Graphic Integer Sequence as canopy of Trees

KrishnenduBasuli

Department of Computer Science, West Bengal State University, INDIA.
krishnendu.basuli@gmail.com

Abstract:- A sequence of nonnegative integers can represent degrees of a graph. Already there is anestablished condition under which the above conclusion is true. It is ourobjectivein this paper to design an algorithm for generation of a spanning tree using Graphic Integer Sequence. And also use this structure for generation of all spanning trees using this information structure.

Keyword: - Canopy, Graphic Integer Sequence {GInS), tree, spanning tree, circuit.

1. Introduction

Historically, in 1847 Kirchhoff developed the theory of trees in order to solve the electrical network equations. Although a physicist, he ably abstracted electrical networks like mathematician in terms of graphs.Ten years later the number of isomers of saturated Hydrocarbons (C_nH_{2n+2}) was expressed by Arthur Cayley as the graphical problem of enumerating the class of "trees" in which each point is of degree 1 or 4. Major application of trees is in the analysis of electrical networks and in data structures where the concept of trees is very important[15,16].About trees or graphs , the most algorithm were developed due to Mayeda[7,8], Seshu[17], Reed[12], Hakimi[3], Deo[1], Knuth[6] and many others. Recent works on tree, graphs and central trees are mainly due toDeo who has also written a decent text book on graph theory [1] and Knuth [6].Graphic Integer Sequence may also be abbreviated as GInS in this paper. Different algorithms related with trees establishes the importance of GInS for better efficiency.

2. Some Fundamental notations about trees of a graph

By a tree, one initially feels a repetitive cross of Christ like structure in one piece consisting of branches with no closed path. The genealogy of a family is often represented by a tree called 'family tree'. The term 'tree' in graph theory owes its origin to an identical concept. Two points can be connected by one point to point connection, three points by two connections and n points can be fully interconnected by n-1 point to point connections. Such a set of interconnections is called a tree, the point to point connections being its branches. This conceptual definition differs somewhat from a formal rigorous definition of tree. However, one must make it a point that there is no perennial conflict between formal and conceptual definitions.

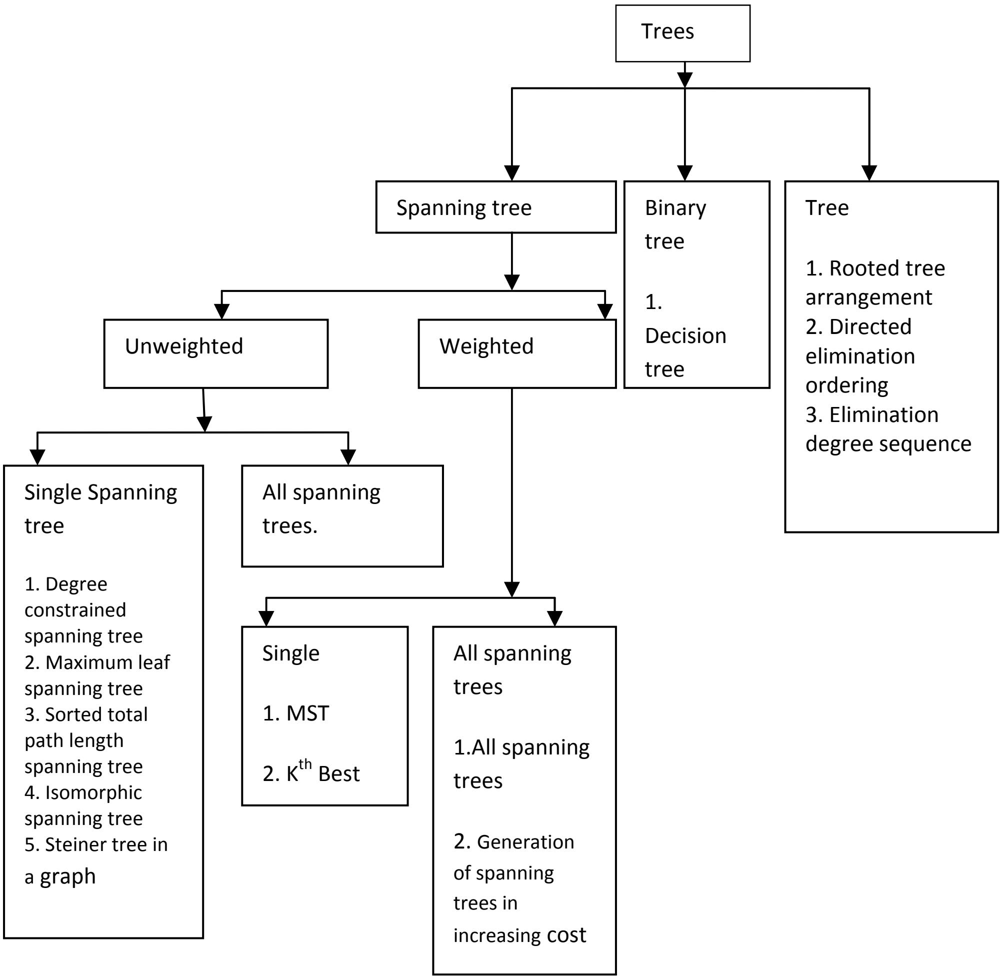

Figure: Canopy of Trees

2.1. Tree definitions:-

A connected graph without any circuits is called tree.

Theorem1:- The following statements are equivalent:

a) G is a tree.
b) G is circuit free but any new edge added to G forms a unique circuit.
c) For every two vertices in G there is a unique path connecting them.
d) G is connected, but if any edge is deleted from G, the connectivity of G is interrupted.
e) G is circuit free and has n-1 edges.
f) G is connected and has n-1 edges.

Proof: -[2].

Another definition of a tree uses the concept of a path. A graph G is a tree if and only if there exist exactly one path between only two vertices of a graph.

Lemma1:- Let T be a tree in a graph G. Then:

i) For Each edge e of G which is not in T, there is a unique circuit in G containing e and edges of T.
ii) For each edge e of G which is in T, there is a unique cut set in G containing h and edges not in T.

2.2. Some Properties of trees

Theorem2:- The incidence matrix D of G(V, E) has rank n-1, where n=|v| [1].

Proof: - Since modulo 2 sums of all the rows D is zero, the rank of D is at most n-1. Let d_j is a row of D corresponding to vertex V_j. suppose we have a linear relation $\sum_{j=1 \text{to } n} X_j d_j=0$ where X_j =0 or 1. The relation has one and only one non-zero solution $X_1=X_2=\ldots\ldots X_n=1$. Thus the rank of D is n-1.

Theorem3:- A sub graphG_s of a connected graph G can be made part of a tree if and only if $G_s$contain no circuit.

Proof: -[17].

3. Classification of tree Generation Algorithm:

Possibly, the most frequently used algorithms in graph theory are the tree generation algorithms. Since there exists several of these which differ in principle as well as in complexity, it is worth to revisit the algorithms to study their relative merits. The efficiency of these algorithms becomes a vital factor due to the fact that number of trees even in a moderate size graph literally very large. Thus, apriori knowledge of these existing algorithm added with estimation of complexity may help a potential user of these algorithms to decide about his choice before getting into the job so that he need not start at into due to his wrong selection of algorithm. In this context, a broad classification of tree generation algorithms is presented here. It is redundant to mention that background material is only to highlight our contribution in this area.

To investigate tree generation algorithms we do here some broad classification of tree generation algorithms depending upon philosophy of design and analysis.

Classification:

Broadly speaking, tree generation algorithms are of two categories,

i) Generation of a single tree of a graph.
ii) Generation of all the trees of a graph.

The evolution of tree generation algorithms shows that although the generation of a single tree and all the trees of a graph can be treated in the same token the two ways are usually mutually exclusive[4]. The algorithms for all tree generation can be classified under the following three heads:-

i) Trees by test and select methods [11].
ii) Elementary tree transformation methods [4, 10].
iii) Decomposition Techniques [6, 9].

4. Generation of a single tree of a graph:-

The existing algorithms are the well-known algorithm for generation of a single spanning tree. The principle is straight forward that if a graph G is connected and circuit less it is its own tree. If G contains a circuit, delete one edge from it. G is still connected. Repeat the operations till G contains no circuits. The remainder graph consisting of n vertices, connected and circuit less is a tree. The BFS and DFS [1, 13] are the well-knownalgorithm for this purpose.

4.1. GInS and Spanning tree of a graph:

We have already shown that how GInSplay the role of deciding factor for being an integer sequence as tree sequence [5]. Hare another algorithm devised for use of GInS as the information structure to generate spanning trees.

Proposed Algorithm: Spanning tree using GInS

Input: Graph G

Output: Spanning Tree

//*E* is the set of the edges and *V* represents set of all vertices in a graph G. *ET* //represents the edges in the tree which will be generated.

Step 1: *deg*[1..*n*] is initialize with 0;

Step 2: *ET* □Φ, *S*□*E*;

Step 3: For all edges (*i*,*j*) ϵ*S* in random order, do

Step 4: Check whether *ET* U (*i*,*j*) is a circuit

i) To check the circuit we find the end vertices of the edge (*i*,*j*).

ii) Assume $V_i$and $V_j$are the end vertices. Then check whether $V_i$and $V_j$belong to same or different connected component.

iii) If $V_i$and$V_j$are belong to different connected component, then there will be no circuit at all, otherwise a circuit will be present.

Step 5: If no circuit is present then,

Step 6: *ET*□□*ET* U (*i*,*j*); *S* □□*S*– (*i*,*j*);

Step 7: *deg*[*i*] = *deg*[*i*] + 1; *deg*[*j*] = *deg*[*j*] + 1;

Step 8: If |*ET*| = |*V*| - 1.

Step 9: Return *deg*[1..*n*].

Step 10: End if

Step 11: Else if circuit present then ignores the edge.

Step 12: *S* □□*S*– (*i*,*j*).

Step 13: End if

Step 14: End for loop.

Step 15: Stop.

In case of single spanning tree generation this algorithm is another option that can be consider. And for all spanning tree generation the proposed algorithm can be extended up to number of spanning tree for the graph G. But it is inherently exponential. So we merge a spatial structure with the GInS to generate the all spanning tree of the graph.

5. Conclusion

We merge graphic integer sequence and SPRIES[14] for generation of privileged columns. If all the vertices are affected by deletion process with graphic integer sequence in count that time only the advantages of special structure can achieves. So Graphic Integer Sequence is very much important for designing the SPRIES. Weighted spanning trees, degree constrained spanning trees are the one to one mapping to GInS.

6. References

[1] Deo N., Graph Theory with applications to Engineering and Computer Science, PHI., 2006.
[2] Even S., Graph Algorithms, Cambridge University Press, SecondEdition, 2012.
[3] Hakimi .S. L, On the Trees of a Graph and Their Generation. J. Franklin Inst. 270, Pp. 347-359, 1961.
[4] Kapoor.S. andRamesh.H. (1995).Algorithms for enumerating all spanning trees of undirected graph and weighted graph. SIAM Journal of Computing, 24, Pp.247-265, 1995.
[5] K. Basuli, S. Naskar and S. S. Sarma, “Generation of all spanning trees in the Limelight CCSIT 2012 in Bangalore, Springer, ISSN.1867- 8211,pp. 188-192, January 2012, India.
[6] Knuth D.E., The art of computer programming, Volume 4, Fascicle 0, 2009.
[7] Mayeda W., Graph Theory, Wiley Inter-science, 1972.
[8] Mayeda W. &Seshu S., Generation of Trees Without Duplications, IEEE Trans, CT-12, pp. 181-185, 1965.
[9] Minty G.J. A., ”A simple algorithm for listing all the trees of a Graph”, IEEE Trans. Circuit Theory, 12,120, 1965.
[10] Pak I. and Postnikov A., Enumeration of Spanning Trees of Graphs, Harvard University, Massachusetts Institute of Technology, 1994.
[11] Peikarski M., Listing of All Possible Trees of a Linear Graph, IEEE Tranas.on Circuit theory, Correspondence, CT-12, Pp.124-125, 1965.
[12] Read R.C. and ArjanR.E.T,"Bounds on backtrack algorithms for listing cycles, paths and spanning trees,”Networks, 5(3) pp.237-252,1975.
[13] Reingold, Nievergelt and Deo, Combinatorial Algorithms: Theory and Practice, PHI, 1977.
[14] S.Naskar, K.Basuli, and S. S.Sarma , “When a Graphic integer sequence represents a Tree Sequence”, , International Journal of Computer Science, ISSN:- 0973-2896, Volume 2, No. 3, p.p.-62-69, 2008.
[15] SenSarma S., Rakshit A., All circuits of a symmetric Graph, Editor Report, IEEE, 1981.
[16] SenSarma S., A. K.Chowdhury, An efficient tree generation algorithm, Journal of Institution of Electronics and Telecommunications Engg. (IETE), Vol 27, No 3,Pp. 105-109,1981.
[17] Seshu S. AND Reed M.B., Linear Graphs and Electrical Networks, Addison-Wesley, Reading, 1961.