Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 28, no. 1, pp. 243–274 (2024)
DOI: 10.7155/jgaa.v28i1.2942

Computing Optimal Leaf Roots of Chordal Cographs in Linear Time *

Van Bang Le *Christian Rosenke*

Institut für Informatik,
Universität Rostock, Rostock, Germany

Submitted: October 2023	Accepted: May 2024	Published: June 2024
Article type: Regular paper	Communicated by: Paolo Ferragina	

Abstract. A graph G is a *k-leaf power*, for an integer $k \geq 2$, if there is a tree T with leaf set $V(G)$ such that, for all distinct vertices $x, y \in V(G)$, the edge xy exists in G if and only if the distance between x and y in T is at most k. Such a tree T is called a *k-leaf root* of G. The computational problem of constructing a k-leaf root for a given graph G and an integer k, if any, is motivated by the challenge from computational biology to reconstruct phylogenetic trees. For fixed k, Lafond [SODA 2022] recently solved this problem in polynomial time.

In this paper, we propose to study *optimal leaf roots* of graphs G, that is, the k-leaf roots of G with *minimum* k value. Thus, all k'-leaf roots of G satisfy $k \leq k'$. In terms of computational biology, seeking optimal leaf roots is more justified as they yield more probable phylogenetic trees. Lafond's result does not imply polynomial-time computability of optimal leaf roots, because, even for optimal k-leaf roots, k may (exponentially) depend on the size of G. This paper presents a linear-time construction of optimal leaf roots for chordal cographs (also known as trivially perfect graphs). Additionally, it highlights the importance of the parity of the parameter k and provides a deeper insight into the differences between optimal k-leaf roots of even versus odd k.

1 Introduction

Leaf powers have been introduced by Nishimura, Ragde and Thilikos [13] to model the phylogeny reconstruction problem from computational biology: given a graph G that represents a set of species with vertices $V(G)$ and the interspecies similarity with edges $E(G)$, how can we reconstruct an evolutionary tree T with a given similarity threshold k? For an integer $k \geq 2$, a *k-leaf root* of G, a tree T with species $V(G)$ as the leaf set and where distinct species $x, y \in V(G)$ have distance at most k in T if and only if they are similar on account of $xy \in E(G)$, is considered a solution to this problem. In case T exists, the graph G is called a *k-leaf power*. The challenge of finding a k-leaf root for given G and k has, yet, been modelled as the *k-leaf power recognition problem*: given G and k, decide if G has a k-leaf root.

*An extended abstract of this work appeared in the conference proceedings of FCT 2023 [11].

E-mail addresses: van-bang.le@uni-rostock.de (Van Bang Le) christian.rosenke@uni-rostock.de (Christian Rosenke)

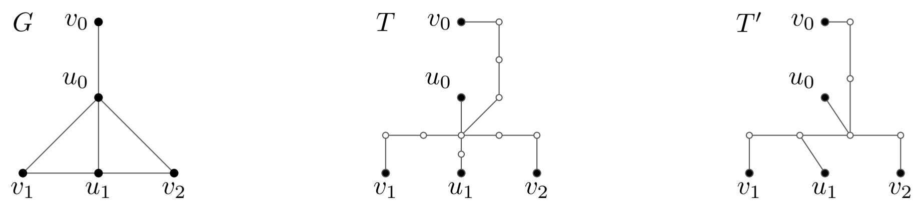

Figure 1: A graph G (left), a 5-leaf root T of G (middle), a 4-leaf root T' of G (right).

For an example, see Figure 1 with the graph G called *dart*. The similarities between the five species can be explained with similarity threshold $k = 5$ using the 5-leaf root T and with $k = 4$ by the 4-leaf root T', both depicted in Figure 1.

For a deeper discourse into the heavily studied field of k-leaf powers, the reader is kindly referred to the survey [14]. Here, we just give a short overview.

Lately, Eppstein and Havvaei [8] showed that k-leaf power recognition for graphs G with n vertices can be solved in $\mathcal{O}(f(k,\omega) \cdot n)$ time with $f(k,\omega)$ exponential in k and ω, the clique number of G. Quite simply put, they reduce k-leaf power recognition to the decision of a certain monadic second order property in a graph derived from G having tree-width bounded by k and ω. Lafond's even more recent algorithm [10] solves k-leaf power recognition in $\mathcal{O}(n^{g(k)})$ time, where $g(k)$ grows superexponentially with k. It applies sophisticated dynamic programming on a tree decomposition of G and exploits structural redundancies in G. Observe that, for fixed k, the latter method runs in polynomial time.

Before these advances, k-leaf power recognition had only been solved for all fixed k between 2 and 6. The 2-leaf powers are exactly the graphs that have just cliques as their connected components, which makes the problem trivial. For $k = 3$ (see [13] and [3]), $k = 4$ (see [13] and [4]), $k = 5$ (see [5]) and $k = 6$ (see [7]) individual algorithms have been developed, all creating a certain (tree-) decomposition of the input graph G and then attempting to fit together candidate k-leaf roots for the components into one k-leaf root for G.

A general controversial aspect of modelling the reconstruction of a phylogenetic tree T with the k-leaf power recognition problem is that k is part of the input. In the biological context, the value of k describes an upper bound on the number of evolutionary events in T that lie between two similar species x, y, thus, species adjacent in the given graph G by an edge xy. Unlike the model suggests, biologists do not always have control over the parameter k. Instead, phylogenetic trees T with as few as possible evolutionary events between all pairs of similar species are preferred. That is because, in reality, a higher number of events between x and y makes a similarity between x and y less likely. Conversely, this means that a k-leaf root of G with a small parameter k models a more probable phylogenetic tree. This paper therefore proposes a subtle change in perspective towards considering the following optimization problem.

Optimal Leaf Root (OLR)
Instance: A graph G.
Output: An *optimal leaf root* T of G, that is, a κ-leaf root of G such that $\kappa \leq k$ for all k-leaf roots of G, or No, if T does not exist.

Subsequently, we use κ to indicate that the respective κ-leaf root is optimal. OLR is in a certain sense an optimization version of k-leaf power recognition. The answer No states that the given graph G is not a k-leaf power for any k and, in particular, not for the given one. Getting an optimal κ-leaf root of G helps to decide if G is a k-leaf power in many cases. A difficulty is that, for all κ and k of different parity and with $2 \leq \kappa < k < 2\kappa - 2$, there are κ-leaf powers that are not k-leaf powers [15]. Then, checking $\kappa \leq k$ does not decide correctly.

As for k-leaf power recognition, there are no known general efficient solutions for OLR. If input was restricted to k-leaf powers with $k \leq K$ for some fixed K, we could repurpose Lafond's algorithm. Testing a given G with all $2 \leq k \leq K$ would finally reveal the minimum κ for which G is κ-leaf power. At that point, a κ-leaf root of G could also be extracted from the algorithm. But that classes of k-leaf powers have not been characterized well for any $k \geq 5$ makes restricting input in the proposed way difficult. Then again, it is unknown how to decide if a given graph G is a k-leaf power for any arbitrary k. And on top of that, the minimum value κ for which a given G is a κ-leaf power, if any, may exponentially depend on the size of G. This means that this brute force searching may take exponentially or even infinitely many runs of Lafond's algorithm.

It is known that, independent of k, all k-leaf powers are strongly chordal, but not vice versa. Ptolemaic graphs are strongly chordal and a *class of unbounded leaf powers*. That is, there is no bound β such that every Ptolemaic graph has a k-leaf root for some $k \leq \beta$. Nevertheless, every Ptolemaic graph on n vertices has a $2n$-leaf root [1, 2]. Later, Theorem 5 shows that, often, this is not optimal.

This paper considers a subclass of Ptolemaic graphs, the chordal cographs, as input to OLR. By definition, chordal cographs form the intersection of the well-known *chordal graphs* and the *cographs* and, thus, they are characterized as the graphs without induced cycles on four vertices and without induced paths on four vertices [9, 16, 17]. Golumbic [9] named this class *trivially perfect graphs* and also characterizes the class as the *comparability graphs of rooted trees*. As the approach in this paper relies on the cotree of chordal cographs for a tree model instead of the rooted tree behind the comparability graph interpretation, sticking to the former, less usual naming of the graph class is more natural here.

As a side effect of Lemma 8, this paper proves that chordal cographs are still a class of unbounded leaf powers. This means that k-leaf power recognition on this class cannot be solved in polynomial time with the algorithm of Lafond or the one of Eppstein and Havvaei. Nevertheless, the following main result of our work states that OLR can be solved in linear time for chordal cographs.

Theorem 1 *Given a chordal cograph G on n vertices and m edges, a κ-leaf root of G with minimum κ can be computed in $\mathcal{O}(n+m)$ time.*

To the best of our knowledge, chordal cographs are, thus, the first class of unbounded leaf powers with a polynomial-time solution for OLR. The linear time leaf root construction works by a divide and conquer algorithm that utilizes the cotree of the input chordal cograph G to (i) recursively identify a (universal) cut vertex u of G, (ii) divide G at u and (iii) conquer by merging recursively computed optimal leaf roots of the connected components of $G - u$ into one optimal leaf root of G.

Please note that while, in general, an OLR-solution does not entirely work for k-leaf power recognition, as elaborated above, our OLR-approach can also be used for linear-time k-leaf power recognition on chordal cographs. The key to this is the ability of our method to solve OLR with a given parity, such that the computed κ-leaf root comes with the minimum κ of the given parity. Hence, if we choose the parity of the given k, we can tell that a given graph G is a k-leaf power if and only if the computed κ-leaf root with κ of the same parity as k satisfies $\kappa \leq k$.

At the end of our approach, in Section 5, Theorem 10 restates our main result with the pronounced generalization of giving a desired parity as an input. The proof of Theorem 10, thus, concludes our elaborations by also showing Theorem 1.

Realizing that the desired parity of κ plays a certain role in our construction, we research this discrepancy here, and show, for certain chordal cographs, that the minimum κ can differ up to 25 percent depending on if it is wanted odd or even. The existence of κ-leaf powers that are not a k-leaf power for any opposite parity $k < 2\kappa - 2$ is shown in [15]. Therefore, 100 percent is a strict upper bound on the difference between minimum odd and even similarity threshold. However, we believe that our examples in Section 6 achieve the uttermost deviation within chordal cographs.

The next section presents basic notation, definitions, and facts on trees and k-leaf powers used in this paper. The optimal leaf root construction method for chordal cographs is introduced in Section 3 and proved correct in Section 4. Section 5 provides a respective linear-time implementation, thus, proving Theorem 1. A deepened evaluation of the difference between chordal cographs with κ-leaf roots of minimum odd versus even κ is carried out in the concluding Section 6.

2 Preliminaries

All considered graphs are finite and without multiple edges or loops. Let $G = (V, E)$ be a graph with vertex set $V(G) = V$ and edge set $E(G) = E$. A *universal vertex* in G is one that is adjacent to all other vertices. If all vertices of G are universal then G is *complete*. A vertex x that is adjacent to exactly one other vertex of G is called a *leaf* and the edge containing x is a *pendant edge*. Two adjacent vertices $x, y \in V(G)$ are *true twins* if $xz \in E(G)$, if and only if $yz \in E(G)$ for all $z \in V(G) \setminus \{x, y\}$.

A graph H is an *induced subgraph* of G if $V(H) \subseteq V(G)$ and $xy \in E(H)$ if and only if $xy \in E(G)$ for all $x, y \in V(H)$. All subgraphs considered in this paper are induced. For $X \subset V(G)$, $G - X$ denotes the induced subgraph H of G with $V(H) = V(G) \setminus X$. If X consists of one vertex x then we write $G - x$ for $G - \{x\}$. Complete subgraphs of G are called *cliques*.

As usual, an *x, y-path* in G is a sequence $v_1, \ldots, v_n$ of distinct vertices from $V(G)$ such that $x = v_1$, $y = v_n$ and $v_i v_{i+1} \in E(G)$ for all $i \in \{1, \ldots, n-1\}$. An x, y-path is called a *cycle* in G if $xy \in E(G)$. The length of the x, y-path, respectively cycle, is the number of its edges, that is, $n-1$ in the x, y-path and n in the cycle. If there is an x, y-path in G for all distinct $x, y \in V(G)$ then G is *connected*. Otherwise, G is *disconnected* and, therefore, composed of *connected components* $G_1, \ldots, G_n$, maximal induced subgraphs of G that are connected. A connected component is *nontrivial* if it has more than one vertex and, otherwise, it is called *isolated vertex*. We call $C \subseteq V(G)$ a *cut set* if $G - C$ has more connected components than G. If C is just a single vertex c then c is a *cut vertex*.

Graphs G and H are isomorphic if a bijection $\sigma : V(G) \to V(H)$ exists with $xy \in E(G)$ if and only if $\sigma(x)\sigma(y) \in E(H)$. If no induced subgraph of G is isomorphic to a graph H then G is *H-free*. *Trees* are the connected cycle-free graphs. This means, a tree T contains exactly one x, y-path for all $x, y \in V(T)$.

In this paper, we learn that, dependent on the given parity, the construction of an optimal leaf root differs in several details. To avoid permanent case distinctions, we use $\pi(i)$ for the parity of an integer i, that is, $\pi(i) = i \bmod 2$.

2.1 Chordal Cographs and Cotrees

Chordal cographs, ccgs for short, are known as the graphs that are free of the path and the cycle on 4 vertices. See the top row of Figure 2 for an example ccg. One particular ccg used in this paper is the *star (with t leaves)*, which consists of the vertices $u, v_1, \ldots, v_t$ for some $t \geq 2$ and the edges $uv_1, \ldots, uv_t$.

Like all cographs, ccgs can be represented with *cotrees* [6]. For an example, see Figure 2. The second row shows the cotree of the example cograph in the first row. For every cograph G, the cotree $\mathcal{T}$ is a rooted tree with leaves $V(G)$ and where every internal node is labelled with ⓪ for *disjoint union* or ① for *full join*. In this way, the leaves define single vertex graphs and every internal node represents the cograph G combining the cographs $H_1, H_2, \ldots, H_n$ of its children with the respective graph operation. More precisely, $V(G) = V(H_1) \cup V(H_2) \cup \cdots \cup V(H_n)$ and $G =$ ⓪$(H_1, H_2, \ldots, H_n)$ means the disconnected cograph on vertex set $V(G)$ and edge set $E(G) = E(H_1) \cup E(H_2) \cup \cdots \cup E(H_n)$ and $G =$ ①$(H_1, H_2, \ldots, H_n)$ means the connected cograph with vertex set $V(G)$ and edge set $E(G) = E(H_1) \cup E(H_2) \cup \cdots \cup E(H_n) \cup \{xy \mid x \in V(H_i), y \in V(H_j), 1 \leq i < j \leq n\}$. The cotree $\mathcal{T}$ is unique, can be constructed in linear time, and has the following properties:

- Every internal node has at least two children.
- No two internal nodes with the same label, ⓪ or ①, are adjacent.
- The subtree $\mathcal{T}_X$ rooted at node X is the cotree of the subgraph G_X induced by the leaves of $\mathcal{T}_X$. If X is labelled with ⓪ then G_X is the disjoint union of the cographs represented by the children of X and if it is labelled with ① then G_X is the full join of the children cographs.
- The cotree of an n-vertex cograph has at most $2n - 1$ nodes.

We mostly work with ccgs without true twins, like G in Figure 2. These graphs have the following properties, as observed in the upper rows of the figure.

Proposition 2 *If G is a ccg without true twins and $\mathcal{T}$ is the cotree of G then every node of $\mathcal{T}$ labelled with ① has exactly two children, one leaf and one node labelled with ⓪.*

Proof: Consider any ①-node X of $\mathcal{T}$. Then X is not a leaf and, thus, has at least two children none of which a ①-node. Since G is free of true twins, there is at most one leaf among the children of X. Distinct leaves x, y would be true twins since, by definition, both are adjacent to all $z \in V(G)$ with the least common ancestor in $\mathcal{T}$ labelled by ①. Because G is free of cycles on four vertices, X also has at most one ⓪-child. Two distinct ⓪-nodes Y, Z would represent two induced subgraphs G_Y and G_Z, where, by definition, G_Y contained two not adjacent vertices a, b and G_Z two not adjacent vertices c, d. Also by definition, there would be edges ac, ad, bc, bd in $E(G)$ and, thus, an induced cycle on four vertices in G. Hence, X having exactly two children, one leaf and one ⓪-node, is the only remaining possibility. □

Proposition 3 (Wolk [16, 17]) *Every connected ccg G without true twins has a unique universal vertex u and $G - u$ is disconnected (that is, u is a cut vertex).*

Proof: Because G is connected, the cotree has a ①-root X. By Proposition 2, X has a leaf child u that, by definition, is universal in G and a ⓪-child Y. Additional universal vertices would be true twins of u. By definition, $G - u = G_Y$ is disconnected and, thus, u is a cut vertex. □

2.2 Diameter, Radius and Center in Trees

Let T be a tree. The following notions are throughout used in the paper:

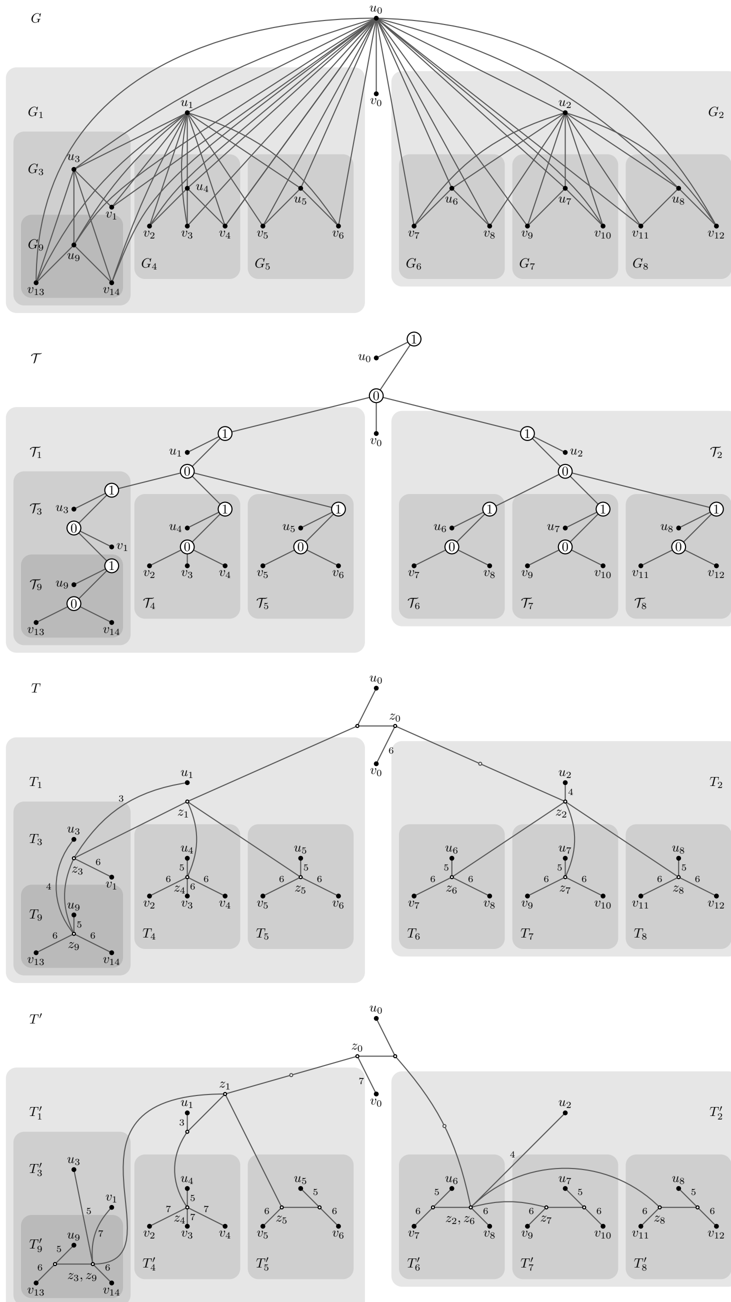

Figure 2: A ccg G (top), the cotree $\mathcal{T}$ of G (2nd row), an optimal (odd) 11-leaf root T of G (3rd row) as computed by Algorithm 1 (with input $p = 1$), and an optimal even 12-leaf root T' of G (bottom) as computed by Algorithm 1 with input $p = 0$.

- The *distance* between two vertices x and y in T, written $\text{dist}_T(x, y)$, is the length of the unique x, y-path in T.
- The *diameter* of T, denoted $\text{diam}(T)$, is the maximum distance between two vertices in T, that is, $\text{diam}(T) = \max\{\text{dist}_T(x, y) \mid x, y \in V(T)\}$.
- A *diametral path* in T is a path of length $\text{diam}(T)$.
- A vertex z is a *center vertex* of T if the maximum distance between z and any other vertex in T is minimum, that is, all $y \in V(T)$ satisfy $\max\{\text{dist}_T(x, z) \mid x \in V(T)\} \leq \max\{\text{dist}_T(x, y) \mid x \in V(T)\}$.
- The *radius* of T, denoted $\text{rad}(T)$, is the maximum distance between a center z and other vertices of T, that is, $\text{rad}(T) = \max\{\text{dist}_T(v, z) \mid v \in V(T)\}$.

For convenience, we define $\pi(T) = \pi(\text{diam}(T))$, the parity of the diameter of T. It is well known for all trees T that $\text{diam}(T) = 2 \cdot \text{rad}(T) - \pi(T)$ and that there is a single center vertex if $\pi(T) = 0$ and two adjacent centers if $\pi(T) = 1$. Furthermore, it is obvious for any diametral path in T that the end-vertices x and y are leaves and the center coincides with the center of T. Thus, we have $\min\{\text{dist}_T(z, x), \text{dist}_T(z, y)\} = \text{rad}(T) - \pi(T)$ and $\max\{\text{dist}_T(z, x), \text{dist}_T(z, y)\} = \text{rad}(T)$ for any center vertex z of T.

We call a center vertex z a *min-max center* of T if, for all center vertices z' of T, $\min\{\text{dist}_T(z, v) \mid v \text{ is a leaf of } T\} \geq \min\{\text{dist}_T(z', v) \mid v \text{ is a leaf of } T\}$. Thus, a min-max center maximizes the distance to the closest leaf of T. For a min-max center z, the *leaf distance* is $d_T^{\min} = \min\{\text{dist}_T(z, v) \mid v \text{ is a leaf of } T\}$. The paper also requires the following technical lemma regarding the radius and the diameter of trees:

Lemma 1 *If $T_1, \ldots, T_s$ are $s \geq 2$ trees with $\text{diam}(T_1) \geq \cdots \geq \text{diam}(T_s)$ then*
(i) $\text{rad}(T_1) \geq \text{rad}(T_2) \geq \cdots \geq \text{rad}(T_s)$,
(ii) *for all $1 \leq i < j \leq s$, if $\text{rad}(T_i) = \text{rad}(T_j)$ then $\pi(T_i) \leq \pi(T_j)$, and*
(iii) *for all $1 \leq i < j \leq s$, $\text{rad}(T_i) - \pi(T_i) \geq \text{rad}(T_j) - \pi(T_j)$.*

Proof: Let $i < j$.

(i) By definition, $2\text{rad}(T_i) - \pi(T_i) = \text{diam}(T_i) \geq \text{diam}(T_j) = 2\text{rad}(T_j) - \pi(T_j)$. Clearly, if $\pi(T_i) = \pi(T_j)$ then we directly get $\text{rad}(T_i) \geq \text{rad}(T_j)$. Otherwise, if only $\pi(T_i) = 1$, we argue that
$$2\text{rad}(T_i) > 2\text{rad}(T_i) - 1 = \text{diam}(T_i) \geq \text{diam}(T_j) = 2\text{rad}(T_j).$$
Finally, if just $\pi(T_j) = 1$ then the diameter of T_i is even and the diameter of T_j is odd, thus, not equal. This immediately tells us that
$$2\text{rad}(T_i) = \text{diam}(T_i) > \text{diam}(T_j) = 2\text{rad}(T_j) - 1$$
and, hence, $2\text{rad}(T_i) \geq 2\text{rad}(T_j)$ and we are done.

(ii) If $\text{rad}(T_i) = \text{rad}(T_j)$, then $2\text{rad}(T_i) - \pi(T_i) = \text{diam}(T_i) \geq \text{diam}(T_j) = 2\text{rad}(T_j) - \pi(T_j) = 2\text{rad}(T_i) - \pi(T_j)$. Hence, $\pi(T_i) \leq \pi(T_j)$.

(iii) By (i), $\text{rad}(T_i) \geq \text{rad}(T_j)$. If $\text{rad}(T_i) > \text{rad}(T_j)$, then $\text{rad}(T_i) - \pi(T_i) > \text{rad}(T_j) - \pi(T_i) \geq \text{rad}(T_j) - 1$. Hence, $\text{rad}(T_i) - \pi(T_i) \geq \text{rad}(T_j) \geq \text{rad}(T_j) - \pi(T_j)$. If $\text{rad}(T_i) = \text{rad}(T_j)$, then, by (ii), $\pi(T_i) \leq \pi(T_j)$. Hence, $\text{rad}(T_i) - \pi(T_i) = \text{rad}(T_j) - \pi(T_i) \geq \text{rad}(T_j) - \pi(T_j)$.

□

2.3 Leaf Powers, Leaf Roots and their Basic Properties

Let $k \geq 2$ be an integer. A graph G is a *k-leaf power* if a *k-leaf root* T of G exists, a tree with leaves $V(G)$ such that, for all distinct vertices $x, y \in V(G)$, xy is an edge in G if and only if $\mathrm{dist}_T(x, y) \leq k$. The example G in Figure 2 therefore is an 11-leaf power because of the 11-leaf root T of G in the third row and also a 12-leaf power by the 12-leaf root T' in the bottom row. Note that the figure shows *compressed* illustrations of T and T' where some long paths of vertices with degree two are depicted by single weighted edges.

It is well-known that

- a complete graph is a k-leaf power for all $k \geq 2$,
- a graph is a k-leaf power if and only if all of its connected components are k-leaf powers, and
- if x, y are true twins in G then G is a k-leaf power if and only if $G - x$ is a k-leaf power.

Note for the last fact that Lemma 7.3 and Corollary 7.4 in [12] imply the possibility to identify and remove all true twins from a graph in linear time. So, in the remainder of the paper, we can simply focus on graphs without true twins.

Since the concept of k-leaf powers is slightly different for odd and even k, we formalize this discrepancy as follows: We say that a k-leaf root is of *even*, respectively *odd parity*, if k is even, respectively odd. A k-leaf root T of G is an *optimal even*, respectively *optimal odd* leaf root if k is even, respectively odd, and all k'-leaf roots of G with k' of the same parity as k satisfy $k \leq k'$. Finally, a k-leaf root T of G is (just) *optimal* if $k \leq k'$ for all k'-leaf roots of G (independent of parity). See the third row of Figure 2 for an optimal odd leaf root T of the example graph G in the same figure and see the bottom row for an optimal even leaf root T' of G. Since T is an 11-leaf root and T' a 12-leaf root, it follows that T is an optimal leaf root of G.

We conclude this section by establishing a few properties for leaf roots as considered in this paper. The first one concerns a bound on the distance between center and leaves in T in case G is connected.

Lemma 2 *Every k-leaf root T of a connected graph satisfies $d_T^{\min} \leq \frac{k}{2}$.*

Proof: Let T be a k-leaf root of a connected graph G, and let z be a min-max center vertex of T. Let T' be any subtree of $T - z$. Then, every pair of leaf v in T' and leaf w outside T' fulfills

$$\mathrm{dist}_T(u, v) = \mathrm{dist}_T(u, z) + \mathrm{dist}_T(z, v) \geq 2 \cdot d_T^{\min}.$$

Thus, if $d_T^{\min}$ was larger than $\frac{k}{2}$, then $\mathrm{dist}_T(u, v) > k$ and G would be disconnected. □

See Figure 2 with the 11-leaf root T of G having $d_T^{\min} = \mathrm{dist}_T(u_0, z_0) = 2 \leq \frac{11}{2}$ and the 12-leaf root T' with $d_{T'}^{\min} = \mathrm{dist}_T(u_0, z_0) = 2 \leq \frac{12}{2}$. Secondly, if G has a universal vertex u then the distance between u and the center in T cannot exceed the difference between k and the radius of T.

Lemma 3 *If G is a non-complete graph with a universal vertex u and T is a k-leaf root of G then $\mathrm{dist}_T(u, z) \leq k - \mathrm{rad}(T) + \pi(T)$ for all center vertices z of T. If $z_1 \neq z_2$ are the center vertices of T, then $\mathrm{dist}_T(u, z_1) \leq k - \mathrm{rad}(T)$ or $\mathrm{dist}_T(u, z_2) \leq k - \mathrm{rad}(T)$.*

Proof: Consider a diametral path P in T, and let x, y be the end vertices of P. Since G is not complete, x and y are not adjacent in G. Hence, u, x and y are different leaves in T. Because T is a tree, the three paths in T, the x, u-path, the y, u-path, and P intersect at a unique (non-leaf) vertex on P, say c.

Let ℓ_1 be the length of the x,c-path and ℓ_2 be the length of the y,c-path in T. Since u is universal in G, we have

$$\mathrm{dist}_T(u,c) + \ell_1 \leq k \quad \text{and} \quad \mathrm{dist}_T(u,c) + \ell_2 \leq k.$$

Let z_1, z_2 be the center vertices of T; possibly with $z_1 = z_2$, if $\mathrm{diam}(T)$ is even. Recall that, since P is a diametral path, the centers of T and P coincide, and that z_1z_2 is an edge on P whenever $z_1 \neq z_2$. Let, without loss of generality, z_1 be on the x,z_2-path (and so z_2 is on the y,z_1-path). Then

$$\begin{aligned}\mathrm{dist}_T(z_1,y) &= \mathrm{rad}(T) = \mathrm{dist}_T(z_2,x) \text{ and}\\ \mathrm{dist}_T(z_1,x) &= \mathrm{rad}(T) - \pi(T) = \mathrm{dist}_T(z_2,y).\end{aligned}$$

Since c is, like z_1 and z_2, on the path P, either the x,z_1-path or the y,z_2-path contains c. Assume first that c is on the x,z_1-path; possibly with $c = z_1$. Then

$$\begin{aligned}\mathrm{dist}_T(u,z_1) &= \mathrm{dist}_T(u,c) + \big(\ell_2 - \mathrm{dist}_T(z_1,y)\big)\\ &= \big(\mathrm{dist}_T(u,c) + \ell_2\big) - \mathrm{rad}(T)\\ &\leq k - \mathrm{rad}(T) \quad \text{and}\\ \mathrm{dist}_T(u,z_2) &= \mathrm{dist}_T(u,c) + \big(\ell_2 - \mathrm{dist}_T(z_2,y)\big)\\ &= \big(\mathrm{dist}_T(u,c) + \ell_2\big) - \big(\mathrm{rad}(T) - \pi(T)\big)\\ &\leq k - \mathrm{rad}(T) + \pi(T).\end{aligned}$$

In the case, where c is on the y,z_2-path, the same arguments yield $\mathrm{dist}_T(u,z_2) \leq k - \mathrm{rad}(T)$ and $\mathrm{dist}_T(u,z_1) \leq k - \mathrm{rad}(T) + \pi(T)$. □

For an illustration of the lemma above, see Figure 2, where the distance of u_0 and the farthest center vertex z_0 of T satisfies $\mathrm{dist}_T(u_0,z_0) = 2 \leq 11 - 10 + 1 = k - \mathrm{rad}(T) + \pi(T)$ and, in T', $\mathrm{dist}_{T'}(u_0,z_0) = 2 \leq 12 - 11 + 1 = k' - \mathrm{rad}(T') + \pi(T')$. Lemma 3 implies upper bounds on radius and diameter of T.

Corollary 4 *If G is a graph with a universal vertex and T is a k-leaf root of G then* $\mathrm{rad}(T) \leq k-1$ *and, in particular,* $\mathrm{diam}(T) \leq 2k-2$.

Proof: Recall that $k \geq 2$ by definition. Hence, for complete graphs, the statement is obvious. So, let G be a non-complete graph with universal vertex u. By Lemma 3, $1 \leq \mathrm{dist}_T(u,z) \leq k - \mathrm{rad}(T)$ for any center vertex z of T. Hence, $\mathrm{rad}(T) \leq k-1$. Since $\mathrm{diam}(T) \leq 2 \cdot \mathrm{rad}(T)$, we have $\mathrm{diam}(T) \leq 2k-2$. □

As a matter of fact, k-leaf roots tend to contain long paths of vertices with degree two. It is reasonable to *compress* such a path $P = v_0, \ldots, v_n$ into a single *weighted* edge $v_0(n)v_n$. Clearly, weighted edges $v_0(n)v_n$ add their weight n to distances in T and, so, $\mathrm{dist}_T(v_0,v_n) = n$.

3 Optimal Leaf Root Construction for CCGs

The aim of this section is the development of an optimal leaf-root construction approach for ccgs G. In very simple terms, we describe a divide and conquer method that splits G into smaller ccgs

$G_1, G_2, \ldots$, recursively obtains their optimal leaf roots $T_1, T_2, \ldots$, and then extends them into an optimal leaf root for G.

We start with introducing two basic leaf root operations and analyze their properties. The first operation, the *extension* of trees, is used to level the recursively found leaf roots $T_1, T_2, \ldots$ on the same k, which is essential for the subsequent composition into one k-leaf root. If T is a tree and $\delta \geq 0$ an integer then $T' = \eta(T, \delta)$ is the tree obtained from T by subdividing every pendant edge δ times, that is, replacing the edge with a new path of length $\delta + 1$ (hence, of $\delta + 1$ edges). The following property of this operation is well-known:

Lemma 4 *If T is a k-leaf root of a graph G and $\delta \geq 0$ an integer then $T' = \eta(T, \delta)$ is a $(k + 2\delta)$-leaf root of G with same center, same min-max center vertices, and* $\operatorname{diam}(T') = \operatorname{diam}(T) + 2\delta$, $\operatorname{rad}(T') = \operatorname{rad}(T) + \delta$, $d^{\min}_{T'} = d^{\min}_{T} + \delta$.

Proof: Note that T and T' have the same set of leaves. For every two leaves u and v, we have $\operatorname{dist}_{T'}(u, v) = \operatorname{dist}_T(u, v) + 2\delta$. Hence, $\operatorname{dist}_{T'}(u, v) \leq k + 2\delta$ if and only if $\operatorname{dist}_T(u, v) \leq k$. Thus, T' is a $k + 2\delta$-leaf root of G. The other statements are obvious from the respective definitions. □

The second operation merges the individual k-leaf roots for the connected components of a graph G into one k-leaf root T for the entire G. The goal here is to minimize the diameter of T, which, in turn, allows making optimizations to the value of k. Assume that G has $s \geq 0$ non-trivial connected components $G_1, \ldots, G_s$ and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$ and let $T_1, \ldots, T_s$ be k-leaf roots for $G_1, \ldots, G_s$ with min-max center vertices $z_1, \ldots, z_s$. If $s > 0$, we define the *critical index* m as the smallest element of $\{1, \ldots, s\}$ with $d^{\min}_{T_m} = \min\{d^{\min}_{T_i} \mid 1 \leq i \leq s\}$ and call T_m the *critical root*. Then, the *merging* $\mu(k, T_1, \ldots, T_s, v_1, \ldots, v_t)$ results in the tree T produced by the following steps:

1. Create a new vertex c.
2. If $s > 0$ then connect c and the center z_m of the critical root by a path of length $\frac{k+\pi(k)}{2} - d^{\min}_{T_m}$. If $\pi(k) = 0$ and $d^{\min}_{T_m} = \frac{1}{2}k$ then this means to identify the vertices c and z_m.
3. For all $i \in \{1, \ldots, m-1, m+1, \ldots, s\}$, connect c and z_i by a path of length $\frac{k-\pi(k)}{2} + 1 - d^{\min}_{T_i}$.
4. For all $j \in \{1, \ldots, t\}$, connect c and v_j by a path of length $\frac{k-\pi(k)}{2} + 1$.

Notice that the μ-operation is sensitive with respect to the parity $\pi(k)$. For one thing, this is necessary to guarantee that all added paths are of integer length, which is done by in- or decreasing odd k. As a side note, we point out that the lengths of added paths are also non-negative by Lemma 2, which makes the μ-operation well-defined. On the other hand, the result is that merging works slightly different for odd and even k. For odd k, all trees $T_1, \ldots, T_s$, including the critical one, are essentially added in the same way by our construction method. This is because, for odd k, $\frac{k+\pi(k)}{2} - d^{\min}_{T_m} = \frac{k-\pi(k)}{2} + 1 - d^{\min}_{T_m}$. The special treatment of the critical root, thus, has an effect only if k is even. Specifically in that case, we can sometimes save one in the diameter of T, if we put the critical root closer to the center of T than the rest. The reason that this optimization works is that, usually, the critical root has the largest diameter.

Lemma 5 *Let G be a graph and $k \geq 2$ an integer. If G is disconnected with $s \geq 0$ non-trivial connected components $G_1, \ldots, G_s$ and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$ and if $T_1, \ldots, T_s$ are k-leaf roots for $G_1, \ldots, G_s$ then $T = \mu(k, T_1, \ldots, T_s, v_1, \ldots, v_t)$ is a k-leaf root of G.*

Proof: From the construction, we immediately see for all distinct vertices x and y from the same component G_i that $\operatorname{dist}_T(x, y) = \operatorname{dist}_{T_i}(x, y)$. Hence, $\operatorname{dist}_T(x, y) \leq k$, if and only if $\operatorname{dist}_{T_i}(x, y) \leq k$, if and only if xy is an edge in G_i.

To prove that T is a k-leaf root for G, it is sufficient to show for all vertices x and y stemming from different components of G that $\mathrm{dist}_T(x, y) > k$. In case of $s > 0$, we assume, without loss of generality, that the critical index is $m = 1$. For the remainder of the proof, we keep in mind that, by the definition of a min-max center, $\mathrm{dist}_{T_i}(x, z_i) - d^{\min}_{T_i} \geq 0$ for all $T_i, 1 \leq i \leq s$ with min-max center z_i and all leaves x of T_i.

For a start, we consider a vertex x in G_1 and v_j for any $j \in \{1, \dots, t\}$ and see

$$\mathrm{dist}_T(x, v_j) = \mathrm{dist}_{T_1}(x, z_1) + \left(\tfrac{k+\pi(k)}{2} - d^{\min}_{T_1}\right) + \left(\tfrac{k-\pi(k)}{2} + 1\right) \geq k + 1.$$

Similarly, for any vertex x in G_i, $1 < i \leq s$, and v_j, we get

$$\mathrm{dist}_T(x, v_j) = \mathrm{dist}_{T_i}(x, z_i) + \left(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T_i}\right) + \left(\tfrac{k-\pi(k)}{2} + 1\right) \geq k + 1.$$

For any vertex x in G_1 and y in G_j with $1 < j \leq s$, it is

$$\begin{aligned}\mathrm{dist}_T(x, y) &= \mathrm{dist}_{T_1}(x, z_1) + \left(\tfrac{k+\pi(k)}{2} - d^{\min}_{T_1}\right) + \left(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T_j}\right) + \mathrm{dist}_{T_j}(z_j, y) \\ &= (k + 1) + \left(\mathrm{dist}_{T_1}(x, z_1) - d^{\min}_{T_1}\right) + \left(\mathrm{dist}_{T_j}(z_j, y) - d^{\min}_{T_j}\right) \geq k + 1.\end{aligned}$$

Similarly, for any vertex x in G_i and y in G_j with $1 < i < j \leq s$, we get

$$\begin{aligned}\mathrm{dist}_T(x, y) &= \mathrm{dist}_{T_i}(x, z_i) + \left(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T_i}\right) + \left(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T_j}\right) + \mathrm{dist}_{T_j}(z_j, y) \\ &= (k + 2 - \pi(k)) + \left(\mathrm{dist}_{T_i}(x, z_i) - d^{\min}_{T_i}\right) + \left(\mathrm{dist}_{T_j}(z_j, y) - d^{\min}_{T_j}\right) \geq k + 1.\end{aligned}$$

Finally, the distance between v_i and v_j with $1 \leq i < j \leq t$ is

$$\mathrm{dist}_T(v_i, v_j) = 2 \cdot \left(\tfrac{k-\pi(k)}{2} + 1\right) \geq k + 1.$$

It follows that $xy \in E(G)$ if and only if $\mathrm{dist}_T(x, y) \leq k$ for all $x, y \in V(G)$. That is, T is a k-leaf root of G. □

The two operations above simplify the description of the following leaf root construction algorithm for ccgs since they hide away many of the technical details. The foundation of the proposed recursive approach is that (i) induced subgraphs of ccgs are ccgs and (ii) every connected ccg without true twins has a unique universal cut vertex (see Proposition 3). Also, recall from Section 2.3 that true twins in graphs can be removed in linear time and, thus, be safely ignored. Therefore, we define for all ccgs G without true twins and a given parity $p \in \{0, 1\}$ the result of the *root operation* $\rho(G, p)$ as the tree T and the number k produced by the following (recursive) procedure:

i. **If G is a star then** let u be the central vertex and $v_1, \dots, v_t$ the leaves of G (with $t \geq 2$ because G does not have true twins)
   1. If $p = 1$ (for odd) then let $T' = \eta(G, 1)$, obtain T by attaching a new leaf to u in T', and return $(T, 3)$.
   2. If $p = 0$ (for even) and $t = 2$ then return $(T, 4)$ with T obtained from a single vertex v by attaching the leaves u, v_1 and v_2 to v with paths of lengths one, two and three, respectively.
   3. If $p = 0$ and $t > 2$ then let $T' = \eta(G, 2)$, obtain T by attaching a new leaf to u in T', and return $(T, 4)$.

ii. **else if** G **is a connected graph then** let u be the universal cut vertex of G (by Proposition 3) and let $G_1, \dots, G_s$ be the $s \geq 1$ non-trivial connected components and $v_1, \dots, v_t$ the $t \geq 0$ isolated vertices of $G - u$.
   1. Recursively find $(T_1, k_1) = \rho(G_1, p), \dots, (T_s, k_s) = \rho(G_s, p)$.
   2. If $s = 1$ then let $k = k_1 + 2(1 - \pi(T_1))$. Otherwise, let

$$\begin{aligned} k_a &= \max\{k_1, \dots, k_s\}, \\ k_b &= \max\{k_i \mid 1 \leq i \leq s, i \neq a\} \text{ and, if } s > 2 \text{ let} \\ k_c &= \max\{k_i \mid 1 \leq i \leq s, i \neq a, i \neq b\}. \end{aligned}$$

   If $p = 1$ (for odd) then let $k = k_a + k_b - 1 - 2 \cdot \pi(T_a) \cdot \pi(T_b)$ and, otherwise,

$$k = \begin{cases} k_a + k_b - 2 \cdot (\pi(T_a) + \pi(T_b) - \pi(T_a) \cdot \pi(T_b)), & \text{if } s = 2 \text{ or } k_a > k_c \\ k_a + k_b - 2 \cdot \pi(T_a) \cdot \pi(T_b), & \text{otherwise.} \end{cases}$$

   3. Get the extended leaf root $T'_i = \eta\left(T_i, \frac{k-k_i}{2}\right)$ for all $i \in \{1, \dots, s\}$ and let $T' = \mu(k, T'_1, \dots, T'_s, v_1, \dots, v_t)$.
   4. Return (T, k) with T obtained from T' by attaching the leaf u to a center vertex of T'.

iii. **else** G **is a disconnected graph and then** let $G_1, \dots, G_s$ be the $s \geq 0$ non-trivial connected components and $v_1, \dots, v_t$ the $t \geq 0$ isolated vertices of G.
   1. Recursively find $(T_1, k_1) = \rho(G_1, p), \dots, (T_s, k_s) = \rho(G_s, p)$.
   2. Let $k = \max\{k_1, \dots, k_s, p + 2\}$ and let $T'_i = \eta\left(T_i, \frac{k-k_i}{2}\right)$ for all $i \in \{1, \dots, s\}$.
   3. Return (T, k) with $T = \mu(k, T'_1, \dots, T'_s, v_1, \dots, v_t)$.

Hence, if the input graph G is not a star then the approach is to firstly divide G into smaller connected subgraphs $G_1, \dots, G_s$ (and isolated vertices $v_1, \dots, v_t$), secondly find corresponding k-leaf roots $T_1, \dots, T_s$ by recursion and the η-operation, and, last, conquer by merging them into a single leaf root of G with the μ-operation. The divide step is simple for disconnected G and, otherwise, is carried out by removing the unique universal cut vertex of G.

The ρ-operation is sensitive to the given parity p for using μ as a subroutine. Observe that p also decides how the resulting k is determined. There are four cases when G is connected and not a star. In the first one, when $s = 1$, the construction is the same for odd and even p and consists of adding $u, v_1, \dots, v_t$ at the correct distance to the center of T_1 and computing k from k_1. Secondly, if $s > 1$ and $p = 1$, the μ-operation has only one way of merging the recursively found leaf roots $T_1, \dots, T_s$ to minimize the diameter of the result T. Then, k widely depends on the two largest values of $k_1, \dots, k_s$. But if $p = 0$, there is one situation that, on the one hand, allows μ to use a smaller diameter for T by prioritizing the critical leaf root and, on the other hand, lets ρ return a slightly better value for k. This happens only when $s = 2$, or whenever the k_c-leaf root, with k_c the third-largest value among $k_1, \dots, k_s$, properly fits into the diametral space of T that is already required for the k_a-leaf root and the k_b-leaf root.

The third and bottom row of Figure 2 illustrate the results $(T, 11)$ of $\rho(G, 1)$ and $(T', 12)$ of $\rho(G, 0)$ on the example G. By recursion, both are produced bottom-up, and it is difficult to follow their assembly at the deeper recursion levels. The highest recursion level of $\rho(G, 1)$, however, has received a 7-leaf root with odd diameter for subgraph G_1 and a 5-leaf root with even diameter for G_2 in Step (ii.1.). In Step (ii.2.), the ρ-procedure determines $k = k_1 + k_2 - 1 = 11$. The extension of the trees in Step (ii.3.) produces the 11-leaf roots T_1 and T_2 for G_1 and G_2, respectively, as shown in Figure 2. Their following merging and the attachment of u_0 in Step (ii.4.) produces the shown 11-leaf root T of G. Similarly, $\rho(G, 0)$ receives an odd-diameter 8-leaf root of G_1 and an even-diameter 6-leaf root of G_2. Since $s = 2$, the critical root can be treated in the special way and, thus, $\rho(G, 0)$ determines $k' = k'_1 + k'_2 - 2 = 12$. After the extension, we get the 12-leaf roots T'_1 for G_1 and T'_2 for G_2 as in Figure 2. Their merging and the attachment of u_0 yields the 12-leaf root T' of G as also illustrated there.

The following statement regards the correctness of our procedure.

Theorem 5 *Let G be a ccg on n vertices and without true twins and let $p \in \{0, 1\}$. Then $(T, k) = \rho(G, p)$ provides a k-leaf root T of G that is optimal for parity p (hence, $\pi(k) = p$) and with $k \leq n + 1$. If G is connected then*
(T1) $\mathrm{rad}(T) = k - 1$*,*
(T2) $d_T^{\min} = 1 + \pi(T)$*, and*
(T3) $\mathrm{diam}(T') \geq \mathrm{diam}(T) + k' - k$ *for all k'-leaf roots T' of G with $\pi(k') = p$.*

The proof of the theorem above stands for the majority of the work behind this paper and it is fairly long. For readability, it has entirely been moved into the separate section that follows.

Note that, with respect to the optimality of the result, Theorem 5 makes a slightly stronger statement than our main result in Theorem 1. In fact, the ρ-operation can find a κ-leaf root with minimum κ for every ccg G simply by choosing the best from $(T, k) = \rho(G, 1)$ and $(T', k') = \rho(G, 0)$. To prove Theorem 1, Section 5 shows how to implement the ρ-operation in linear time.

4 The Proof of Theorem 5

To make reading the proof of Theorem 5 easier, we break it down into the proofs of several propositions. In principle, the proof works by induction on the number of vertices and, below, we begin with the first proposition that serves the base case.

Proposition 6 *Let G be a star on $n \geq 3$ vertices and let $p \in \{0, 1\}$ be a given parity. Then $(T, k) = \rho(G, p)$ provides a k-leaf root T of G that is optimal for parity p (hence, $\pi(k) = p$) with $k = 4 - p \leq n + 1$ and satisfying* (T1)*,* (T2)*, and* (T3)*.*

Proof: We start by observing that G does not have true twins and, thus, $\rho(G, p)$ is well-defined. Moreover, since G is not complete, every 3-leaf root of G is an optimal (odd) leaf root and every 4-leaf root is an optimal even leaf root of G. By the same reason, any k'-leaf root T' of G satisfies $\mathrm{diam}(T') \geq k' + 1$.

We first consider the odd case, where $k = 3 = 4 - \pi(k)$, and show that the tree T returned in Step (i.1.) is a 3-leaf root of G satisfying the claimed conditions. In fact, T is just $\eta(G, 1)$ with an additional leaf attached for u. This is obviously a 3-leaf root of G with $\mathrm{rad}(T) = 2 = k - 1 < n$, $\mathrm{diam}(T) = 4$ and, thus, $\pi(T) = 0$ and $d_T^{\min} = 1 = 1 + \pi(T)$. Moreover, for any k'-leaf root T' of G, we see that

$$\mathrm{diam}(T') \geq k' + 1 = 4 + k' - 3 = \mathrm{diam}(T) + k' - k,$$

which settles the odd case.

In the even case, we have $k = 4$. If G has exactly two leaves v_1, v_2 attached to the central vertex u then Step (i.2.) returns the tree T where the leaves u, v_1 and v_2 are attached to a vertex v with paths of lengths one, two and three, respectively. Obviously, T is a 4-leaf root of G with $\mathrm{rad}(T) = 3 = k - 1 = n$, $\mathrm{diam}(T) = 5$ and, thus, $\pi(T) = 1$ and $d_T^{\min} = 2 = 1 + \pi(T)$. Every even k'-leaf root T' of G fulfills

$$\mathrm{diam}(T') \geq k' + 1 = 5 + k' - 4 = \mathrm{diam}(T) + k' - k,$$

which settles this case, too.

Finally, let $p = 0$ and $t \geq 3$, where Step (i.3.) constructs T from $\eta(G, 2)$ by attaching a new leaf to u. Similar to the odd case, T is obviously a 4-leaf root of G with $\text{rad}(T) = 3 = k - 1 < n$, $\text{diam}(T) = 6$ and, thus, $\pi(T) = 0$ and $d_T^{\min} = 1 = 1 + \pi(T)$. To see (T3), suppose to the contrary that $\text{diam}(T') < \text{diam}(T) + k' - k = 6 + k' - 4 = k' + 2$ for some even k'-leaf root T' of G. Consider the vertex w on the v_1, v_2-path in T' that is closest to v_3 and let $d_i = \text{dist}_{T'}(v_i, w)$ for all $i \in \{1, 2, 3\}$. Because of the diameter bound and since v_1, v_2 and v_3 are pairwise non-adjacent in G, we have $k'+1 \leq d_i+d_j < k'+2$ for all $1 \leq i < j \leq 3$. This means $d_1+d_2 = d_1+d_3 = d_2+d_3 = k'+1$. Hence, $2(d_1 + d_2 + d_3) = 3(k' + 1)$, which implies that $3(k' + 1)$ is even. This is a contradiction to the fact that k' is even and, thus, the last case is settled. Thus, Proposition 6 has been shown. □

With the following proposition, we only care about non-trivial connected input graphs and, for the moment, refrain from showing the optimality of (T, k) or even Property (T3).

Proposition 7 *Let G be a connected ccg on n vertices and without true twins and let $p \in \{0, 1\}$ be a given parity. Then $(T, k) = \rho(G, p)$ provides a k-leaf root T of G with $\pi(k) = p$ and $k \leq n + 1$ and satisfying* (T1) *and* (T2).

Proof: The proof works by induction on the number of vertices in G. The smallest connected ccg without true twins is the star with two leaves (since we ignore graphs with just one vertex). In Proposition 6, this base case has already been settled. Moreover, Proposition 6 allows to assume that G is not a larger star in the following induction step.

Next, let G have more than three vertices. Because G is connected and not a star, we are in Case ii. of the procedure for ρ. Again because G is connected and without true twins, it has a unique universal cut vertex by Proposition 3. Hence, $H = G - u$ is disconnected. Since G is not a star, H has $s \geq 1$ non-trivial connected components $G_1, \ldots, G_s$ and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$.

Note that, as for G, every G_i is a connected ccg without true twins but with fewer vertices than G. This means that the induction hypothesis holds and, thus, Step (ii.1.) provides a k_i-leaf root T_i of parity p satisfying (T1) and (T2) for every G_i, $1 \leq i \leq s$. We observe that $k_i \geq 3$ for all $i \in \{1, \ldots, s\}$ as none of $G_1, \ldots, G_s$ is a complete graph (since they are neither isolated vertices nor contain true twins).

At this point, we assume, without loss of generality, that $\text{diam}(T_1) \geq \text{diam}(T_2) \geq \cdots \geq \text{diam}(T_s)$. In addition to a simplified argumentation below, we get that

$$k_1 \geq k_2 \geq \cdots \geq k_s \text{ and also } k_1 - \pi(T_1) \geq k_2 - \pi(T_2) \geq \cdots \geq k_s - \pi(T_s).$$

This is firstly because of Lemma 1, which implies

$$\begin{aligned}
&\text{rad}(T_1) \geq \text{rad}(T_2) \geq \cdots \geq \text{rad}(T_s) \text{ and} \\
&\text{rad}(T_1) - \pi(T_1) \geq \text{rad}(T_2) - \pi(T_2) \geq \cdots \geq \text{rad}(T_s) - \pi(T_a),
\end{aligned}$$

and secondly because of the induction hypothesis $k_i = \text{rad}(T_i) + 1$, $1 \leq i \leq s$. Hence, if $s > 1$ then Step (ii.2.) selects $k_a = k_1$, $k_b = k_2$ and, if it exists, $k_c = k_3$.

Next, let k be the number computed in Step (ii.2.). We observe that $\pi(k) = p$. For $s = 1$, the values k and k_1 differ by an even number, thus, have the same parity p. If $s > 1$ and $p = 1$ then k is the odd sum of the three odd numbers k_1, k_2 and -1 and one even number. In the last two cases, where $s > 1$ and $p = 0$, the even value of k is the result of adding the even k_1, the even k_2 and one more even number. Moreover, it is easy to see that $k \geq k_i, 1 \leq i \leq s$. If $s = 1$ then

k is at least k_1. Otherwise, k is at least $k_1 + k_2 - 3$ with $k_1 \geq k_2 \geq \cdots \geq k_s \geq 3$. This makes $k \geq k_i, 1 \leq i \leq s$, even if $k_1 = k_2 = 3$.

By Lemma 4 and the induction hypothesis, Step (ii.3.) produces a k-leaf root $T'_i = \eta\left(T_i, \frac{k-k_i}{2}\right)$ of G_i for all $i \in \{1, \ldots, s\}$ such that

$$\begin{array}{rcll} \mathrm{diam}(T'_i) & = & \mathrm{diam}(T_i) + k - k_i & \text{by Lemma 4,} \\ \mathrm{rad}(T'_i) & = & \mathrm{rad}(T_i) + \frac{k-k_i}{2} = (k_i - 1) + \frac{k-k_i}{2} = \frac{k+k_i}{2} - 1 & \text{by Lemma 4 and (T1),} \\ k - 1 & \leq & |V(G_i)| + (k - k_i) & \text{by } k_i \leq |V(G_i)| + 1 \text{ and} \\ d^{\min}_{T'_i} & = & d^{\min}_{T_i} + \frac{k-k_i}{2} = (1 + \pi(T_i)) + \frac{k-k_i}{2} & \text{by Lemma 4 and (T2).} \end{array}$$

Then, Lemma 5 tells us that Step (ii.3.) continues by assembling a k-leaf root T' of H with the μ-operation. Observe that, under the given assumption that $\mathrm{diam}(T_1) \geq \mathrm{diam}(T_2) \geq \cdots \geq \mathrm{diam}(T_s)$, the μ-operation chooses the critical index $m = 1$. This happens because μ uses m to select the first tree in $T'_1, \ldots, T'_s$ that minimizes $d^{\min}_{T'_m} = 1 + \pi(T_m) + \frac{k-k_m}{2}$. In fact, if there was $i \in \{2, \ldots, s\}$ with $\pi(T_i) - \frac{k_i}{2} < \pi(T_1) - \frac{k_1}{2}$ then we would have $k_1 - k_i < 2(\pi(T_1) - \pi(T_i))$. Since $k_1 \geq k_i$ and both are of the same parity and $\pi(T_1), \pi(T_i) \in \{0, 1\}$, we could conclude that $k_1 = k_i$ and $\pi(T_1) = 1$ and $\pi(T_i) = 0$. But that would imply the contradiction

$$\begin{aligned} \mathrm{diam}(T_1) = 2\mathrm{rad}(T_1) - \pi(T_1) &= 2(k_1 - 1) - 1 < 2(k_1 - 1) - 0 \\ &= 2(k_i - 1) - \pi(T_i) = 2\mathrm{rad}(T_i) - \pi(T_i) = \mathrm{diam}(T_i). \end{aligned}$$

In the following, we show that $\mathrm{rad}(T') = k - 1 \leq n$ and, for that purpose, we firstly determine the diameter of T'. A diametral path in T' connects two leaves x and y, thus, vertices of H. To find the length of a diametral path, thus, the diameter of T', we subsequently analyze all possible origins of x, y in H.

For leaves x and y of the same $G_i, 1 \leq i \leq s$, the longest possible connecting path has length

$$\mathrm{diam}(T'_i) = 2\mathrm{rad}(T'_i) - \pi(T'_i) = 2\left(\tfrac{k+k_i}{2} - 1\right) - \pi(T_i) = k + k_i - \pi(T_i) - 2.$$

The longest path in T' connecting a leaf x in G_1 (recall the critical index $m = 1$) and $y = v_j, 1 \leq j \leq t$ has length

$$\begin{aligned} \ell_1 &= \left(\mathrm{rad}(T'_1) + \tfrac{k+\pi(k)}{2} - d^{\min}_{T'_1}\right) + \tfrac{k-\pi(k)}{2} + 1 \\ &= k + \mathrm{rad}(T'_1) - d^{\min}_{T'_1} + 1 \\ &= k + \left(\tfrac{k+k_1}{2} - 1\right) - \left(1 + \pi(T_1) + \tfrac{k-k_1}{2}\right) + 1 \\ &= k + k_1 - \pi(T_1) - 1. \end{aligned}$$

Similarly, a longest path of T' connecting a leaf x in G_i with $i \in \{2, \ldots, s\}$ and $y = v_j, 1 \leq j \leq t$ has length

$$\begin{aligned} \ell_i &= \left(\mathrm{rad}(T'_i) + \tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T'_i}\right) + \tfrac{k-\pi(k)}{2} + 1 \\ &= k - \pi(k) + \mathrm{rad}(T'_i) - d^{\min}_{T'_i} + 2 \\ &= k - \pi(k) + \left(\tfrac{k+k_i}{2} - 1\right) - \left(1 + \pi(T_i) + \tfrac{k-k_i}{2}\right) + 2 \\ &= k - \pi(k) + k_i - \pi(T_i). \end{aligned}$$

Next, let x be a leaf in G_1 (again, recall that the critical index m is 1) and y a leaf in G_i with $i \in \{2, \ldots, s\}$ that are most distant from each other in T'. The path connecting the leaves x and y has length

$$\begin{aligned}\ell_{1,i} &= \big(\mathrm{rad}(T'_1) + \tfrac{k+\pi(k)}{2} - d^{\min}_{T'_1}\big) + \big(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T'_i} + \mathrm{rad}(T'_i)\big)\\ &= k + (\mathrm{rad}(T'_1) - d^{\min}_{T'_1}) + (\mathrm{rad}(T'_i) - d^{\min}_{T'_i}) + 1\\ &= k + \tfrac{k+k_1}{2} - 1 - \big(1 + \pi(T_1) + \tfrac{k-k_1}{2}\big) + \tfrac{k+k_i}{2} - 1 - \big(1 + \pi(T_i) + \tfrac{k-k_i}{2}\big) + 1\\ &= k + k_1 - \pi(T_1) + k_i - \pi(T_i) - 3.\end{aligned}$$

The leaves x in G_i and y in G_j with $i, j \in \{2, \ldots, s\}$ and $i < j$ that are farthest from each other in T' have distance

$$\begin{aligned}\ell_{i,j} &= \big(\mathrm{rad}(T'_i) + \tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T'_i}\big) + \big(\tfrac{k-\pi(k)}{2} + 1 - d^{\min}_{T'_j} + \mathrm{rad}(T'_j)\big)\\ &= k - \pi(k) + (\mathrm{rad}(T'_i) - d^{\min}_{T'_i}) + (\mathrm{rad}(T'_j) - d^{\min}_{T'_j}) + 2\\ &= k - \pi(k) + \tfrac{k+k_i}{2} - 1 - \big(1 + \pi(T_i) + \tfrac{k-k_i}{2}\big) + \tfrac{k+k_j}{2} - 1 - \big(1 + \pi(T_j) + \tfrac{k-k_i}{2}\big) + 2\\ &= k - \pi(k) + k_i - \pi(T_i) + k_j - \pi(T_j) - 2.\end{aligned}$$

Finally, any $x = v_i$ and $y = v_j$, $1 \le i < j \le t$, have distance

$$\mathrm{dist}_{T'}(v_i, v_j) = 2\big(\tfrac{k-\pi(k)}{2} + 1\big) = k - \pi(k) + 2.$$

Before we can determine the diameter of T', we need to sort out the longest paths of T' from those given above. If $t > 0$ then we have that

$$\ell_1 = k + k_1 - \pi(T_1) - 1 > (k + k_1 - 2) - \pi(T_1) = 2\mathrm{rad}(T'_1) - \pi(T'_1) = \mathrm{diam}(T'_1).$$

This leads to $\ell_1 > \mathrm{diam}(T'_1) \ge \mathrm{diam}(T'_2) \ge \cdots \ge \mathrm{diam}(T'_s)$. Moreover, in case of $t \ge 2$, we also find that

$$\ell_1 - \mathrm{dist}_{T'}(v_i, v_j) = (k + k_1 - \pi(T_1) - 1) - (k - \pi(k) + 2) = k_1 + \pi(k) - \pi(T_1) - 3$$

for all $1 \le i < j \le t$. If $k_1 = 3$ then, because $\pi(k) = \pi(k_1) = 1$, we get $\ell_1 - \mathrm{dist}_{T'}(v_i, v_j) = 1 - \pi(T_1) \ge 0$, which means $\ell_1 \ge \mathrm{dist}_{T'}(v_i, v_j)$. This holds even more so if $k_1 > 3$.

If $s \ge 2$ then

$$\begin{aligned}\ell_{1,2} - \ell_{1,i} &= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) - (k + k_1 - \pi(T_1) + k_i - \pi(T_i) - 3)\\ &= ((k_2 - 1) - \pi(T_2)) - ((k_i - 1) - \pi(T_i))\\ &= (\mathrm{rad}(T_2) - \pi(T_2)) - (\mathrm{rad}(T_i) - \pi(T_i))\\ &\ge 0 \text{ for all } i \ge 2 \text{ by Lemma 1},\\ \ell_{1,2} - \mathrm{diam}(T'_1) &= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) - (k + k_1 - \pi(T_1) - 2)\\ &= k_2 - \pi(T_2) - 1 > 0 \text{ since } k_2 \ge 3, \text{ and}\\ \text{if } t > 0 \text{ then } \ell_{1,2} - \ell_1 &= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) - (k + k_1 - \pi(T_1) - 1)\\ &= k_2 - \pi(T_2) - 2 \ge 0 \text{ because } k_2 \ge 3 \text{ and, if also}\\ 2 \le i \le s \text{ then } \ell_{1,2} - \ell_i &= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) - (k - \pi(k) + k_i - \pi(T_i))\\ &\ge k_1 - \pi(T_1) - 3 + \pi(k) \text{ because } k_2 - \pi(T_2) \ge k_i - \pi(T_i)\end{aligned}$$

Moreover, if $s > 2$ then

$$\begin{aligned}
\ell_{1,2} - \ell_{2,3} &= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) - (k - \pi(k) + k_2 - \pi(T_2) + k_3 - \pi(T_3) - 2)\\
&= ((k_1 - 1) - \pi(T_1)) - ((k_3 - 1) - \pi(T_3)) + \pi(k) - 1\\
&= (\mathrm{rad}(T_1) - \pi(T_1)) - (\mathrm{rad}(T_3) - \pi(T_3)) + \pi(k) - 1\\
&\geq 0 \text{ if } \pi(k) = 1 \text{ by Lemma 1 and}\\
\ell_{2,3} - \ell_{i,j} &= (k - \pi(k) + k_2 - \pi(T_2) + k_3 - \pi(T_3) - 2) - (k - \pi(k) + k_i - \pi(T_i) + k_j - \pi(T_j) - 2)\\
&= \big(((k_2 - 1) - \pi(T_2)) - ((k_i - 1) - \pi(T_i))\big) + \big(((k_3 - 1) - \pi(T_3)) - ((k_j - 1) - \pi(T_j))\big)\\
&= \big((\mathrm{rad}(T_2) - \pi(T_2)) - (\mathrm{rad}(T_i) - \pi(T_i))\big) + \big((\mathrm{rad}(T_3) - \pi(T_3)) - (\mathrm{rad}(T_j) - \pi(T_j))\big)\\
&\geq 0 \text{ if } 1 < i < j \leq s \text{ by Lemma 1.}
\end{aligned}$$

It remains to compare $\ell_{1,2}$ and $\ell_{2,3}$ in the even case where $\pi(k) = 0$:

$$\begin{aligned}
\ell_{1,2} - \ell_{2,3} &= ((k_1 - 1) - \pi(T_1)) - ((k_3 - 1) - \pi(T_3)) + \pi(k) - 1\\
&= (k_1 - \pi(T_1)) - (k_3 - \pi(T_3)) - 1 \begin{cases} \geq 0, & \text{if } k_1 - \pi(T_1) > k_3 - \pi(T_3),\\ = -1, & \text{otherwise, that is } k_1 - \pi(T_1) = k_3 - \pi(T_3). \end{cases}
\end{aligned}$$

We are now ready to estimate the diameter of T' and, thus, the radius. If $s = 1$ (and therefore $t > 0$) then

$$\mathrm{diam}(T') = \max\{\mathrm{diam}(T'_1), \ell_1, \mathrm{dist}_{T'}(v_i, v_j) \mid 1 \leq i < j \leq t\}.$$

From the observations above, $\mathrm{diam}(T') = \ell_1$. In Step (ii.2.), we see that $k = k_1 + 2(1 - \pi(T_1))$ and so

$$\ell_1 = k + k_1 - \pi(T_1) - 1 = k + (k - 2 + 2\pi(T_1)) - \pi(T_1) - 1 = 2k - 3 + \pi(T_1).$$

Obviously, $\pi(T') = 1 - \pi(T_1)$, which means that $\mathrm{rad}(T') = k - 1$. Note also that, since $n \geq |V(G_1)| + 2$ (because $\{u, v_1\} \subseteq V(G) \setminus V(G_1)$)

$$k - 1 \leq |V(G_1)| + (k - k_1) \leq (n - 2) + (2 - 2\pi(T_1)) \leq n.$$

If $s \geq 2$ then let

$$\begin{aligned}
m_1 &= \max\{\mathrm{diam}(T'_1), \dots, \mathrm{diam}(T'_s)\},\\
m_2 &= \max\{\ell_{i,j} \mid 1 \leq i < j \leq s\},\\
m_3 &= \max\{\ell_1, \dots, \ell_t\},\\
m_4 &= \max\{\mathrm{dist}_{T'}(v_i, v_j) \mid 1 \leq i < j \leq t\}.
\end{aligned}$$

Then, apparently, $\mathrm{diam}(T') = \max\{m_1, m_2, m_3, m_4\}$. By the above arguments, $\ell_{1,2} \geq \ell_1 \geq \max\{m_1, m_4\}$. If $p = \pi(k) = 1$ then $\ell_{1,2} \geq m_2$ and, moreover, $\ell_{1,2} \geq m_3 + k_1 - \pi(T_1) - 3 + \pi(k) \geq m_3$, since $k_1 + \pi(k) \geq 4$. Thus, in this case, $\mathrm{diam}(T') = \ell_{1,2}$.

Otherwise, if $p = \pi(k) = 0$, we first consider the case $k_1 - \pi(T_1) > k_3 - \pi(T_3) \geq 2$. Again, $\mathrm{diam}(T') = \ell_{1,2}$ since $\ell_{1,2} \geq m_2$ and $\ell_{1,2} \geq m_3 + k_1 - \pi(T_1) - 3 \geq m_3$, as $k_1 - \pi(T_1) \geq 3$.

Finally, if $p = \pi(k) = 0$ and $k_1 - \pi(T_1) = k_3 - \pi(T_3)$, then we know that $\ell_{2,3} > \ell_{1,2}$. In this case, $\mathrm{diam}(T') = \ell_{2,3}$, as $\ell_{2,3} \geq m_2$ and $\ell_{2,3} \geq \ell_{1,2} + 1 \geq m_3 + k_1 - \pi(T_1) - 3 + 1 \geq m_3$, because $k_1 \geq 3$. Thus,

$$\mathrm{diam}(T') = \begin{cases} \ell_{2,3}, & \text{if } p = 0 \text{ and } s > 2 \text{ and } k_1 - \pi(T_1) = k_3 - \pi(T_3),\\ \ell_{1,2}, & \text{otherwise,} \end{cases}$$

according to the considerations above. In the first case, where

$$\text{diam}(T') = \ell_{2,3} = k - \pi(k) + k_2 - \pi(T_2) + k_3 - \pi(T_3) - 2,$$

the value $k - \pi(k) + k_2 + k_3 - 2$ is even because of $p = 0$, and, therefore, $\pi(T') = \pi(T_2) + \pi(T_3) - 2\pi(T_2) \cdot \pi(T_3)$. Since $k_1 \geq k_2 \geq k_3$ are of the same parity and because

$$k_1 - \pi(T_1) \geq k_2 - \pi(T_2) \geq k_3 - \pi(T_3) = k_1 - \pi(T_1),$$

it must be the case that $k_1 = k_2 = k_3$ and $\pi(T_1) = \pi(T_2) = \pi(T_3)$. Moreover, Step (ii.2.) sets $k = k_1 + k_2 - 2 \cdot \pi(T_1) \cdot \pi(T_2)$ in this case. Because $k_1 = k_3$ and $\pi(T_1) = \pi(T_3)$, we get

$$\ell_{2,3} = k - 2 + (k_1 + k_2 - 2 \cdot \pi(T_1) \cdot \pi(T_2)) - (\pi(T_2) + \pi(T_3) - 2\pi(T_2) \cdot \pi(T_3)) = 2k - 2 - \pi(T')$$

and, hence, $\text{rad}(T') = k - 1$. Moreover, since $n \geq |V(G_1)| + |V(G_2)| + 1$ (because $u \in V(G) \setminus (V(G_1) \cup V(G_2))$), we have

$$\begin{aligned}(k-1) + (k-1) &\leq (|V(G_1)| + (k - k_1)) + (|V(G_2)| + (k - k_2)) \\ &= (|V(G_1)| + |V(G_2)|) + (k - k_1 - k_2) + k \\ &\leq (n-1) - 2 \cdot \pi(T_1) \cdot \pi(T_2) + k \\ &\leq n + (k-1),\end{aligned}$$

which means that $k - 1 \leq n$.

In the other case, where

$$\text{diam}(T') = \ell_{1,2} = k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3,$$

we distinguish between $p = 0$ and $p = 1$. If $p = 0$ then $k + k_1 + k_2 - 3$ is odd and $\pi(T') = 1 - \pi(T_1) - \pi(T_2) + 2 \cdot \pi(T_1) \cdot \pi(T_2)$. Moreover, Step (ii.2.) sets $k = k_1 + k_2 - 2 \cdot (\pi(T_1) + \pi(T_2) - \pi(T_1) \cdot \pi(T_2))$, and we get

$$\begin{aligned}\ell_{1,2} &= k - 2 + (k_1 + k_2 - 2(\pi(T_1) + \pi(T_2) - \pi(T_1) \cdot \pi(T_2)) - (1 - \pi(T_1) - \pi(T_2) + 2\pi(T_1) \cdot \pi(T_2)) \\ &= 2k - 2 - \pi(T'),\end{aligned}$$

which again means, $\text{rad}(T') = k - 1$. Otherwise, if $p = 1$ then $k + k_1 + k_2 - 3$ is even and $\pi(T') = \pi(T_1) + \pi(T_2) - 2 \cdot \pi(T_1) \cdot \pi(T_2)$. In this final case, Step (ii.2.) defines $k = k_1 + k_2 - 1 - 2 \cdot \pi(T_1) \cdot \pi(T_2))$ and, thus,

$$\ell_{1,2} = k - 2 + (k_1 + k_2 - 1 - 2 \cdot \pi(T_1) \cdot \pi(T_2)) - (\pi(T_1) + \pi(T_2) - 2 \cdot \pi(T_1) \cdot \pi(T_2)) = 2k - 2 - \pi(T')$$

and we can conclude that $\text{rad}(T') = k - 1$ holds in every case. Similarly to the previous case, we also get

$$\begin{aligned}(k-1) + (k-1) &\leq (|V(G_1)| + (k - k_1)) + (|V(G_2)| + (k - k_2)) \\ &= (|V(G_1)| + |V(G_2)|) + (k - k_1 - k_2) + k \\ &\leq (n-1) - (2 \cdot \pi(T_1) \cdot \pi(T_2) + 1) + k \\ &\leq n + (k-1),\end{aligned}$$

and, again, $k - 1 \leq n$.

With T', the algorithm has created a k-leaf root of $H = G - u$ with $\text{rad}(T') = k - 1 \leq n$ and $\pi(k) = p$. Step (ii.4.) extends T' to a tree T by attaching the universal vertex u of G as a leaf pendent to a center vertex of T'. Obviously, T is a k-leaf root of G with $\pi(k) = p$ and Property (T1) $\text{rad}(T) = k - 1 \leq n$ because $\text{dist}_T(x, y) = \text{dist}_{T'}(x, y)$ for all $x, y \in V(H) = V(G) - u$ and

$$\text{dist}_T(u, x) \leq \text{rad}(T) + 1 = \text{rad}(T') + 1 = (k - 1) + 1 = k.$$

Moreover, T has Property (T2) $d_T^{\min} = 1 + \pi(T)$. If $\pi(T) = 0$ then the center of T is a single vertex z, which is also the min-max center. Since z is adjacent to the leaf u, we get $d_T^{\min} = 1$. Otherwise, if $\pi(T) = 1$ then T has two center vertices z_1 and z_2 where, without loss of generality, u is attached to z_1. That makes z_2 the min-max center with distance $d_T^{\min} = 2$ to leaf u. This concludes the proof of Proposition 7. □

With the following proposition, we summarize what we have so far and also cover the disconnected case.

Proposition 8 *Let G be a ccg on n vertices and without true twins and let $p \in \{0, 1\}$ be a given parity. Then $(T, k) = \rho(G, p)$ provides a k-leaf root T of G with $\pi(k) = p$ and $k \leq n + 1$. If G is connected then (T, k) satisfies* (T1) *and* (T2)*.*

Proof: With the Propositions 6 and 7, we have shown for connected input graphs G that $(T, k) = \rho(G, p)$ returns a k-leaf root T of G with $\pi(k) = p$ and that has the properties (T1) and (T2). It remains to handle disconnected input. In this case, G has $s \geq 0$ non-trivial connected components $G_1, \ldots, G_s$ and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$.

If $s = 0$ then G consists of $n \geq 2$ isolated vertices and, thus, Step (iii.2.) sets $k = 2$ for even parity and $k = 3$, otherwise. Then, by Lemma 5, Step (iii.3.) returns a k-leaf root T of G.

For $s > 0$, every G_i is a connected ccg without true twins. By Proposition 7, Step (iii.1.) provides a k_i-leaf root T_i of parity p for every G_i, $1 \leq i \leq s$. The value assigned to k in Step (iii.2.) is the maximum $k_m = \max\{k_1, \ldots, k_s\}$. Since $k_m \leq |V(G_m)| + 1$ and $|V(G_m)| < n$, we have $k \leq n + 1$. According to Lemma 4, the same step produces a k-leaf root $T_i' = \eta\left(T_i, \frac{k - k_i}{2}\right)$ of G_i for all $i \in \{1, \ldots, s\}$. Finally, by Lemma 5, Step (iii.3.) returns a k-leaf root T of G.

Notice that, the k-leaf root T returned by $\rho(G, p)$ for disconnected input graphs G does not need to satisfy (T1) and (T2). □

The following, last proposition finalizes the proof of Theorem 5 because it shows that the k-leaf root T returned by $\rho(G, p)$ is optimal with respect to both the value of k and the diameter of T.

Proposition 9 *Let G be a ccg without true twins and let $p \in \{0, 1\}$ be a given parity. Then $(T, k) = \rho(G, p)$ provides a k-leaf root T of G that is* optimal *for parity p (hence, $\pi(k) = p$). If G is connected then (T, k) satisfies* (T3)*.*

Proof: That $(T, k) = \rho(G, p)$ provides a k-leaf root T of G with $\pi(k) = p$ has just been established. We only need to show that T fulfills (T3) for connected input graphs G and that k is always parity-optimal.

We begin with the proof of (T3) and, like for Proposition 8, this works by induction on the number of vertices in G. Again, for the base case, Proposition 6 handles the star with two leaves, the smallest connected ccg without true twins and more than one vertex. Proposition 6 settles the theorem for all stars.

Now, let G have more than three vertices. That G is connected and not a star leads us to Case ii. of ρ, again. Borrowing the argumentation from the proof of Proposition 8, we immediately get that

- G has a unique universal cut vertex u and $H = G - u$ has $s \geq 1$ non-trivial connected components $G_1, \ldots, G_s$, each a ccg without true twins, and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$,
- by induction hypothesis, Step (ii.1.) provides k_i-leaf roots T_i of parity p for all $G_i, 1 \leq i \leq s$, which are optimal and satisfy (T3), here,
- $k_1 \geq k_2 \geq \cdots \geq k_s$, if, without loss of generality, we assume

$$\operatorname{diam}(T_1) \geq \operatorname{diam}(T_2) \geq \cdots \geq \operatorname{diam}(T_s),$$

- Step (ii.3.) produces k-leaf roots $T_i' = \eta\left(T_i, \frac{k-k_i}{2}\right)$ of all $G_i, 1 \leq i \leq s$ with

$$k = \begin{cases} k_1 + 2(1 - \pi(T_1)), & \text{if } s = 1, \\ k_1 + k_2 - 1 - 2 \cdot \pi(T_1) \cdot \pi(T_2) & \text{if } s \geq 2, p = 1, \\ k_1 + k_2 - 2 \cdot \pi(T_1) \cdot \pi(T_2), & \text{if } s > 2, p = 0, k_1 - \pi(T_1) = k_3 - \pi(T_3), \\ k_1 + k_2 - 2 \cdot (\pi(T_1) + \pi(T_2) - \pi(T_1) \cdot \pi(T_2)), & \text{otherwise, and} \end{cases}$$

$$\begin{aligned} \operatorname{diam}(T_i') &= \operatorname{diam}(T_i) + k - k_i, \\ \operatorname{rad}(T_i') &= \operatorname{rad}(T_i) + \tfrac{k-k_i}{2} = \tfrac{k+k_i}{2} - 1, \end{aligned}$$

- Step (ii.3.) produces $T' = \mu(k, T_1', \ldots, T_s', v_1, \ldots, v_t)$ with

$$\operatorname{diam}(T') = \begin{cases} k + k_1 - \pi(T_1) - 1, & \text{if } s = 1, \\ k + k_2 - \pi(T_2) + k_3 - \pi(T_3) - 2, & \text{if } s > 2, p = 0, k_1 - \pi(T_1) = k_3 - \pi(T_3), \\ k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3, & \text{otherwise, and} \end{cases}$$

- $(T, k) = \rho(G, p)$, as assembled in Step (ii.4.), represents a k-leaf root of G with $\operatorname{diam}(T) = \operatorname{diam}(T')$.

Next, consider an arbitrary k'-leaf root R of G and, for all $i \in \{1, \ldots, s\}$, let R_i be the smallest subtree of R with leaf set $V(G_i)$.

To see that R_i and R_j are vertex-disjoint for all distinct $i, j \in \{1, \ldots, s\}$, assume there is a vertex c that belongs to both R_i and R_j. Since $V(G_i)$ and $V(G_j)$ are disjoint, c is not a leaf. Because G_i and G_j are connected, there are adjacent $v, w \in V(G_i)$ and adjacent $x, y \in V(G_j)$ such that c is on both, the path in R_i between v and w, which has length at most k', and the path in R_j between x and y, which has length at most k', too. Without loss of generality, $\operatorname{dist}_{R_i}(c, v) \leq \frac{k'}{2}$ and $\operatorname{dist}_{R_j}(c, x) \leq \frac{k'}{2}$, which implies the contradiction $\operatorname{dist}_R(v, x) \leq k'$. Then vx would be an edge in G while, at the same time, there are no edges between G_i and G_j in $G - u$.

Clearly, R_i is a k'-leaf root of G_i and, since T_i has (T3), $\operatorname{diam}(R_i) \geq \operatorname{diam}(T_i) + k' - k_i$. Since $\operatorname{diam}(T_i') = \operatorname{diam}(T_i) + k - k_i$ is in the listing above, we get

$$\operatorname{diam}(R_i) \geq \operatorname{diam}(T_i) + k' - k_i = (\operatorname{diam}(T_i') - k + k_i) + k' - k_i = \operatorname{diam}(T_i') + k' - k.$$

Now that we have established lower bounds on the diameters of the subtrees $R_1, \ldots, R_s$, we want to use them to obtain a tight lower bound on the diameter of R.

For this, let u_i be the unique universal vertex of G_i by Proposition 3. By Lemma 3, there is a center vertex z_i of R_i with $\text{dist}_{R_i}(u_i, z_i) \leq k' - \text{rad}(R_i)$. Let x_i and y_i be the two leaves of a diametral path D_i in R_i, thus, a path of length $\text{diam}(R_i)$. Then, the center vertex z_i of T_i is on D_i and $\text{dist}_{R_i}(u_i, z_i) = \text{dist}_R(u_i, z_i)$.

Next, we determine a long path in R and, with the length of this path, imply a lower bound on $\text{diam}(R)$. First, suppose $s = 1$. Then $\text{dist}_R(z_1, v_1) > \text{rad}(R_1)$ for, otherwise,

$$\text{dist}_R(u_1, v_1) \leq \text{dist}_R(u_1, z_1) + \text{dist}_R(z_1, v_1) \leq (k' - \text{rad}(R_1)) + \text{rad}(R_1) = k',$$

and u_1 and v_1 would be adjacent in G. Since R is a tree, the path X between v_1 and x_1 and the path Y between v_1 and y_1 share a common subpath with one endpoint v_1 and the other, say c, on D_1. If Y is longer than X then c is on the x_1, z_1-path and, thus, Y contains z_1. Otherwise, if X is at least as long as Y then c is on the z_1, y_1-path and z_1 is on X. This means, the length ℓ of the longest path among X and Y fulfills

$$\ell \geq \text{dist}_R(v_1, z_1) + \text{rad}(R_1) > 2\text{rad}(R_1) \geq \text{diam}(R_1) \geq \text{diam}(T_1') + k' - k.$$

By the bound on the length ℓ of one path in R we get $\text{diam}(R) > \text{diam}(T_1') + k' - k$, and we conclude

$$\begin{aligned}\text{diam}(R) &\geq \text{diam}(T_1') + k' - k + 1 = (2\text{rad}(T_1') - \pi(T_1')) + k' - k + 1\\ &= (k + k_1 - 2) - \pi(T_1') + k' - k + 1\\ &= (k + k_1 - \pi(T_1) - 1) + k' - k\\ &= \text{diam}(T) + k' - k,\end{aligned}$$

as claimed.

If $s \geq 2$ then $\text{dist}_R(z_i, z_j) \geq \text{rad}(R_i) + \text{rad}(R_j) - k' + 1$ for all $1 \leq i < j \leq s$ since, otherwise,

$$\begin{aligned}\text{dist}_R(u_i, u_j) &\leq \text{dist}_R(u_i, z_i) + \text{dist}_R(z_i, z_j) + \text{dist}_R(z_j, u_j)\\ &\leq (k' - \text{rad}(R_i)) + (\text{rad}(R_i) + \text{rad}(R_j) - k') + (k' - \text{rad}(R_j)) = k',\end{aligned}$$

and u_i and u_j would be adjacent in G.

Similar to the argument for $s = 1$, we consider certain paths in R and select the longest among them. Here, we have four paths XX, XY, YX and YY going from a vertex of $\{x_i, y_i\}$ to a vertex of $\{x_j, y_j\}$. Since R is a tree, the four path share a common subpath with one endpoint, say c, on D_i and the other, say c', on D_j. With a similar argument as above, we get that a longest path among XX, XY, YX and YY contains z_i and z_j. For example, if YY is at least as long as the other three paths then c is on the x_i, z_i-path and c' is on the z_j, x_j-path and, thus, YY contains both, z_i and z_j. The length $\ell_{i,j}$ of the selected long path fulfills

$$\begin{aligned}\ell_{i,j} &\geq \text{rad}(R_i) + \text{dist}_R(z_i, z_j) + \text{rad}(R_j)\\ &\geq \text{rad}(R_i) + \big(\text{rad}(R_i) + \text{rad}(R_j) - k + 1\big) + \text{rad}(R_j)\\ &= \text{diam}(R_i) + \pi(R_i) + \text{diam}(R_j) + \pi(R_j) - k' + 1.\end{aligned}$$

Because $\mathrm{diam}(R) \geq \ell_{1,2}$ and $\pi(R_1) \geq 0$ and $\pi(R_2) \geq 0$, we find that

$$\begin{aligned}
\mathrm{diam}(R) &\geq \mathrm{diam}(R_1) + \mathrm{diam}(R_2) - k' + 1 \\
&\geq (\mathrm{diam}(T_1) + k' - k_1) + (\mathrm{diam}(T_2) + k' - k_2) - k' + 1 \\
&= (2\mathrm{rad}(T_1) - \pi(T_1) - k_1) + (2\mathrm{rad}(T_2) - \pi(T_2) - k_2) + k' + 1 \\
&= (2k_1 - 2 - \pi(T_1) - k_1) + (2k_2 - 2 - \pi(T_2) - k_2) + k' + (k - k) + 1 \\
&= (k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3) + k' - k.
\end{aligned}$$

Hence, if $p = 1$ or $s = 2$ or ($s > 2$ and) $k_1 - \pi(T_1) > k_3 - \pi(T_3)$ then (as $\mathrm{diam}(T) = k + k_1 - \pi(T_1) + k_2 - \pi(T_2) - 3$, here) we already get

$$\mathrm{diam}(R) \geq \mathrm{diam}(T) + k' - k.$$

In the remaining case, we have $p = 0$ and $s > 2$ and $k_1 - \pi(T_1) = k_3 - \pi(T_3)$. Like in the proof of Proposition 7, we can conclude that $k_1 = k_2 = k_3$ and $\pi(T_1) = \pi(T_2) = \pi(T_3)$, here. Thus, $\mathrm{diam}(T_1) = \mathrm{diam}(T_2) = \mathrm{diam}(T_3)$ and $\mathrm{diam}(T_1') = \mathrm{diam}(T_2') = \mathrm{diam}(T_3')$, too. If we write $\delta = \mathrm{diam}(T_1') + k' - k$ then $\mathrm{diam}(R_i) \geq \delta$ for all $i \in \{1, 2, 3\}$.

Here, we show that at least one of the three length $\ell_{1,2}, \ell_{1,3}, \ell_{2,3}$ is at least $2\delta - k' + 2$. If $\mathrm{diam}(R_i) + \pi(R_i) > \delta$ for some $i \in \{1, 2, 3\}$, then

$$\begin{aligned}
\ell_{i,j} &\geq \mathrm{diam}(R_i) + \pi(R_i) + \mathrm{diam}(R_j) + \pi(R_j) - k' + 1 \\
&\geq (\delta + 1) + \delta - k' + 1 \\
&\geq 2\delta - k' + 2
\end{aligned}$$

for all $j \in \{1, 2, 3\} \setminus \{i\}$ and we are done, already. Therefore, consider that $\mathrm{diam}(R_i) + \pi(R_i) \leq \delta$ for all $i \in \{1, 2, 3\}$. This means, for all $i \in \{1, 2, 3\}$, that $\delta \leq \mathrm{diam}(R_i) \leq \delta - \pi(R_i)$, which is only possible if $\mathrm{diam}(R_i) = \delta$ and $\pi(R_i) = 0$. In other words, the diameters of R_1, R_2 and R_3 are equal to the even number δ. Hence, $\mathrm{rad}(R_1) = \mathrm{rad}(R_2) = \mathrm{rad}(R_3) = \frac{\delta}{2}$. If $\mathrm{dist}_R(z_i, z_j) \geq \delta - k' + 2$ for any distinct $i, j \in \{1, 2, 3\}$, then

$$\begin{aligned}
\ell_{i,j} &\geq \mathrm{rad}(R_i) + \mathrm{dist}_R(z_i, z_j) + \mathrm{rad}(R_j) \\
&\geq \tfrac{\delta}{2} + (\delta - k' + 2) + \tfrac{\delta}{2} \\
&= 2\delta - k' + 2
\end{aligned}$$

and we are done, again. So, finally, assume that $\mathrm{dist}_R(z_i, z_j) \leq \delta - k' + 1$ for all $i, j \in \{1, 2, 3\}$. Then, since we have

$$\mathrm{dist}_R(z_i, z_j) \geq \mathrm{rad}(R_i) + \mathrm{rad}(R_j) - k' + 1 = \tfrac{\delta}{2} + \tfrac{\delta}{2} - k' + 1$$

from before, $\mathrm{dist}_R(z_1, z_2) = \mathrm{dist}_R(z_1, z_3) = \mathrm{dist}_R(z_2, z_3) = \delta - k' + 1$. Let w be the last vertex on the z_1, z_2-path in R that is closest to z_3 and let δ_i be the distances between z_i and w for all $i \in \{1, 2, 3\}$. Then, for all $1 \leq i < j \leq 3$, it must be

$$\delta_i + \delta_j = \mathrm{dist}_R(z_i, z_j) = \delta - k' + 1.$$

Solving the system of the three linear equations above yields

$$\delta_1 = \delta_2 = \delta_3 = \tfrac{1}{2}(\delta - k' + 1),$$

which implies that k' is odd, a contradiction to $\pi(k') = \pi(k) = p = 0$.

So at least one length, say $\ell_{1,2}$, is at least $2\delta - k' + 2$. This means that

$$\begin{aligned}
\text{diam}(R) &\geq \ell_{1,2} \geq 2\delta - k' + 2 \\
&= (\text{diam}(T_1') + k' - k) + (\text{diam}(T_3') + k' - k) - k' + 2 \\
&= (\text{diam}(T_1) + k - k_1 + k' - k) + (\text{diam}(T_3) + k - k_3 + k' - k) - k' + 2 \\
&= (2\text{rad}(T_1) - \pi(T_1) + k' - k_1) + (2\text{rad}(T_3) - \pi(T_3) + k' - k_3) - k' + 2 \\
&= (2\text{rad}(T_1) - \pi(T_1) - k_1) + (2\text{rad}(T_3) - \pi(T_3) - k_3) + k' + (k - k) + 2 \\
&= k + (k_1 - 2 - \pi(T_1)) + (k_3 - 2 - \pi(T_3)) + k' - k + 2 \\
&= (k + k_1 - \pi(T_1) + k_3 - \pi(T_3) - 2) + k' - k.
\end{aligned}$$

Because here, $\text{diam}(T) = k + k_1 - \pi(T_1) + k_3 - \pi(T_3) - 2$, we have now shown that $\text{diam}(R) \geq \text{diam}(T) + k' - k$.

It remains to prove that the value of k in $(T, k) = \rho(G, p)$ is optimal with respect to leaf roots of G with parity p. For the connected case, this can be seen as follows. By Corollary 4, $2k' - 2 \geq \text{diam}(R)$ holds for every k'-leaf roots R of G. This and Property (T3) of T leads to

$$2k' - 2 \geq \text{diam}(R) \geq \text{diam}(T) + k' - k.$$

Since $\text{diam}(T) = 2\text{rad}(T) - \pi(T)$ and $\text{rad}(T) = k - 1$ by Property (T1) of T, we get

$$\text{diam}(T) + k' - k = (2\text{rad}(T) - \pi(T)) + k' - k = 2(k - 1) - \pi(T) + k' - k \geq k + k' - 3.$$

Together, this implies $2k' - 2 \geq k + k' - 3$, which means that $k' \geq k - 1$. Because both, k and k', have the same parity, we get $k' \geq k$. That is, T is an optimal k-leaf root of G with parity p.

Finally, let G be a disconnected graph with $s \geq 0$ non-trivial connected components $G_1, \ldots, G_s$ and $t \geq 0$ isolated vertices $v_1, \ldots, v_t$ such that $s + t \geq 2$. By Proposition 8, $(T, k) = \rho(G, k)$ results from finding $(T_1, k_1) = \rho(G_1, p), \ldots, (T_s, k_s) = \rho(G_s, p)$ in Step (iii.1.) and assembling them to a k-leaf root T for G with $k = \max\{k_1, \ldots, k_s, p + 2\}$ in the Steps (iii.2.) and (iii.3.). After handling the connected case above, we know that T_i is an optimal k_i-leaf root of G_i with parity p for all $i \in \{1, \ldots, s\}$.

Clearly, if $s = 0$ then a $(p + 2)$-leaf root is the best possible for parity p, and we are done. Otherwise, assume, without loss of generality, that $k = k_1$. Moreover, let R be a p-parity k'-leaf root of G and, for all $i \in \{1, \ldots, s\}$, let R_i be the smallest subtree of R with leaf set $V(G_i)$. Then R_1 is a k'-leaf root of G_1. Since k_1 is optimal for G_1, we have that $k' \geq k_1 = k$, thus, T is an optimal k-leaf root of G with parity p. □

With the four propositions above, Theorem 5 has been proven. □

5 Linear Time Leaf Root Construction for CCGs

The algorithm in this section is an implementation of the ρ-operation from Section 3. Here, the recursive subdivision of the input ccg G is replaced with a post-order iteration of the cotree of G. Moreover, the tree representation of the output leaf root T has to necessarily be *compressed* by the algorithm. This means that T is encoded with a denser representation, where long paths of degree two-vertices are compressed into single weighted edges. Otherwise, the size of T alone would be quadratic in the input length. But before we go into the details, we analyze the used submodules and show that the operations η and μ run efficiently.

Lemma 6 *Let T be a compressed tree with n leaves and with explicitly given min-max center z, center Z, diameter* $\operatorname{diam}(T)$*, and leaf-distance $d_T^{\min}$. For all integers $\delta \geq 0$, the compressed tree $T' = \eta(T, \delta)$ with min-max center z', center Z', diameter* $\operatorname{diam}(T')$*, and leaf distance $d_{T'}^{\min}$ can be computed in $\mathcal{O}(n)$ time.*

Proof: By definition, T' is obtained from T by replacing all n pendant edges with new paths of length $\delta + 1$, each. In the compressed encoding, this takes just n modifications of the weights of edges. More precisely, if $v(\ell)x$ is a pendent edge of T representing a path of length ℓ that ends at leaf x (where we simply take $\ell = 1$ for an unweighted edge) then T' has the same pendent edge $v(\ell')x$ with the new weight $\ell' = \ell + \delta$. The n constant-time modifications take $\mathcal{O}(n)$ time, altogether.

According to Lemma 4, $z' = z$, $Z' = Z$, $\operatorname{diam}(T') = \operatorname{diam}(T) + 2\delta$ and $d_{T'}^{\min} = d_T^{\min} + \delta$, which takes just constant time to compute. □

Lemma 7 *Let $s \geq 0$ be an integer and, for all $i \in \{1, \ldots, s\}$, let T_i be a given, compressed tree with explicitly given min-max center z_i, center Z_i, diameter* $\operatorname{diam}(T_i)$*, and leaf-distance $d_{T_i}^{\min}$. For all integers $k \geq 2$ and vertices $v_1, \ldots, v_t$, the merged compressed tree $T' = \mu(k, T_1, \ldots, T_s, v_1, \ldots, v_t)$ with center Z' and diameter* $\operatorname{diam}(T')$ *can be computed in $\mathcal{O}(s + t)$ time.*

Proof: According to definition, the computation of T' by μ starts with finding the critical index m, which is the smallest number in $\{1, \ldots, s\}$ with $d_{T_m}^{\min} = \min\{d_{T_i}^{\min} \mid 1 \leq i \leq s\}$. Since $d_{T_i}^{\min}$ is explicitly known for all given trees, m is found after iterating the input once in $\mathcal{O}(s)$ time.

Then, every tree $T_i, 1 \leq i \leq s$ is attached at z_i to a new vertex c by a path of at most $\frac{k-\pi(k)}{2} + 1 - d_{T_i}^{\min}$ edges and every vertex $v_j, 1 \leq j \leq t$ is attached to c with a path of length $\frac{k-\pi(k)}{2} + 1$. Each of these paths is compressed into one edge of the respective weight and, hence, this takes $\mathcal{O}(s + t)$ time.

It remains to show that Z' and $\operatorname{diam}(T')$ can be computed alongside T' without consuming essentially more computing time. Both parameters require the identification of a diametral path in T'. We recall that every diametral path P connects two leaves of the tree T' and, therefore, is either entirely included in one of the trees $T_1, \ldots, T_s$ or contains c and connects leaves that stem from different input trees or vertices $v_1, \ldots, v_t$. To be able to cover the first case, we search a tree T_a with $\operatorname{diam}(T_a) = \max\{\operatorname{diam}(T_i) \mid 1 \leq i \leq s\}$. This works in $\mathcal{O}(s)$ time since the diameters are given. If, at the end, P turns out to be part of T_a then $Z' = Z_a$ and $\operatorname{diam}(T') = \operatorname{diam}(T_a)$, which is computed in $\mathcal{O}(1)$ time.

For the other possibility, we firstly think of the case where P connects two leaves v_i and v_j, $1 \leq i < j \leq t$. Then, obviously, $\operatorname{diam}(T') = k - \pi(k) + 2 \geq \operatorname{diam}(T_a)$ and $Z' = \{c\}$, which can be decided and computed in $\mathcal{O}(1)$ time.

Secondly, one or both end vertices of P may be leaves of the trees $T_1, \ldots, T_s$. To efficiently find P under this condition, we hook additional computations into the iteration of these trees during the evaluation of the μ-operation. More precisely, while a tree T_i is attached to c by a path of length p_i (that is, $p_i = \frac{k+\pi(k)}{2} - d_{T_i}^{\min}$ for $i = m$ and, otherwise, $p_i = \frac{k-\pi(k)}{2} + 1 - d_{T_i}^{\min}$), we memorize the T'-distance

$$d_i = \max\{\operatorname{dist}_{T'}(c, v) \mid v \text{ is leaf in } T_i\} = p_i + \operatorname{rad}(T_i) = p_i + \left\lceil \frac{\operatorname{diam}(T_i)}{2} \right\rceil$$

between c and a farthest leaf of T_i. As $\operatorname{diam}(T_i)$ is known in advance for all $i \in \{1, \ldots, s\}$, this takes just $\mathcal{O}(1)$ additional time per iteration and, thus, $\mathcal{O}(s)$ time in total. We also define

$$d_0 = \begin{cases} \mathrm{dist}_{T'}(c, v_1) = \frac{k-\pi(k)}{2} + 1, & \text{if } t > 0, \\ 0, & \text{otherwise,} \end{cases}$$

the T'-distance between c and any present leaf $v_1, \ldots, v_t$. Observe for later that, if $t > 0$ then

$$\begin{aligned} d_i - d_0 &= p_i + \mathrm{rad}(T_i) - \tfrac{k-\pi(k)}{2} + 1 \\ &\geq \tfrac{k+\pi(k)}{2} - d^{\min}_{T_i} + \mathrm{rad}(T_i) - \tfrac{k-\pi(k)}{2} + 1 \\ &\geq \tfrac{(k+\pi(k))-(k-\pi(k))}{2} + 1 \text{ since } d^{\min}_{T_i} \leq \mathrm{rad}(T_i) \\ &= \pi(k) + 1 \geq 1, \end{aligned}$$

hence, $d_i > d_0$ for all $i \in \{1, \ldots, s\}$.

After the evaluation of μ and the associated completion of T', we determine values i_1 and i_2 such that

$$d_{i_1} = \max\{d_i \mid 0 \leq i \leq s\} \text{ and } d_{i_2} = \max\{d_i \mid 0 \leq i \leq s, i \neq i_1\},$$

which takes $\mathcal{O}(s)$ time. If a diametral path P of T' runs through c and has at least one end vertex in one of the trees $T_1, \ldots, T_s$ then, obviously, the length of this path is $d = d_{i_1} + d_{i_2}$. We can detect this situation in $\mathcal{O}(1)$ time by checking $d > \mathrm{diam}(T_a)$ and $t > 1 \Rightarrow d > k - \pi(k) + 2$. In this case, we infer that $\mathrm{diam}(T') = d$ and Z' is on P. More precisely, if $Z' = \{z'_1, z'_2\}$ (with $z'_1 = z'_2$ if $\pi(T') = 0$) then Z' is situated on the longer subpath of P with z'_1 at distance $\delta_1 = \left\lceil \frac{d_{i_1} - d_{i_2}}{2} \right\rceil$ from c and z'_2 at distance $\delta_2 = \left\lfloor \frac{d_{i_1} - d_{i_2}}{2} \right\rfloor$.

To conclude the proof, we show that Z' is situated on the recently inserted weighted edge $c(p_{i_1})z_{i_1}$, or more precisely, $p_{i_1} \geq \delta_1 \geq \delta_2 \geq 0$. That $\delta_2 \geq 0$ holds since, otherwise, $d_{i_1} < d_{i_2}$. Similarly, we note that, if one of the end vertices of P is a leaf of $v_1, \ldots, v_t$ then $i_2 = 0$ since, otherwise, $d_0 = d_{i_1} < d_{i_2}$. So, i_1 is in $\{1, \ldots, s\}$, and we further observe that

$$2\mathrm{rad}(T_{i_1}) - \pi(T_{i_1}) = \mathrm{diam}(T_{i_1}) \leq \mathrm{diam}(T_a) < d_{i_1} + d_{i_2} = p_{i_1} + \mathrm{rad}(T_{i_1}) + d_{i_2},$$

because T_a has the largest diameter among the given s trees. This means that $\mathrm{rad}(T_{i_1}) \leq p_{i_1} + d_{i_2}$. Using this for the estimation of δ_1, we get

$$\begin{aligned} \delta_1 &= \left\lceil \tfrac{d_{i_1} - d_{i_2}}{2} \right\rceil \leq \tfrac{p_{i_1} + \mathrm{rad}(T_{i_1}) - d_{i_2}}{2} \leq \tfrac{p_{i_1} + (p_{i_1} + d_{i_2}) - d_{i_2}}{2} \\ &= p_{i_1}. \end{aligned}$$

Hence, if $\delta_1 = \delta_2 = 0$ then $z'_1 = z'_2 = c$. If $\delta_1 = p_{i_1}$ then $z'_1 = z_{i_1}$ and, if $\delta_1 > \delta_2$ in this case, then the edge $c(p_{i_1})z_{i_1}$ is split into the weighted edge $c(p_{i_1} - 1)z'_2$ and the edge $z'_2 z'_1$. Otherwise, if also $\pi(T') = 0$, then the center $z'_1 = z'_2$ is inserted by splitting the edge $c(p_{i_1})z_{i_1}$ into the weighted edges $c(\delta_1)z'_1$ and $z'_1(p_{i_1} - \delta_1)z_{i_1}$. In the final case, $c(p_{i_1})z_{i_1}$ is replaced by inserting the center edge $z'_1 z'_2$ between $c(\delta_2)z'_2$ and $z'_1(p_{i_1} - \delta_1)z_{i_1}$. Altogether, this implies that Z' and $\mathrm{diam}(T')$ are always found in $\mathcal{O}(s)$ time. □

Knowing the computational complexities of η and μ, we can design an efficient implementation of the main procedure in Algorithm 1. We decided to implement the recursive definition of $\rho(G, p)$ as an iterative traversal of the cotree $\mathcal{T}$ of G. The primary reason for this is that the recursive partition of G is almost trivial using $\mathcal{T}$. For instance, connected and disconnected graphs G can easily be distinguished with the cotree. The former have a root labelled by ① and the latter by ⓪. Likewise, we can detect small input graphs, stars, with $\mathcal{T}$ by checking if the root is labelled with ① and if the only child that is labelled with ⓪ has just leaf-children. For that reason, the main loop of Algorithm 1 is basically divided into three parts, the first two for connected subgraphs of G and the third one for disconnected input. We like to point out, however, that the third part is executed at most once in the first loop-iteration, if G is disconnected. All recursively generated subgraphs of G are connected and, thus, processed by the first or second part.

Recall that, for connected graphs G of sufficient size, the ρ-operation divides G at the unique universal vertex u, to recurse into the non-trivial connected components $G_1, \ldots, G_s$ of $G - u$, and to conquer by merging the according leaf roots $T_1, \ldots, T_s$ into a parity-optimal solution for G. This divide and conquer procedure is translated into a traversal of the cotree $\mathcal{T}$ as follows. Since input consists of ccgs without true twins, we rely on Proposition 2. That means that nodes with the label ①, like the root X of $\mathcal{T}$, always have exactly one leaf-child, say u, and one child with label ⓪, say Y. The leaf u marks the unique universal vertex in G and Y has children $Z_1, \ldots, Z_s$ with label ① and leaf-children $v_1, \ldots v_t$ that represent the non-trivial connected components $G_1, \ldots, G_s$ and the isolated vertices $v_1, \ldots, v_t$ of $G - u$. The chosen post-order traversal of $\mathcal{T}$ makes sure that, before processing X (and Y), the nodes $Z_1, \ldots, Z_s$ have been visited and finished. Because we use a stack to pass interim results upwards, we always find leaf roots $T_1, \ldots, T_s$ for $G_1, \ldots, G_s$ on the stack (in reverse order), when we need to compute a leaf root T for the subgraph that corresponds to $\mathcal{T}_X$.

We present the details of our construction in Algorithm 1: `OptimalLeafRoot`. The following theorem summarizes our results.

Theorem 10 *Given a chordal cograph G on n vertices and m edges and $p \in \{0, 1\}$, a compressed κ-leaf root of G with minimum integer κ of parity p can be computed in $\mathcal{O}(n + m)$ time.*

Proof: Let G be a ccg and $p \in \{0, 1\}$ a parity. We can assume that G is free of true twins as, otherwise, we can remove x for every pair x, y of true twins in G in linear time and, when (T, k) has been computed, insert every x pendant to the parent of the corresponding twin leaf y into T. The cotree $\mathcal{T}$ of G is computed in Line 1, which works in linear time $\mathcal{O}(n + m)$ as discussed in Section 2.

According to Lemma 6, there is a constant c_1 such that $\eta(T, \delta)$ takes at most $c_1 \cdot n$ computation steps for input trees T with n leaves. In the same way, c_2 is a constant such that $\mu(k, T_1, \ldots, T_s, v_1, \ldots, v_t)$ requires not more than $c_2(s + t)$ steps as promised by Lemma 7.

We assume the existence of a constant c_3 that can be used as an upper bound on the number of steps that the algorithm requires for any of the following operational blocks: Firstly, the base case in Line 7 is, in fact, solved by a loop that adds $t + 1$ leaves to the leaf root T. Here c_3 is supposed to be an upper bound on the costs for one loop iteration. The computation of Line 7 as a whole then requires at most $(t + 1) \cdot c_3$ steps. Secondly, the Lines 2 and 22 contain **for**-loops where c_3 shall be a bound on the number of steps needed to go from one iteration to the next. This includes, for instance, traversing the tree $\mathcal{T}$ to visit the next node X and to push a pointer to (T_i, k_i) onto the stack $\mathcal{S}$. Thirdly, certain lines like Line 10 and Line 13 describe single iterated operations. Beyond the costs of advancing the loop here, c_3 is supposed to additionally bound the computational expenses for one entire iteration. In Line 10, for example, this means the cost of popping from the stack and, in Line 13 the necessary steps to update a maximum are included. Finally, c_3 is also meant to cover the costs for decomposing the cotree like in Line 4, checking certain conditions as in Line 15 and for simple arithmetic computations like for the value of k

Algorithm 1: `OptimalLeafRoot`

Input: A ccg $G = (V, E)$ without true twins and a parity $p \in \{0, 1\}$
Output: A pair (T, k) of a k-leaf root T of G with smallest p-parity integer k.

```
initialize empty stack $\mathcal{S}$ and compute the cotree $\mathcal{T}$ of $G$
foreach node $X$ visited traversing $\mathcal{T}$ in post-order do
    if $X$ is labelled with ① then
        let $Y$ be the ⓪-child and $u$ the leaf-child of $X$
        let $s$ be the number of ①-children and $v_1, \dots, v_t$ the leaf-children of $Y$
        if $s = 0$ then                          // Case i., base case, input is a star
            build $T$ like Case i., Section 3 for star on edges $uv_1, \dots, uv_t$
            push $(T, 4 - p)$ onto $\mathcal{S}$
        else                                     // Case ii., input is a connected graph
            foreach $i \in \{s, s-1, \dots, 1\}$ do pop $(T_i, k_i)$ from $\mathcal{S}$
            if $s = 1$ then $k \leftarrow k_1 + 2(1 - \pi(T_1))$
            else
                $k_a \leftarrow \max\{k_1, \dots, k_s\}$
                $k_b \leftarrow \max\{k_i \mid 1 \le i \le s, i \ne a\}$
                if $p = 1$ then $k \leftarrow k_a + k_b - 1 - 2 \cdot \pi(T_a) \cdot \pi(T_b)$
                else
                    if $s > 2$ and $k_a > \max\{k_i \mid 1 \le i \le s, i \ne a, i \ne b\}$ then
                        $k \leftarrow k_a + k_b - 2 \cdot \pi(T_a) \cdot \pi(T_b)$
                    else $k \leftarrow k_a + k_b - 2 \cdot (\pi(T_a) + \pi(T_b) - \pi(T_a) \cdot \pi(T_b))$
                end
            end
            foreach $i \in \{1, \dots, s\}$ do $T'_i \leftarrow \eta(T_i, (k - k_i)/2)$
            $T \leftarrow \mu(k, T'_1, \dots, T'_s, v_1, \dots, v_t)$
            attach $u$ as a leaf to a center of $T$
            push $(T, k)$ onto $\mathcal{S}$
        end
    end
    if $X$ is a ⓪-node with no parent then        // Case iii., disconnected input
        let $s$ be the number of ①-children and $v_1, \dots, v_t$ the leaf-children of $X$
        foreach $i \in \{s, s-1, \dots, 1\}$ do pop $(T_i, k_i)$ from $\mathcal{S}$
        $k \leftarrow \max\{k_1, \dots, k_s, p + 2\}$
        foreach $i \in \{1, \dots, s\}$ do $T'_i \leftarrow \eta(T_i, (k - k_i)/2)$
        $T \leftarrow \mu(k, T'_1, \dots, T'_s, v_1, \dots, v_t)$
        push $(T, k)$ onto $\mathcal{S}$
    end
end
pop $(T, k)$ from $\mathcal{S}$
return $(T, k)$
```

starting at Line 15.

We begin the proof with connected input graphs. The idea is to use induction on the height h of the cotree $\mathcal{T}$ to show that, then, there is a constant c such that Algorithm 1 requires $c(n+m)$ computation steps to put $(T,k) = \rho(G,p)$ on top of the stack $\mathcal{S}$ before finishing the last visited node of $\mathcal{T}$ by the post-order-loop in Lines 2 to 27. More precisely, we choose $c = c_1 + c_2 + 9c_3$. This would mean that the computation in this case takes $\mathcal{O}(n+m)$ steps.

In any case, Algorithm 1 starts with decomposing the cotree below node X by enumerating the $s+t$ children of X in $(s+t)\cdot c_3$ time. Next, checking if $s = 0$ in c_3 time allows deciding between the base case, which we handle with the induction start, and the general condition, which we prove with the induction step.

Since G is connected and free of true twins, the induction start is at $h = 2$ where, according to Section 2.1, the cotree has a ①-root X with leaf child u and a ⓪-child that has the leaf-children $v_1, \ldots, v_t$, only. Then G is the star with center u and leaves $v_1, \ldots, v_t$, that is, $n = t+1$ and $m = t$. This means, Algorithm 1 iterates just one ①-node (the root X), heads directly into Line 7, and, accordingly, handles this case like Case i. of the ρ-operation. We argue above that this takes $(t+2)\cdot c_3$ time for composing the leaf root and for advancing the loop. Afterwards, it pushes $(T,k) = \rho(G,p)$ on top of the stack $\mathcal{S}$ in c_3 time before the for-loop finishes X, the last visited node of $\mathcal{T}$. Since $s = 0$, $t = n-1$, and $c > 3c_3$, the $(4-p)$-leaf root T is constructed in time

$$(s+t)\cdot c_3 + c_3 + (t+2)\cdot c_3 = c_3(n+m) + 2c_3 \leq 3c_3(n+m) < c(n+m).$$

In the induction step, the height h of $\mathcal{T}$ is at least 3 and G is not a star. After decomposing the cotree and checking the condition $s > 0$, the algorithm enters Line 10 because G is again connected and, thus, the cotree $\mathcal{T}$ has a ①-root X. According to Section 2.1, the root has a leaf-child u and a ⓪-child Y with, in turn, ①-children $Z_1, \ldots, Z_s$, $s \geq 1$. Due to post-order traversal, the algorithm consecutively iterates through the subtrees $\mathcal{T}_{Z_1}, \ldots, \mathcal{T}_{Z_s}$ in that order before entering Y and then X. As the cotrees $\mathcal{T}_{Z_1}, \ldots, \mathcal{T}_{Z_s}$ have height less than h, they fulfill the induction hypothesis. This means for all $i \in \{1, \ldots, s\}$ that the algorithm requires not more than $c(n_i + m_i) + c_3$ steps to output $(T_i, k_i) = \rho(G_i, p)$ on top of the stack $\mathcal{S}$, where G_i is the connected component of $G - u$ that corresponds to $\mathcal{T}_{Z_i}$ with $n_i = |V(G_i)|$ vertices and $m_i = |V(G_i)|$ edges. As the isolated vertices $v_1, \ldots, v_t$ and the universal vertex u with its $n-1$ adjacent edges are not included in $G_1, \ldots, G_s$ and as, thus, $n_1 + \cdots + n_s = n-t-1$ and $m_1 + \cdots + m_s = m-n+1$, it takes $c((n-t-1)+(m-n+1)) + s\cdot c_3 = c\cdot(m-t) + s\cdot c_3$ steps to put the entire sequence $(T_1,k_1), \ldots, (T_s,k_s)$ on top of the stack $\mathcal{S}$. Algorithm 1 pops this list of interim results from the stack in reverse order in Line 10. Although unmentioned for clarity in the listing of Algorithm 1, every tree T_i, $1 \leq i \leq s$ is retrieved together with the explicit information on center, min-max center, diameter, and leaf distance.

After that, it is easy to check that k is computed exactly like in Case ii. in the listing of the ρ-operation in Section 3. With the up to three max-oparations, three conditionals, and four arithmetic computations, this is done in at most $(3s+7)\cdot c_3$ steps.

Since $T_1, \ldots, T_s$ have $n-t-1$ leaves, altogether, Line 22 computes in $c_1(n-t-1) + s\cdot c_3$ time the same extended k-leaf roots $T'_1, \ldots, T'_s$ as ρ. Notice that the subroutine η includes min-max center z'_i, center Z'_i, diameter $\operatorname{diam}(T'_i)$, and leaf distance $d^{\min}_{T'_i}$ for every tree T'_i, $1 \leq i \leq s$. Based on that, the Lines 23 and 24 produce with $c_2(s+t) + c_3$ steps the same merged k-leaf root T as ρ does. The subroutine μ explicitly returns only the center $Z = \{z_1, z_2\}$ and the diameter $\operatorname{diam}(T)$ with T. However, the min-max center z and the leaf distance $d^{\min}_T$ are immediately determined in

Line 24 due to attaching u. If $\pi(T) = 0$ then $z = z_1 = z_2$ and $d_T^{\min} = 1$ and, otherwise, z is the center that is not chosen as the parent of u and $d_T^{\min} = 2$. The costs for these computations are bounded by c_3.

After all that, the result $(T, k) = \rho(G, p)$, including z, Z, $\operatorname{diam}(T)$, and $d_T^{\min}$, is pushed onto $\mathcal{S}$ in Line 25. This finishes the processing of X, the last visited node of $\mathcal{T}$.

Since $n \geq s + t$ and $n \geq 3$ and $c = c_1 + c_2 + 9c_3$, the total computation time is

$$\begin{aligned}
&((s+t)\cdot c_3) + c_3 + (c(m-t) + s\cdot c_3) + ((3s+7)\cdot c_3) + (c_1(n-t-1) + s\cdot c_3) + (c_2(s+t) + c_3) \\
&= c(m-t) + c_1(n-t-1) + c_2(s+t) + c_3(6s+t+9) \\
&< cm + c_1 n + c_2 n + c_3(9n) = (c_1 + c_2 + 9c_3)n + cm = c(n+m).
\end{aligned}$$

This completes the induction.

For a disconnected graph G, we also show that the loop in Lines 2 to 27 puts $(T, k) = \rho(G, p)$ on top of the stack $\mathcal{S}$. The cotree $\mathcal{T}$ of G has a ⓪-root X, which has ①-children $Z_1, \ldots, Z_s$, $s \geq 0$ and leaf-children $v_1, \ldots, v_t$ such that $s + t \geq 2$. Notice that X, as the root of $\mathcal{T}$, has no parent. Then, as argued before, the post-order traversal consecutively iterates through $\mathcal{T}_{Z_1}, \ldots, \mathcal{T}_{Z_s}$ before entering the node X in Line 29. Since the graphs $G_1, \ldots, G_s$ are connected, we have seen above that iterating the children of X takes at most $c(n_1+m_1)+\cdots+c(n_s+m_s)$ time, where $n_i = |V(G_i)|$ and $m_i = |E(G_i)|$. This time, $n - t = n_1 + \cdots + n_s$ and $m - t = m_1 + \cdots + m_s$ and, thus, at most $c(n+m)$ time is required to put the sequence $(T_1, k_1) = \rho(G_1, p), \ldots, (T_s, k_s) = \rho(G_s, p)$ on top of the stack $\mathcal{S}$. Again unmentioned, every tree T_i, $1 \leq i \leq s$ comes with explicitly designated min-max center, center, diameter and leaf distance. Altogether, it, thus, takes linear time to prepare the interim results. Like in Case iii. of the ρ-operation, the value of k is the maximum among $\{k_1, \ldots, k_s, p + 2\}$. So, by Lemmas 6 and 7, $(T, k) = \rho(G, p)$ is correctly computed in Lines 32 and 33. Then Line 34 puts (T, k) on top of $\mathcal{S}$, which finishes X, the last visited node of $\mathcal{T}$. With a similar argument as for the connected case above, we see that this is all done in linear time.

Thus, the algorithm takes $\mathcal{O}(n + m)$ time in each of the above cases to finally reach Line 37, where $(T, k) = \rho(G, p)$ waits accessible at the top of $\mathcal{S}$. Hence, because G is a ccg without true twins, Theorem 5 tells us that Algorithm 1 provides an optimal k-leaf root T of G with $\pi(k) = p$ at this point. This proves Theorem 10. ☐

Observe that Theorem 10 restates our main result from Theorem 1 in a slightly generalized form. By running Algorithm 1: `OptimalLeafRoot` once for every parity $p \in \{0, 1\}$, we clearly find an optimal leaf root of the input graph G in linear time. Hence, the proof of Theorem 10 settles Theorem 1, too.

6 Conclusion

With Theorem 10, we have shown that the OLR problem is linear-time solvable for chordal cographs. Our work also provides a linear-time solution for the k-leaf power recognition problem on chordal cographs. Specifically, for a given ccg G and an integer k, it is sufficient to compute $(T, \kappa) = \rho(G, \pi(k))$ (in linear time with Algorithm 1) to see by $\kappa \leq k$ if G is a k-leaf power.

We conclude the paper by exploring the differences in the construction of odd and even leaf-roots. As we have seen, merging three or more even leaf roots sometimes requires a slightly stronger increase in k than for odd leaf roots. This can accumulate to an arbitrary big gap between k and k' of an optimal odd k-leaf root and an optimal even k'-leaf root of a given ccg. For example, consider the (infinite) series $F_1, F_2, F_3, \ldots, F_i, \ldots$ of ccgs defined as follows. Let F_0 be the path on three vertices and for all integers $i > 0$ define

$$F_i = t_i \textcircled{1} ((x_i \textcircled{1} (F_{i-1} \textcircled{0} u_i)) \textcircled{0} (y_i \textcircled{1} (F_{i-1} \textcircled{0} v_i)) \textcircled{0} (z_i \textcircled{1} (F_{i-1} \textcircled{0} w_i)))$$

with $g \in \{t_i, u_i, v_i, w_i, x_i, y_i, z_i\}$ denoting a graph with the single vertex g. By Section 2, $F_1, F_2, \ldots$ are a family of ccgs and, apparently, all these graphs are connected and without true twins.

Lemma 8 *For all integers $i \geq 1$, the graph F_i is a (odd) k_i-leaf power for $k_i = 2^{i+2} - 1$ but not a $(k_i - 2)$-leaf power and a (even) k'_i-leaf power for $k'_i = k_i + 2^i - 1$ but not a $(k'_i - 2)$-leaf power.*

Proof: We show by induction on i that $(T_i, k_i) = \rho(F_i, 1)$ and $(T'_i, k'_i) = \rho(F_i, 0)$ with $\pi(T_i) = \pi(T'_i) = 0$. This proves the lemma because, by Theorem 5, ρ provides an optimal leaf root of the given parity.

For the induction start at $i = 1$, we leave it to the reader to check that $(T_1, k_1) = \rho(F_1, 1)$ and $(T'_1, k'_1) = \rho(F_1, 1)$ provide a k_1-leaf root T_1 of F_1 with $k_1 = 7$ and $\operatorname{diam}(T_1) = 8$ and a k'_1-leaf root T'_1 with $k'_1 = 8$ and $\operatorname{diam}(T'_1) = 10$.

For the induction step, let $i > 1$ and assume that $(T_{i-1}, k_{i-1}) = \rho(F_{i-1}, 1)$ with $\pi(T_{i-1}) = 0$ and $(T'_{i-1}, k'_{i-1}) = \rho(F_{i-1}, 0)$ with $\pi(T'_{i-1}) = 0$.

We complete the proof by following the constructions of $\rho(F_i, 1)$ and $\rho(F_i, 0)$ as introduced in Section 3. First, note that

$$F_i = t_i \textcircled{1} (X_i \textcircled{0} Y_i \textcircled{0} Z_i)$$

has the universal cut vertex t_i and that $F_i - t_i$ consists of the three mutually isomorphic connected components

$$\begin{aligned} X_i &= x_i \textcircled{1} (F_{i-1} \textcircled{0} u_i), \\ Y_i &= y_i \textcircled{1} (F_{i-1} \textcircled{0} v_i), \\ Z_i &= z_i \textcircled{1} (F_{i-1} \textcircled{0} w_i). \end{aligned}$$

Each of them has a universal cut vertex x_i, y_i or z_i and all of $X_i - x_i$, $Y_i - y_i$, $Z_i - z_i$ consist of the non-trivial connected component F_{i-1} and an isolated vertex u_i, v_i or w_i. This means that the leaf root construction for X_i, Y_i, Z_i works according to the case $s = 1$ and, up to isomorphism, we therefore have

$$\begin{aligned} (T, k) &= \rho(X_i, 1) = \rho(Y_i, 1) = \rho(Z_i, 1) \text{ with} \\ k &= k_{i-1} + 2(1 - \pi(T_{i-1})) = 2^{i+1} + 1 \text{ and} \\ (T', k') &= \rho(X_i, 0) = \rho(Y_i, 0) = \rho(Z_i, 0) \text{ with} \\ k' &= k'_{i-1} + 2(1 - \pi(T'_{i-1})) = 2^{i+1} + 2^{i-1}. \end{aligned}$$

Like in the proof of Proposition 8, we get that a diametral path has length ℓ_1 in both constructions and, thus,

$$\begin{aligned} \operatorname{diam}(T) &= 2k - 3 + \pi(T_{i-1}) = 2^{i+2} - 1 \text{ and} \\ \operatorname{diam}(T') &= 2k' - 3 + \pi(T'_{i-1}) = 2^{i+2} + 2^i - 3. \end{aligned}$$

Hence, $\pi(T) = \pi(T') = 1$.

Now get back to the leaf root construction for F_i. If $p = 1$ then our procedure handles the only case with $s \geq 2$. We get

$$\begin{aligned}(T_i, k_i) &= \rho(F_i, 1) \text{ with} \\ k_i &= 2k - 1 - 2\pi(T)^2 = 2^{i+2} - 1\end{aligned}$$

as claimed and, like in the proof of Proposition 8, a diametral path is of length $\ell_{1,2}$, thus,

$$\begin{aligned}\text{diam}(T_i) &= k_i + 2k - 2\pi(T) - 3 \\ &= (2^{i+2} - 1) + 2(2^{i+1} + 1) - 5 \\ &= 2^{i+3} - 4,\end{aligned}$$

which means $\pi(T_i) = 0$. If $p = 0$ then the procedure works the case with $s > 2$ and where $k - \pi(T)$ equally stands for all three (isomorphic) k-leaf roots T of the subgraphs X_i, Y_i, Z_i. This means that

$$\begin{aligned}(T'_i, k'_i) &= \rho(F_i, 0) \text{ with} \\ k'_i &= 2k' - 2\pi(T')^2 = (2^{i+2} + 2^i) - 2 \\ &= (2^{i+2} - 1) + 2^i - 1 = k_i + 2^i - 1,\end{aligned}$$

as claimed. Moreover, like in the proof of Proposition 8, a diametral path in T'_i has length $\ell_{2,3}$. Here, we get

$$\begin{aligned}\text{diam}(T'_i) &= k'_i - \pi(k'_i) + 2k' - 2\pi(T') - 2 \\ &= (2^{i+2} + 2^i - 2) + 2(2^{i+1} + 2^{i-1}) - 4 \\ &= 2^{i+3} + 2^{i+1} - 6,\end{aligned}$$

thus, $\pi(T_i) = 0$. This completes the proof. □

This means that, although odd and even leaf root construction follows the same approach, there are k-leaf powers of odd k among the chordal cographs that have optimal even k'-leaf roots with k' roughly $1.25k$.

Acknowledgments

We are grateful to the three anonymous reviewers for their careful reading and their valuable comments and suggestions that significantly improved the quality of the paper.

References

[1] Andreas Brandstädt and Christian Hundt. Ptolemaic graphs and interval graphs are leaf powers. In *LATIN 2008: Theoretical Informatics, 8th Latin American Symposium*, LNCS, pages 479–491, 2008. doi:10.1007/978-3-540-78773-0_42.

[2] Andreas Brandstädt, Christian Hundt, Federico Mancini, and Peter Wagner. Rooted directed path graphs are leaf powers. *Discrete Mathematics*, 310(4):897–910, 2010. doi:10.1016/j.disc.2009.10.006.

[3] Andreas Brandstädt and Van Bang Le. Structure and linear time recognition of 3-leaf powers. *Information Processing Letters*, 98(4):133–138, 2006. doi:10.1016/j.ipl.2006.01.004.

[4] Andreas Brandstädt, Van Bang Le, and R. Sritharan. Structure and linear-time recognition of 4-leaf powers. *ACM Transactions on Algorithms*, 5(1):11:1–11:22, 2008. doi:10.1145/1435375.1435386.

[5] Maw-Shang Chang and Ming-Tat Ko. The 3-Steiner root problem. In *Graph-Theoretic Concepts in Computer Science, 33rd International Workshop, WG 2007*, LNCS, pages 109–120, 2007. doi:10.1007/978-3-540-74839-7_11.

[6] Derek G. Corneil, H. Lerchs, and L. Stewart Burlingham. Complement reducible graphs. *Discrete Applied Mathematics*, 3(3):163–174, 1981. doi:10.1016/0166-218X(81)90013-5.

[7] Guillaume Ducoffe. The 4-Steiner root problem. In *Graph-Theoretic Concepts in Computer Science - 45th International Workshop, WG 2019*, LNCS, pages 14–26, 2019. doi:10.1007/978-3-030-30786-8_2.

[8] David Eppstein and Elham Havvaei. Parameterized leaf power recognition via embedding into graph products. *Algorithmica*, 82(8):2337–2359, 2020. doi:10.1007/s00453-020-00720-8.

[9] Martin Charles Golumbic. Trivially perfect graphs. *Discrete Mathematics*, 24(1):105–107, 1978. doi:10.1016/0012-365X(78)90178-4.

[10] Manuel Lafond. Recognizing k-leaf powers in polynomial time, for constant k. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022*, pages 1384–1410. SIAM, 2022. doi:10.1137/1.9781611977073.58.

[11] Van Bang Le and Christian Rosenke. Computing optimal leaf roots of chordal cographs in linear time. In *Fundamentals of Computation Theory - 24th International Symposium, FCT 2023*, volume 14292 of *LNCS*, pages 348–362. Springer, 2023. doi:10.1007/978-3-031-43587-4_25.

[12] Ross M. McConnell. Linear-time recognition of circular-arc graphs. *Algorithmica*, 37(2):93–147, 2003. doi:10.1007/s00453-003-1032-7.

[13] Naomi Nishimura, Prabhakar Ragde, and Dimitrios M. Thilikos. On graph powers for leaf-labeled trees. *Journal of Algorithms*, 42(1):69–108, 2002. doi:10.1006/jagm.2001.1195.

[14] Christian Rosenke, Van Bang Le, and Andreas Brandstädt. *Leaf powers*, page 168–188. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2021. doi:10.1017/9781108592376.011.

[15] Peter Wagner and Andreas Brandstädt. The complete inclusion structure of leaf power classes. *Theor. Comput. Sci.*, 410(52):5505–5514, 2009. doi:10.1016/j.tcs.2009.06.031.

[16] E.S Wolk. The comparability graph of a tree. *Proceedings of the American Mathematical Society*, 13:789–795, 1962. doi:10.1090/S0002-9939-1962-0172273-0.

[17] E.S Wolk. A note on "the comparability graph of a tree". *Proceedings of the American Mathematical Society*, 16:17–20, 1965. doi:10.1090/S0002-9939-1962-0172273-0.