Physica A 406 (2014) 236–243

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Enumeration of spanning trees in planar unclustered networks[☆]

Yuzhi Xiao[a,b], Haixing Zhao[b,*], Guona Hu[b], Xiujuan Ma[a,b]

[a] *School of Computer Science, ShaanXi Normal University, Xi'an, ShaanXi, 710062, PR China*

[b] *School of Computer, Qinghai Normal University, Xining, Qinghai, 810008, PR China*

HIGHLIGHTS

- Given a new linear algorithm to count the number of spanning trees of planar network.
- We obtain an upper bound and a lower bound for the numbers of spanning trees.
- We obtain the number of spanning trees of a special outerplanar network by the method.
- The counting method can be adapted for other planar networks.

ARTICLE INFO

Article history:
Received 28 August 2013
Received in revised form 2 March 2014
Available online 22 March 2014

Keywords:
Complex networks
Spanning trees
Computational complexity
Planar networks

ABSTRACT

Among a variety of subgraphs, spanning trees are one of the most important and fundamental categories. They are relevant to diverse aspects of networks, including reliability, transport, self-organized criticality, loop-erased random walks and so on. In this paper, we introduce a family of modular, self-similar planar networks with zero clustering. Relevant properties of this family are comparable to those networks associated with technological systems having low clustering, like power grids, some electronic circuits, the Internet and some biological systems. So, it is very significant to research on spanning trees of planar networks. However, for a large network, evaluating the relevant determinant is intractable. In this paper, we propose a fairly generic linear algorithm for counting the number of spanning trees of a planar network. Using the algorithm, we derive analytically the exact numbers of spanning trees in planar networks. Our result shows that the computational complexity is $O(t)$, which is better than that of the matrix tree theorem with $O(m^2t^2)$, where t is the number of steps and m is the girth of the planar network. We also obtain the entropy for the spanning trees of a given planar network. We find that the entropy of spanning trees in the studied network is small, which is in sharp contrast to the previous result for planar networks with the same average degree. We also determine an upper bound and a lower bound for the numbers of spanning trees in the family of planar networks by the algorithm. As another application of the algorithm, we give a formula for the number of spanning trees in an outerplanar network with small-world features.

☆ Research supported by the National Natural Science Foundation of China (No. 61164005), the National Basic Research Program of China (No. 2010CB334708) and the Program for Changjiang Scholars and Innovative Research Team in Universities (No. IRT1068), and the Nature Science Foundation from Qinghai Province (No. 2012-Z-943).

* Corresponding author.
E-mail address: h.x.zhao@163.com (H. Zhao).

http://dx.doi.org/10.1016/j.physa.2014.03.028
0378-4371/

1. Introduction

In recent years, a great deal of effort has been devoted to the study of subgraphs, including motifs [1,2], communities [3,4], loops [5], cliques [6], and so on. Among a variety of subgraphs, spanning trees are one of the most important. A spanning tree of a connected network is a minimal set of edges that connect every vertex [7]. The study of spanning trees is closely related to various aspects of networks, such as reliability [8,9], optimal synchronization [10] and random walks [11]. On the other hand, spanning trees have numerous connections with other interesting problems associated with networks, such as dimer coverings [12], Potts model [13,14], and the origin of fractality for fractal scale-free networks [15,16]. Thus, it is of great interest to determine the exact number of spanning trees in a given network [17].

Due to the complexity and diversity of networks, a general approach for counting spanning trees of a generic network is not available. Many efforts have been devoted to enumerating spanning trees in some specific networks by using different techniques according to their special structures [18–29]. Many of the proposed models display both high clustering and the small-world effect. But, it is well known that many real-life networks present an average path length in logarithmic scale with the number of nodes and a degree distribution which follows a power law. Often, these networks also have a modular and self-similar structure and, in some cases, usually associated with topological restrictions, their clustering is low and they are almost planar [30–34]; for examples, power grids, some electronic circuits, the Internet and some biological systems. However, investigation on the number of spanning trees in planar unclustered networks still is rare. In view of the distinct structure, as compared to networks with high clustering, it is of great interest to examine spanning trees in planar unclustered networks using a generic algorithm.

In this paper, we study general planar networks including the outerplanar graphs discussed in Refs. [28–31]. At first, we propose a generic linear algorithm to count the number of spanning trees of a general planar network. Using the algorithm, we derive analytically the exact numbers of spanning trees in planar networks. Then, we determine an upper bound and a lower bound for the numbers of spanning trees in the family of planar networks by the algorithm. Finally, we introduce a family of modular, self-similar networks with zero clustering and small-world features, and also obtain exact numbers of spanning trees. Our method and procedure for employing the decimation technique to enumerate spanning trees are generalized and can be easily extended to other planar networks.

2. Algorithm for counting spanning trees in planar unclustered networks

2.1. Algorithm description

Recall that a graph is planar if it can be drawn on the plane with no edges crossing [34].

In this paper, we consider the following planar graphs:

Let C_m be a cycle with m vertices and let $C(t)$ be a 2-connected planar graph with t cycles C_m such that any two C_m's at most have one common edge. We obtain the planar graph $C(t)$ by the following steps.

Step 1. Let $C(t)$ be a planar graph. At $t = 0$, $C(0)$ has two vertices and one generating edge, called the initial edge.
Step 2. For $t \geq 1$, $C(t)$ is obtained from $C(t-1)$ and a path of length $m-1$ by identifying the two final vertices of the path and the end vertices of an edge, called the t-th common edge, of $C(t-1)$, respectively.

Clearly, $C(t)$ has $(m-2)t+2$ vertices and $(m-1)t+1$ edges. The girth of a graph is the length of the shortest cycles. Here the girth of $C(t)$ is m.

For a graph G, $N_{ST}(G)$ denotes the number of spanning trees of G. When $m = 3$, the authors in Ref. [35] have given an algorithm for counting the number of spanning trees in $C(t)$. In this section, we generalize their results and provide a linear algorithm for counting the number of spanning trees of the planar graphs shown above for $m \geq 4$. By theoretical analysis, we give the planar graphs with minimum and maximum numbers of spanning trees in $C(t)$. The entropy of the spanning trees of the planar graphs $C(t)$ is from $\frac{\ln(m-1)}{m-2}$ to $\frac{\ln \frac{m+\sqrt{m^2-4}}{2}}{m-2}$ as t tends to infinity.

For $m \geq 4$ and $t \geq 1$, a general algorithm to count the number of spanning trees of the planar graphs $C(t)$ is given by the following algorithm.

Algorithm A. Initial condition: Let $C(t)$ be a planar graph with t cycles C_m's. Let p_i be the path of length $m-1$ added at the ith step in $C(t)$, $i = 1, 2, \ldots, t$. All edges of $C(t)$ are weighted by a pair of real numbers (1, 1). Here, $N_{ST}(t)$ denotes the number of spanning trees of $C(t)$.

Step 1. Take $i = t$.
Step 2. Let the weights of the edge of p_i be denoted by $(x_1, y_1), (x_2, y_2), \ldots, (x_{m-1}, y_{m-1})$, respectively. The weight of the ith common edge is denoted by (x_m, y_m). Deleting p_i, we get x_m and y_m by the following formulas.

$$\begin{cases} x_m := y_1x_2x_3\ldots x_{m-1}x_m + x_1y_2x_3\ldots x_{m-1}x_m + x_1x_2y_3\ldots x_{m-1}x_m + \cdots + x_1x_2x_3\ldots x_{m-1}y_m, \\ y_m := y_1x_2x_3\ldots x_{m-1}y_m + x_1y_2x_3\ldots x_{m-1}y_m + x_1x_2y_3\ldots x_{m-1}y_m + \cdots + x_1x_2x_3\ldots y_{m-1}y_m. \end{cases} \tag{1}$$

Step 3. If $i = 1$, then stop, and output $N_{ST}(t) = x_m$. Otherwise, $i := i - 1$, go step 2.