

Spanning Trees of Recursive Scale-Free Graphs

C. Tyler Diggans,^{1,2,3} Erik M. Boltt,^{1,4} and Daniel ben-Avraham^{1,2}

¹Clarkson Center for Complex Systems Science, Clarkson University, Potsdam, NY 13699

²Department of Physics, Clarkson University, Potsdam, NY 13699

³Air Force Research Laboratory: Information Directorate, Rome, NY 13441

⁴Department of Electrical and Computer Engineering, Clarkson University, Potsdam, NY 13699

We present a link-by-link rule-based method for constructing all members of the ensemble of spanning trees for any recursively generated, finitely articulated graph, such as the DGM net. The recursions allow for many large-scale properties of the ensemble of spanning trees to be analytically solved exactly. We show how a judicious application of the prescribed growth rules selects for certain subsets of the spanning trees with particular desired properties (small-world or extended diameter, degree distribution, etc.), and thus provides solutions to several optimization problems. The analysis of spanning trees enhances the usefulness of recursive graphs as sophisticated models for everyday life complex networks.

I. INTRODUCTION

Recursive, finitely articulated models of complex networks mimic many key properties of real-world networks. At the same time, these models are amenable to *exact* analysis, providing important insights into the nature of everyday life complex networks. In this letter, we revisit the subject of spanning trees for recursive scale-free nets more generally, and provide a method for generating all spanning trees, enabling a more complete analysis of this important set of subgraphs.

The method presented here is applicable to any recursive generative network model that produces finitely articulated graphs, but we focus on the DGM net [1] as a representative example of this class and denote the ν^{th} generation of this net as $F_\nu(1, 2)$ in our notation [2–4].

The DGM net was first introduced as a pseudofractal scale-free web (PSW) in [1], and is equivalent to the (1, 2)-flower graph [2]. It is a small-world complex network with hierarchical structure, a scale-free degree distribution with power law exponent $\gamma \approx 2.57$, and a high clustering coefficient of $C = 4/5$, matching several characteristic benchmarks of real-life complex nets. Its recursive nature and the finite articulation have enabled exact analytical studies of the statistics of cycles [5], diffusion [6], percolation [3], spectral properties [7], and minimum dominating sets [8], shedding light on the analogous properties in real-life complex nets.

There are two equivalent ways to construct the DGM net: a hub pasting approach and a link-by-link growth process [1, 2]. Previous work has exploited the hub pasting approach of constructing this and similar models to enumerate the spanning trees (and calculate the subsequent *tree entropy* [9] of the ensembles) [10–13], but additional analysis is made possible by turning to the link-based approach.

We provide an explicit process for efficiently generating the whole ensemble of spanning trees using the same recursive link-by-link processes that create the original network, except that the links are assigned an R or G , which stand for “real” or “ghost” respectively. We use

the term ghost to indicate that the link is part of the original graph, but not part of the spanning tree, and we represent these links by dashed lines in all the figures.

Furthermore, specific trees with optimal characteristics can be selected (with high probability) by heuristically (and/or stochastically) applying the link-based growth rules, solving various optimal spanning tree problems, which are NP-hard in general [14–16].

The Minimum Spanning Tree (MST) problem on weighted networks has garnered a lot of attention [17], because of its relevance to the famed *traveling salesman* problem [18, 19], as well as useful applications in data science [20–26]. However, the MST is not clearly defined for unweighted networks. Common generalizations of this optimization problem for the undirected and unweighted case include the dense, sparse, and minimum routing cost spanning tree problems. Many other variations of optimal spanning trees exist [15, 16, 27–36], and such trees are relevant to an ever widening diversity of applications from optical network design [14, 37] to networked oscillator synchronization [38].

We show that for recursive, finitely articulated networks, such optimal spanning trees (and indeed the whole ensemble of spanning trees) can be created alongside the original network with minimal overhead.

However, prior to exploring the main contribution of this work, we mention two points that have not previously been discussed, which are best addressed within the hub pasting paradigm.

II. HUB PASTING AND TYPICAL DIAMETER

Select (with replacement) three members from the ensemble of trees from generation $\nu - 1$ and identify them at the hubs, as illustrated in Fig 1 (a). A cycle forms because of this identification (shown in bold). To obtain a spanning tree of $F_\nu(1, 2)$, remove one of the links of this cycle. The ambiguity of this link removal complicates the enumeration of spanning trees, but provides insight into the typical tree diameter or span.

In [5], the typical cycle length of $F_\nu(1, 2)$ was shown to scale as 2^ν . Thus, the typical spanning tree of $F_\nu(1, 2)$ contains a path of at least of order 2^ν . Given that the maximum diameter of any tree is also of order 2^ν (see below), we conclude that the typical tree has a diameter of order 2^ν . It is interesting that while the $(1, 2)$ -flower is small-world, the diameter of the typical spanning tree scales like $\sim 2^\nu \sim N^{\log_3 2}$ (where the order of $F_\nu(1, 2)$ is $N = \frac{1}{2}3^\nu + \frac{3}{2} \sim 3^\nu$) and is therefore *not* small-world (section IV shows how to ensure selection of a small-world spanning tree).

Additionally, we note that the “copy machine” algorithm outlined in Appendix C of [6] can be adapted to efficiently generate adjacency matrices of spanning trees of $F_\nu(1, 2)$.

III. LINK-BY-LINK CONSTRUCTION

The *link-by-link* approach to spanning tree construction will mirror the general link-by-link construction process of the recursive net in question, while keeping track of two different kinds of links: “real” links (R) that belong to the spanning tree, and “ghost” links (G) that belong to the original network, but not the spanning tree.

For the example of $F_\nu(1, 2)$, at each generation, an R -link is either replaced by a G -link and two R -links (Rule # 1 in Fig. 1 (b)), or the R -link is replaced by an R -link and a path of one R -link and one G -link in either order (Rule # 2). Similarly, a G -link is always replaced by a G -link and a path of one R -link and one G -link (Rule # 3). Other recursive scale-free network models such as the generic (u, v) -flower graph [2] or a similar construction described in [39] may require a larger set of rules with more cases to consider, making the subsequent analysis more tedious (as seen below), but the process is, in essence, the same.

Denoting by R_ν (G_ν) the number of real (ghost) links in generation ν , the construction rules in Fig. 1 imply

$$\begin{aligned} R_{\nu+1} &= 2R_\nu + G_\nu, & R_0 &= 1 \\ G_{\nu+1} &= R_\nu + 2G_\nu, & G_0 &= 0, \end{aligned} \quad (1)$$

leading to $R_\nu = \frac{1}{2}(3^\nu + 1)$ and $G_\nu = \frac{1}{2}(3^\nu - 1)$. This is consistent with the fact that $R_\nu = N - 1$ (where N is the order of $F_\nu(1, 2)$) and $G_\nu + R_\nu = M = 3^\nu$ (M is the size, or number of links, of $F_\nu(1, 2)$) [1].

Each sequence of link updatings results in a single unique spanning tree; moreover, each updating rule is uniquely reversible. Finally, the reverse rules applied (in an appropriate manner) to an arbitrary spanning tree of $F_\nu(1, 2)$ generate a spanning tree of $F_{\nu-1}(1, 2)$ (i.e. no loops are created, and neither is the tree disconnected by the reverse rule application). It follows that there is a one-to-one correspondence between spanning trees and valid generating sequences of rules, and therefore, these rules generate the complete set of labeled spanning trees.

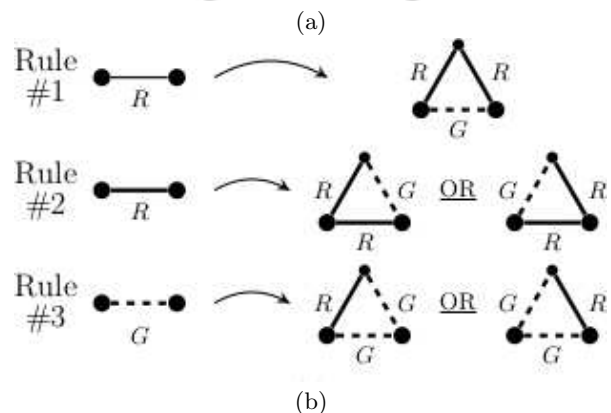
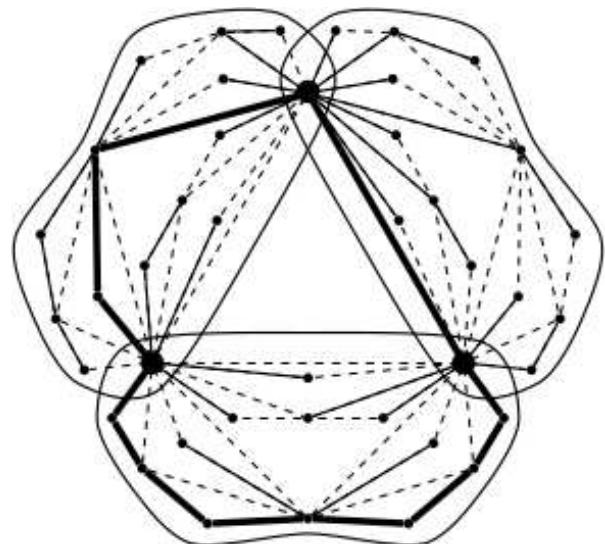


FIG. 1: As with the regular flower graph, there are two methods of constructing spanning trees of $(1, 2)$ -flower graphs. (a) The hub pasting approach (here requiring an additional step) entails choosing $1 + 2 = 3$ elements from the previous generation of spanning trees and identifying pairs of hub nodes (as shown; this, however, results in a single cycle (bolded), which must be broken by removing one of its links. (b) The *link-by-link* approach involves applying one of three rules to each link in each generation recursively. Each sequence of valid rule operations results in a unique spanning tree, allowing for the enumeration *and generation* of all spanning trees in the ensemble.

Let T_ν be the number of labeled spanning trees of generation ν . Then,

$$T_{\nu+1} = (3^{R_\nu} \cdot 2^{G_\nu}) T_\nu, \quad (2)$$

where the term in parenthesis accounts for the 3 (2) ways of updating R (G) links. Together with the expressions for R_ν and G_ν and the initial condition of $T_0 = 1$, we get

$$T_\nu = \left(\frac{3}{2}\right)^{\frac{\nu}{2}} 6^{(3^\nu - 1)/4}.$$

In terms of the order of the graph, N ,

$$T_\nu = \left(\sqrt{\frac{3}{2}} \right)^{\log_3(2N-3)} 6^{\left(\frac{N}{2}-1\right)} \sim \frac{2^C}{6} N^C \sqrt{6}^N,$$

where $C = (1 - \log_3(2))/2$ is a constant. Although a different numbering convention for the generation was used, this confirms the result found in [10].

In comparison, the total number of all labeled trees of order N is N^{N-2} [40]. Amusingly, the functional dependence of T_ν with N is closer to that of *unlabeled* trees of order N [41].

Consider now the general (u, v) -flower. Letting $w = u + v$ (note the difference in v and ν), we obtain the recursion for the enumeration of all labelled spanning trees [analogous to Eq. (2)]: $T_{\nu+1} = w^{R_\nu} \cdot (uv)^{G_\nu} T_\nu$ with $T_0 = 1$, which leads to

$$T_\nu = w^{\frac{w-2}{w-1} \left(\frac{w^\nu-1}{w-1} + \frac{\nu}{w-2} \right)} \cdot (uv)^{\frac{1}{w-1} \left(\frac{w^\nu-1}{w-1} - \nu \right)}.$$

This gives the tree entropy [9] of the (u, v) -flower:

$$\lim_{N \rightarrow \infty} \frac{\ln T_\nu}{N} = \frac{\ln(w)}{w-1} + \frac{w-2}{w-1} \ln(uv).$$

In particular, the $(1, 2)$ -flower is then the one with the least entropy [10].

IV. SPANNING TREE SUBSETS

In many applications of spanning trees, one is not concerned with simply the number of trees, but one seeks an optimal spanning tree for some purpose, e.g., minimal branching vertex spanning tree (MBVST), dense spanning tree (DST), etc. Within the process outlined, such optimal trees can be selected for from various subsets of the ensemble of spanning trees, which share certain properties, by restricting how the rules are applied.

For example, consider the maximum diameter of any spanning tree from the ensemble in generation ν . Denoting this ensemble property by s_ν^{\max} , the recursion $s_{\nu+1}^{\max} = 2s_\nu^{\max} + 2$ is obtained, since there will be an additional ghost link (which extends the span by one) on either end of the joined pair of spans from the previous generation. Using the initial condition $s_2^{\max} = 5$ (the first generation for which the necessary two G -links exist) yields a maximum diameter of $s_\nu^{\max} = 7 \cdot 2^{\nu-2} - 2$.

If rule # 1 is applied to all R links (excluding rule # 2) then the diameter scales $\sim 2^\nu$. Further reducing the ambiguity of rule #3 by choosing the R link to extend the diameter whenever possible (e.g. choosing the R link to be adjacent to nodes with smaller degree), a member of the subset of spanning trees with maximal diameter is obtained, see Fig. 2 (a) for one of eight such spanning trees of $F_4(1, 2)$ with $s_4^{\max} = 26$. In fact, the application of any finite proportion of rule # 1 to R -links will result in *non* small-world spanning tree of these small-world networks.

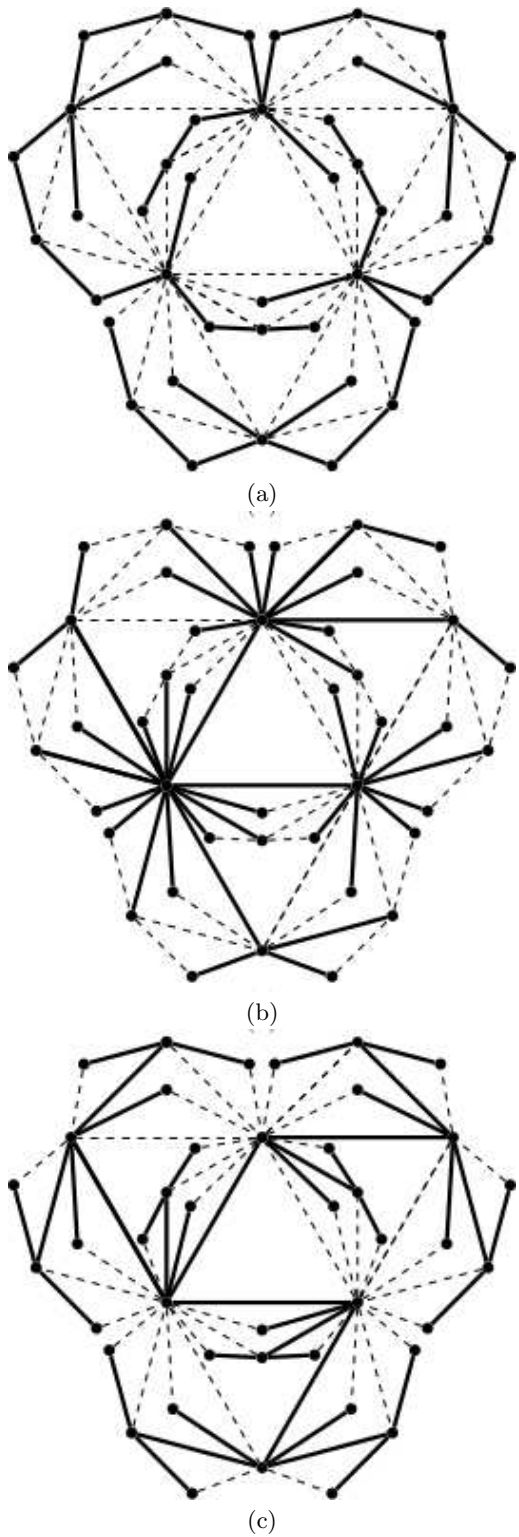


FIG. 2: Representative examples of extremal trees that can be selected from within the set of spanning trees of $F_\nu(1, 2)$: (a) using rule #'s 1 and 3 only, where the ambiguity of rule # 3 is reduced by maximizing the diameter when possible (breaking ties randomly) resulting in a MBVST; (b) using rules #'s 2 and 3 only with $p = 1$ resulting in a DST; and (c) using rule #'s 2 and 3 only with $p = 0$ leading to a balanced tree with a more homogeneous degree distribution.

Contrast this with the subset of spanning trees that is created by never applying rule # 1, where the diameter is then logarithmic in the order. In this class of trees where only rule #'s 2 and 3 are applied (small-world spanning trees), if one chooses the R links to always be adjacent to the largest degree node of the original edge (breaking ties randomly), then one obtains a member of the subset of spanning trees with the smallest diameter, i.e. a dense spanning tree having minimal Wiener index [35, 42], shown in Fig. 2 (b).

Letting a parameter p control the proportion of instances (when applying rule #'s 2 and 3) where the R -link is attached to the largest degree node, the tree in Fig. 2 (b) is a representative of the set created using $p = 1$; the tree in Fig. 2 (c) is representative of $p = 0$. The parameter p , then alters the degree distribution of the spanning tree, while retaining the small-world property, and values of $p \in (0, 1)$ then define probability distributions over this subset of small-world trees, providing a useful tool for approaching the Minimal k -Branching Vertex problem (k-MBVST) [34].

V. CONCLUSION

In summary, a general approach for obtaining spanning trees of recursive finitely articulated graphs has been presented through the example of the DGM net. Basic properties of the ensemble of all spanning trees, such as the typical/maximum diameter and the number of labeled trees are obtained exactly. One can then select a subset of spanning trees from the ensemble with desirable properties such as small-world features, certain degree distributions, and maximal/minimal diameter, effectively solving many optimal spanning tree problems on such finitely articulated models.

Perhaps the most interesting related open question would be finding the number of *unlabelled* spanning trees of recursive finitely articulated graphs. While finding this number is, in general, an intractable problem, we hope that the recursive properties of these graphs might provide a convenient foothold for such a problem.

-
- [1] Sergey N. Dorogovtsev, Alexander V. Goltsev, and José F. F. Mendes. “pseudofractal scale-free web”. *Phys. Rev. E*, 65(6):066122, jun 2002.
- [2] Hernán D. Rozenfeld, Shlomo Havlin, and Daniel ben-Avraham. Fractal and transfractal recursive scale-free nets. *New J. Phys.*, 9, 12 2006.
- [3] Hernán D. Rozenfeld and Daniel ben Avraham. Percolation in hierarchical scale-free nets. *Phys. Rev. E*, 75:061102, 2007.
- [4] C. Tyler Diggans, Erik M. Boltt, and Daniel ben Avraham. Stochastic and mixed flower graphs. *Phys. Rev. E*, 101:052315, May 2020.
- [5] Hernán Rozenfeld, Joseph E. Kirk, Erik M. Boltt, and Daniel ben-Avraham. Statistics of cycles: How loopy is your network? *J. Phys. A*, 38, 03 2004.
- [6] Erik M. Boltt and Daniel ben-Avraham. “what is special about diffusion on scale-free nets”. *New J. Phys.*, 2005.
- [7] Pinchen Xie, Zhong-Zhi Zhang, and Francesc Comellas. On the spectrum of the normalized laplacian of iterated triangulations of graphs. *Appl. Math. Computation*, 273:1123–1129, 2016.
- [8] Liren Shan, Huan Li, and Zhong-Zhi Zhang. Domination number and minimum dominating sets in pseudofractal scale-free web and sierpinski graph. *Theor. Comput. Sci.*, 677:12–30, 2017.
- [9] Russell Lyons. Asymptotic enumeration of spanning trees. *Combinatorics Probability and Computing*, 14(4):491–522, 2005.
- [10] Zhongzhi Zhang, Hongxiao Liu, Bin Wu, and Shuigeng Zhou. Enumeration of spanning trees in a pseudofractal scale-free web. *EPL (Europhysics Letters)*, 90(6):68002, jun 2010.
- [11] Zhongzhi Zhang, Hongxiao Liu, Bin Wu, and Tao Zou. Spanning trees in a fractal scale-free lattice. *Physical Review E*, 83(1):016116, 2011.
- [12] Zhongzhi Zhang, Bin Wu, and Francesc Comellas. The number of spanning trees in apollonian networks. *Discrete applied mathematics*, 169:206–213, 2014.
- [13] Weigang Sun, Shuai Wang, and Jingyuan Zhang. Counting spanning trees in prism and anti-prism graphs. *J. Appl. Anal. Comput.*, 6(1):65–75, 2016.
- [14] Luisa Gargano, Pavol Hell, Ladislav Stacho, and Ugo Vaccaro. Spanning trees with bounded number of branch vertices. In Peter Widmayer, Stephan Eidenbenz, Francisco Triguero, Rafael Morales, Ricardo Conejo, and Matthew Hennessy, editors, *Automata, Languages and Programming*, pages 355–365, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
- [15] Gábor Salamon and Gábor Wiener. On finding spanning trees with few leaves. *Information Processing Letters*, 105(5):164–169, 2008.
- [16] Selene Silvestri, Gilbert Laporte, and Raffaele Cerulli. A branch-and-cut algorithm for the minimum branch vertices spanning tree problem. *Computers & Operations Research*, 81:322–332, 2017.
- [17] R.L. Graham and Pavol Hell. On the history of the minimum spanning tree problem. *Annals of the History of Computing*, 7(1):43–57, 1985.
- [18] Michael Held and Richard M. Karp. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162, 1970.
- [19] T. L. Magnanti and R. T. Wong. Network design and transportation planning: Models and algorithms. *Transportation Science*, 18(1):1–55, 1984.
- [20] Larry S. Shapiro. *Clustering using maximum affinity spanning trees*, page 61–99. Distinguished Dissertations in Computer Science. Cambridge University Press, 1995.
- [21] Xu Y., Olman V., and Xu D. Clustering gene expression data using a graph-theoretic approach: an application of minimum spanning trees. *Bioinformatics*, 4:536–45, 2002.
- [22] Sudhir Varma and Richard Simon. Iterative class discovery and feature selection using minimal spanning trees. *BMC Bioinformatics*, 5(126), 2004.

- [23] Qiang He and Chee-Hung Henry Chu. A new image segmentation technique using maximum spanning tree. In Valentin E. Brimkov, Reneta P. Barneva, and Herbert A. Hauptman, editors, *Combinatorial Image Analysis*, pages 197–204, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [24] Laurent Galluccio, Olivier Michel, Pierre Comon, and Alfred O. Hero. Graph based k-means clustering. *Signal Processing*, 92(9):1970–1984, 2012.
- [25] Seong Eun Maeng, Hyung Wooc Choi, and Jae Woo Lee. Complex networks and minimal spanning trees in international trade network. *Intl. J. of Modern Physics: Conference Series*, 16:51–60, 2012.
- [26] J. Lovelace Rainbolt and M. Schmitt. The use of minimal spanning trees in particle physics. *Journal of Instrumentation*, 12(02), feb 2017.
- [27] B.Y. Wu and K.-M Chao. *Minimum routing cost spanning trees*, pages 41–84. Discrete Mathematics and its Applications. CRC Press, 2004.
- [28] Andrey A. Dobrynin, Roger Entringer, and Ivan Gutman. Wiener index of trees: Theory and applications. *Acta Applicandae Mathematicae*, 66(3):211, 05 2001.
- [29] Dong-Hee Kim, Jae Dong Noh, and Hawoong Jeong. Scale-free trees: The skeletons of complex networks. *Phys. Rev. E*, 70:046126, Oct 2004.
- [30] Raffaele Cerulli, Monica Gentili, and Antonio Iossa. Bounded-degree spanning tree problems: Models and new algorithms. *Computational Optimization and Applications*, 42:353–370, 04 2009.
- [31] Ricardo Silva, Diego Silva, Mauricio Resende, Geraldo Mateus, José Gonçalves, and Paola Festa. An edge-swap heuristic for generating spanning trees with minimum number of branch vertices. *Optimization Letters*, 8, 04 2014.
- [32] László Székely, Stephan Wagner, and Hua Wang. *Problems related to graph indices in trees*, pages 3–30. Springer International Publishing, 2016.
- [33] Tao Li, Yingqi Gao, Qiankun Dong, and Hua Wang. Degree sums and dense spanning trees. *PLOS ONE*, 12(9):1–12, 09 2017.
- [34] Massinissa Merabet, Jitamitra Desai, and Miklos Molnar. A generalization of the minimum branch vertices spanning tree problem. In Jon Lee, Giovanni Rinaldi, and A. Ridha Mahjoub, editors, *Combinatorial Optimization*, pages 338–351, Cham, 2018. Springer International Publishing.
- [35] Mustafa Ozen, Goran Lesaja, and Hua Wang. Globally optimal dense and sparse spanning trees, and their applications. *Statistics, Optimization & Information Computing*, 8(2):328–345, May 2020.
- [36] Vladimir Božović, Žana Kovijanić Vukićević, Goran Popivoda, Rong-Ying Pan, and Xiao-Dong Zhang. Extreme wiener indices of trees with given number of vertices of maximum degree. *Discrete Applied Mathematics*, 304:23–31, 2021.
- [37] Alfredo Marín. Exact and heuristic solutions for the minimum number of branch vertices spanning tree problem. *European Journal of Operational Research*, 245(3):680–689, 2015.
- [38] C. Tyler Diggans, Jeremie Fish, Abd AlRahman R. Al-Momani, and Erik M. Bollt. The essential synchronization backbone problem. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(11):113142, 2021.
- [39] Zhong-Zhi Zhang, Li-Li Rong, and Shuigeng Zhou. A general geometric growth model for pseudofractal scale-free web. *Physica A*, 377(1):329 – 339, 2007.
- [40] Arthur Cayley. *A theorem on trees*, volume 13 of *Cambridge Library Collection - Mathematics*, page 26–28. Cambridge University Press, 2009.
- [41] Richard Otter. The number of trees. *Annals of Mathematics*, 49(3):583–599, 1948.
- [42] Harry Wiener. Structural determination of paraffin boiling points. *Journal of the American chemical society*, 69(1):17–20, 1947.