The complexity of wheel graphs with multiple edges and vertices

Abstract

In this paper, we focus on calculate the number of spanning trees of the general wheel graphs, which means the original wheel graphs adding large amount of vertices and edges. Particularly, we introduce the C-graph and deduce a new equation that computing the spanning trees by removing C-graphs instead of edges. In Addition, we test our results by Kirchhoff's matrix-tree theorem in some simple cases and provide the tree entropy of the general wheel graphs. Finally, we analyse the relation between the wheel graph and double-wheel graphs and propose the idea of calculating the spanning trees of double-wheel graphs.

Keywords: the number of spanning trees; wheel graph; iterate relations; tree entropy; double-wheel graphs; complex network

2010 Mathematics Subject Classification: 53C25; 83C05; 57N16

1 Introduction

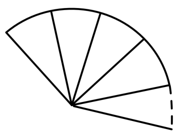
A spanning tree is a minimal connected subgraph containing all vertices of a connected graph. The spanning tree of a given graph is not unique. By traversing from different vertices, different spanning trees can be obtained. The number of spanning trees, also known as complexity, plays an important role in the development of graph theory. We denote the number of the spanning trees of one graph G by using the symbol $\tau(G)$. As a key feature of graphs, the concept of spanning trees has been applied in many areas, such as chemistry, physics, computer science and so on. Thanks to Kirchhoff's effort, a feasible way to find the number of spanning trees of graphs has been given. His matrix tree theorem[6] shows that the cofactor of a graph's Laplacian matrix is equal to its number of spanning trees.

Deriving closed formulas of the number of spanning trees for various graphs has attracted the attention of a lot of researchers. Cayley who gives the famous Cayley's theorem[1] is one of them. After that, the number of spanning trees of graphs like Sierpinski[11], Apollonian[14], ladder[5][9], fan[2], wheel[3][9] have all been found.

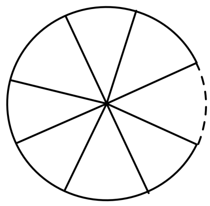
Feussner found a technical to calculate the complexity of simple graph by deleting and merging two edges. Assume G is a planar graph and e is an edge of G, we can obtain the following equation[6][4]:

$$\tau(G) = \tau(G - e) + \tau(G \cdot e)$$

where $G - e$ means spanning trees don't contain the chosen edge, and $\tau(G.e)$ represents that two vertices of this edge merge together. Based on the Feussner's iterative formula , Mohammad Hassan Shirdareh Haghighi got the Recuesive Relation of the spanning trees of the ordinary fan graph and wheel graph, and then obtain its number of spanning trees. However, the simple iterative



(a) F(n)



(b) W(n)

Figure 1: the simple fan and wheel graph

formula cannot deal with the complexity of wheel graphs with multiple edges and vertices(called the general wheel graph). Further research was conducted on the number of spanning trees for wheel and fan graphs. Hajor Sabbani found that it wound be more interesting and practical when several connections between two vertices, so he treated the number of spanning trees of multi-edge wheel graph. [10]

The number of spanning trees in general wheel graphs has a wide range of applications compared to simple wheel graphs. In a communication network, there are usually multiple connections between two components because when there is only one connection between two components. When this connection fails, it can be connected through other lines in these two components. At the same time, as the number of edges between the two points increases, the information transmission speed is faster. Therefore, the reliability and robustness of this communication device are ensured. At the same time, the spanning tree of the graph after adding points and edges can be used for the study of electrical networks, with the number of edges (or points) and the conductivity of each edge. The effective conductance between two specific vertices, x, can be written as the number of spanning trees after adding points or edges, that is, a spanning forest with exactly two components, each containing exactly one vertex x, y . Gujun Wang popularized the fan graph spanning tree and applied it to the iterative Apollonian networks. He assigned weight A to edges between two outer vertices of the wheel graph and weight B to edges between center vertex and one outer vertex, thereby calculating the number of weighted wheel graph spanning trees[13][12] and the weighted wheel graph spanning trees has widely used in random walks.[15]

In this essay, we aim to solve this problem. To begin with, we introduce a simplification method by which the spanning trees of a graph can be divided into 2 groups and we only need to discuss 2 easier ones. According to general wheel graph's structure, we make a generalization to this method.

By applying it, we are able to find an iterative relation between the wheel graph and its simplified graph, and obtain the formula depending on the iterative equation. Then, we consider double-wheel graph because it has the special qualities.[7] Eventually, we uncover the relationship of the number of spanning trees between the general wheel graphs and the double-wheel graphs, and compute the value of tree entropy. Tree entropy shows the relationship between the vertices and the number of the spanning trees.[8].

2 Construction of graphs

2.1 Related symbol

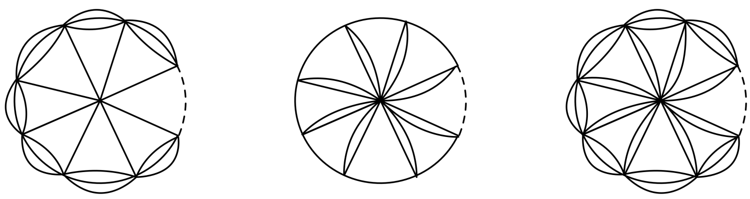
Before constructing the entire wheel graph with multiple points and edges, we need to represent the features of the entire graph. We need to represent how many edges or points there are between adjacent vertices on this graph:

- e_1: Number of edges between two outer vertices.
- e_2: Number of edges between center vertex and one outer vertex.
- d_1: The edge between two outer vertices is divided into d_1 lines.
- d_2: The edge between a center vertex is divided into d_2 lines.
- n: The general wheel graph is consisted of n general fan graphs.

We will use $W_{d_1,d_2,e_1,e_2}(n)$ to represent the general wheel graph.

2.2 The process of constructing $W_{d_1,d_2,e_1,e_2}(n)$

From the preceding part of the text, we know the basic forms of fan and wheel graphs $W_{1,1,1,1}(n)$. Now, we make the general wheel graph. First, we can add edges on the outer and central vertex of the basic wheel graph to get $W_{1,1,1,e_2}(n)$. Also, we can add edges on the adjacent outer vertices to get $W_{1,1,e_1,1}(n)$: Combining the above two cases, we can get $W_{1,1,e_1,e_2}(n)$:



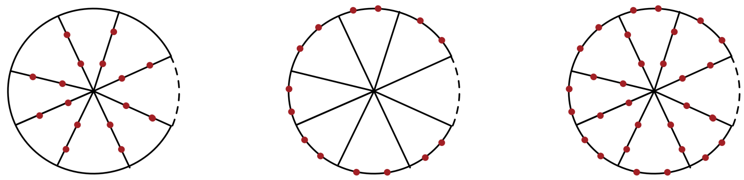
(a) $W_{1,1,1,e_2}(n)$ (b) $W_{1,1,e_1,e_2}(n)$ (c) $W_{1,1,e_1,e_2}(n)$

Figure 2: the multi-edge wheel graphs

In addition to adding edges between adjacent points, we also add a fixed number of points to the edges, which helps to check this path. We begin with the edges between the outer and central vertex and get $W_{1,d_2,1,1}(n)$. Then, we can add points on the edges between adjacent outer vertices and get $W_{d_1,1,1,1}(n)$. Combining the above two cases, we can get $W_{d_1,d_2,1,1}(n)$:

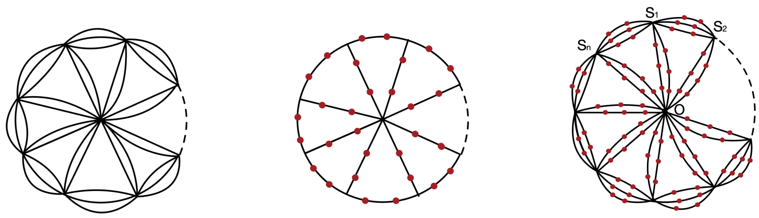
At last, we mix all the added elements together,from $W_{d_1,d_2,1,1}(n)$, and $W_{1,1,e_1,e_2}(n)$ to $W_{d_1,d_2,e_1,e_2}(n)$:

From the initial fan graph $W_{1,1,1,1}(n)$ to the general fan graph $W_{d_1,d_2,e_1,e_2}(n)$, we can clearly see that their difference is that between two vertices, from a simple line to a complex graph with



(a) $W_{1,d_2,1,1}(n)$ (b) $W_{d_1,1,1,1}(n)$ (c) $W_{d_1,d_2,1,1}(n)$

Figure 3: the wheel graphs with multi vertices on one edge



(a) $W_{1,1,e_1,e_2}(n)$ (b) $W_{d_1,d_2,1,1}(n)$ (c) $W_{d_1,d_2,e_1,e_2}(n)$

Figure 4: wheel graphs with multiple edges and vertices

multiple vertices and edges. We name the complex graph as "C-graph". Let $C_{d,e}$ be an undirected and connected graph formed by 2 vertices u, v which are linked by e identical paths with $d-1$ vertices on them. We will delete C-graph instead of a line in the next part.

2.3 Other needed graph

In the process of solving the main result, we need to deal with some associated graph because some new graphs appeared when we delete a "C-graph" of $W_{d_1,d_2,e_1,e_2}(n)$ or merging two vertices. There are four types of graphs that need to be recognized

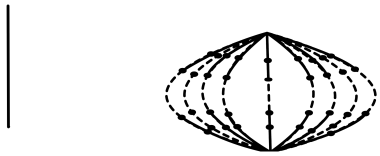
- $G_{d_1,d_2,e_1.e_2}(n)$ =$W_{d_1,d_2,e_1,e_2}(n+1) - C_{d_2,e_2}(s_1, s_n)$;
- $X_{d_1,d_2,e_1.e_2}(n)$ =$W_{d_1,d_2,e_1,e_2}(n+1) \cdot C_{d_2,e_2}(s_1, s_n)$;
- $Y_{d_1,d_2,e_1.e_2}(n)$ =$X_{d_1,d_2,e_1,e_2}(n+1) \cdot C_{d_1,e_1}(O, s_1) - L_{d_1,e_1}(O)$
- $Z_{d_1,d_2,e_1.e_2}(n)$ =$Y_{d_1,d_2,e_1,e_2}(n) - C_{d_2,e_2}(O, s_{n+1})$

L_{d_1,e_1} is a self-loop obtained by merging u, v of $C_{d,e}(u, v)$. The removal method will be provided in the following part.

3 Main results

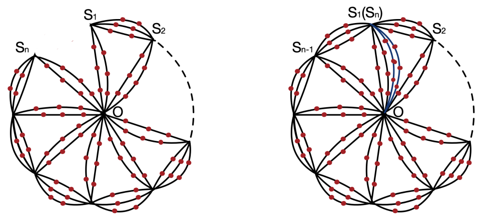
3.1 Generalization of Feussner's iterative formula

In the original wheel graph, there is only one edge between two vertices. When the number of edges and the number of points between these two vertices increase, we can see "C-graph" between two vertices). Let G be an undirected and connected graph which contains at least one C-graph, then let



(a) (simple line) (b) C-graph

Figure 5: from a simple line to a complex graph between two points



(a) $G_{d_1,d_2,e_1,e_2}(n-1)$ (b) $X_{d_1,d_2,e_1,e_2}(n-1)$

Figure 6: G graph and X graph (blue edges denote C_{d_1,e_1})

$G - C_{d,e}(u,v)$ be a new graph obtained by deleting all vertices of $C_{d,e}(u,v)$ except u and v from G, and $G \cdot C_{d,e}(u,v)$ be a new graph obtained by merging u and v in $G - C_{d,e}(u,v)$.

Lemma 3.1. For convenience, we denote the new vertex by u. The number of spanning trees of G is

$$\tau(G) = ed^{e-1}(G \cdot C_{d,e}(u,v)) + d^e \tau(G - C_{d,e}(u,v))$$

Proof. We can divide the proof into two parts:

- Merging C and G means that there is always a path between C and G that is connected, so there is a total of $\tau(C_{d,e})$ ways to merge them. Look at figure 6, the red path is connected and that is one case for spanning trees of $C_{d,e}$. We assume that there are e identical paths connecting u and v, each of these paths is divided into d parts. Except the red path which is chose arbitrarily, other points separate "C-graph" into 2 connected graphs such that u and v is not connected. Hence, we can deduce that the number of spanning trees of "C-graph" is ed^{e-1}.
$$\tau(C_{d,e}) = ed^{e-1}$$
- Due to the addition of d points and e edges in the middle of the C-graph, there are d^e methods for making the C-graph disconnected.

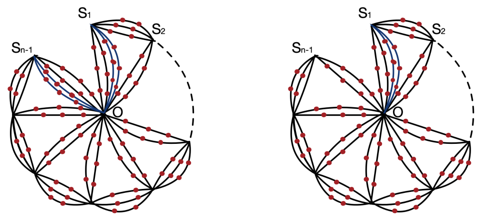
□

3.2 the complexity of general fan graph

The task of this article is to calculate the complexity of wheel graphs with multiple edges and vertices $W_{d_1,d_2,e_1,e_2}(n)$. However, the final iterative formula of the general wheel graph includes the general fan graph.

Lemma 3.2. The iterative formula of the general fan graphs is

$$\tau(\mathrm{F}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = (2\mathrm{e}_2\mathrm{d}_1^{\mathrm{e}_1}\mathrm{d}_2^{\mathrm{e}_2-1} + \mathrm{e}_1\mathrm{d}_1^{\mathrm{e}_1-1}\mathrm{d}_2^{\mathrm{e}_2})\tau(\mathrm{F}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-1)) - \mathrm{e}_2^2\mathrm{d}_1^{2\mathrm{e}_1}\mathrm{d}_2^{2\mathrm{e}_2-2}\tau(\mathrm{F}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-2))$$



(a) $Y_{d_1,d_2,e_1,e_2}(n-2)$ (b) $Z_{d_1,d_2,e_1,e_2}(n-2)$

Figure 7: Y graph and Z graph (blue edges denote C_{d_2,e_2})

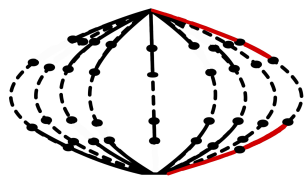
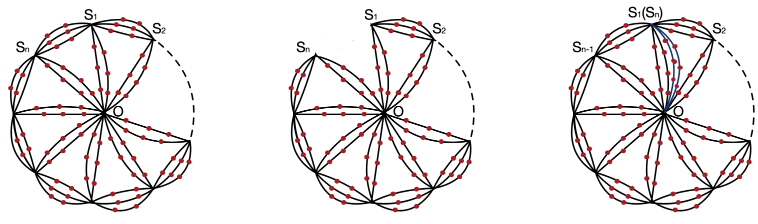


Figure 8: A spanning tree of "C-graph"

3.3 Solution

Firstly, we remove all the edges(it forms a C-graph) between the outer vertices of $W_{d_1,d_2,e_1,e_2}(n)$. According to Lemma 3.1, we can get

$$\begin{aligned}\tau(\mathrm{W}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) &= d_2^{e_2}\tau(W_{d_1,d_2,e_1,e_2}(n) - C_{d_2,e_2}(s_1,s_n)) + e_2 d_2^{e_2-1}\tau(W_{d_1,d_2,e_1,e_2}(n)\cdot C_{d_1,e_1}(s_1,s_n)) \\ &= d_2^{e_2}\tau(F_{d_1,d_2,e_1,e_2}(n-1)) + e_2 d_2^{e_2-1}\tau(X_{d_1,d_2,e_1,e_2}(n-1))\end{aligned}$$



(a) $W_{d_1,d_2,e_1,e_2}(n)$ (b) $F_{d_1,d_2,e_1,e_2}(n-1)$ (c) $X_{d_1,d_2,e_1,e_2}(n-1)$

Figure 9: Delete all the edges between S_n and S_1

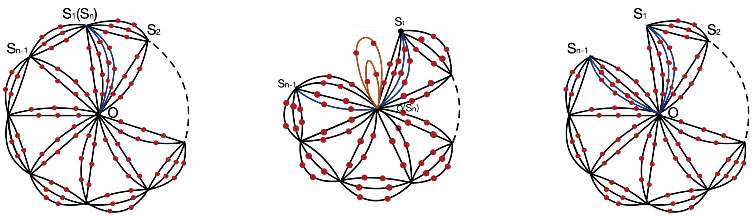
Then it forms $X_{d_1,d_2,e_1,e_2}(n-1)$, we need to remove the blue part on the X-graph(See (c) of Figure 9) because the C-graph on the blue part is a different type of C-graph compared with the other C-graph between the center vertex and the outer vertex.

$$\tau(\mathrm{X}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = d_1^{e_1}\tau(X_{d_1,d_2,e_1,e_2}(n-1) - C_{d_1,e_1}(O,s_1)) + e_1 d_1^{e_1-1}\tau(X_{d_1,d_2,e_1,e_2}(n-1)\cdot C_{d_1,e_1}(O,s_1))$$

After merging $X_{d_1,d_2,e_1,e_2}(n-1)$ and $C_{d_1,e_1}(O,s_1)$, we can get a graph with self loop(See (b) of Figure 10). The self loop is originated from the black C-graph of $X_{d_1,d_2,e_1,e_2}(n-1)$, which is between the vertices $S_1(S_n)$ to O. It is easy to prove that there are $d_1^{e_1}$ ways to make the self loop unconnected. We use $L_{d,e}$ to represent the self loop with e edges and d vertices on each edge:

$$\tau(X_{d_1,d_2,e_1,e_2}(n-1)\cdot C_{d_1,e_1}(O,s_1) = d_1^{e_1}\tau(X_{d_1,d_2,e_1,e_2}(n-1)\cdot C_{d_1,e_1}(O,s_1) - L_{d_1,e_1}(O))$$

We can get $Y_{d_1,d_2,e_1,e_2}(n-1)$ after the self cycle removed. Also, the coefficient need to be multiplied:



(a) $X_{d_1,d_2,e_1,e_2}(n-1))$ (b) the graph with a self-loop (c) $Y_{d_1,d_2,e_1,e_2}(n-1)$

Figure 10: The process of converting X graph to Y graph

$$\tau(\mathrm{X}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}) = \mathrm{d}_1^{\mathrm{e}_1}\tau(\mathrm{W}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-1)) + \mathrm{e}_1\mathrm{d}_1^{2\mathrm{e}_1-1}\tau(\mathrm{Y}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-1))$$

Next, $Y_{d_1,d_2,e_1,e_2}(n-1)$ is needed to be solved. See Figure 11, it includes a blue C-graph and a green C-graph, which is originally from the outer side.

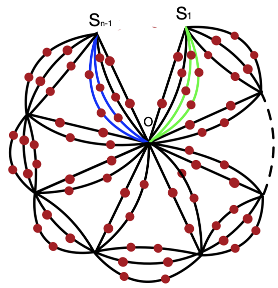


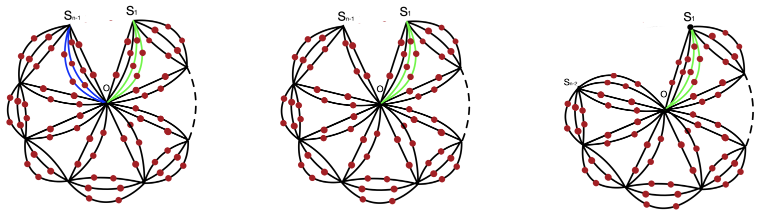
Figure 11: $Y_{d_1,d_2,e_1,e_2}(n)$

We remove the blue C-graph of $Y_{d_1,d_2,e_1,e_2}(n-1)$ to get the relationship with $Y_{d_1,d_2,e_1,e_2}(n-1)$ and $Z_{d_1,d_2,e_1,e_2}(n)$.The method for removing self loops is the same as above:

$$\begin{aligned}\tau(\mathrm{Y}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) &= d_2^{e_2}\tau(Y_{d_1,d_2,e_1,e_2}(n) - C_{d_2,e_2}(O,s_{n+1})) + e_2 d_2^{e_2-1}\tau(Y_{d_1,d_2,e_1,e_2}(n)\cdot C_{d_2,e_2}(O,s_{n+1}))\\ &= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(n)) + e_2 d_1^{e_1} d_2^{e_2-1}\tau(Y_{d_1,d_2,e_1,e_2}(n)\cdot C_{d_2,e_2}(O,s_{n+1}) - L_{d_1,e_1}(O))\\ &= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(n)) + e_2 d_1^{e_1} d_2^{e_2-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1))\end{aligned}$$

At last, we delete the green C-graph of $Z_{d_1,d_2,e_1,e_2}(n)$. $F_{d_1,d_2,e_1,e_2}(n-1)$ and $Y_{d_1,d_2,e_1,e_2}(n)$ can represent $Z_{d_1,d_2,e_1,e_2}(n-1)$ while the result of $F_{d_1,d_2,e_1,e_2}(n-1)$ has been known in Lemma 3.2.

$$\tau(\mathrm{Z}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = \mathrm{d}_2^{\mathrm{e}_2}\tau(\mathrm{F}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-1)) + \mathrm{e}_2\mathrm{d}_1^{\mathrm{e}_1}\mathrm{d}_2^{\mathrm{e}_2-1}\tau(\mathrm{Z}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n}-1))$$



(a) $Y_{d_1,d_2,e_1,e_2}(n)$ (b) $Z_{d_1,d_2,e_1,e_2}(n)$ (c) $Y_{d_1,d_2,e_1,e_2}(n-1)$

Figure 12: Delete the blue part of $Y_{d_1,d_2,e_1,e_2}(n)$

Then, we can get the following relationship(the specific proof process is in appendix)

$$e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1)) = d_2^{e_2}\tau(F_{d_1,d_2,e_1,e_2}(n)), n = 2,3,\dots$$

Based on the above proof process, by combining the obtained equations, we can obtain the following equation system

$$\begin{cases} \tau(\mathrm{W}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = d_2^{e_2}\tau(F_{d_1,d_2,e_1,e_2}(n-1)) + e_2 d_2^{e_2-1}\tau(X_{d_1,d_2,e_1,e_2}(n-1)) \\ \tau(\mathrm{X}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = d_1^{e_1}\tau(W_{d_1,d_2,e_1,e_2}(n-1)) + e_1 d_1^{2e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1)) \\ \tau(\mathrm{Y}_{\mathrm{d}_1,\mathrm{d}_2,\mathrm{e}_1,\mathrm{e}_2}(\mathrm{n})) = \dfrac{d_2^{e_2}}{e_1 d_1^{e_1-1}}\tau(F_{d_1,d_2,e_1,e_2}(n+1)) \end{cases}$$

After eliminating the $\tau(X_{d_1,d_2,e_1,e_2}(n))$ and $(Y_{d_1,d_2,e_1,e_2}(n))$ variables, we can obtain the final iterative result, when $n \geq 4$:

$$\begin{aligned} \tau(W_{d_1,d_2,e_1,e_2}(n)) = d_2^{e_2}\tau(F_{d_1,d_2,e_1,e_2}(n-1)) + e_2 d_1^{e_1} d_2^{e_2-1}\tau(W_{d_1,d_2,e_1,e_2}(n-1)) \\ + e_2 d_1^{e_1} d_2^{2e_2-1}\tau(F_{d_1,d_2,e_1,e_2}(n-2)) \end{aligned}$$

then calculate the number of spanning trees of the general wheel graph through iterative equations:

$$\begin{aligned} \tau(W_{d_1,d_2,e_1,e_2}(n)) =& 2^{-n}(d_1^{e_1-1}d_2^{e_2-1})^n[(e_1d_2 + 2d_1e_2 + \sqrt{e_1d_2(e_1d_2+4d_1e_2)})^n + \\ & (e_1d_2 + 2d_1e_2 - \sqrt{e_1d_2(e_1d_2+4d_1e_2)})^n] - 2(d_1^{e_1}d_2^{e_2-1}e_2)^n \end{aligned}$$

3.4 The result of the double-wheel graph

Double-wheel graph is a graph obtained by combining the central point of 2 wheel graphs(See Figure 13). It can be used in many areas since it has been proved it is graceful including in radio astronomy, X-ray crystallography, cryptography, and experimental design. Assume that DW is a double-wheel graph basing on wheel graph W_1 and W_2, then the number of spanning trees of DW is

$$\tau(DW) = \tau(W_1)\tau(W_2)$$

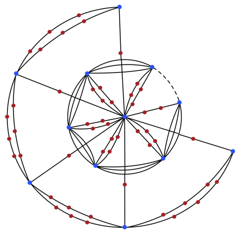


Figure 13: Double-wheel graph

4 Analysis

4.1 Result test

Let d_1, d_2, e_1, e_2 all be 1, then we get the iterative equation about the number of the spanning trees of the simplest wheel graph.

$$\tau(\mathrm{W}_{1,1,1,1}(\mathrm{n})) = \tau(\mathrm{G}_{1,1,1,1}(\mathrm{n}-1)) + \tau(\mathrm{W}_{1,1,1,1}(\mathrm{n}-1)) + \tau(\mathrm{G}_{1,1,1,1}(\mathrm{n}-2))$$

which is the same as the result given by Mohammad Hassan Shirdareh Haghighi. Also, when $e_1 = 1, d_1 = 1, d_2 = 1$, we can get

$$\tau(\mathrm{W}_{1,1,1,\mathrm{e}_2}(\mathrm{n})) = \tau(\mathrm{F}_{1,1,1,\mathrm{e}_2}(\mathrm{n}-1)) + \mathrm{e}_2\tau(\mathrm{W}_{1,1,1,\mathrm{e}_2}(\mathrm{n}-1)) + \mathrm{e}_2\tau(\mathrm{F}_{1,1,1,\mathrm{e}_2}(\mathrm{n}-2))$$

Hajor Sabbani proved this conclusion, which is consistent with our conclusion

4.2 Tree entropy

The quantity $h = \lim\limits_{n\to\infty} \frac{ln(\tau(X_n))}{|VX_n|}$ shows the the asymptotic tree number entropy of the graph sequence X_n. where VX_n is the number of the vertices of the graph. In $W_{d_1,d_2,e_1,e_2}(n)$, we can find nC_{d_2,e_2} and nC_{d_1,e_1}. So $|VX_n|$ is $n(e_1(d_1-1)+e_2(d_2-1)+1)+1$. For the general wheel graph, $h = \frac{ln(\frac{1}{2}d_1^{e_1-1}d_2^{e_2-1}(d_2e_1+2d_1e_2+\sqrt{d_2e_1(d_2e_1+4d_1e_2)}))}{e_1(d_1-1)+e_2(d_2-1)+1}$.

5 Conclusion

In this paper we find the number of the spanning trees of the general wheel graphs by using the method of iteration and the formula. We deduce based on the famous Feussner's equation. This approach is also suitable for solving the double-wheel graph, and then we give our corresponding answer. Additionally, we calculate the value of tree entropy of the general wheel graphs and provide some simple verified cases in Analysis part.

6 Appendix

$$e_1d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1)) = d_2^{e_2}\tau(G_{d_1,d_2,e_1,e_2}(n)), n = 2, 3, \ldots$$

Proof. By definition, we get

$$\tau(Y_{d_1,d_2,e_1,e_2}(1)) = d_2^{e_2}\tau(Y_{d_1,d_2,e_1,e_2}(1) - C_{d_2,e_2}(O,s_2)) + e_2 d_2^{e_2-1}\tau(Y_{d_1,d_2,e_1,e_2}(1)\cdot C_{d_2,e_2}(O,s_2))$$
$$= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{e_1} d_2^{e_2-1}\tau(Y_{d_1,d_2,e_1,e_2}(1)\cdot C_{d_2,e_2}(O,s_2) - L_{d_1,e_1}(O))$$
$$= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{e_1} d_2^{e_2-1}(d_1^{e_1}\tau(Y_{d_1,d_2,e_1,e_2}(1)\cdot C_{d_2,e_2}(O,s_2) - L_{d_1,e_1}(O) -$$
$$C_{d_1,e_1}(O,s_1)) + e_1 d_1^{e_1-1}\tau((Y_{d_1,d_2,e_1,e_2}(1)\cdot C_{d_2,e_2}(O,s_2) - L_{d_1,e_1}(O))\cdot C_{d_1,e_1}(O,s_1)))$$
$$= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{e_1} d_2^{e_2-1}(d_1^{e_1}\tau(C_{d_2,2e_2}(O,s_1)) + e_1 d_1^{e_1-1}\tau(L_{d_2,2e_2}(O))$$
$$= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{e_1} d_2^{e_2-1}(2e_2 d_1^{e_1} d_2^{2e_2-1} + e_1 d_1^{e_1-1} d_2^{2e_2})$$
$$= d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{2e_1-1} d_2^{3e_2-2}(2e_2 d_1 + e_1 d_2)$$

By lemma 1,

$$\begin{aligned}\tau(G_{d_1,d_2,e_1,e_2}(2)) &= e_1 d_1^{e_1-1}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_2 d_1^{e_1} d_2^{e_2-1}\tau(G_{d_1,d_2,e_1,e_2}(1)) \\ &= e_1 d_1^{e_1-1}\tau(Z_{d_1,d_2,e_1,e_2}(1)) + e_1 e_2 d_1^{3e1-2} d2^{2e2-2}(2d_1 e_2 + e_1 d_2)\end{aligned}$$

Therefore, it holds $e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(1)) = d_2 z\ e_2\tau(G_{d_1,d_2,e_1,e_2}(2))$.

Assume that it holds $e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-2)) = d_2^{e_2}\tau(G_{d_1,d_2,e_1,e_2}(n-1))$, then we get:

$$\begin{aligned}e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1)) &= e_1 d_1^{e_1-1} d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(n-1)) + e_2 d_1^{e_1} d_2^{e_2-1} * e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-2)) \\ &= e_1 d_1^{e_1-1} d_2^{e_2}\tau(Z_{d_1,d_2,e_1,e_2}(n-1)) + e_2 d_1^{e_1} d_2^{2e_2-1}\tau(G_{d_1,d_2,e_1,e_2}(n-1)) \\ &= d_2 e_2\tau(G_{d_1,d_2,e_1,e_2}(n))\end{aligned}$$

Thus, $e_1 d_1^{e_1-1}\tau(Y_{d_1,d_2,e_1,e_2}(n-1)) = d_2 e_2\tau(G_{d_1,d_2,e_1,e_2}(n)), n = 2, 3, \ldots$. □

References

[1] Arthur Cayley. *A theorem on trees*, volume 23. Cambridge University Press, 1889.

[2] s.N. Daoud. The deletion-contraction method for counting the number of spanning trees of graphs. *The European Physical Journal Plus*, 130, 10 2015.

[3] s.N. Daoud. *Complexity of graphs generated by wheel graph and their asymptotic limits*, volume 25. 08 2017.

[4] W. Feussner. Zur berechnung der stromstärke in netzförmigen leitern. *Annalen der Physik*, 320(12):385–394, 1904.

[5] Mohammad Haghighi and Khodakhast Bibak. Recursive relations for the number of spanning trees. *Applied Mathematical Sciences (Ruse)*, 3, 01 2009.

[6] G. Kirchhoff. *Ueber die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Vertheilung galvanischer Ströme geführt wird.* 1847.

[7] Ronan Le Bras, Carla P. Gomes, and Bart Selman. Double-wheel graphs are graceful. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, page 587–593. AAAI Press, 2013.

[8] Russell Lyons. Asymptotic enumeration of spanning trees, 2005.

[9] Fei Ma and Bing Yao. An iteration method for computing the total number of spanning trees and its applications in graph theory. *Theoretical Computer Science*, 708, 11 2017.

[10] Hajar Sahbani and Mohamed El Marraki. On the number of spanning trees in graphs with multiple edges. *Journal of Applied Mathematics and Computing*, 55(1):245–255, 2017.

[11] Masato Shinoda, Elmar Teufl, and Stephan G. Wagner. Uniform spanning trees on sierpinski graphs. *arXiv: Probability*, 2013.

[12] Elmar Teufl and Stephan Wagner. The number of spanning trees in self-similar graphs. *Annals of Combinatorics*, 15(2):355–380, 2011.

[13] Gujun Wang and Feng Zhu. Counting spanning trees of generalized n-edges apollonian networks. *International Journal of Modern Physics C*, 0(0):2350119, 0.

[14] Zhongzhi Zhang, Bin Wu, and Francesc Padró. The number of spanning trees in apollonian networks. *Discrete Applied Mathematics*, 169, 09 2012.

[15] Feng Zhu, Meifeng Dai, Yujuan Dong, and J. Liu. Random walk and first passage time on a weighted hierarchical network. *International Journal of Modern Physics C*, 25:1–10, 2014.