THE LIMITED VISIBILITY LOCALIZATION GAME

ANTHONY BONATO, TRENT G. MARBACH, JOHN MARCOUX, AND JD NIR

Abstract. We consider a variant of the Localization game where the cops have limited visibility. For a nonnegative integer k, we study the k-visibility Localization game and the corresponding k-visibility localization number ζ_k. This game is connected to the k-truncated metric dimension and isoperimetric boundary conditions on the host graph. We give bounds on k-visibility localization numbers related to maximum degree, isoperimetric inequalities, and other graph parameters. For all k, we give a family of trees with unbounded ζ_k values. Extending results known for the localization number, we show that for $k \geq 2$, every tree contains a subdivision with $\zeta_k = 1$.

1. Introduction

Pursuit-evasion games are combinatorial models for detecting or neutralizing an adversary's activity on a graph. Examples include Cops and Robbers and the Localization game. In such models, pursuers (often called *cops*) attempt to capture an evader (called the *robber*) loose on the vertices of a graph. The rules governing how players move and how the adversary is captured depend on the variant under study. Such games are motivated by foundational topics in computer science, discrete mathematics, and artificial intelligence, such as robotics and network security. For surveys of pursuit-evasion games, see the books [4, 10]; see Chapter 5 of [4] for more on the Localization game.

In the Localization game, the robber is invisible, and the cops use distance probes to determine their exact location. The Localization game was first introduced for one cop in [15, 27]. The game in its present form was first considered in the paper [15] and subsequently studied in several papers, including [2, 5, 6, 7, 12, 13, 14, 28].

Limiting the robber's visibility has been studied in the context of the Cops and Robbers game. For a nonnegative integer k, in k-visibility Cops and Robbers, the robber is visible to the cops only when a cop is distance at most k. The case when $k = 0$ has been studied [18, 19, 30], as has the case when $k = 1$ [32, 33, 34], and a recent paper studied the cases $k \geq 1$ [16]. A variant of the Localization game was introduced in [9], in which the cops have visibility of 1. The goal of the present paper is to extend this to larger ranges of visibility. Such variants have surprising connections to various width parameters and isoperimetry in graphs; see [11] for connections to isoperimetry in complete trees.

Let k be a nonnegative integer. The *k-visibility Localization game* is played over a sequence of discrete time-steps; a *round* of the game is a move by the cops and the subsequent move by the robber. The robber occupies a vertex of the graph, and in each round may move to a neighboring vertex or remain on their current vertex. A move for the cops is a placement of cops on a set of vertices (note that the cops are not limited to moving to neighboring vertices). The players move on alternating time-steps, with the robber going first. In each round, the cops $C_1, C_2, \ldots, C_m$ occupy a set of vertices $u_1, u_2, \ldots, u_m$ and each cop sends out a *cop probe* d_i, where $1 \leq i \leq m$. If a cop C_i is at distance j from the vertex of the robber, where $0 \leq j \leq k$, then $d_i = j$. In all other cases, the cop probe returns no information, and we set $d_i = *$. Hence, in each round, the cops determine

---

2020 *Mathematics Subject Classification.* 05C57,05C12.

Key words and phrases. localization number, limited visibility, pursuit-evasion games, isoperimetric inequalities, graphs.

a *distance vector* $D = (d_1, d_2, \dots, d_m)$ of cop probes. Relative to the cops' position, there may be more than one vertex x with the same distance vector that the robber may occupy. We refer to such a vertex x as a *candidate of* D or simply a *candidate*.

The cops win if they have a strategy that, after a finite number of rounds, determines a unique candidate, at which time we say that the cops *capture* the robber. We assume the robber is *omniscient*, in the sense that they know the entire strategy for the cops. If the robber evades capture, then the robber wins. For a graph G, define the *k-visibility localization number* of G, written $\zeta_k(G)$, to be the least positive integer m for which m cops have a winning strategy in the k-visibility Localization game.

For a graph G of order n and diameter $d = \mathrm{diam}(G)$, a probe of $*$ by a $(d-1)$-visibility cop can only represent a distance of d, and so $\zeta(G) = \zeta_{d-1}(G)$. It is also straightforward to see that $\zeta_0(G) \le n - 1$. It then follows that

$$\zeta(G) = \zeta_{d-1}(G) \le \zeta_{d-2}(G) \le \cdots \le \zeta_0(G) \le n - 1,$$

as a winning cop strategy in the k-visibility Localization game will also be winning in the $(k+1)$-visibility Localization game.

The Localization game is connected to the *metric dimension* of a graph, which is the minimum size of a *resolving set*, a set of vertices $v_1, v_2, \dots, v_m$ such that the distance vector is unique to each vertex of G. Hence, the metric dimension is the number of cop probes needed to locate the robber in one round of the Localization game. The metric dimension of graphs was introduced in [22, 29] independently. The k-visibility Localization game is similarly related to the k-truncated metric dimension of a graph, introduced in [20, 21], in which a resolving set uses a metric where distances larger than k are indistinguishable. The k-truncated metric dimension has been studied in, for example, [1, 17, 24].

We note that Bernshteyn and Lee [3] introduced the zero-visibility search game, which is similar to the zero-visibility Localization game, except that the robber is captured only if it shares a vertex with a cop. As an example in which the parameters differ, two cops are required to win the zero-visibility search game on the graph P_2, whereas the zero-visibility Localization game requires only one cop.

The paper is organized as follows. In the next section, we study a variant of the k-visibility localization number, called the k-proximity number, which will be used to derive bounds and to connect it to an isoperimetric function. Section 3 focuses on trees, and for each k we give a family of trees with unbounded ζ_k values. We also consider a threshold problem, determining how fast k must grow so that ζ_k remains bounded on a growing sequence of trees. Section 4 shows that every tree contains a subdivision with $\zeta_k = 1$ when $k \ge 2$. This extends an analogous result about 1-visibility localization numbers of trees [15] to k-visibility localization numbers. We finish with a discussion of future directions.

In a forthcoming paper, we analyze the k-visibility localization number of Cartesian grids. A result of that work not captured in this investigation is a separation of the ζ_k parameters: for distinct i and j, there exists a graph $G = G(i,j)$ such that $\zeta_i(G) \ne \zeta_j(G)$.

All graphs we consider are finite, simple, and undirected. The variable k stands for a nonnegative integer. Unless otherwise stated, we only consider connected graphs. The set of vertices that share an edge with x is denoted $N(x)$, and we refer to vertices in $N(x)$ as *neighbors* of x. We define $N[x] = N(x) \cup \{x\}$. For a set S of vertices, $N[S] = \bigcup_{u \in S} N[u]$. For a graph G, let $\Delta(G)$ be the maximum degree of a vertex in G, and for $k \ge 0$ define $\Delta_k(G)$ to be the maximum number of vertices within distance k of any vertex $u \in V(G)$. Note that $\Delta_0(G) = 1$ for all G. For further background on graph theory, see [31].

2. PROXIMITY AND ISOPERIMETRIC BOUNDS

We begin by introducing the *k-proximity game* as a tool for analyzing ζ_k. The k-proximity game has the same rules as the k-visibility Localization game; however, in addition to winning by locating the robber, the cops may also win by having any cop be within distance k of the robber. We denote the minimum number of cops required to win the k-proximity game on a graph G as $\operatorname{prox}_k(G)$. This is a generalization of the *blind Localization game* first considered in [12] corresponding to the case that $k = 1$, which was later reintroduced as the one-proximity game in [9]. The k-proximity game is also analogous to the *k-visibility seeing cop-number* introduced in [16]. Note that similar to the localization number, we have $\operatorname{prox}_{k-1}(G) \geq \operatorname{prox}_k(G)$ for $k \geq 1$.

We first prove Theorem 1, which shows that while the k-proximity number and k-visibility localization numbers are not necessarily the same, there is an upper and lower bound on ζ_k in terms of prox_k. Recall that $\Delta_k(G)$ is defined as the maximum number of vertices within distance k of any vertex $u \in V(G)$.

Theorem 1. *For a graph G and $k \geq 0$,*

$$\operatorname{prox}_k(G) \leq \zeta_k(G) \leq \Delta_k(G) \operatorname{prox}_k(G).$$

Proof. The lower bound holds because, for any strategy used by the cops to identify the robber in the k-visibility Localization game, the same number of cops playing the k-proximity game in the same way can identify the robber's position.

For the upper bound, we simulate a k-proximity cop by using at most $\Delta_k(G)$ k-visibility localization cops to probe each vertex visible by the k-proximity cop. By following a winning strategy in the k-proximity game, some k-proximity cop must come within distance k of the robber. In the corresponding cop move for the k-visibility game, the cop probes the robber's exact location. □

Noting that $\Delta_0(G) = 1$ for all G, we find that $\zeta_0(G) = \operatorname{prox}_0(G)$ for any graph, which also follows from the definitions of the games: being within distance 0 of the robber is identical to locating the robber, so the two conditions for the k-proximity cops to win are identical to the condition for the k-visibility Localization cops to win.

When G is a tree, the k-proximity and k-visibility localization numbers are even more closely linked.

Lemma 2. *For a tree T, we have that*

$$\operatorname{prox}_k(T) \leq \zeta_k(T) \leq \operatorname{prox}_k(T) + 1.$$

Before we prove Lemma 2, we must consider how the k-visibility Localization and k-proximity games behave on subtrees. A graph property f is *monotone* with respect to subgraphs if $f(G') \leq f(G)$ for each subgraph G' of G. In general, as in the Localization game, the k-visibility localization number of a graph G is not necessarily monotone with respect to subgraphs, even if only induced subgraphs are considered; see Exercises 5.6 and 5.7 in [4] for an example in the Localization game. However, when G is a tree, we observe that the properties in question are monotone with respect to subgraphs.

Lemma 3. *For a tree T with subtree T', $\zeta_k(T') \leq \zeta_k(T)$ and $\operatorname{prox}_k(T') \leq \operatorname{prox}_k(T)$.*

Proof. Let $f : V(T) \to V(T')$ map $f(v) = v$ for each $v \in V(T')$ and, for $u \in V(T) \setminus V(T')$, let f map u to the unique vertex in $V(T')$ minimizing the distance (in T) from u. To play either game on T', the cops follow their winning strategy on T. Whenever their strategy calls for probing a vertex $u \in V(T) \setminus V(T')$, they instead probe $f(u)$ and add the distance (in T) from u to $f(u)$ to the distance they receive. If this distance is greater than k, then the cops treat the probe as though it returned $*$.

Since for any v in $V(T')$ and u in $V(T) \setminus V(T')$ the unique shortest path from u to v must pass through $f(u)$, this strategy yields the exact same information as the strategy on T, but with the additional information that the robber cannot be on any of the vertices in $V(T) \setminus V(T')$. Thus, the candidate sets formed in this strategy will be subsets of the candidate sets formed in the strategy on T, so as the strategy succeeds on T, it will also succeed on T'. $\square$

We can now prove Lemma 2.

Proof of Lemma 2. Theorem 1 provides the lower bound. Note that when $k = 0$, $\zeta_k(T) = \mathrm{prox}_k(T)$, and so the statement clearly holds. We therefore assume for the rest of the proof that $k \geq 1$ and describe a strategy to use $\mathrm{prox}_k(T) + 1$ cops to win the k-visibility Localization game.

Choose a vertex $r \in T$ and root T at r. We use one cop, say C_r, to probe r each round. Let $T_1, T_2, \ldots, T_m$ be the connected components of $T - r$. To avoid immediate capture at r, the robber must be on T_i for some i. Notice that the robber is unable to leave T_i without going through r, at which point C_r would probe a distance of 0 and capture the robber, so the robber must remain on T_i.

By Lemma 3, $\mathrm{prox}_k(T_i) \leq \mathrm{prox}_k(T)$ for each i. Our goal is to determine which T_i contains the robber and move C_r to the root of that T_i. As the height of T_i is strictly smaller than T, this strategy inductively pushes the robber to a leaf vertex where they will be captured. The cops other than C_r follow a winning k-proximity game strategy on T_i. As this strategy is winning and T_i is finite, there is some maximum number of rounds required under this strategy to capture the robber. If these cops follow this strategy for that many rounds without ever probing a non-$*$ distance, then the cops know the robber is not located on T_i and proceed to T_{i+1}. However, if the cops do probe a non-$*$ distance, they must be able to determine if the robber is located in T_i or is merely within distance k of some vertex in T_i.

To demonstrate how they accomplish this, suppose that the cops probe a vertex v in T_i and find the robber is within distance k of v, say distance d_1. Let d_2 be the distance that the cop C_r probed to the robber, with either $d_2 \leq k$ or $d_2 = *$, and let d_3 be the distance between v and r. If $d_2 = 0$, then the robber is captured by the cop C_r, so we may assume that $1 \leq d_2 \leq k$ or $d_2 = *$. If the robber is located outside of T_i, then r is on the unique path between v and the robber, requiring $d_2 < d_1$, so if $d_2 \geq d_1$, the cops know the robber is located in T_i. If $d_2 < d_1$, we have that $d_2 \neq *$. In this case, we claim that the robber is located in T_i if and only if $d_2 + d_3 - d_1 > 0$. Consider the walk in T from the robber's location to r and then to v. By the triangle inequality, this walk, which has length $d_2 + d_3$, is at least as long as d_1. As T is a tree, equality holds if and only if the path from the robber to v goes through r, which occurs if and only if the robber is located outside of T_i. We conclude if $d_2 = *$, $d_2 \geq d_1$, or $d_2 < d_1$ and $d_2 + d_3 - d_1 > 0$ then the robber is located in T_i, and if $d_2 < d_1$ and $d_2 + d_3 - d_1 = 0$, then the robber is not located within T_i. In either case, by executing a winning k-proximity strategy on T_i, the cops can determine whether the robber is located in T_i or that they should move on to T_{i+1}.

Suppose the cops determine that the robber is on T_i. If $d_2 = 1$, then the robber is captured, either because $d_1 = 0$ or they are at the unique vertex at distance $d_3 - 1$ from v and distance 1 from r. If $d_2 \geq 2$, then after the robber's next move, then the robber will still be within T_i. In this case, C_r can safely switch to probing the root of T_i, trapping the robber in a tree of strictly smaller height. By repeating this process, the cops inductively reduce the height of the tree until they capture the robber at a leaf vertex. $\square$

For a set of vertices S, let $N(S)$ denote the set of vertices in $V(G) \setminus S$ with a neighbor in S. We will refer to this as the *border* of S. The *vertex-isoperimetric function* of a graph G at volume k is defined as

$$\Phi_V(G, k) = \min_{S \subseteq V(G): |S| = k} |N(S)|,$$

which was defined on graphs by [23]. In the literature, the term *isoperimetric inequality* is commonly defined as a lower bound on $|N(S)|$, typically for a general set S. Let $\Phi_V(G) = \max_k \Phi_V(G,k)$; this value is referred to in the literature as the *isoperimetric peak*.

Theorem 4. *For an n-vertex graph G with $G \neq K_n$,*

$$\zeta_k(G) \geq \mathrm{prox}_k(G) > \frac{\Phi_V(G)}{\Delta_k(G)}.$$

Proof. Suppose that we play the k-proximity game with $\frac{\Phi_V(G)}{\Delta_k(G)}$ or fewer cops, and assume for contradiction that the cop player can win. During the present round, denote the set of vertices that the robber could occupy before the cops have probed any vertices as S_1, after the cops have probed their vertices as S_2, and after the robber has moved as S_3 (which is equal to S_1 in the subsequent round). Since each cop can detect the robber on at most $\Delta_k(G)$ vertices on each round, cumulatively there are at most $\lfloor \frac{\Phi_V(G)}{\Delta_k(G)} \rfloor \cdot \Delta_k(G) \leq \Phi_V(G)$ vertices of S_1 in which the robber would be caught by the cop if the robber occupied that vertex. Therefore, $|S_2| \geq |S_1| - \Phi_V(G)$.

Let $x \geq 1$ be the maximum value such that $\Phi_V(G,x) = \Phi_V(G)$. We claim that our condition $G \neq K_n$ implies $x > 1$. By definition $\Phi_V(G,1) = \min_v N(\{v\}) = \delta(G)$, and so if $x = 1$ we have that $\delta(G) = \Phi_V(G,1) = \Phi_V(G)$. Consider a set $S \subseteq V(G)$ of cardinality $|S| = n - \delta(G)$. The complement of S, written S^c, contains $\delta(G)$ vertices, each with degree at least $\delta(G)$ and at most $\delta(G) - 1$ neighbors in S^c. It follows that each vertex of S^c has a neighbor in S, and so $N(S) = S^c$. Since this observation holds for all sets S, it follows that $\Phi(G, n - \delta(G)) = \delta(G)$, and since $x = 1$ is the maximum value with $\Phi(G,x) = \delta(G)$, it must be that $n - \delta(G) \leq 1$. Equivalently, $\delta(G) \geq n - 1$, which only occurs if $G = K_n$.

Therefore, we may assume that the maximum x such that $\Phi_V(G,x) = \Phi_V(G)$ satisfies $x > 1$. We claim, by induction, that on each round $|S_1| \geq x + \Phi_V(G)$. For the base case, we show that $|V(G)| \geq x + \Phi_V(G)$, since $|S_1| = |V(G)|$ in the first round. Adding i vertices B to a set A may decrease the border of A by at most i vertices, namely, $N(A \cup B) \supseteq N(A) \setminus B$. It follows that $\Phi_V(G, x+i) \geq \Phi_V(G) - i$. In particular, when $i = \Phi_V(G) - 1$, this yields $\Phi_V(G, x + \Phi_V(G) - 1) \geq 1$. Since the border of the set of all vertices, $N(V(G))$, is empty, we know $\Phi_V(G,z) = 0$ for $z \geq |V(G)|$. Combining this with the previous statement, we must have that $x + \Phi_V(G) - 1 < |V(G)|$, from which it follows that $x + \Phi_V(G) \leq |V(G)|$.

This means that at the start of the game, we have $|S_1| = |V(G)| \geq x + \Phi_V(G)$ vertices that the robber could occupy. We assume that the cop player can win the k-visibility Localization game on G, and thus the number of vertices the robber can safely occupy eventually tends to zero. The set of vertices the robber may occupy only decreases after the cop player's move, so $|S_1| \geq x + \Phi_V(G)$ unless we had $|S_2| \leq x + \Phi_V(G)$ in the previous round. Thus there must be some round where we have $|S_1| \geq x + \Phi_V(G)$ and $|S_2| \leq x + \Phi_V(G)$. We suppose the present round is such a round.

Since we know $|S_2| \geq |S_1| - \Phi_V(G)$ and we have assumed that $|S_1| \geq x + \Phi_V(G)$, we have that $|S_2| \geq (x + \Phi_V(G)) - \Phi_V(G) = x > 1$, so at least two vertices are candidates and the robber is not captured this round. Suppose that $|S_2| = x + p$, for some $p \geq 0$. We also have that $|N(S_2)| \geq \Phi_V(G, x+p) \geq \Phi_V(G) - p$, and so

$$|S_3| = |S_2| + |N(S_2)| \geq (x+p) + (\Phi_V(G) - p) = x + \Phi_V(G).$$

Therefore, if we start with $x + \Phi_V(G)$ or more vertices where the robber can safely hide before the current cop move, then there will be $x + \Phi_V(G)$ or more vertices for the robber before the next move, and the robber will not be caught. Since we start with $|V(G)| \geq x + \Phi_V(G)$ vertices where the robber could be, the cops will never be able to reduce the size of the unknown vertices below $x > 1$, and so the robber will never be caught. This completes the contradiction. $\square$

A path-decomposition of a graph G is a sequence of subsets of $V(G)$, say X_i, such that for each $e \in E(G)$ there is an i such that $e \in X_i$ and such that $X_i \cap X_k \subseteq X_j$ whenever $i < j < k$. The pathwidth of a graph G, $\mathrm{pw}(G)$, is the minimum of $\max_i |X_i|$ over all path-decompositions.

Lemma 5 ([9])**.** *For a graph G and positive integer k,*

$$\mathrm{prox}_k(G) \leq \zeta_k(G) \leq \mathrm{pw}(G),$$

and for $k = 0$ we have

$$\mathrm{prox}_0(G) = \zeta_0(G) \leq \mathrm{pw}(G) + 1.$$

We have the following corollary to Theorem 4 and Lemma 5 for the 0-visibility localization number.

Corollary 6. *For an n-vertex graph G with $G \neq K_n$,*

$$\Phi_V(G) + 1 \leq \mathrm{prox}_0(G) = \zeta_0(G) \leq \mathrm{pw}(G) + 1.$$

In the complete graph case, we have that $\Phi_V(K_n) = \mathrm{prox}_0(K_n) = \zeta_0(K_n) = \mathrm{pw}(K_n) = n - 1$.

We note that $\mathrm{pw}(G)$ is equal to the vertex-separation number [25], which is an isoperimetric-type property of graphs. Since $\mathrm{prox}_0(G) = \zeta_0(G)$ is bounded above and below by isoperimetric-type properties, we can think of this value as intrinsically measuring some kind of isoperimetric-type property of the graph G.

3. TREES

The original Localization game, in which the cops have unrestricted vision, is completely understood on trees. As was proved first in [27] and later in [13], the localization number of trees is at most two, and trees with localization number 1 are fully characterized. In contrast, it was shown in [9] that the 1-visibility localization number of trees is unbounded. This also holds for the k-visibility localization number.

Let $T(h, q)$ denote the complete q-ary tree of height h; that is, the rooted tree where each vertex of distance less than h from the root has q children. The isoperimetric peak of $T(h, q)$ has been studied, and will allow us to understand the k-visibility Localization game on trees.

Theorem 7 ([11])**.** *For integers $h \geq 1$ and $q \geq 2$,*

$$\Phi_V(T(h,q)) \geq \begin{cases} h & \text{for } q \geq 5, \\ h + 1 - \log_q(2h) & \text{for } q \in \{3, 4\}, \\ \frac{h-3}{2} - \log_2(\sqrt{3h}) & \text{for } q = 2. \end{cases}$$

When we fix q and k to be constants and take $h \geq k$, it follows that $\Delta_k(T(h,q)) = \frac{q(q-1)^k - 2}{q-2}$ is also constant. Applying Theorem 4 and Theorem 7 in this case yields a lower bound of $\zeta_k(T(h,q))$ that tends to infinity with h, which we capture in the following result.

Theorem 8. *If d and k are nonnegative integers, then there exists a tree T such that $\zeta_k(T) > d$.*

Note that when q and k are fixed constants, this lower bound is asymptotic to the radius of the tree, $\mathrm{rad}(T)$. We show in Theorem 11 that there is an upper bound on ζ_k that is asymptotic to the radius of the tree for k constant. As such, the bounds are asymptotically tight for such complete q-ary trees. These bounds are not asymptotically tight when k changes with the radius of the tree, however. Noting that $\zeta_k(G)$ weakly decreases when only k increases, in Theorem 10 we find the threshold on k in terms of the radius required for the k-visibility localization number to diverge on trees, and show that a class of graphs exists that diverge beyond this threshold.

We begin with an example problem that illustrates the case when k is large.

Theorem 9. *For a tree T, if $k = \mathrm{rad}(T)$, then $\zeta_k(T) \leq 2$.*

Proof. We give a strategy in which two cops with visibility $\mathrm{rad}(T)$ capture the robber. Choose a vertex r_1 that has distance at most $\mathrm{rad}(T)$ to each vertex, and root T at r_1. The first cop probes r_1, which will always return a distance $d_1 \leq \mathrm{rad}(T)$. The robber is restricted to a subtree of T rooted at some neighbor of r_1: if they attempt to move from one neighbor of r_1 to another, they must pass through r_1, at which point $d_1 = 0$ and the first cop locates the robber. The second cop sequentially probes the neighbors of r_1 until it encounters a vertex r_2 with distance $d_2 < d_1$. The cops now know that the robber is located within the subtree rooted at r_2 and can repeat the process on this subtree. In this way, the cops confine the robber to subtrees of decreasing height until the robber is captured on a leaf vertex. □

Consider a sequence of trees $(T_n)_{n\geq 1}$ of increasing radius. Theorem 8 tells us for fixed k, it may be that $\lim_{n\to\infty} \zeta_k(T_n) \to \infty$. However, if we allow k to grow as $k = k(n) = \mathrm{rad}(T_n)$, then by Theorem 9 we have that $\lim_{n\to\infty} \zeta_k(T_n) \leq 2$. In the following theorem, the main result of this section, we determine how fast k must grow as a function of $\mathrm{rad}(T)$ to ensure $\zeta_k(T_n)$ remains bounded.

Theorem 10. *For every sequence of trees $(T_n)_{n\geq 1}$ with $\mathrm{rad}(T_n) \to \infty$ and every positive integer valued function f such that $f(n) = \Omega(\sqrt{n})$, there exists a constant d such that if $k = k(n) = f(\mathrm{rad}(T_n))$, then*

$$\limsup_{n\to\infty} \zeta_k(T_n) \leq d.$$

For every positive integer valued function f such that $f(n) = o(\sqrt{n})$ there exists a sequence of trees $(T_n)_{n\geq 1}$ such that, with $k = k(n) = f(\mathrm{rad}(T_n))$,

$$\lim_{n\to\infty} \zeta_k(T_n) = \infty.$$

The upper bound of Theorem 10 follows from the following general upper bound for $\zeta_k(T)$ in terms of k and $\mathrm{rad}(T)$.

Theorem 11. *For any tree T and any integer $k \geq 1$,*

$$\zeta_k(T) \leq \left\lceil \frac{\mathrm{rad}(T) + k}{k^2} \right\rceil + 1 \leq \frac{\mathrm{rad}(T)}{k^2} + 3.$$

Proof. We give a strategy in which $\left\lceil \frac{\mathrm{rad}(T)+k}{k^2} \right\rceil + 1$ cops capture the robber in the k-visibility Localization game. Choose a vertex r of T with distance at most $\mathrm{rad}(T)$ from any other vertex, and root T at r. Our strategy broadly works as follows. The cop player's first goal is "detecting" the robber by probing a vertex that returns a non-$*$ distance. To detect the robber, we first describe how to use the probes on a single path from r to a leaf of T. Next, we describe how to move from one leaf to the next, "sweeping" across T. We apply this detection scheme not to T itself, but to the subtrees rooted at the neighbors of r. As in other proofs, once we determine the subtree containing the robber, we apply the strategy recursively, forcing the robber to a leaf where they are captured.

First, consider the following strategy to protect a path from r to a leaf of T of length ℓ using $\left\lceil \frac{\ell+k}{k^2} \right\rceil$ cops. We claim that, by implementing this strategy, the robber cannot, after a finite number of rounds, enter the path without being detected by some cop probe within distance k. Consider the set of vertices $S_k = \{s_i\}_{i=0}^{\lfloor \ell/k \rfloor}$ on the path such that s_i is at distance ik from r. If ℓ/k is not an integer, then add the leaf to S_k as $s_{\lceil \ell/k \rceil}$. We claim that if each of these vertices is probed at least once every k rounds, then after each vertex has been probed once, the cops can detect the robber entering the path. Suppose that the cops begin by probing the vertices along the path at $s_0, s_k, s_{2k}, \ldots, s_{\left\lceil \frac{\ell-k^2+k}{k^2} \right\rceil k}$. Note that this placement uses

$$1+\left\lceil \frac{\ell-k^2+k}{k^2} \right\rceil = 1+\left\lceil \frac{\ell+k}{k^2}-1 \right\rceil = \left\lceil \frac{\ell+k}{k^2} \right\rceil$$

probes. On subsequent rounds, if a cop probed s_i on the previous round, then they probe s_{i+1} on this round. After a cop probes the leaf, they return to the root and probe r.

If ℓ/k is an integer, then S_k contains $1+\ell/k = 1+\lceil \ell/k \rceil$ vertices, and otherwise it contains $2+\lfloor \ell/k \rfloor = 1+\lceil \ell/k \rceil$ vertices. Over k rounds, the cops probe a total of

$$k \cdot \left\lceil \frac{\ell+k}{k^2} \right\rceil = k \cdot \left\lceil \frac{\ell/k+1}{k} \right\rceil \geq \lceil \ell/k+1 \rceil = 1+\lceil \ell/k \rceil$$

vertices, where we use the inequality $x \cdot \left\lceil \frac{y}{x} \right\rceil \geq \lceil y \rceil$. Thus, this strategy guarantees that every vertex in S_k is probed at least once every k moves.

Now suppose the cops have implemented this strategy for at least k rounds so that every vertex in S_k has been probed at least once. Assume for contradiction that the robber enters the path at some vertex v between the vertices s_i and s_{i+1} and then leaves the path, all without any cop probing a vertex within distance k of the robber. Let t denote the last round in which a cop probes s_{i+1} before the robber reaches v. Let the robber's position on round t be u_1, and the robber's position on round $t+k$ be u_2. To go undetected on round t, u_1 must be at distance at least $k+1$ from s_{i+1}. To avoid being detected on round $t-1$, when s_i was probed, u_1 must also be at distance at least k from s_i. First suppose u_1 is at distance at most $\frac{k}{2}$ from v. There is exactly one path from u_1 to s_i and exactly one path from u_1 to s_{i+1}, both of which must pass through v. Thus, we can say that $d(s_i,v) = d(s_i,u_1) - d(v,u_1) \geq k - \frac{k}{2} = \frac{k}{2}$ and similarly, $d(s_{i+1},v) = d(s_{i+1},u_1) - d(v,u_1) \geq k+1-\frac{k}{2} = \frac{k}{2}+1$. This gives

$$d(s_i,s_{i+1}) = d(s_i,v) + d(v,s_{i+1}) \geq \frac{k}{2}+\frac{k}{2}+1 = k+1,$$

but s_i was chosen to be at distance at most k from s_{i+1}. Thus, we conclude u_1 is at distance greater than $k/2$ from v. Similar analysis shows u_2 is at distance greater than $k/2$ from v. But then from round t to round $t+k$ the robber has traveled a distance of $d(u_1,v)+d(v,u_2) > k/2+k/2 = k$, which is a contradiction. We conclude that the robber cannot enter the path at any vertex v. The robber cannot enter the path at any $s_i \in S_k$, because if they are at distance $k+1$ when s_i is probed, they cannot reach s_i before it is probed again. We conclude that after k rounds if the robber enters the path, some cop will probe a vertex within distance k of the robber.

Next, we extend the strategy provided above, which detects the robber if they move to a path of T, to a strategy that detects the robber on T. Order the root-to-leaf paths in T using a depth-first-search ordering of the leaves. All but one cop, say C^*, will probe these paths. The cops use the first k rounds to clear the first path using the method described above. During this time, C^* probes r every round. After the first k rounds, the cops probing the first path will now focus on the second path. They spend the next k rounds clearing the second path using the method described above, while C^* alternates between probing r and the vertex furthest from r shared between the first and second paths. The cops continue this strategy, spending k rounds per path while C^* alternates between probing r and the last vertex shared between the current and previous path.

We claim that this detection strategy works; that is, after finitely many rounds, some cop will probe a vertex within distance k of the robber. To prove this claim, we show that from round $jk+1$ to round $(j+1)k$; the robber can only stay undetected by occupying vertices on paths later in the order than path j. As T contains a finite number of leaves, the robber will eventually be detected. Because the vertices were chosen by depth-first search, the vertices of path j partition T into the vertices on paths that come before j and those on paths that come after j, so the robber cannot access the vertices on paths before j without first occupying a vertex on path j. The initial portion of path j, up to at most distance k from where path j diverges from path $j+1$, has been probed for at least the last k rounds, so the robber is unable to enter undetected by the previous analysis. As C^* probes the junction between path j and path $j+1$ every other round, and $k \geq 1$, the robber cannot access vertices close to the intersection of the paths either. The remaining vertices of path j are only adjacent to vertices in paths that came before j, so the claim holds by induction.

To capture the robber, the cop player applies this detection strategy to the subtree rooted at each neighbor of r, with the modification that C^* probes r whenever it would probe the root of the subtree. The cop player's goal is to determine a neighbor r' of r such that the robber is contained in the subtree of the descendants of r', with root taken to be r'. Suppose the detection strategy is played on the subtree of descendants of r', and some cop's probe returned a distance of at most k to the robber. A cop probes r' once in every k rounds, and the cop C^* probes r once every two rounds, and might probe a vertex close to r' on the other round. All other vertices that are probed by the cops have distance more than k to vertices outside of this subtree. Therefore, the cop who probed a distance of at most k to the robber knows the robber is on this subtree unless it was one of these specially mentioned probes. In the latter case, the cops probe r and r' in the next round. If the robber was not on this subtree in the last round, then the cop on r will probe a distance of at most k, say d_1, and the cop on r' will probe a higher value, say d_2, which has either $d_2 = d_1 + 1$ or $d_2 = *$, both of which indicate to the cops that the robber is not on this subtree. If the robber is in this subtree, then either both cops probe $*$, or $d_2 < d_1$, or $d_2 = k$ and $d_1 = *$.

The cops can then probe r and r', ensuring the robber is not on either vertex, and then restart the strategy on the subtree rooted at r'. As this subtree has a strictly smaller radius, the claim follows by induction, noting that a tree of radius one is a star with k-visibility localization number 1 when $k \geq 1$.

Suppose the robber plays only on the subtree rooted at some r'. In that case, the cops will eventually use the detection strategy on that subtree, detect the robber, and make progress by limiting the robber to a smaller subtree. To avoid this, the robber will attempt to move from one neighbor of r to another. As C^* probes r every other round and $k \geq 1$, they will detect the robber at distance 1 from r. When this occurs, the cops adopt a straightforward strategy: C^* probes r every round, while the other cops probe the neighbors of r. As long as C^* received a distance of 1 on the previous round if a cop probes a neighbor of r and receives either no distance or a distance larger than C^*, then the cops know that the robber is not located on that subtree. If C^* stops receiving a distance of 1, the cops restart their detection strategy having eliminated at least $\lceil \frac{\mathrm{rad}(T)+k}{k^2} \rceil$ subtrees that could contain the robber. We conclude that the robber approaching r only speeds up the cops' progress to find the subtree on which the robber resides. Therefore, the robber's best strategy is to stay on one subtree of r where they will eventually be detected, and the cops will make progress. ☐

The following corollary provides a bound in the case $k = 0$.

Corollary 12. *For any tree T, $\zeta_0(T) \leq \mathrm{rad}(T) + 1$.*

Proof. Similar to the proof of Theorem 11, we order the paths from a central root vertex to the leaves by depth-first-search and sequentially search each root-to-leaf path by placing a cop on each vertex of the path. ☐

To prove the lower bound in Theorem 10, we begin with the following construction.

Theorem 13. *For each $h \geq 1$, $q \geq 3$, and $k \geq 1$, there exists a tree T of height* $\mathrm{rad}(T) = h(k+1)$ *with*

$$\zeta_k(T) > \frac{\Phi_V(T(h,q))}{4(k+1)} \geq \frac{h - \log_q(2h)}{4(k+1)} \geq \frac{\mathrm{rad}(T)}{4(k+1)^2} - \frac{\log(\mathrm{rad}(T))}{4(k+1)\log(q)}.$$

Proof. Let T be the result of taking $T(h,q)$ and subdividing each edge k times. Note that the resulting tree has height $\mathrm{rad}(T) = h(k+1)$: initially, the path from root to leaf had h edges, each of which was subdivided into $k+1$ edges, for a total of $h(k+1)$ edges in a path from root to leaf in T. The vertices of T that were originally vertices of $T(h,q)$ will be called *major* vertices of T. Each vertex of T is distance at most k from at most two major vertices.

Given a winning strategy for the k-proximity game on T that uses $\mathrm{prox}_k(T)$ cops, we will construct a winning strategy for the 0-proximity game on $T(h,q)$ that uses $4(k+1)\,\mathrm{prox}_k(T)$ cops. By Theorem 4, the 0-proximity game on $T(h,q)$ requires more than $\Phi_V(T(h,q))$ cops for the cops to be able to win. From Theorem 7, $\Phi_V(T(h,q)) \geq h - \log_q(2h)$ when $q \geq 3$ and $h \geq 1$. This result gives us the desired bound: since $4(k+1)\,\mathrm{prox}_k(T)$ cops can win the 0-proximity game on $T(h,q)$, we have $4(k+1)\,\mathrm{prox}_k(T) > h - \log_q(2h)$ or

$$\begin{aligned}\mathrm{prox}_k(T) &> \frac{h - \log_q(2h)}{4(k+1)} \\ &= \frac{\mathrm{rad}(T)/(k+1)}{4(k+1)} - \frac{\log_q(2\,\mathrm{rad}(T)/(k+1))}{4(k+1)} \\ &\geq \frac{\mathrm{rad}(T)}{4(k+1)^2} - \frac{\log(\mathrm{rad}(T))}{4(k+1)\log(q)}\end{aligned}$$

and Theorem 1 implies that this many cops are needed for the k-visibility Localization game as well.

We refer to the 0-proximity game on $T(h,q)$ with $4(k+1)\,\mathrm{prox}_k(T)$ cops as the *shadow* game and the k-proximity game on T with $\mathrm{prox}_k(T)$ cops as the *main* game. We will create a mapping from the positions of the cops on the main game during certain rounds to the vertices of the shadow game at certain rounds. If the robber is not caught when the cops play on these vertices of the shadow game during these rounds, then we can show that there is a robber strategy in the main game that also avoids capture. That is, we use the shadow game as an oracle for the main game.

We partition the rounds of the main game into periods of $k+1$ rounds. For period $\alpha \geq 1$, which runs from round $(\alpha-1)(k+1)+1$ to round $\alpha(k+1)$, let S_α be the set of major vertices such that some cop probed a vertex of distance at most k from that major vertex during period α. There are at most two major vertices of distance k from each vertex, there are $k+1$ rounds per period, and $\mathrm{prox}_k(T)$ vertices are probed during each round of the main game. Therefore, S_α contains at most $2(k+1)\,\mathrm{prox}_k(T)$ vertices.

The cops in the shadow game adopt the following strategy. On round α of the shadow game, they observe how the cops in the main game play during the $k+1$ rounds of period α. In the shadow game, the cops probe $S_{\alpha-1} \cup S_\alpha$ (where we take $S_0 = \varnothing$). As $|S_\alpha| \leq 2(k+1)\,\mathrm{prox}_k(T)$ for each α, this move requires at most $4(k+1)\,\mathrm{prox}_k(T)$ cops. Thus, it suffices to demonstrate that this strategy captures the robber in the shadow game.

For the sake of contradiction, suppose that the robber cannot be captured in the shadow game. Specifically, assume the robber occupies a sequence of vertices $\{w_i\}_{i=1}^{\infty}$ such that $d(w_i, w_{i+1}) \leq 1$ for $i \geq 1$ and no cop in the shadow game probes w_i on round i. Suppose that the robber in the main game chooses to implement the following strategy: they begin the game at w_1. For each $i \geq 1$, if $w_i \neq w_{i+1}$ the robber spends rounds $(i-1)(k+1)+1$ to $i(k+1)$ moving from w_i to w_{i+1}, and if $w_i = w_{i+1}$ then the robber remains stationary during these rounds. In either case, the robber is on w_{i+1} at the end of round $i(k+1)$. The robber in the main game is invisible, and the cops in the main game win as soon as a probe returns a non-$*$ distance, so this change in the robber's strategy does not affect how the cops play in the main game. Because $\operatorname{prox}_k(T)$ cops are playing the main game, this robber strategy does not avoid capture. Suppose the robber was captured in the main game on some round $(j-1)(k+1)+\beta'$, with $1 \leq \beta' \leq k+1$. Since $k \geq 1$, we can assume the cop player wins the main game because a probe returns a non-$*$ distance, as otherwise we can just extend the game by one additional round, in which a cop plays on the last known position of the robber and will probe a non-$*$ distance, since the robber may be distance at most 1 from this vertex.

So on round $(j-1)(k+1)+\beta'$ in the main game, the robber is on a vertex u when a cop probes vertex v such that $d_T(u,v) \leq k$. The robber was on vertex u in the path connecting the two major vertices w_j and w_{j+1}, and this cop was on vertex v in the path between the two major vertices, say y_1 and y_2, noting that we have $w_j = w_{j+1}$ if the robber was on a major vertex or $y_1 = y_2$ if this cop was on a major vertex. Since $d_T(u,v) \leq k$, it must be that one of w_i or w_{i+1} is equal to one of y_1 or y_2. However, since $\{y_1, y_2\} \subseteq S_j$, this means that one of w_i or w_{i+1} is in S_j. In the shadow game, S_j was probed on round j and again on round $j+1$. Thus, if $w_i \in S_j$, the robber would have been captured in the shadow game on round j, and if $w_{i+1} \in S_j$, the robber would have been captured in the shadow game on round $j+1$. However, this contradicts our assumption that the robber was not captured in the shadow game. We then have that the proposed cop strategy in the shadow game with $4(k+1)\operatorname{prox}_k(T)$ cops is guaranteed to capture the robber, and so $\operatorname{prox}_0(T(h,q)) \leq 4(k+1)\operatorname{prox}_k(T)$, as claimed. □

Next, we use this construction to bound $\zeta_k(T(h,q))$.

Corollary 14. *For $q \geq 3, k \geq 1$, and $h' \geq k+1$,*

$$\zeta_k(T(h',q)) > \frac{1}{4} \cdot \frac{h' - (k+1)(1+\log_q(h'))}{(k+1)^2}.$$

Proof. Set $h = \lfloor \frac{h'}{k+1} \rfloor$. The complete q-ary tree of height h' contains T, the subdivision of $T(h,q)$ that we analyzed in Theorem 13, noting that $q \geq 3, k \geq 1$, and

$$h = \left\lfloor \frac{h'}{k+1} \right\rfloor \geq \left\lfloor \frac{k+1}{k+1} \right\rfloor = 1.$$

Theorem 13 and Lemma 3 yield

$$\zeta_k(T(h',q)) \geq \zeta_k(T) > \frac{h - \log_q(2h)}{4(k+1)}.$$

As $h \geq \frac{h'}{k+1} - 1$, we have

$$\zeta_k(T(h',q)) > \frac{1}{4} \cdot \frac{h' - (k+1)(1+\log_q(h'))}{(k+1)^2}$$

as claimed. □

Noting that $\operatorname{rad}(T(h,q)) = h$ and $h + k = \Theta(h)$ when $h \geq k+1$, Theorem 11 together with Corollary 14 yield the following result.

Corollary 15. *For $q \geq 3, k \geq 1$, and $h(k) = h \geq k+1$, when k tends to infinity,*

$$\zeta_k(T(h,q)) = \Theta\left(\frac{h}{k^2}\right).$$

In fact, so long as $q \geq 3$, $k \geq 1$, and $h(k) = h \geq k+1$, these results imply that we know the asymptotic value of $\zeta_k(T(h,q))$ when k tends to infinity up to a multiplicative factor of 4, as

Corollary 14 yields $\limsup_{h\to\infty} \zeta_k(T(h,q))/\left(\frac{h}{k^2}\right) \geq \frac{1}{4}$ and $\liminf_{h\to\infty} \zeta_k(T(h,q))/\left(\frac{h}{k^2}\right) \leq 1$ from Theorem 11.

We complete this section with a proof of Theorem 10.

Proof of Theorem 10. We begin with the case that $f = \Omega(\sqrt{n})$. There is $\alpha \in \mathbb{R}^+$ and $N \in \mathbb{N}$ such that for all $n \geq N$, $f(n) \geq \alpha\sqrt{n}$. There are $\beta, \gamma > 0$ such that $\beta \leq f(n) \leq \gamma$ for all integers $1 \leq n \leq N$ since this set of values is finite. By Theorem 11, if $\mathrm{rad}(T_n) < N$,

$$\zeta_k(T_n) \leq \left\lceil \frac{\mathrm{rad}(T_n) + k}{k^2} \right\rceil + 1 \leq \frac{N+\gamma}{\beta^2} + 2.$$

For $\mathrm{rad}(T_n) \geq N$, we have that

$$\zeta_k(T_n) \leq \left\lceil \frac{\mathrm{rad}(T_n) + k}{k^2} \right\rceil + 1 \leq \frac{\mathrm{rad}(T_n)}{f(\mathrm{rad}(T_n))^2} + \frac{1}{f(\mathrm{rad}(T_n))} + 2 \leq \frac{1}{\alpha^2} + \frac{1}{\alpha\sqrt{N}} + 2.$$

Thus, by setting

$$d = \max\left(\frac{N+\gamma}{\beta^2} + 2, \frac{1}{\alpha^2} + \frac{1}{\alpha\sqrt{N}} + 2\right)$$

we have that

$$\limsup_{n\to\infty} \zeta_k(T_n) \leq d.$$

Now assume $f = o(\sqrt{n})$. Let $T_n = T(n,3)$, and set $k_n = f(n)$. For sufficiently large n we have $n \geq \sqrt{n} + 1 > k_n + 1$, so Corollary 15 yields

$$\zeta_{k_n}(T_n) = \Theta\left(\frac{n}{k_n^2}\right) = \Theta\left(\frac{n}{f(n)^2}\right).$$

As $f(n) = o(\sqrt{n})$, $f(n)^2 = o(n)$, and thus $\lim_{n\to\infty} \zeta_{k_n}(T_n) = \infty$ as required. □

4. Subdivisions of graphs

The relationship between subdividing a graph and the Localization game has been studied [15], and it is known that for any graph G, the graph obtained by subdividing each edge of G $3n$ times, $G^{1/3n}$, has $\zeta(G^{1/3n}) = 1$. The central idea of the cop's strategy on $G^{1/3n}$ is that if the cop probes all the original vertices of G one at a time, the robber will eventually be identified as being close to a probed vertex, and cannot move to another original vertex of G without the cop identifying its location. The issue with extending this strategy to the k-visibility Localization game for a fixed k is that, typically, we consider graphs for which $3n$ is much larger than k. This renders the strategy from the original game ineffective, since the cop can no longer see the entirety of any path of length $3n$. Therefore, we need to develop a new strategy for graph subdivision in the limited-visibility game.

To remedy this, we will instead consider what happens if we subdivide the edges non-uniformly. That is, we do not necessarily subdivide each edge the same number of times. The advantage of this approach is that if we separate the vertices of a graph into two sets and subdivide all the edges that go from one set to the other a large number of times, then in the proximity game, the cops can first clear these subdivided edges and then focus on clearing one of the sets of vertices. If we think of vertices the cop player has cleared as "clean" and vertices that could contain the robber as "contaminated," it will take a large number of rounds for the contaminated area to spread back over the subdivided edges to recontaminate the clean area. This time allows the cops to essentially ignore half of the graph while the contamination spreads back over the subdivided edges. Our main results are summarized in the following two theorems.

Theorem 16. *For a tree T and positive integer k, there is a subdivision T' of T with* $\mathrm{prox}_k(T') = 1$.

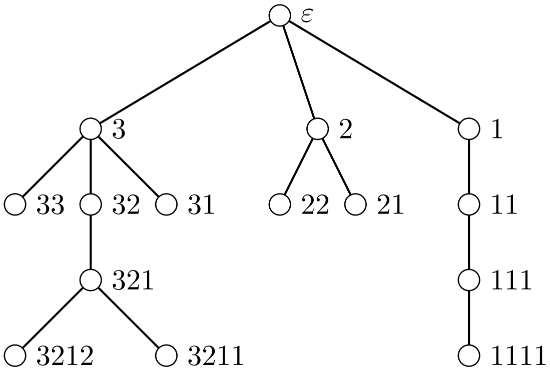

FIGURE 1. A tree labeled according to the proof of Theorem 16.

Theorem 17. *For a tree T and positive integer $k > 1$, there is a subdivision T' of T with $\zeta_k(T') = 1$.*

One may ask why $k = 1$ is forbidden in Theorem 17 but not Theorem 16. It can be shown that the class of graphs G with $\zeta_1(G) = 1$ is exactly the set of caterpillars [26]. As a result, it is not always possible to subdivide a tree T to reduce $\zeta_1(T)$ to 1.

We begin with the proof of Theorem 16. In the following, let $\mathrm{Children}(u)$ denote the set of children of u in some rooted tree (that is, the neighbors of u that are further from the root), $\mathrm{Parent}(u)$ denote the parent vertex of a non-root vertex u (that is, the unique neighbor of u closer to the root), and $\mathrm{Desc}(u)$ be the set of descendants of u (that is, the set of vertices v for which the unique path from the root to v passes through u, including u itself).

Proof of Theorem 16. Choose a vertex r of T with distance at most $\mathrm{rad}(T)$ from each other vertex as the root of T. Assume that $k < \mathrm{rad}(T)$, as otherwise probing r immediately returns a non-$*$ distance. Each vertex will be labeled with a word on the alphabet of the nonnegative integers. The root is labeled ε. We iteratively label the vertices of T by labeling the children of a vertex u as ui, for $1 \leq i \leq |\mathrm{Children}(u)|$. See Figure 1 for an example of this labeling.

For each edge $(\mathrm{Parent}(v), v)$ in T, we need to determine x_v, the number of times we will subdivide this edge when constructing T'. We first describe the strategy implemented by the cop player, as the choice of x_v is determined by this strategy. Let $P_v = (v_0 = \mathrm{Parent}(v), v_1, v_2, \ldots, v_{x_v+1} = v)$ denote the path of length $x_v + 1$ in T' between $\mathrm{Parent}(v)$ and v, and also denote by P'_v the subpath of P_v with the first $k+1$ vertices removed, i.e. $P'_v = (v_{k+1}, v_{k+2}, \ldots, v_{x_v+1})$.

To define each x_v, it is necessary to know the number of rounds required for the strategy to clear the descendant subtree of any given vertex. For any vertex u in T, let t_u be the number of rounds needed to ensure the robber is not on the subtree $\mathrm{Desc}(u)$ under the assumptions

1. before the first round, P_u does not contain the robber;
2. the robber does not move from a vertex outside of $\mathrm{Desc}(u) \cup P'_u$ onto a vertex in $\mathrm{Desc}(u) \cup P'_u$ during these t_u rounds; and
3. if the robber moves from a vertex in $\mathrm{Desc}(u)$ to a vertex outside of $\mathrm{Desc}(u)$, then the cop will capture the robber by the end of the t_u rounds.

The strategy on T' is defined recursively as follows. Let u be some vertex of T such that the robber is known to be on some vertex in $\mathrm{Desc}(u)$. For each i, starting with $i = |\mathrm{Children}(u)|$ and decreasing i by one until $i = 1$, the cop plays to ensure that the edges along the path $P_{ui} = (u_0 = u, u_1, u_2, \ldots, u_{x_{ui}+1} = ui)$ are clear while ensuring that the robber cannot move from an uncleared path onto u without being captured. To do this, in round 1 of this procedure, the cop probes $u = u_0$, ensuring that the robber is at distance at least $k+1$ from u before their move and distance at least k from u after their move. Over the next $2k+1$ rounds, the robber may move from their current position to u and then distance k along any path from u, including a path the cop has already cleared, but no further. We describe how the cop uses these $2k+1$ rounds to begin clearing P_{ui}.

For $2 \leq \tau \leq 2k+1$, on round τ the cop probes the vertex $u_{(2\tau-2)k}$ of path P_{ui} if $\tau \cdot k < x_{ui}+1$, and probes ui otherwise. In the former case, because the cop probes $u_{2\tau k}$ on round $\tau+1$, which is within distance k from the vertices $u_{(2\tau-1)k}, \ldots, u_{(2\tau+1)k}$, the robber can only be on a vertex u_j after moving in round τ if $j \geq (2\tau-1)k$. If the cop probes vertex ui, it is because ui is within distance k from all vertices $u_{(2\tau-1)k}, \ldots, ui$, so the robber cannot be on any vertex of P_{ui}. Thus after these $2k+1$ rounds, if the robber is on P_{ui}, they must either be on a vertex between u_{4k^2+1} and ui, inclusive, or have moved from an uncleared vertex through u to one of the first k vertices of P_{ui}, as previously mentioned.

We now repeat the procedure, starting with the cop probing vertex u in round $(2k+1)+1$, ensuring the robber is not within distance k of u, including along a cleared path. In round $(2k+1)+\tau$ with $2 \leq \tau \leq 2k+1$, the cop probes the vertex $u_{4k^2+(2\tau-1)k}$ of path P_{ui} if $4k^2+(2\tau-1)k < x_{ui}+1$, and probes ui otherwise. We continue repeating this procedure, with the cop probing either vertex $u_{(4k^2+k)\alpha-k}$ or ui at the end of repetition α of the procedure, at which point $(2k+1)\alpha$ rounds have taken place. The first repetition where ui will be probed is when $\alpha = \lceil \frac{x_{ui}+1+k}{4k^2+k} \rceil$, and so we consider P_{ui} cleared after these $y_{ui} = (2k+1)\lceil \frac{x_{ui}+1+k}{4k^2+k} \rceil \leq \frac{x_{ui}}{2k-1}$ rounds, although we note that those vertices of distance at most k from u may contain the robber at this point. These vertices of distance at most k from u will be cleared during the next cop move, unless they are already clear.

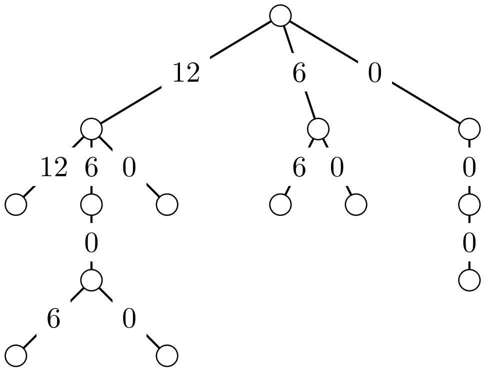

Figure 2. The tree from Figure 1 with labels representing the number of subdivisions for a $k=2$ strategy.

This process is repeated for each i, from $i = |\text{Children}(u)|$ until $i=1$, at which point all the paths P_{ui} have been cleared, noting that this cop strategy does allow the robber to enter a path P_{ui} through the vertex ui after this path had been cleared. Define $x_{u0}=1$, and

$$x_{ui} = \left(\sum_{i'<i} y_{ui'} \right) + \max_{i'<i} \{t_{ui'}\},$$

noting that this value is well defined since $x_{ui'}$ is defined for $i'<i$, since $y_{ui'}$ is defined once $x_{ui'}$ is defined, and since the $t_{ui'}$ are defined inductively. Notice that after we have cleared path P_{ui} in the above procedure, there will be a further $\sum_{i'<i} y_{ui'}$ rounds until all the paths P_{ui} have been cleared, and so the robber may have moved along P_{ui} from ui to be at the vertex of distance $x_{ui} - \sum_{i'<i} y_{ui'} = \max_{i'<i} t_{ui'}$ from u, but no closer. Thus, after all of these paths have been cleared, in the next t_{uj} rounds, we may clear the subtree $\text{Desc}(uj)$, where j is the smallest index such that the robber may still be in $\text{Desc}(uj)$. At this point, the robber may have moved close to

the root but has had only just enough time to reach it. We then repeat this procedure from the start, again clearing the paths P_{ui} for all i, and then clearing the next subtree $\mathrm{Desc}(u(j+1))$, until all subtrees have been cleared. Since the robber only has enough time to reach u before we restart, the robber cannot move between two subtrees, ensuring that the tree will be cleared by the end of this procedure. The robber will be captured using this procedure if they started in $\mathrm{Desc}(u)$. Define t_u to be the length of time it took to run this procedure.

In this iterative approach, starting with u as the root vertex, each edge (v, vi) of the graph has its parameter x_{vi} explicitly defined. The tree T' with edge (v, vi) subdivided x_{vi} times requires exactly one cop to win, and so the proof is complete. □

As an immediate implication of Lemma 2 and Theorem 16, it is possible to subdivide any tree to achieve $\zeta_k(T) \leq 2$, but when is it possible for $\zeta_k(T)$ to be 1? If we consider the structure of a tree in which every edge has been subdivided a large number of times, then we can see that locally the tree will resemble a *spider graph*, a tree with exactly one vertex of degree larger than two. This will prove advantageous for the cop since the k-visibility localization number of a spider graph is 1 for any $k \geq 2$, as we will see in Lemma 18.

Lemma 18. *If S is a spider graph and $k \geq 2$, then $\zeta_k(S) = 1$.*

Proof. Let c be the unique vertex of S of degree larger than two and mark all branches as "uncleared." To locate the robber using one probe per round, the cop player alternates between two strategies.

In Strategy A, the cop selects a leaf ℓ of an uncleared branch and probes it. If the probe returns a non-$*$ distance $d \leq d(\ell, c)$, then cop locates the robber at the unique vertex at distance d from ℓ. Otherwise, the cop now alternates probing c and the vertex along the path from ℓ to c at distance $k-1$ closer to c than the previous probed vertex along this path. As before, if probing a vertex along the path yields a non-$*$ distance at most the distance to c, the robber is located. If the cop player reaches the vertex on the branch within distance $k-1$ from c without ever receiving a non-$*$ distance, they mark this branch as "cleared" because the robber could not have been located on this branch. The cop then selects a new uncleared branch and repeats the process.

If the cop receives a non-$*$ probe after probing c, they move to Strategy B, described below. If they probe a vertex v within distance $k-1$ of c and receive a distance larger than $d(v, c)$, the cop marks the branch as cleared, then probes c and is guaranteed to receive a non-$*$ distance, at which point they switch to Strategy B.

In Strategy B, we assume the cop has just probed c and received a distance d. Each branch of length less than d is marked "cleared" as the robber cannot be on such a branch. The cop selects an uncleared branch and probes the vertex at distance $d+1$ along that branch. If the probe returns a distance of 0 or 2, the robber has been located. Otherwise, that branch is cleared, and the cop now probes c. If the probe at c returns a non-$*$ distance, the cop repeats this process; otherwise, they return to Strategy A.

Each strategy either locates the robber or clears a branch after a finite number of rounds. Because the robber cannot occupy a branch when it is cleared and S has a finite number of branches, to prove this strategy locates the robber it suffices to show the robber cannot move from an uncleared branch to a cleared branch without being captured. Note that in both strategies, the cop probes c every other round. To move from an uncleared branch to a cleared branch, the robber must pass through c. If the robber occupies c on the round the cop probes it, they will be captured. Thus, the robber must be adjacent to c when it is probed. As $k \geq 2$, this causes the probe to return a distance of 1, so the cop will implement Strategy B. If the robber now moves to c, the cop will probe a vertex at distance $2 \leq k$ from c, receive a distance of 2, and locate the robber at c.

We conclude that the robber cannot move from an uncleared branch to a cleared branch; thus, each branch will eventually be cleared and the robber located. □

Now consider what happens if, before we subdivide to reduce the k-proximity number to 1, we subdivide the tree so that all vertices of degree greater than 2 are very far apart. The tree is now locally a spider, and if the tree's edges have been subdivided sufficiently many times, the robber cannot escape to another high-degree vertex before the cop locates them. This is exactly the strategy described below in the proof of Theorem 17.

Proof of Theorem 17. Take every edge of T and subdivide it $\frac{Nk}{2}+k+1$ times, where N is the smallest value such that H_N (the N^{th} harmonic number) is at least $\frac{4\Delta(T)}{k}$. Next, apply the subdivision described in Theorem 16 to obtain T'. The tree T' now satisfies $\operatorname{prox}_k(T') = 1$, and the vertices of degree greater than 2 in T' are all sufficiently far apart for our purposes. When the cop wins the proximity game and sees the robber, we assume the cop is on a high degree vertex (degree greater than 2) or the cop is not on a high degree vertex but the distance from the cop to the robber is greater than the distance from the cop to the nearest high degree vertex. We do this since if the cop sees the robber far from a high-degree vertex, then the strategy plays out similarly. Now the cop uses the fact that this is locally a spider graph and applies the strategy from Lemma 18, but rather than using the leaf of the branch, the cop uses the furthest vertex the robber could have reached if they were on this branch when the first probe returned a non-$*$ distance. If the robber cannot evade for more than $\frac{Nk}{2}+1$ rounds after being seen, then the cop can use this strategy to eventually capture the robber.

We now show that the robber cannot evade for more than $\frac{Nk}{2}+1$ rounds once the cop has won the proximity game. First, consider how long it will take to clear the first branch. Since the robber could have initially been on only one vertex on the branch, there are now three possible locations, and since $k \geq 2$ the cop can clear this in one move. For the first $\frac{k}{4}$ branches, the cop will be able to do this, and if we account for the fact that every other round the cop will probe the high-degree vertex, we can see that this will take at most two rounds per branch or $\frac{k}{2}$ rounds total. At this point, the "contamination" extends for $k+1$ steps on each branch, and it will now take two rounds to clear each branch; once we include the rounds in which the cop probes the high-degree vertex, this becomes at most four rounds. We can find that $\frac{k}{8}$ branches can be cleared like this before it takes more moves, then the contamination will stretch across a length of $2k+1$ on each branch while the cop clears the next $\frac{k}{12}$ branches in six rounds each, and so on. Continuing this pattern, we can see that to clear all the branches in time, it must be that $\frac{k}{4}\sum_{i=1}^{N}\frac{1}{i} \geq \Delta(T)$, which implies that we need $H_N \geq \frac{4\Delta(T)}{k}$. Thus, the furthest that the robber could have moved is $\frac{Nk}{2}+1$, which gives the desired result. ☐

Since any graph can be subdivided enough to make a graph that is locally a spider, we can apply the concept from Theorem 17 any time there is a class of graphs which is closed under subdivisions and where it is possible to subdivide enough to reduce k-proximity number to 1. This notion is formalized in Corollary 19.

Corollary 19. *For any family of graphs $\mathcal{F}$ that is closed under subdivisions, if for all $G \in \mathcal{F}$ there is an integer $k > 1$ such that there exists a subdivision G' of G with $\operatorname{prox}_k(G') = 1$, then there exists a graph $G'' \in \mathcal{F}$ which is also a subdivision of G such that $\zeta_k(G'') = 1$.*

As an example, the class of cycle graphs $\{C_n\}_{n\geq 3}$ is closed under subdivisions, and every cycle C can be subdivided in a way that reduces $\operatorname{prox}_2(C)$ to 1. Therefore, by Corollary 19 it is possible to subdivide any cycle C in a way that reduces $\zeta_2(C)$ to 1.

5. Future directions

As mentioned, in a forthcoming paper, we analyze the k-visibility localization number of Cartesian grids. A key result of that work is demonstrating that for distinct i and j, there exist graphs G such that $\zeta_i(G) \neq \zeta_j(G)$.

While we proved that for any tree T and any $k \geq 2$, there is a subdivision T' of T satisfying $\zeta_k(T) = 1$, it remains an open question whether the same holds for general graphs. Further, we do not know how to characterize the k-visibility localization number for trees. A recent paper [8] considered the localization numbers of locally finite graphs, which are countable graphs in which each vertex has finite degree. A direction for future research is to consider the ζ_k numbers of locally finite trees and graphs.

Acknowledgments

The authors were supported by NSERC.

References

[1] Z. Bartha, J. Komjáthy, J. Raes, Sharp bound on the truncated metric dimension of trees, *Discrete Mathematics* (2023) **346** 113410.

[2] N.C. Behague, A. Bonato, M.A. Huggan, T.G. Marbach, B. Pittman, The localization capture time of a graph, *Theoretical Computer Science* **911** (2022) 80-91.

[3] A. Bernshteyn, E. Lee, Searching for an intruder on graphs and their subdivisions, *Electronic Journal of Combinatorics* **29**(3) (2022) P3.9.

[4] A. Bonato, *An Invitation to Pursuit-Evasion Games and Graph Theory*, American Mathematical Society, Providence, Rhode Island, 2022.

[5] A. Bonato, M.A. Huggan, T. Marbach, The localization number of designs, *Journal of Combinatorial Designs* **29** (2021) 175–192.

[6] A. Bonato, M.A. Huggan, T. Marbach, The localization number and metric dimension of graphs of diameter 2, *Contributions to Discrete Mathematics* **18** (2023) 29–52.

[7] A. Bonato, W. Kinnersley, Bounds on the localization number, *Journal of Graph Theory* **94** (2020) 1–18.

[8] A. Bonato, F. Lehner, T. Marbach, JD Nir, The Localization game on locally finite trees, In: *Proceedings of EUROCOMB'23*, 2023.

[9] A. Bonato, T. Marbach, M. Molnar, JD Nir, The one-visibility Localization game, *Theoretical Computer Science* **978** (2023) 114186.

[10] A. Bonato, R.J. Nowakowski, *The Game of Cops and Robbers on Graphs*, American Mathematical Society, Providence, Rhode Island, 2011.

[11] A. Bonato, L. Mandic, T. Marbach, M. Ritchie, *The isoperimetric peak of complete trees*, Preprint 2025.

[12] B. Bosek, P. Gordinowicz, J. Grytczuk, N. Nisse, J. Sokół, M. Śleszyńska-Nowak, Localization game on geometric and planar graphs, *Discrete Applied Mathematics* **251** (2018) 30–39.

[13] B. Bosek, P. Gordinowicz, J. Grytczuk, N. Nisse, J. Sokół, M. Śleszyńska-Nowak, Centroidal localization game, *Electronic Journal of Combinatorics* **25**(4) (2018) P4.62.

[14] A. Brandt, J. Diemunsch, C. Erbes, J. LeGrand, C. Moffatt, A robber locating strategy for trees, *Discrete Applied Mathematics* **232** (2017) 99–106.

[15] J. Carraher, I. Choi, M. Delcourt, L.H. Erickson, D.B. West, Locating a robber on a graph via distance queries, *Theoretical Computer Science* **463** (2012) 54–61.

[16] N.E. Clarke, D. Cox, C. Duffy, D. Dyer, S.L. Fitzpatrick, M.E. Messinger, Limited visibility Cops and Robber, *Discrete Applied Mathematics* **282** (2020) 53–64.

[17] T. Datema, *Extremal graphs for threshold metric dimension*, Bachelors thesis, Delft University of Technology, 2022.

[18] D. Dereniowski, D. Dyer, R. Tifenbach, B. Yang, Zero-visibility cops & robber and the pathwidth of a graph, *Journal of Combinatorial Optimization* **29** (2015) 541–564.

[19] D. Dereniowski, D. Dyer, B. Yang, The complexity of zero-visibility cops and robber, *Theoretical Computer Science* **607** (2015) 135–148.

[20] R.M. Frongillo, J. Geneson, M.E. Lladser, R.C. Tillquist, E. Yi, Truncated metric dimension for finite graphs, *Discrete Applied Mathematics* (2022) **320** 150–169.

[21] J. Geneson, E. Yi, The distance-k dimension of graphs, Preprint 2025.

[22] F. Harary, R.A. Melter, On the metric dimension of a graph, *Ars Combinatoria* (1976) **2** 191–195.
[23] L.H. Harper, Optimal Numberings and Isoperimetric Problems on Graphs, *Journal of Combinatorial Theory* **1** (1966) 385–393.
[24] J. Jiři, *Parameterized Algorithms for the Truncated Metric Dimension problem*, Bachelors thesis, Czech Technical University, 2023.
[25] N. Kinnersley, The vertex separation number of a graph equals its path-width. *Information Processing Letters* **42** (1992) 345–350
[26] M. Molnar, *A new variant of the Localization game*, Masters thesis, Toronto Metropolitan University, 2023.
[27] S. Seager, Locating a robber on a graph, *Discrete Mathematics* **312** (2012) 3265–3269.
[28] S. Seager, Locating a backtracking robber on a tree, *Theoretical Computer Science* **539** (2014) 28–37.
[29] P.J. Slater, Leaves of trees, in *Proceedings of the Sixth Southeastern Conference on Combinatorics, Graph Theory and Computing (Florida Atlantic Univ., Boca Raton, Fla., 1975)*, pp. 549–559, Congress. Numer., No. XIV, Utilitas Math., Winnipeg, MB.
[30] R. Tosić, Vertex-to-vertex search in a graph, In: *Proceedings of the Sixth Yugoslav Seminar on Graph Theory, Dubrovnik* (1985) 233–237.
[31] D.B. West, *Introduction to Graph Theory, 2nd edition*, Prentice Hall, 2001.
[32] B.T. Yang, Computing the one-visibility cop-win strategies for trees, *Combinatorial optimization and application*, *Lecture Notes in Computer Science* **13135** (2021) 125–139.
[33] B.T. Yang, One-visibility cops and robber on trees: optimal cop-win strategies, *Theoretical Computer Science* **928** (2022) 27–47.
[34] B.T. Yang, T. Akter, One-visibility cops and robber on trees, *Theoretical Computer Science* **886** (2021) 139–156.

(A1,A2,A3) Toronto Metropolitan University, Toronto, Canada

(A4) Oakland University, Rochester, U.S.A.
Email address, A1: `(A1) abonato@torontomu.ca`
Email address, A2: `(A2) trent.marbach@gmail.com`
Email address, A3: `(A3) jmarcoux@torontomu.ca`
Email address, A4: `(A4) jdnir@oakland.edu`