q+2)}N_q(g) + (q+1)H_q(g), \end{aligned} \tag{46}$$

which can be rewritten as

$$\begin{aligned} & H_q(g+1) - \frac{(q+1)(3q+7)}{2(q+3)}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} - \frac{q+1}{q+3} \\ & = (q+1)\left\{H_q(g) - \frac{(q+1)(3q+7)}{2(q+3)}\left[\frac{(q+1)(q+2)}{2}\right]^g \right. \\ & \quad \left. - \frac{q+1}{q+3}\right\}. \end{aligned} \tag{47}$$

With the initial condition $H_q(0) = \frac{(q+1)^2}{(q+2)}$, Equation (47) is solved to obtain (45). ■

Theorem 5.1 shows that for $g \to \infty$, the dependence of mean hitting time $H_q(g)$ on the number $N_q(g)$ of vertices in graph $\mathcal{G}_g(g)$ is $H_q(g) \sim N_q(g)$, which implies that the $H_q(g)$ behaves linearly with $N_q(g)$.

5.3. The number of spanning trees

A spanning tree of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N vertices is a subgraph of $\mathcal{G}$, which is a tree including all the N vertices. Let $\tau(\mathcal{G})$ denote the number of spanning trees in graph $\mathcal{G}$. It has been shown [12, 14] that $\tau(\mathcal{G})$ can be expressed in terms of the $N - 1$ nonzero eigenvalues for normalized Laplacian matrix of $\mathcal{G}$ and the degrees of all vertices in $\mathcal{G}$:

$$\tau(\mathcal{G}) = \frac{\prod_{i\in\mathcal{V}} d_i \prod_{i=2}^{N} \sigma_i(\mathcal{G})}{\sum_{i\in\mathcal{V}} d_i}. \tag{48}$$

The number of spanning trees is an important graph invariant. In the sequel, we will use the above-obtained eigenvalues to determine this invariant for graph $\mathcal{G}_q(g)$.

Theorem 5.2. *Let $\tau_q(g) = \tau(\mathcal{G}_q(g))$ be the number of spanning trees in graph $\mathcal{G}_q(g)$. Then, for all $g \geq 0$,*

$$\begin{aligned}\tau_q(g) = {} & 2^{\frac{2(q+1)}{q(q+3)^2}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} - \left(\frac{q+1}{q+3}\right)g - \frac{(q+1)^2(q+2)}{q(q+3)^2}} \\ & \cdot (q+2)^{\frac{2(q^2+2q-1)}{q(q+3)^2}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} + \left(\frac{q+1}{q+3}\right)g + \frac{q^3+2q^2-q+2}{q(q+3)^2}}. \end{aligned} \tag{49}$$

Proof. First, by Theorem 4.4, we derive the relation for the product of all the nonzero eigenvalues for normalized Laplacian matrix for graph $\mathcal{G}_q(g+1)$ and $\mathcal{G}_q(g)$:

$$\begin{aligned} & \prod_{i=2}^{N_q(g+1)} \sigma_i(g+1) \\ & = \left(\frac{2}{q+1}\right)^{M_q(g)-N_q(g)} \left(\frac{q+2}{q+1}\right)^{(q-1)M_q(g)+N_q(g)} \prod_{i=2}^{N_q(g)} \frac{\sigma_i(g)}{q+1} \\ & = \frac{2^{M_q(g)-N_q(g)}(q+2)^{(q-1)M_q(g)+N_q(g)}}{(q+1)^{qM_q(g)+N_q(g)-1}} \prod_{i=2}^{N_q(g)} \sigma_i(g). \end{aligned} \tag{50}$$

Second, we derive the relation be between the product of degrees of all vertices in $\mathcal{G}_q(g+1)$ and the product of degrees of all vertices in $\mathcal{G}_q(g)$. For $\mathcal{G}_q(g+1)$, the degree of all the new vertices in $\mathcal{V}'_{g+1}$ that were generated at iteration $g+1$ is $q+1$, while for each i of those old vertices in $\mathcal{V}_g$, we have $d_i(g+1) = (q+1)d_i(g)$. Then,

$$\begin{aligned} \prod_{i\in\mathcal{V}_{g+1}} d_i(g+1) & = \prod_{i\in\mathcal{V}'_{g+1}} d_i(g+1) \prod_{i\in\mathcal{V}_g} d_i(g+1) \\ & = (q+1)^{qM_q(g)} \prod_{i\in\mathcal{V}_g} (q+1)d_i(g) \\ & = (q+1)^{qM_q(g)+N_q(g)} \prod_{i\in\mathcal{V}_g} d_i(g). \end{aligned} \tag{51}$$

Finally, the sum of degrees of all vertices in $\mathcal{G}_q(g)$ is equal to $2M_q(g)$. Then, combining Equations (5), (48), (50) and (51), we obtain the following recursive relation for $\tau_q(g+1)$ and $\tau_q(g)$:

$$\tau_q(g+1) = 2^{M_q(g)-N_q(g)+1}(q+2)^{(q-1)M_q(g)+N_q(g)-1}\tau_q(g). \tag{52}$$

Considering the expressions for $M_q(g)$ and $N_q(g)$ in Equations (5) and (6), we obtain

$$\begin{aligned}\tau_q(g+1) = {} & 2^{\frac{q+1}{q+3}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} - \frac{q+1}{q+3}} \\ & \times (q+2)^{\frac{q^2+2q-1}{q+3}\left[\frac{(q+1)(q+2)}{2}\right]^{g+1} + \frac{q+1}{q+3}} \tau_q(g). \end{aligned} \tag{53}$$

By using (40) and the fact that the degree of all nodes in $\mathcal{G}_q(0)$ is $q+1$, one obtains $\tau_q(0) = \tau(\mathcal{K}_{q+2}) = (q+2)^q$. With this result, Equation (53) is solved to yield (49). ■

6. CONCLUSION

For many graph products of two graphs, one can analyze the structural and spectral properties of the resulting graph, expressing them in terms of those corresponding the two graphs. Because of this strong advantage, many authors have used graph products to generate realistic networks with cycles at different scales. In this paper, by iteratively using the edge corona product, we proposed a minimal model for complex networks called simplicial networks, which can capture group interactions in real networks, characterized by a parameter q. We then provided an extensive analysis for relevant topological properties of the model, most of which are dependent on q. We show that the resulting networks display some remarkable characteristics of real networks, such as nontrivial higher-order interaction, power-law distribution of vertex degree, small diameter and high clustering coefficient.

Furthermore, we found exact expressions for all the eigenvalues and their multiplicities of the transition probability matrix and normalized Laplacian matrix of our proposed networks. Using these obtained eigenvalues, we further evaluated mixing time, as well as mean hitting time for random walks on the networks. The former scales sublinearly with the vertex number, while the latter behaves linearly with the vertex number. The sublinear scaling of mixing time is contrary to previous knowledge that mixing time scales at most logarithmically with the vertex number. We also used the obtained eigenvalues to determine the number of spanning trees in the networks. Thus, in addition to the advantage of networks generated by other graph products, the proposed networks have another obvious advantage that both the eigenvalues and their multiplicities of relevant matrix can be analytically and exactly determined, since for previous networks created by graph products, the eigenvalues are only obtained recursively at most. The explicit expression for each eigenvalue facilitates to study those dynamical processes determined by one or several particular eigenvalues, such as mixing time considered here.

It should be mentioned that many real networks are weighted with variable edge length [26]. For example, in scientific collaboration networks, the collaboration strength between collaborators can be weighted by the number of papers they coauthored. It is thus necessary to model these realistic networks by weighted simplicial complexes [16]. In future, as the case of corona product [57], one can also define extended edge corona product of graphs and use it to build weighted scale-free networks with rich properties matching those of real-world networks [3].

DATA AVAILABILITY STATEMENT

No new data were generated or analyzed in support of this research.

ACKNOWLEDGMENTS

National Natural Science Foundation of China (61803248, 61872093, U19A2066 and U20B2051); National Key R & D Program of China (2018YFB1305104 and 2019YFB2101703); Shanghai Municipal Science and Technology Major Project (2018SHZDZX01 and 2021SHZDZX03); ZJLab; Fudan Undergraduate Research Opportunities Program (to Y.W).

REFERENCES

[1] Aldous, D. and Fill, J. (1993) Reversible Markov chains and random walks on graphs. *J. Theor. Probab.*, 2, 91–100.

[2] Barabási, A.-L. and Albert, R. (1999) Emergence of scaling in random networks. *Science*, 286, 509–512.

[3] Barrat, A., Barthelemy, M., Pastor-Satorras, R. and Vespignani, A. (2004) The architecture of complex weighted networks. *Proc. Natl. Acad. Sci. U. S. A.*, 101, 3747–3752.

[4] Barriere, L., Comellas, F., Dalfo, C. and Fiol, M.A. (2016) Deterministic hierarchical networks. *J. Phys. A*, 49, 225202.

[5] Barriere, L., Comellas, F., Dalfó, C. and Fiol, M.A. (2009) The hierarchical product of graphs. *Discrete Appl. Math*, 157, 36–48.

[6] Barrière, L., Dalfó, C., Fiol, M.A. and Mitjana, M. (2009) The generalized hierarchical product of graphs. *Discrete Math.*, 309, 3871–3881.

[7] Benson, A.R., Abebe, R., Schaub, M.T., Jadbabaie, A. and Kleinberg, J. (2018) Simplicial closure and higher-order link prediction. *Proc. Natl. Acad. Sci. U. S. A.*, 115, E11221–E11230.

[8] Bhat, U., Krapivsky, P.L., Lambiotte, R. and Redner, S. (2016) Densification and structural transitions in networks that grow by node copying. *Phys. Rev. E*, 94, 062302.

[9] Bianconi, G. and Dorogovstev, S.N. (2020) The spectral dimension of simplicial complexes: a renormalization group theory. *J. Stat. Mech*, 2020, 014005.

[10] Bianconi, G. and Ziff, R.M. (2018) Topological percolation on hyperbolic simplicial complexes. *Phys. Rev. E*, 98, 052308.

[11] Chakrabarti, D., Wang, Y., Wang, C., Leskovec, J. and Faloutsos, C. (2008) Epidemic thresholds in real networks. *ACM Trans. Inform. Syst. Secur*, 10, 13.

[12] Chen, H. and Zhang, F. (2007) Resistance distance and the normalized Laplacian spectrum. *Discrete. Appl. Math*, 155, 654–661.

[13] Chung, F. and Linyuan, L. (2002) The average distances in random graphs with given expected degrees. *Proc. Natl. Acad. Sci. U. S. A.*, 99, 15879–15882.

[14] Chung, F. (1997) Spectral graph theory, regional conference series in math. *CBMS Amer. Math. Soc*, 92.

[15] Condamin, S., Bénichou, O., Tejedor, V., Voituriez, R. and Klafter, J. (2007) First-passage times in complex scale-invariant media. *Nature*, 450, 77–80.

[16] Courtney, O.T. and Bianconi, G. (2017) Weighted growing simplicial complexes. *Phys. Rev. E*, 95, 062301.

[17] Devriendt, K. and Van Mieghem, P. (2019) The simplex geometry of graphs. *J. Complex Netw.*, 7, 469–490.

[18] Dorogovtsev, S.N., Goltsev, A.V. and Mendes, J.F.F. (2002) Pseudofractal scale-free web. *Phys. Rev. E*, 65, 066122.

[19] Frucht, R. and Harary, F. (1970) On the corona of two graphs. *Aequationes Math.*, 4, 322–325.

[20] Ghoshal, G., Zlatić, V., Caldarelli, G. and Newman, M.E.J. (2009) Random hypergraphs and their applications. *Phys. Rev. E*, 79, 066118.

[21] Girvan, M. and Newman, M.E.J. (2002) Community structure in social and biological networks. *Proc. Natl. Acad. Sci. U. S. A.*, 99, 7821–7826.

[22] Giusti, C., Pastalkova, E., Curto, C. and Itskov, V. (2015) Clique topology reveals intrinsic geometric structure in neural correlations. *Proc. Natl. Acad. Sci. U. S. A.*, 112, 13455–13460.

[23] Hatcher, A. (2002) *Algebraic Topology*. Cambridge University Press.

[24] Haynes, T.W. and Lawson, L.M. (1993) Applications of E-graphs in network design. *Networks*, 23, 473–479.

[25] Haynes, T.W. and Lawson, L.M. (1995) Invariants of E-graphs. *Int. J. Comput. Math.*, 55, 19–27.

[26] He, Z., Mao, S., Kompella, S. and Swami, A. (2015) Minimum Time Length Scheduling Under Blockage and Interference in Multi-hop Mmwave Networks. In *Proc. 2015 IEEE Global Communications Conf.*, pp. 1–7. IEEE, New York, USA.

[27] Horstmeyer, L. and Kuehn, C. (2020) Adaptive voter model on simplicial complexes. *Phys. Rev. E*, 101, 022305.

[28] Hou, Y.S. and Wai-Chee (2010) The spectrum of the edge corona of two graphs. *Electron. J. Linear Algebra*, 20, 586–594.

[29] Hunter, J.J. (2014) The role of Kemeny's constant in properties of Markov chains. *Commun. Stat. Theor. Methods*, 43, 1309–1321.

[30] Imrich, W. and Klavžar, S. (2000) *Product Graphs: Structure and Recognition*. Wiley.

[31] Jain, S. and Seshadhri, C. (2017) A Fast and Provable Method for Estimating Clique Counts using Turán's Theorem. In *Proc. 26th International Conf. World Wide Web*, Perth, Australia, 3–7 April, pp. 441–449. International World Wide Web Conferences Steering Committee Republic and Canton of Geneva, Switzerland.

[32] Jin, Y., Li, H. and Zhang, Z. (2017) Maximum matchings and minimum dominating sets in Apollonian networks and extended Tower of Hanoi graphs. *Theoret. Comput. Sci*, 703, 37–54.

[33] Kemeny, J.G. and Snell, J.L. (1976) *Finite Markov Chains*. Springer, New York.

[34] Klamt, S., Haus, U.-U. and Theis, F. (2009) Hypergraphs and cellular networks. *PLoS Comput. Biol*, 5, e1000385.

[35] Klemm, K. and Stadler, P. F. (2006) Statistics of cycles in large networks. *Phys. Rev. E*, 73, 025101.

[36] Lacopini, I., Petri, G., Barrat, A. and Latora, V. (2019) Simplicial models of social contagion. *Nat. Commun.*, 10, 2485.

[37] Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C. and Ghahramani, Z. (2010) Kronecker graphs: an approach to modeling networks. *J. Mach. Learn. Res.*, 11, 985–1042.

[38] Leskovec, J. and Faloutsos, C. (2007) Scalable Modeling of Real Graphs Using Kronecker Multiplication. In *Proc. 24th Int. Conf. Machine Learning*, Corvalis Oregon, USA, 20–24 June, pp. 497–504. Association for Computing Machinery, New York, NY, United States.

[39] Levene, M. and Loizou, G. (2002) Kemeny's constant and the random surfer. *Am. Math. Mon*, 109, 741–745.

[40] Levin, D.A., Peres, Y. and Wilmer, E.L. (2008) *Markov Chains and Mixing Times*. American Mathematical Society, Providence, RI.

[41] Lovàsz, L., Lov, L. and Of Paul Erdos (1996) Random walks on graphs: a survey. *Combinatorics*, 8, 1–46.

[42] Lv, Q., Yi, Y. and Zhang, Z. (2015) Corona graphs as a model of small-world networks. *J. Stat. Mech. Theory. E*, 2015, P11024.

[43] Mahdian, M. and Xu, Y. (2007) Stochastic Kronecker Graphs. In *Proc. 5th Int. Conf. Algorithms and Models for the Web-Graph*, San Diego, CA, 11–12 December, pp. 453–466, Springer, Berlin.

[44] Mahdian, M. and Xu, Y. (2011) Stochastic Kronecker graphs. *Random Struct. Algorithms*, 38, 453–466.

[45] Matamalas, J.T., Gómez, S. and Arenas, A. (2020) Abrupt phase transition of epidemic spreading in simplicial complexes. *Phys. Rev. Res.*, 2, 012049.

[46] Millán, A.P., Torres, J.J. and Bianconi, G. (2019) Synchronization in network geometries with finite spectral dimension. *Phys. Rev. E*, 99, 022307.

[47] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D. and Alon, U. (2002) Network motifs: simple building blocks of complex networks. *Science*, 298, 824–827.

[48] Mitzenmacher, M., Pachocki, J., Peng, R., Tsourakakis, C. and Xu, S. C. (2015) Scalable Large Near-Clique Detection in Large-Scale Networks via Sampling. In *Proc. 21th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Sydney, NSW, Australia, 10–13 August, pp. 815–824. Association for Computing Machinery, New York, NY, USA.

[49] Mohaisen, A., Yun, A. and Kim, Y. (2010) Measuring the Mixing Time of Social Graphs. In *Proc. 10th ACM SIGCOMM Conf. Internet Measurement*, 1–3 November, pp. 383–389. Association for Computing Machinery, New York, NY, USA.

[50] Newman, M.E.J. (2003) The structure and function of complex networks. *SIAM Rev.*, 45, 167–256.

[51] Noh, J.D. and Rieger, H. (2004) Random walks on complex networks. *Phys. Rev. Lett*, 92, 118701.

[52] Ollivier, Y. (2009) Ricci curvature of markov chains on metric spaces. *J. Funct. Anal.*, 256, 810–864.

[53] Parsonage, E., Nguyen, H.X., Bowden, R., Knight, S., Falkner, N. and Roughan, M. (2011) Generalized Graph Products for Network Design and Analysis. In *2011 19th IEEE Int. Conf. Network Protocols*, pp. 79–88. IEEE.

[54] Patania, A., Petri, G. and Vaccarino, F. (2017) The shape of collaborations. *EPJ Data Sci.*, 6, 18.

[55] Patel, R., Agharkar, P. and Bullo, F. (2015) Robotic surveillance and Markov chains with minimal weighted Kemeny constant. *IEEE Trans. Autom. Control*, 60, 3156–3167.

[56] Petri, G. and Barrat, A. (2018) Simplicial activity driven model. *Phys. Rev. Lett*, 121, 228301.

[57] Qi, Y., Li, H. and Zhang, Z. (2018) Extended corona product as an exactly tractable model for weighted heterogeneous networks. *Comput. J.*, 61, 745–760.

[58] Qi, Y., Yi, Y. and Zhang, Z. (2019) Topological and spectral properties of small-world hierarchical graphs. *Comput. J.*, 62, 769–784.

[59] Ravasz, E. and Barabási, A.-L. (2003) Hierarchical organization in complex networks. *Phys. Rev. E*, 67, 026112.

[60] Redner, S. (2001) *A Guide to First-Passage Processes*. Cambridge University Press.

[61] Reimann, M.W., Nolte, M., Scolamiero, M., Turner, K., Perin, R., Chindemi, G., Dłotko, P., Levi, R., Hess, K. and Markram, H. (2017) Cliques of neurons bound into cavities provide a missing link between structure and function. *Front. Comput. Neurosci*, 11, 48.

[62] Rozenfeld, H.D., Kirk, J.E., Bollt, E.M. and Ben-Avraham, D. (2005) Statistics of cycles: how loopy is your network? *J. Phys. A*, 38, 4589.

[63] Salnikov, V., Cassese, D. and Lambiotte, R. (2018) Simplicial complexes and complex systems. *Eur. J. Phys.*, 40, 014001.

[64] Sinclair, A. (1992) Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combin. Probab. Comput*, 1, 351–370.

[65] Skardal, P.S. and Arenas, A. (2019) Abrupt desynchronization and extensive multistability in globally coupled oscillator simplexes. *Phys. Rev. Lett.*, 122, 248301.

[66] Tsourakakis, C. (2015) The K Clique Densest Subgraph Problem. In *Proc. 24th Int. Conf. World Wide Web*, Florence, Italy, 18–22 May, pp. 1122–1132. ACM.

[67] Watts, D.J. and Strogatz, S.H. (1998) Collective dynamics of 'small-world' networks. *Nature*, 393, 440–442.

[68] Weichsel, P.M. (1962) The Kronecker product of graphs. *Proc. Am. Math. Soc.*, 13, 47–52.

[69] Wuchty, S., Oltvai, Z.N. and Barabási, A.-L. (2003) Evolutionary conservation of motif constituents in the yeast protein interaction network. *Nat. Genet.*, 35, 176.

[70] Yi, Y., Zhang, Z. and Patterson, S. (2020) Scale-free loopy structure is resistant to noise in consensus dynamics in complex networks. *IEEE Trans. Cybern.*, 50, 190–200.

[71] Zhang, Z., Rong, L. and Zhou, S. (2007) A general geometric growth model for pseudofractal scale-free web. *Phys. A*, 377, 329–339.

[72] Zhang, Z. and Wu, B. (2015) Pfaffian orientations and perfect matchings of scale-free networks. *Theoret. Comput. Sci*, 570, 55–69.