



Entropy and Enumeration of Subtrees in a Cactus Network

Lixin Dong¹, Haixing Zhao^{1*} and Hong-Jian Lai²

¹ School of Computer, Qinghai Normal University, Xining, China, ² Department of Mathematics, West Virginia University, Morgantown, WV, United States

For a given network, the number of spanning trees is a key parameter to measure its reliability in edge failure cases, while the number of subtrees is a key parameter to measure its reliability in both vertex and edge failures cases. Zhang et al. investigated the entropy of spanning trees, spanning forests and connected spanning subgraphs of Koch networks. In this paper, we extend Koch networks to 3-cactus networks, and study the entropy of subtrees of 3-cactus networks. We present a linear algorithm to count the number of subtrees in a 3-cactus network, determine the upper and lower bounds of the entropy of subtrees of these networks and characterize those attaining the extremal values. As an application, a linear algorithm is developed to count the number of subtrees in Koch networks, with complexity $O(g)$, where g is the number of iterations. Finally, we determine the entropy of subtrees of Koch networks.

OPEN ACCESS

Edited by:

Yang-Hui He,
City University of London,
United Kingdom

Reviewed by:

Praveen Agarwal,
Anand International College of
Engineering, India
Simon Chiossi,
Fluminense Federal University, Brazil

*Correspondence:

Haixing Zhao
h.x.zhao@163.com

Specialty section:

This article was submitted to
Mathematical and Statistical Physics,
a section of the journal
Frontiers in Physics

Received: 23 June 2020

Accepted: 24 August 2020

Published: 08 October 2020

Citation:

Dong L, Zhao H and Lai H-J (2020)
Entropy and Enumeration of Subtrees
in a Cactus Network.
Front. Phys. 8:575648.
doi: 10.3389/fphy.2020.575648

Keywords: cactus networks, Koch networks, the number of subtrees, entropy of subtrees, reliability

1. INTRODUCTION

The number of subtrees of a connected network is well-studied, as an important parameter to measure the reliability of a network for both vertex and edge failures. For a network G , let p be the probability of failure of a vertex in G , and q be the probability of failure of an edge in G . The reliability of the network G , denoted by $R(G; p, q)$, is the probability that the remaining vertices can communicate with each other. Zhao [1] showed that for given numbers m and n , among the networks with n vertices and m edges, if $p \rightarrow 0$ and $q \rightarrow 1$ with p and $1 - q$ are equivalent infinitesimals, then networks with more subtrees are more reliable. Therefore, it is of interest to develop efficient algorithms to count the number of subtrees in a network and, in a given family of networks, characterize the networks with extremal number of subtrees. Denote by $\eta(G)$ the number of subtrees of G . For a vertex $v \in V(G)$, let $\eta_v(G)$ denote the number of subtrees containing v in G . Székely and Wang [2] determined the maximum and minimum values of $\eta(G)$ and characterized all extremal networks among all trees G with given number of vertices. Yan and Yeh [3] developed a linear algorithm to count the number of subtrees of a tree. In the same paper, they also characterized all trees having diameter at least d with maximum number of subtrees, as well as all trees having maximum degree at least Δ with minimum number of subtrees. On the other hand, Kirk and Wang [4] characterized all trees with a given maximum degree that have the maximum number of subtrees. Zhang and Zhang [5] and Zhang et al. [6, 7] characterized the trees attaining the maximum and minimum numbers of subtrees among all trees with the given degree sequence. Xiao et al. [8] showed that $5 + n + 2^{n-3} \leq \eta(T) \leq 2^{n-1} + n - 1$ for any tree T with n vertices. Yang et al. [9, 10] studied the number of subtrees in spiro and polyphenyl hexagonal chains and hexagonal and phenylene chains, respectively. For more results on this topic, we refer the readers to references [11–17].

Koch networks, introduced by Zhang et al. [18], are typical fractal networks. Zhang et al. showed that Koch networks have some important properties in real life networks, such as a power law degree distribution with exponent between 2 and 3, a large clustering coefficient and a small average path length. However, the number of subtrees in Koch networks has not been determined. In this paper, we investigate the entropy and enumeration of subtrees in 3-cactus networks, which is a generalization of Koch networks. We first establish a linear algorithm to count the number of subtrees in an arbitrary 3-cactus network. Then we characterize the 3-cactus networks with upper and lower bounds of the entropy of subtrees. Finally, as an application, we obtain the entropy of subtrees in Koch networks.

2. PRELIMINARIES

We first introduce some general notation and definitions that will be used throughout the paper. Undefined notation and terminology will follow Bondy and Murty [19].

We follow Zhang et al. [18] and Wu et al. [20] to define Koch networks. Denote by $G_{m,g}$ the Koch network after g iterations, which has N_g vertices, E_g edges and L_g triangles. At $g = 0$, $G_{m,0}$ is a triangle with three vertices labeled by X , Y , and Z , respectively. This triangle is called the initial triangle, and its three vertices are called hub vertices. For $g \geq 1$, $G_{m,g}$ is obtained from $G_{m,g-1}$ by attaching m triangles to each of the three vertices in each triangle in $G_{m,g-1}$. The Koch networks for $m = 2$ are shown in **Figure 1**.

By definition, $L_0 = 1$ and for $g \geq 1$, $L_g = (3m + 1)L_{g-1}$. This implies that $L_g = (3m + 1)^g$. Likewise, it is routine to derive that $N_g = 2(3m + 1)^g + 1$ and $E_g = 3(3m + 1)^g$. More properties of Koch networks can be found in Zhang et al. [18].

To extend the notion of Koch networks, we introduce 3-cactus networks.

Definition 1. Let $\mathcal{G}(t)$ be the set of all 3-cactus networks with t triangles. We define a 3-cactus network $G_t \in \mathcal{G}(t)$ by the following recursive iterations.

Step 1: When $t = 0$, G_0 is isomorphic to K_1 , the graph with one vertex and no edges. The only vertex in G_0 is called the **initial vertex** of G_t .

Step 2: Assume that $t \geq 1$ and a 3-cactus network G_{t-1} is already generated by $(t - 1)$ iterations. Then, a 3-cactus network G_t can be obtained from G_{t-1} and a new triangle by identifying a vertex of the new triangle and a vertex of the G_{t-1} .

Let G be a subnetwork of G' . If the vertex set (or edge set) of G is a proper subset of the vertex set (or edge set) of G' , G is called a *proper subnetwork* of G' . Let $d_G(u)$ denote the degree of vertex u in G , and (v_1, v_2, v_3) denote the triangle with vertices v_1 , v_2 , and v_3 . A triangle (v_1, v_2, v_3) in G is called a *d-pendant triangle* if two of the vertices v_1, v_2, v_3 have degree 2 in G and the third vertex has degree equals d in G for some integer $d \geq 3$. A *d-pendant triangle* is also called a *pendant triangle*.

By Definition 1, a network $G_t \in \mathcal{G}(t)$ will contain a pendant triangle with a vertex of degree at least 4. Define a 3-cactus networks $G_t \in \mathcal{G}(t)$ to be in **Class I** if G_t contains a 4-pendant triangle, and in **Class II** if G_t does not contain a 4-pendant triangle. The following observations follow from Definition 1.

Observation 1. Let $t \geq 0$ be an integer, and let $G_t \in \mathcal{G}(t)$ be a network formed by Definition 1 using t iterations. Each of the following holds.

(i) The 3-cactus networks is a generalization of Koch networks, as building a Koch network $G_{m,g}$ using g iterations amounts to $(3m + 1)^g$ iterations in building a 3-cactus network.

(ii) G_t has t triangles, $2t + 1$ vertices and $3t$ edges.

(iii) If (u_0, u_1, u_2) is a pendant triangle of G_t with $d_{G_t}(u_1) = d_{G_t}(u_2) = 2$, then $t \geq 2$ and u_0 is a cut vertex of G_t .

(iv) Every Class II 3-cactus network G contains two pendant triangles (u_0, u_1, u_2) and (u_0, v_1, v_2) sharing a common vertex u_0 .

Proof: It suffices to justify (iv) as all others are direct consequences from Definition 1. Let G be a Class II 3-cactus network. By Definition 1, $G \in \mathcal{G}(t)$ for some $t \geq 3$, and if $G \in \mathcal{G}(3)$, then G has a vertex of degree 6 and all other vertices of degree 2, and so (iv) holds. Assume that $t \geq 4$ and (iv) holds for smaller values of t . By Definition 1, G contains a pendant triangle (x, z_1, z_2) with $d_G(z_1) = d_G(z_2) = 2$. Since G is of Class II, $d_G(x) > 4$. Let $G' = G - \{z_1, z_2\}$. Then $G' \in \mathcal{G}(t - 1)$. If G' is of Class II, then by induction, G' has two pendant triangles (u_0, u_1, u_2) and (u_0, v_1, v_2) sharing a common vertex u_0 . But as G is of Class II, u_0 cannot be a vertex of degree 4 in a 4-pendant triangle in G . If $x \notin \{u_0, u_1, u_2, v_1, v_2\}$, then (u_0, u_1, u_2) and (u_0, v_1, v_2) are two pendant triangles in G , and so (iv) holds and we are done. Hence we assume that $x \notin \{u_0, u_1, u_2, v_1, v_2\}$. If $x = u_0$, then (u_0, u_1, u_2) , (u_0, v_1, v_2) , and (u_0, z_1, z_2) are three pendant triangles with a common vertex u_0 , implying (iv) also. Hence we must have $x \in \{u_1, u_2, v_1, v_2\}$. By symmetry, we assume that $x = u_1$. Then the two triangles (u_0, u_1, u_2) and (u_1, z_1, z_2) sharing a common vertex $x = u_1$ with $d_G(u_1) = 4$, and so (u_1, z_1, z_2) is a 4-pendant triangle. This implies that G is of Class I, a contradiction. Therefore, we assume that G' is of Class I, and so G' has a 4-pendant triangle (u'_0, u'_1, u'_2) with $d_{G'}(u'_1) = d_{G'}(u'_2) = 2$. Since G is of Class II, we may assume that $x \in \{u'_0, u'_1, u'_2\}$. Thus either $x = u'_0$, whence G has two pendant triangles (x, u'_1, u'_2) and (x, z_1, z_2) satisfying (iv); or $x \in \{u'_1, u'_2\}$, whence G is of Class I, contrary to the assumption that G is of Class II. \square

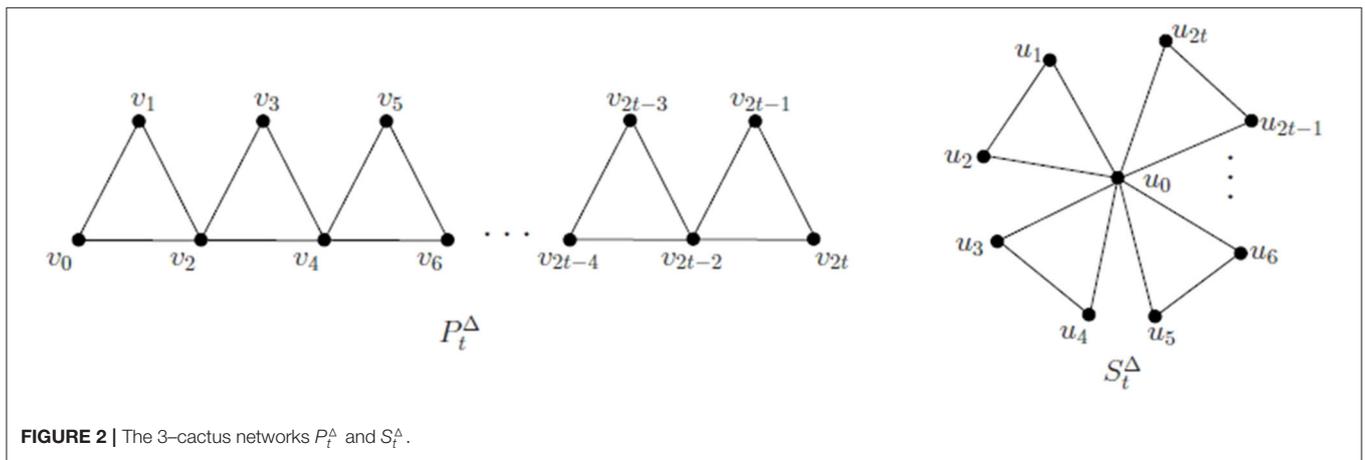
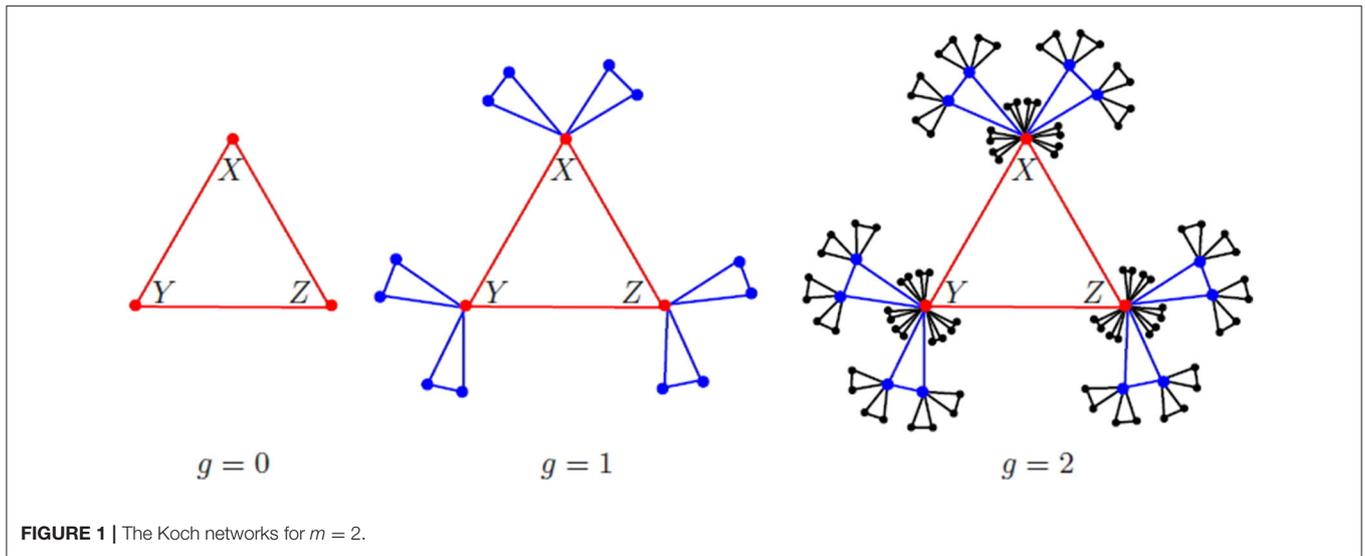
Thus every Koch network is a 3-cactus network. However, there exist 3-cactus networks that are not Koch networks, as shown by the *triangle-path* P_t^Δ and the *triangle-star* S_t^Δ are two special classes of 3-cactus networks with t triangles. Examples of these 3-cactus networks are depicted in **Figure 2**.

3. ALGORITHM FOR COUNTING THE NUMBER OF SUBTREES

In this section, we present a linear algorithm to count the number of subtrees in a 3-cactus network and its proof of the correctness.

Algorithm A: Initial condition: Let $G_t = (V, E)$ be a 3-cactus network with t triangles with the vertex set $V = \{u_0, u_1, v_1, u_2, v_2, \dots, u_t, v_t\}$ and the edge set $E = \{e_1, e_2, \dots, e_{3t}\}$, where u_i and v_i are the vertices added at the i -th iteration, for each $i = 1, 2, \dots, t$. Let $\eta(G_t)$ be the number of subtrees of G_t . For every v , we assign an ordered pair of real numbers $(1, 0)$ as the initial weight of the vertex.

Step 1. Set $k = t$.



Step 2. Let (a_1, b_1) , (a_2, b_2) , and (a_3, b_3) denote the weights of u_k , v_k and the common neighbor of u_k and v_k , respectively. Delete the vertices u_k and v_k , and reset the value of (a_3, b_3) by the following:

$$\begin{cases} a_3 := a_3(3a_1a_2 + a_1 + a_2 + 1), \\ b_3 := b_1 + b_2 + b_3 + a_1a_2 + a_1 + a_2. \end{cases} \quad (1)$$

Step 3. If $k = 1$, then stop, output $\eta(G_t) = a_3 + b_3$. Otherwise, set $k := k - 1$ and go to Step 2.

As there will be t iterations using Step 2, the complexity of Algorithm A is $O(t)$. Since running Algorithm A to a 3-cactus networks $G_t \in \mathcal{G}(t)$ can be performed in a reverse order of building G_t as described in Definition 1, it is possible to arrange the iterations in Step 2 during the running of Algorithm A so that when Algorithm A terminates, only the initial vertex of G_t is left.

Example 1. As examples, it is inspected that $\eta(G_0) = 1$. Let $G_1 = (u_0, u_1, u_2)$ be a triangle. Then G_1 has three subtrees avoiding u_0 , namely, $\{u_1\}$, $\{u_2\}$ and the complete graph on $\{u_1, u_2\}$, in G_1 that avoids the vertex u_0 , and six subtrees containing u_0 , induced

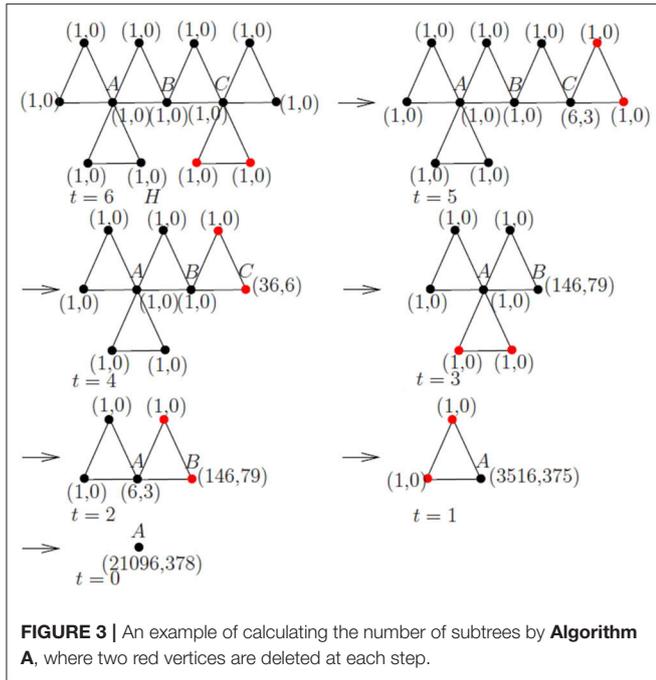
by the vertex subset $\{u_0\}$ or by the edge subsets, $\{u_0u_1\}$, $\{u_0u_2\}$, $\{u_0u_1, u_0u_2\}$, $\{u_0u_1, u_1u_2\}$, $\{u_0u_2, u_1u_2\}$. It follows that $\eta(G_1) = 3 + 6 = 9$. On the other hand, using Algorithm A, with each of u_0, u_1, u_2 assigned the initial weight $(1, 0)$, by (1) , we have $a_3 = 6$ and $b_3 = 3$. By Step 3 of Algorithm A, we also have $\eta(G_1) = 9$.

Example 2. Let H be the 3-cactus network generated by 6 iterations, as depicted in **Figure 3**. Using Algorithm A to H , the steps of calculating the number of subtrees of H are shown in **Figure 3**, which shows $\eta(H) = 21,096 + 378 = 21,474$.

Theorem 1 proves the correctness of Algorithm A.

Theorem 1. Let G_t be a 3-cactus network generated by t steps with vertex set $V = \{u_0, u_1, v_1, u_2, v_2, \dots, u_t, v_t\}$. Let (a, b) denote the weight of vertex u_0 . Each of the following holds when Algorithm A stops.

- (i) The value of a , denoted by $\eta_{u_0}(G_t)$, is the number of subtrees containing vertex u_0 in G_t .
- (ii) The value of b , denoted by $\eta(G_t - u_0)$, is the number of subtrees that do not contain vertex u_0 in G_t .



(iii) The value of $a + b$, denoted by $\eta(G_t)$, is the number of subtrees in G_t .

Proof: We argue by induction on t . If $t = 1$, then G_1 is a triangle. By Algorithm A, we have that $a = 6$ and $b = 3$, and by inspection, $\eta(G_1) = 9$. Thus, the result holds for $t = 1$. Suppose that $k \geq 2$ and the result holds for all values of $t < k$.

The structural of G_t when $t = k \geq 2$ is depicted in Figure 4, and we shall use Figure 4 and its notation to illustrate our arguments. Thus, H_1, H_2 and H_3 are the three vertex disjoint subgraphs of G_t such that H_1 contains exactly one vertex u_1 in the initial 3-cycle $G_1 = (u_0, u_1, v_1)$, H_2 contains exactly one vertex v_1 in $G_1 = (u_0, u_1, v_1)$ and H_3 contains exactly one vertex u_0 in $G_1 = (u_0, u_1, v_1)$, as described in Definition 1.

Relabel the vertices u_1, v_1 , and u_0 as w_1, w_2 , and w_3 , when each of them is considered as a vertex in H_1, H_2 , and H_3 , respectively. In the rest of the proof of this theorem, we keep in mind that $u_1 = w_1, v_1 = w_2$, and $u_0 = w_3$.

As $G_1 = (u_0, u_1, v_1)$ is an initial 3-cycle of G_t , we can view w_i as the initial vertex of the 3-cactus network H_i . For each $i \in \{1, 2, 3\}$, since each H_i is a 3-cactus network generated by at most $t - 1$ iterations, by applying Algorithm A to H_i ending at the initial vertex w_i of H_i , we obtain the weights $(a_1, b_1), (a_2, b_2), (a_3, b_3)$ of the vertices w_1, w_2 , and w_3 , respectively. By the induction hypothesis, we have, for $i \in \{1, 2, 3\}$, both (A) and (B) of the following.

- (A) a_i is the number of subtrees containing vertex w_i in H_i ,
 - (B) b_i is the number of subtrees that do not contain vertex w_i in H_i .
- (2)

At this stage of Algorithm A, the graph to be computed is $G_1 = (u_0, u_1, v_1)$ with the weights assigned to u_0, u_1, v_1 being (a_3, b_3) ,

(a_1, b_1) , and (a_2, b_2) , respectively. Running Algorithm A at the last iteration, we delete the vertices u_1 and v_1 , and reset the weight (a, b) to the vertex u_0 by (1),

$$a = a_3(3a_1a_2 + a_1 + a_2 + 1) \text{ and} \\ b = b_1 + b_2 + b_3 + a_1a_2 + a_1 + a_2. \tag{3}$$

Let \mathcal{T} denote the collection of subtrees of G_t . Define

$$\mathcal{T}_0 = \{T \in \mathcal{T} : w_3 \in V(T) \text{ and } w_1, w_2 \notin V(T)\}, \\ \mathcal{T}_3 = \{T \in \mathcal{T} : w_1, w_2, w_3 \in V(T)\}, \\ \mathcal{T}_6 = \{T \in \mathcal{T} : w_1, w_2 \in V(T) \text{ and } w_3 \notin V(T)\},$$

and for $i \in \{1, 2\}$, define

$$\mathcal{T}_i = \{T \in \mathcal{T} : w_3, w_i \in V(T) \text{ and } w_{3-i} \notin V(T)\}, \\ \mathcal{T}_{3+i} = \{T \in \mathcal{T} : w_i \in V(T) \text{ and } w_3, w_{3-i} \notin V(T)\}, \\ \mathcal{T}_7 = \{T \in \mathcal{T} : w_1, w_2, w_3 \notin V(T)\}.$$

Thus $\mathcal{T} = \cup_{i=0}^7 \mathcal{T}_i$ is a partition. By (A) and (B) in (2), $|\mathcal{T}_0| = a_3$, $|\mathcal{T}_4| = a_1$, $|\mathcal{T}_5| = a_2$ and $|\mathcal{T}_7| = b_1 + b_2 + b_3$. As the edge u_0u_1 lies in every tree in \mathcal{T}_1 , we have $|\mathcal{T}_1| = a_1a_3$. Likewise, $|\mathcal{T}_2| = a_2a_3$ and $|\mathcal{T}_6| = a_1a_2$. Since every tree in \mathcal{T}_3 contains exactly two edges in G_1 and there are three pairs of such edges in G_1 , we have $|\mathcal{T}_3| = 3a_1a_2a_3$. By definition and (3),

$$\eta_{u_0}(G_t) = \sum_{i=0}^3 |\mathcal{T}_i| = a_3 + a_1a_3 + a_2a_3 + 3a_1a_2a_3 \\ = a_3(3a_1a_2 + a_1 + a_2 + 1) = a.$$

With a similar argument and using $|\mathcal{T}_7| = b_1 + b_2 + b_3$, we have

$$\eta(G_t - u_0) = \sum_{i=4}^7 |\mathcal{T}_i| = a_1 + a_2 + a_1a_2 + b_1 + b_2 + b_3 = b.$$

It follows that

$$\eta(G_t) = \sum_{i=0}^7 |\mathcal{T}_i| = \eta_{u_0}(G_t) + \eta(G_t - u_0) = a + b.$$

This shows that when Algorithm A terminates, it outputs the values as stated in Theorem 1 (i), (ii), and (iii), and justifies the correctness of Algorithm A. \square

4. UPPER AND LOWER BOUNDS OF THE NUMBER OF SUBTREES IN A 3-CACTUS NETWORK

The following lemma [8] will be used in our proof.

Lemma 1 ([8]). *Let G be a connected network with n ($n \geq 3$) vertices and let G_1 and G_2 be two subnetworks of G such that $G = G_1 \cup G_2$, $V(G_1) \cap V(G_2) = \{v\}$ and $E(G_1) \cap E(G_2) = \emptyset$. Then*

$$\eta(G) = \eta(G_1) + \eta(G_2) + (\eta(G_1) - \eta(G_1 - v) - 1) \\ (\eta(G_2) - \eta(G_2 - v) - 1) - 1. \tag{4}$$

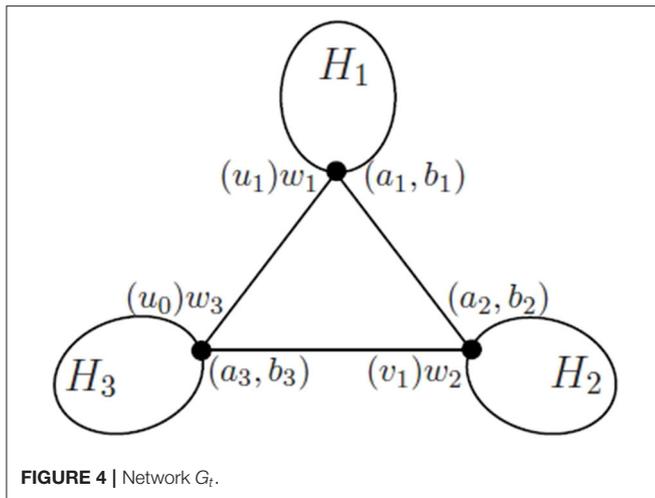


FIGURE 4 | Network G_t .

Theorem 2. For an integer t , S_t^Δ is the unique 3-cactus network with the maximum number of subtrees in $\mathcal{G}(t)$.

Proof: We prove the theorem by induction on t , and observe that the theorem holds when $t \in \{1, 2\}$. Let $k \geq 3$ be an integer and assume that the result is true for $t < k$. We consider that case when $t = k$.

Let G' be a 3-cactus network with k triangles, and let (u, v, w) be a pendant triangle such that $d_{G'}(u) \geq 4$ and $d_{G'}(w) = d_{G'}(v) = 2$. By Observation 1(iii), u is a cut vertex. Let $G'_1 = G' - \{w, v\}$ and $G'_2 = (u, v, w)$.

By the induction hypothesis, $\eta(G'_1) \leq \eta(S_{k-1}^\Delta)$, where the equality holds if and only if $G'_1 = S_{k-1}^\Delta$. If $G'_1 = S_{k-1}^\Delta$, then $G'_1 - u = (k-1)P_2$, where P_2 is the path with 2 vertices. If $G'_1 \neq S_{k-1}^\Delta$, then $(k-1)P_2$ must be a proper subnetwork of $G'_1 - u$, which implies $\eta(G'_1 - u) > \eta((k-1)P_2)$. Since $G'_2 = (u, v, w)$ and $G'_2 - u = P_2$, it follows from (4) that $\eta(G') \leq \eta(S_k^\Delta)$, where the equality holds if and only if $G' = S_k^\Delta$, which completes the proof. \square

The next lemma presents some observations which follow from Algorithm A.

Lemma 2. Let $G_t \in \mathcal{G}(t)$ be a 3-cactus network with a pendant triangle (u, v, w) such that $d_{G_t}(w) = d_{G_t}(v) = 2$, and let $G_{t-1} = G_t - \{w, v\}$. Then

- (i) $\eta(G_t) = 5\eta_u(G_{t-1}) + \eta(G_{t-1}) + 3$;
- (ii) $\eta_w(G_t) = 4\eta_u(G_{t-1}) + 2$.

\square

Lemma 3. Let $t \geq 3$ be an integer and $G \in \mathcal{G}(t)$ be a Class II 3-cactus network. Then there exists a Class I 3-cactus network $G' \in \mathcal{G}(t)$ satisfying $\eta(G') < \eta(G)$.

Proof: Let $G \in \mathcal{G}(t)$ be a Class II 3-cactus network. By Observation 1(iv), G contains two pendant triangles (u_0, u_1, u_2) and (u_0, u, w) such that $d_G(u_0) \geq 6$ and $d_G(u_1) = d_G(u_2) = d_G(u) = d_G(w) = 2$, as depicted in Figure 5A. Let H be the graph depicted in Figure 5. Denote by a_0 (b_0 , respectively) the number of subtrees containing u_0 (not containing u_0) in H . By Algorithm A, when deleting the vertices u_1 and u_2 from G , the weight of

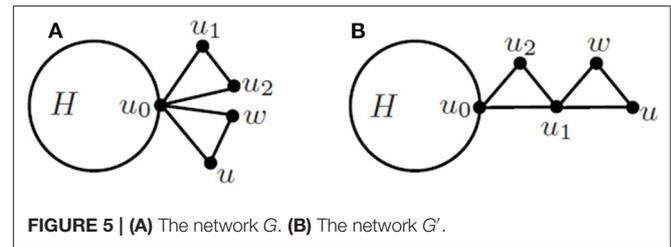


FIGURE 5 | (A) The network G . (B) The network G' .

vertex u_0 , denoted by (a_{u_0}, b_{u_0}) , can be reset as follows: $a_{u_0} = 6a_0$ and $b_{u_0} = b_0 + 3$. Then in the step of deleting the vertices u_0 and w from G , the weight of vertex u is given by $\eta_u(G) = 4a_{u_0} + 2$, $\eta(G - u) = 2a_{u_0} + b_{u_0} + 1$. Thus, we have $\eta(G) = 36a_0 + b_0 + 6$.

Let G' denote the graph depicted in Figure 5B, which is obtained from H by identifying vertex u_0 of H and a vertex of degree 2 in P_2^Δ . By Algorithm A, when deleting the vertices u_0 and u_2 from G' , the weight of vertex u_1 , denoted by (a_{u_1}, b_{u_1}) , is obtained as $a_{u_1} = 4a_0 + 2$, $b_{u_1} = 2a_0 + b_0 + 1$. Then in the step of deleting the vertices u_1 and w from G' , the weight of vertex u is given by $\eta_u(G') = 4a_{u_1} + 2$, $\eta(G' - u) = 2a_{u_1} + b_{u_1} + 1$. Thus, we have $\eta(G') = 26a_0 + b_0 + 16$. Algebraic computation yields that $\eta(G) - \eta(G') = 10a_0 - 10 > 0$, and so the lemma is proved. \square

Theorem 3. For an integer t , P_t^Δ is the unique 3-cactus network with the minimum number of subtrees in $\mathcal{G}(t)$.

Proof: By Lemma 3, within the family $\mathcal{G}(t)$, for every 3-cactus network G in Class II, there is always a 3-cactus network G' in Class I containing fewer subtrees. Thus, we only need to show that P_t^Δ is the unique 3-cactus network with the minimum number of subtrees, among all the networks of $\mathcal{G}(t)$ in Class I.

We prove the conclusion by induction on t . Let $G_t \in \mathcal{G}(t)$ be a 3-cactus network with a pendant triangle (u, v, w) such that $d_{G_t}(u) = 4$ and $d_{G_t}(w) = d_{G_t}(v) = 2$. We will prove the following results: (i) P_t^Δ is the unique 3-cactus network with the minimum number of subtrees in $\mathcal{G}(t)$; (ii) $\eta_w(G_t)$ attains the minimum value if and only if $G_t = P_t^\Delta$.

The results hold for $t = 1$ and 2. Suppose that $k \geq 3$ and the results are true for $t < k$, we consider the case when $t = k$. Let $G_{k-1} = G_k - \{w, v\}$. Since $d_{G_{k-1}}(u) = 2$, by Lemma 2 and the induction hypothesis, we have that

$$\eta_w(G_k) = 4\eta_u(G_{k-1}) + 2 \geq 4\eta_u(P_{k-1}^\Delta) + 2$$

and

$$\eta(G_k) = 5\eta_u(G_{k-1}) + \eta(G_{k-1}) + 3 \geq 5\eta_u(P_{k-1}^\Delta) + \eta(P_{k-1}^\Delta) + 3,$$

where the equalities hold if and only if $G_k = P_k^\Delta$. \square

With Algorithm A, we can compute the numbers of subtrees of S_t^Δ and P_t^Δ , respectively. Let u_0 be the center of S_t^Δ and v_0 be a vertex of degree 2 in a pendant triangle of P_t^Δ . From Algorithm A, it follows that

$$\begin{cases} \eta_{u_0}(S_t^\Delta) = 6\eta_{u_0}(S_{t-1}^\Delta), & \eta_{u_0}(S_0^\Delta) = 1, \\ \eta(S_t^\Delta - u_0) = \eta(S_{t-1}^\Delta - u_0) + 3, & \eta(S_0^\Delta - u_0) = 0, \end{cases}$$

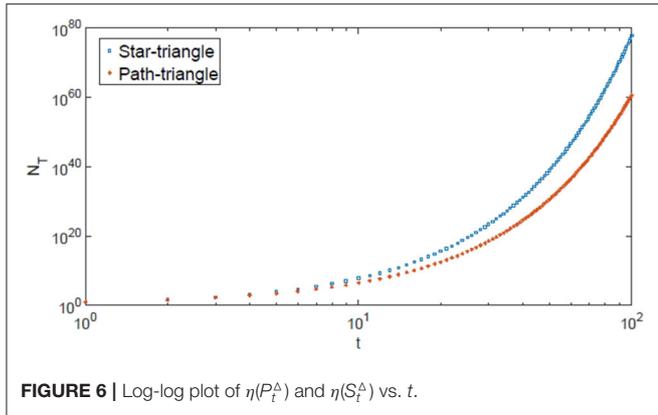


FIGURE 6 | Log-log plot of $\eta(P_t^\Delta)$ and $\eta(S_t^\Delta)$ vs. t .

and

$$\begin{cases} \eta_{v_0}(P_t^\Delta) = 4\eta_{v_0}(P_{t-1}^\Delta) + 2, & \eta_{v_0}(P_0^\Delta) = 1, \\ \eta(P_t^\Delta - v_0) = \eta(P_{t-1}^\Delta - v_0) + 2\eta_{v_0}(P_{t-1}^\Delta) + 1, & \eta(P_0^\Delta - v_0) = 0. \end{cases}$$

By algebraic manipulations, we obtain that

$$\eta(S_t^\Delta) = 6^t + 3t \tag{5}$$

and

$$\eta(P_t^\Delta) = \frac{25}{9} \times 4^t - \frac{t}{3} - \frac{16}{9}. \tag{6}$$

By Theorems 2 and 3, $\eta(S_t^\Delta)$ and $\eta(P_t^\Delta)$ represent the upper and lower bounds of the number of subtrees in a 3-cactus network with t triangles, respectively. As shown in (5) and (6), the numbers of subtrees in S_t^Δ and P_t^Δ grow exponentially as the number of triangles increases, as seen in Figure 6, where the curves of values of $\eta(S_t^\Delta)$ and $\eta(P_t^\Delta)$ vs. t are plotted.

5. THE ENTROPY OF SUBTREES OF KOCH NETWORKS

The entropy of spanning trees has been well-studied, as seen in Lyons et al. [21] and Zhang et al. [22, 23], among others. The entropy of subtrees can be similarly defined.

Definition 2. Let G be a network with $N(G)$ vertices. The entropy of subtrees of G is defined as follows:

$$E(G) = \lim_{N(G) \rightarrow \infty} \frac{\ln \eta(G)}{N(G)}.$$

Let T be a tree with n vertices. Székely and Wang [2] and Yan and Yeh[3] obtained the results of the number of subtrees in a tree as follows: $\frac{n(n+1)}{2} = \eta(P_n) \leq \eta(T) \leq \eta(K_{1,n-1}) = 2^{n-1} + n - 1$, where P_n and $K_{1,n-1}$ are the path and star with n vertices, respectively. So, we have

$$E(K_{1,n-1}) = \lim_{n \rightarrow \infty} \frac{\ln(2^{n-1} + n - 1)}{n} = \ln 2$$

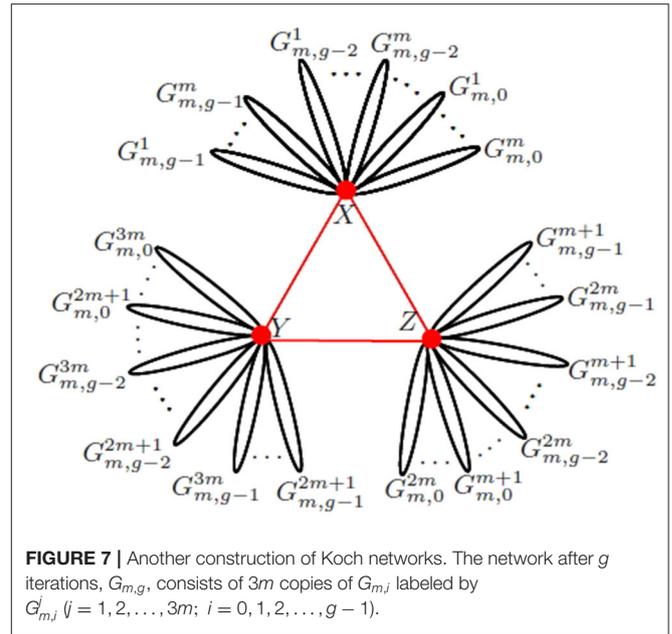


FIGURE 7 | Another construction of Koch networks. The network after g iterations, $G_{m,g}$, consists of $3m$ copies of $G_{m,i}$ labeled by $G_{m,i}$ ($i = 1, 2, \dots, 3m; i = 0, 1, 2, \dots, g - 1$).

and

$$E(P_n) = \lim_{n \rightarrow \infty} \frac{\ln \frac{n(n+1)}{2}}{n} = 0.$$

From Equations (5) and (6), we can calculate the entropies of subtrees of S_t^Δ and P_t^Δ , respectively, as follows:

$$E(S_t^\Delta) = \lim_{t \rightarrow \infty} \frac{\ln(6^t + 3t)}{2t + 1} = \frac{\ln 6}{2}$$

and

$$E(P_t^\Delta) = \lim_{t \rightarrow \infty} \frac{\ln(\frac{25}{9} \times 4^t - \frac{t}{3} - \frac{16}{9})}{2t + 1} = \ln 2.$$

Therefore, for a tree T with n vertices and a 3-cactus network G_t with t triangles, we have

$$E(P_n) \leq E(T) \leq E(K_{1,n-1}) = E(P_t^\Delta) \leq E(G_t) \leq E(S_t^\Delta).$$

Now we calculate the entropy of subtrees of Koch networks. The following definition and notation will be used in this section.

Let $\alpha(x)$ and $\beta(x)$ be two infinities when x is close to infinity. If $\lim_{x \rightarrow \infty} \frac{\alpha(x)}{\beta(x)} = 1$, we say that $\alpha(x)$ and $\beta(x)$ are equivalent infinities, and write $\alpha(x) \sim \beta(x)$ ($x \rightarrow \infty$).

For Koch networks, another construction is given as follows. The network $G_{m,g}$ is obtained from $G_{m,i}$ ($i = 0, 1, 2, \dots, g - 1$) and the initial triangle by adding m copies of $G_{m,i}$ to each of the three hub vertices of the initial triangle, as shown in Figure 7.

According to the construction, $G_{m,g}$ has the form illustrated in Figure 8, where H_g is the subnetwork of $G_{m,g}$ obtained from m copies of $G_{m,i}$ ($i = 0, 1, 2, \dots, g - 1$) by identifying mg hub

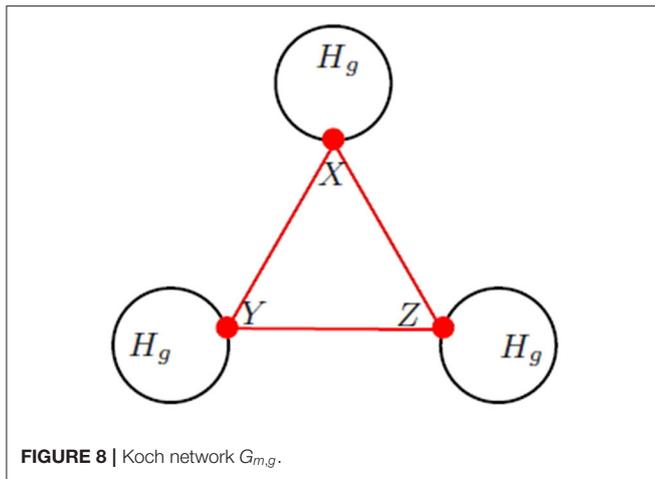


FIGURE 8 | Koch network $G_{m,g}$.

vertices X in m copies of $G_{m,i}$ ($i = 0, 1, 2, \dots, g-1$). Then $\eta_X(H_g)$ and $\eta(H_g - X)$ are given by

$$\begin{cases} \eta_X(H_g) = \left(\prod_{i=0}^{g-1} \eta_X(G_{m,i})\right)^m, \\ \eta(H_g - X) = m \sum_{i=0}^{g-1} \eta(G_{m,i} - X). \end{cases} \quad (7)$$

By Algorithm A to the initial triangle (X, Y, Z) , we get that

$$\begin{cases} \eta_X(G_{m,g}) = \eta_X(H_g)[3(\eta_X(H_g))^2 + 2\eta_X(H_g) + 1], \\ \eta(G_{m,g} - X) = 3\eta(H_g - X) + (\eta_X(H_g))^2 + 2\eta_X(H_g) \end{cases} \quad (8)$$

and initial conditions are $\eta_X(G_{m,0}) = 6$ and $\eta(G_{m,0} - X) = 3$. So, we arrive at

$$\eta(G_{m,g}) = 3\eta_X(H_g)[(\eta_X(H_g))^2 + \eta_X(H_g) + 1] + 3\eta(H_g - X). \quad (9)$$

From Equation (7), we obtain

$$\begin{cases} \eta_X(H_g) = \eta_X(H_{g-1})(\eta_X(G_{m,g-1}))^m, \\ \eta(H_g - X) = \eta(H_{g-1} - X) + m \eta(G_{m,g-1} - X). \end{cases} \quad (10)$$

According to the above theoretical analysis, we give a linear algorithm for counting the number of subtrees in a Koch network, as follows.

Algorithm B: Initial condition: Let $G_{m,g}$ be a Koch network generated by g iterations and X, Y and Z be three hub vertices in $G_{m,g}$. Let $\eta(G_{m,g})$ denote the number of subtrees of $G_{m,g}$.

Step 1. Take $k = 1, a = 6, b = 3, x = 1$ and $y = 0$.

Step 2. Suppose that, at iteration k , the number of subtrees containing X in H_k and the number of subtrees not containing X in H_k are denoted by x and y , respectively. One gets the values x and y as

$$\begin{cases} x : = x(a)^m, \\ y : = y + mb. \end{cases}$$

Step 3. Suppose that a and b denote the number of subtrees containing X in $G_{m,k}$ and the number of subtrees not containing X in $G_{m,k}$, respectively. We get a and b by

$$\begin{cases} a : = x(3x^2 + 2x + 1), \\ b : = 3y + x^2 + 2x. \end{cases}$$

Step 4. If $k = g$, then stop, output $\eta(G_{m,g}) = a + b$. Otherwise, $k : = k + 1$, go to Step 2.

It is not difficult to see that the complexity of the above algorithm is $O(g)$. In order to calculate the entropy of subtrees of Koch networks, we need the following lemma:

Lemma 4. Let $G_{m,g}$ be a Koch network generated by g iterations. Then

$$\eta(G_{m,g}) \sim 3^{(3m+1)g-1} \times 6^{3m(3m+1)g-1} \quad (g \rightarrow \infty).$$

Proof: From Equation (10), we have the following recurrence relations:

$$\begin{cases} \eta_X(H_g) = \eta_X(H_{g-1})[\eta_X(H_{g-1})(3(\eta_X(H_{g-1}))^2 + 2\eta_X(H_{g-1}) + 1)]^m, \\ \eta(H_g - X) = (3m + 1)\eta(H_{g-1} - X) + m[(\eta_X(H_{g-1}))^2 + 2\eta_X(H_{g-1})]. \end{cases}$$

It is easy to obtain $\eta_X(H_g) \sim 3^m \times (\eta_X(H_{g-1}))^{3m+1}$ ($g \rightarrow \infty$). Note that $\eta_X(H_1) = 6^m$.

It follows that

$$\eta_X(H_g) \sim 3^{\frac{1}{3}[(3m+1)g-1]} \times 6^{m(3m+1)g-1} \quad (g \rightarrow \infty). \quad (11)$$

Since $\eta(H_1 - X) = 3m$, we have

$$\begin{aligned} \eta(H_g - X) &= 3m(3m + 1)^{g-1} + m \sum_{i=1}^{g-1} (3m + 1)^{g-1-i} [(\eta_X(H_i))^2 \\ &\quad + 2\eta_X(H_i)] \\ &\leq 3m(3m + 1)^{g-1} + m(g - 1)(3m + 1)^{g-2} \\ &\quad [(\eta_X(H_{g-1}))^2 + 2\eta_X(H_{g-1})] \\ &\sim m(g - 1)(3m + 1)^{g-2} (\eta_X(H_{g-1}))^2 \quad (g \rightarrow \infty). \end{aligned} \quad (12)$$

From Equations (11) and (12), we get that $\lim_{g \rightarrow \infty} \frac{\eta(H_g - X)}{(\eta_X(H_g))^3} = 0$.

From Equation (9), it is easy to see that

$$\eta(G_{m,g}) \sim 3(\eta_X(H_g))^3 \quad (g \rightarrow \infty). \quad (13)$$

From Equations (11) and (13), we conclude that

$$\eta(G_{m,g}) \sim 3^{(3m+1)g-1} \times 6^{3m(3m+1)g-1} \quad (g \rightarrow \infty).$$

The proof is thus complete. \square

From Lemma 4, we can calculate the entropy of subtrees of Koch networks, as

$$\begin{aligned} E(G_{m,g}) &= \lim_{g \rightarrow \infty} \frac{\ln \eta(G_{m,g})}{2(3m + 1)^g + 1} \\ &= \lim_{g \rightarrow \infty} \frac{\ln[3^{(3m+1)g-1} \times 6^{3m(3m+1)g-1}]}{2(3m + 1)^g + 1} = \frac{\ln 6}{2} - \frac{\ln 2}{2(3m + 1)}. \end{aligned}$$

Next, we can compare the entropy of subtrees of Koch networks with those of P_t^Δ and S_t^Δ . Observe that $\ln 2 < \frac{\ln 6}{2} - \frac{\ln 2}{2(3m+1)} < \frac{\ln 6}{2}$ and $\lim_{m \rightarrow \infty} E(G_{m,g}) = \lim_{m \rightarrow \infty} (\frac{\ln 6}{2} - \frac{\ln 2}{2(3m+1)}) = \frac{\ln 6}{2}$. Thus, the following inequality holds.

$$E(P_t^\Delta) < E(G_{m,g}) < E(S_t^\Delta).$$

The entropy of subtrees of Koch networks reflects the fact that although the number of subtrees of Koch networks grows exponentially, the growth rate is lower than that of the triangle-star and higher than that of the triangle-path. This result suggests that if the vertex failure probability p satisfies $p \rightarrow 0$, the edge failure probability q satisfies $q \rightarrow 1$, and p and $1-q$ are equivalent infinitesimals, then Koch networks are less reliable than the triangle-star, and more reliable than the triangle-path. Noticing that $E(G_{m,g})$ can be considered as a function of the variable m , it is easy to see that the entropy of subtrees of Koch networks tends to the maximum entropy of subtrees of all 3-cactus networks as m tends to infinity.

6. CONCLUSIONS

In this paper, we have given the definition of the entropy of subtrees, which is used to compare the average number of subtrees for networks of different sizes. We have established a linear algorithm for counting the number of subtrees in any 3-cactus network, and characterized the 3-cactus networks with upper and lower bounds of the entropy of subtrees among all 3-cactus networks with t triangles. In order to avoid exponential computation, we have also proposed a linear algorithm for calculating the number of subtrees in Koch networks. Finally, we have determined the entropy of subtrees of Koch networks which tends to the maximum entropy of subtrees of all 3-cactus networks as m tends to infinity.

REFERENCES

- Zhao HX. *Research on subgraph polynomials and reliability of networks* (Ph.D. thesis), School of Computer, Northwestern Polytechnical University, Xi'an, China (2004).
- Székely LA, Wang H. On subtrees of trees. *Adv Appl Math.* (2005) **34**:138–55. doi: 10.1016/j.aam.2004.07.002
- Yan WG, Yeh Y-N. Enumeration of subtrees of trees. *Theoret Comput Sci.* (2006) **369**:256–68. doi: 10.1016/j.tcs.2006.09.002
- Kirk R, Wang H. Largest number of subtrees of trees with a given maximum degree. *SIAM J Discrete Math.* (2008) **22**:985–95. doi: 10.1137/070687736
- Zhang XM, Zhang XD. The minimal number of subtrees with a given degree sequence. *Graphs Comb.* (2015) **31**:309–18. doi: 10.1007/s00373-013-1383-z
- Zhang XM, Zhang XD, Gray D, Wang H. Trees with the most subtrees—an algorithmic approach. *J Comb.* (2012) **3**:207–23. doi: 10.4310/JOC.2012.v3.n2.a5
- Zhang XM, Zhang XD, Gray D, Wang H. The number of subtrees of trees with given degree sequence. *J Graph Theory.* (2013) **73**:280–95. doi: 10.1002/jgt.21674
- Xiao YZ, Zhao HX, Liu Z, Mao YP. Trees with large numbers of subtrees. *Int J Comput Math.* (2017) **94**:372–85. doi: 10.1080/00207160.2015.1109640
- Yang Y, Liu HB, Wang H, Fu HS. Subtrees of spiro and polyphenyl hexagonal chains. *Appl Math Comput.* (2015) **268**:547–60. doi: 10.1016/j.amc.2015.06.094
- Yang Y, Liu HB, Wang H, Deng AS, Magnant C. On algorithms for enumerating subtrees of hexagonal and phenylene chains. *Comput J.* (2017) **60**:690–710. doi: 10.1093/comjnl/bxw091
- Axenovich M, Osang G. Unavoidable subtrees. *Discrete Math.* (2013) **313**:924–30. doi: 10.1016/j.disc.2013.01.015
- Christou M, Crochemore M, Flouri T, Iliopoulos CS, Janoušek J, Melichar B, et al. Computing all subtree repeats in ordered trees. *Inform Process Lett.* (2012) **112**:958–62. doi: 10.1016/j.ipl.2012.09.001
- Li S, Wang S. Further analysis on the total number of subtrees of trees. *Electron J Comb.* (2012) **19**:P48. doi: 10.37236/2186
- Martin DM, Thatte BD. The maximum agreement subtree problem. *Discrete Appl Math.* (2013) **161**:1805–17. doi: 10.1016/j.dam.2013.02.037
- Székely LA, Wang H. Extremal values of ratios: distance problems vs. subtree problems in trees II. *Discrete Math.* (2014) **322**:36–47. doi: 10.1016/j.disc.2013.12.027
- Vince A, Wang H. The average order of a subtree of a tree. *J Combin Theory B.* (2010) **100**:161–70. doi: 10.1016/j.jctb.2009.05.006
- Wang H. *Some results on trees* (Ph.D. thesis), Department of Mathematics, University of South Carolina, Columbia, United States (2005).

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

LD, HZ, and H-JL contributed the conception and design of the study. LD organized the literature, wrote the simulation programs, and performed the design of figures. LD and HZ wrote the first draft of the manuscript. All authors contributed to the manuscript revision, read, and approved the submitted version.

FUNDING

This work was supported by National Science Foundation of China (Grant No. 11661069), the Program for Changjiang Scholars and Innovative Research Team in Universities (Grant No. IRT-15R40), the Key Laboratory of Qinghai Province (Grant No. 2013-Z-Y17), the Research Fund for the Chunhui Program of Ministry of Education of China, the Nature Science Foundation from Qinghai Province (Grant No. 2018-ZJ-718), and the Research Foundation from Qinghai Normal University (Grant No. 2018zr010).

ACKNOWLEDGMENTS

The authors were very grateful to the referees and the editor for their valuable comments and suggestions, which significantly improved the presentation of this manuscript.

18. Zhang ZZ, Gao SY, Chen LC, Zhou SG, Zhang HJ, Guan JH. Mapping Koch curves into scale-free small-world networks. *J Phys A Math Theor.* (2010) **43**:395101. doi: 10.1088/1751-8113/43/39/395101
19. Bondy JA, Murty USR. *Graph Theory With Applications*. New York, NY; North-Holland: The Macmillan Press Ltd. (1976).
20. Wu B, Zhang ZZ, Chen GR. Properties and applications of Laplacian spectra for Koch networks. *J Phys A Math Theor.* (2012) **45**:025102. doi: 10.1088/1751-8113/45/2/025102
21. Lyons R, Peled R, Schramm O. Growth of the number of spanning trees of the Erdős-Rényi giant component. *Combin Probab Comput.* (2008) **17**:711–26. doi: 10.1017/S0963548308009188
22. Zhang ZZ, Liu HX, Wu B, Zou T. Spanning trees in a fractal scale-free lattice. *Phys Rev E.* (2011) **83**:016116. doi: 10.1103/PhysRevE.83.016116
23. Zhang ZZ, Wu B, Comellas F. The number of spanning trees in Apollonian networks. *Discrete Appl Math.* (2014) **169**:206–13. doi: 10.1016/j.dam.2014.01.015

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Dong, Zhao and Lai. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.