Find the Center of a Tree

Epari Aneesh
iwm2014502@iiita.ac.in
Kshitij Tripathi
ism2014501@iiita.ac.in
Rishabh Kumar
icm2014004@iiita.ac.in

Abstract - The objective of this paper is to find the center of a tree. This paper describes an efficient approach that is by removing the leaf nodes of a tree level by level, until size of tree is less than or equal to two.

Index Terms – Tree, one-centered, bi-centered, leaf nodes

I. Introduction

A centered tree is a tree with only one center, and a bicentered tree is a tree with two centers. Given a graph, the eccentricity of a vertex v is defined as the greatest distance from v to any other vertex. A center of a graph is a vertex with minimal eccentricity. Radius is the minimal eccentricity value in the tree. Diameter is the maximal eccentricity value in the tree. A graph can have an arbitrary number of centers. Furthermore, the periphery and center of a graph are the set of vertices whose eccentricity is the diameter and radius, respectively. However, there are only two possibilities:
The tree has precisely one center (centered trees).
The tree has precisely two centers (bicentered trees). In this case, the two centers are adjacent.

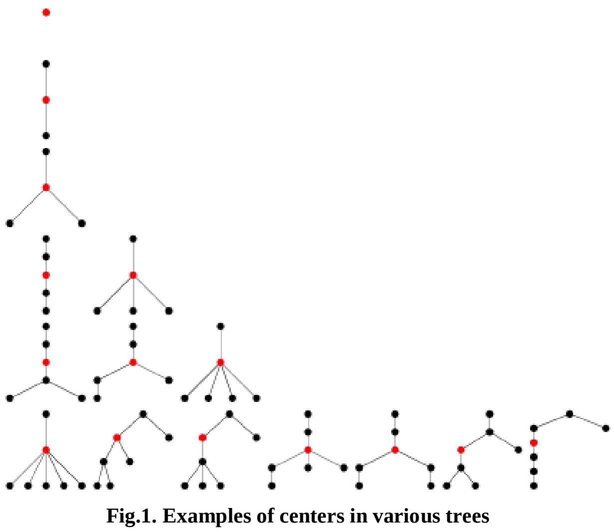

Fig.1. Examples of centers in various trees

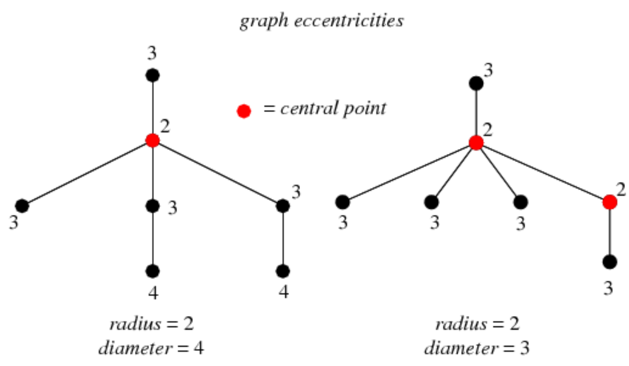

Fig.2. Examples of radius, diameters in various trees

II. Motivation

The problem finds its applications in a lot of varied fields. Finding the center of a graph is useful in facility location problems where the goal is to minimize the worst-case distance to the facility. For example, placing a hospital at a central point reduces the longest distance the ambulance has to travel. Also, for planning pipeline systems, example in a city water tanks should be located in the center of a graph so as to minimize time and distance among the residents.

III. Algorithms

A. Algorithm 1: Naive

This algorithm follows from the naïve definition of radius, eccentricity and center of a tree.

Input: Adjacency List of the tree

Output: Center/s of the tree

1. Let V be the number of vertices
2. Let D[V] [V] be the shortest distance matrix
3. Let E[V] be the eccentricity matrix
4. Let C be the set of centers
5. Let rad be radius of tree, initialized to positive infinity
6. Let diam be diameter of tree, initialized to positive

infinity

7. Compute D using All-Pair Shortest Path algorithm, Floyd Warshall will suffice here
8. For i from 1 to V
   For j from 1 to V
   E[i] = max(E[i], D[i][j])
9. For i from 1 to V
   rad = min (rad, E[i])
10. For i from 1 to V
    If (E[i] is rad)
    Add i to set c
11. Return c

Time Complexity: O (V * V * V) where V is the number of vertices the tree.

Memory Complexity: O V * V) where V is the number of vertices the tree

B. Algorithm 2:

Input: Adjacency List of the tree

Output: Center/s of the tree

Algorithm:

1. Get the degree of vertex from the adjacency list, store in degree array of vertices.
2. Traverse the degree list and do:
   If degree of v is 1
   Push v to the queue
3. Do while queue size is not less than 3
   Pop the queue front, u
   traverse the neighbor's v of u
   degree[v] -= 1;
   if (degree[v] is 1)
   push v to the queue
4. The remaining 1/ 2 vertices in the queue would be center/s of the tree

Time Complexity: O(V) where V is the number of vertices the tree.

Memory Complexity: O(V) where V is the number of vertices the tree

IV. Implementation and Results

The performance analysis was done for increasing number of vertices from 1 to 1000000 for the algorithm 2 and from 1 to 250 for the algorithm 1. The values were plotted after taking an average over 10 trials. The plotting was done with the help of GNUPlot tool. Input for the graph was provided randomly.

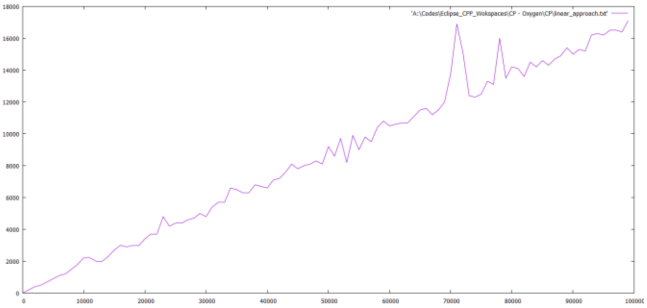

Fig.3. Time complexity analysis of Algorithm 1

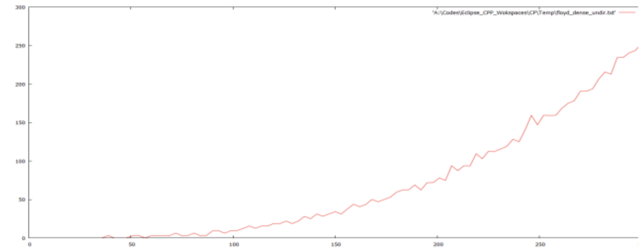

Fig.4. Time complexity Analysis of Algorithm 2

V. Conclusion

Two approaches were analyzed and implemented. Algorithm 1 which is the naïve brute force takes O (n * n * n) where n is the number of vertices. It also uses O (n * n) memory overhead. This approach directly follows from the mathematical definitions of center and diameter of a tree. Algorithm 2 is the efficient approach proposed which works in O(n) time and O(n) space follows from the level by level removal of leaves, since they correspond to higher eccentricity values. The remaining 1/ 2 vertices determine the center of the tree.

VI. References

[1]"Graph Eccentricity -- from Wolfram MathWorld", *Mathworld.wolfram.com*, 2017. [Online]. Available: http://mathworld.wolfram.com/GraphEccentricity.html. [Accessed: 29- Sep- 2017].

[2]"Centered tree", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Centered_tree#cite_note-1. [Accessed: 29- Sep- 2017].

[3]"Graph Center -- from Wolfram MathWorld", *Mathworld.wolfram.com*, 2017. [Online]. Available: http://mathworld.wolfram.com/GraphCenter.html. [Accessed: 29- Sep- 2017].