BACHELOR

On the threshold metric dimension

Raes, Jarvi

Award date:
2021

Department of Mathematics and Computer Science

On the threshold metric dimension

Author:
Jarvi Raes (1014564)

Supervisor:
J. Komjathy

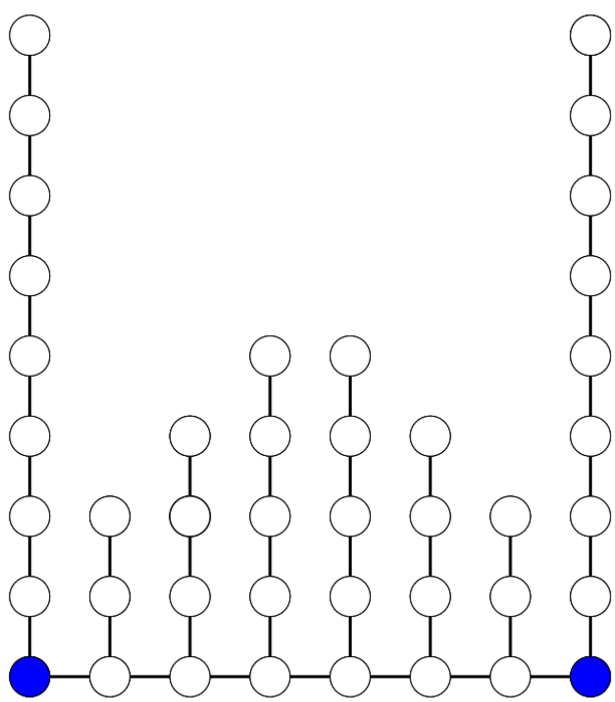

Figure 1: The largest graph possible with two sensors and threshold $k = 8$

August 17, 2021

Contents

1 Introduction

1.1 Motivation

Our current society is often called a networked society because of the many networks that surround us. Examples of those networks are things like traffic networks and the internet. On these networks, many processes are able to spread such as computer viruses, news on Facebook, retweeting on Twitter, viral videos and many more. In many contexts, finding the source is very important: finding the designer of a computer virus, the source of fake news and finding patient zero, a topic that has shown to be useful during the start of the COVID-19 pandemic in the beginning of 2020. The information known in these source localization problems normally consists of the infection time of a small subset of the network. The *metric dimension* is a notion in combinatorics first defined in 1975 by Slater [17] and a year later independently by Harary and Melter [4]. The metric dimension models the spreading in networks as graphs where the source spreads along edges starting at a source vertex and the information is given by the distances between sensor vertices and the source vertex. Recently, it has been connected to the deterministic version of the source-localization problem [23]. While spreading in most networks is not a deterministic problem, the metric dimension can still be used to give estimates of the number of sensor vertices required in some settings. Ever since its introduction, the metric dimension has been studied extensively [3, 16, 22]. Ever since, the metric dimension has had real life applications such as the representation of chemical compounds [8], fire protection [6] and the navigation of robots in networks [9]. Moreover, several other variants of the metric dimension have been researched such as the k-metric dimension [21], strong metric dimension [14] and local metric dimension[15]. There is also the topic of this thesis, the *threshold-k metric dimension*. The threshold-k metric dimension is a variant of the metric dimension where sensor vertices can only tell their distance to the source vertex if they are within a distance of k to it, where k is a positive integer. This topic is of interest for processes that mutate or change over time such as viruses and gossip. Most of the focus on this topic is on the locating-dominating code, where the maximal distance the sensor vertices can be from the source vertex is equal to one. The notion of the locating-dominating code was first introduced by Slater [19], who showed that it was a linear problem and found a lower bound [18], which was later improved by Blidia et al. [1] and later again by Slater [20]. A different kind of topic on networks is the source obfuscation problem. In this problem, the goal is to spread something along a network in such a way that a few spy vertices cannot detect the source vertex. Examples of the applications of this problem are in the anonymization of transactions of Bitcoin. Similarly to the source localization problem there are also uses of the metric dimension in the source obfuscation problem as it states the number of spy vertices required such that detection of the source vertex is guaranteed. For example, a recent paper [12] found that the addition of connections in the network has the ability to significantly increase the metric dimension, implying that the spy vertices will likely have more difficulty finding the source vertex.

1.2 Algorithmic Aspect

It is well-known that finding the metric dimension of arbitrary graphs is an NP-hard problem [10] and only approximable up to $log(n)$ [5], meaning that it becomes impossibly hard to find the metric dimension on very large graphs. As such studies usually focus on certain types of graphs where the metric dimension is easier to find, such as wheels [2] and circulant graphs [7]. A big focus is also on trees, for which it has been shown that the metric dimension is linear to calculate [10]. There is also research on the asymptotic behaviour of certain trees, such as uniform random trees [13] critical Galton-Watson trees and linear preferential attachment trees [11].

1.3 Our contribution

For the threshold-k metric dimension in trees, most of the research is focused on the threshold-1 metric dimension and the threshold-∞ metric dimension, referred to as the locating-dominating code and metric dimension respectively. However, not much is known about the threshold-k metric dimension of trees for other values other than that it has to be somewhere between the metric dimension [10] and the lower bound of the locating-dominating code [20]v of the tree. In this thesis we study these threshold-k metric dimensions for which not much is known in more detail. We start by finding the largest possible trees that have a certain threshold-k metric dimension. After we have done this, we use the size of these graphs to find the sharp lower bound that works for arbitrary k, and later we further develop this lower bound for $k = 2$.

1.4 Methodology

To find the lower bound of the threshold-k metric dimension for arbitrary k we first find the largest trees that have a certain threshold-k metric dimension. We do this by finding a 'skeleton' of the tree, characteristics that this tree must have. These characteristics are found by showing that trees that do not follow these characteristics can be transformed in certain ways such that the threshold-k metric dimension does not increase, while the number of vertices increases. The first transformation moves the vertices that are only measured by a single sensor vertex to be connected only to that sensor vertex, which allows us to do the other transformations. The first of these other transformations moves sensor vertices with certain properties between them closer together. This cause these sensor vertices to get close enough to each other so that the second transformation can be done. This transformation adds an additional vertex to the tree. After these transformations the sensor vertices can still determine the source vertex, proving that the tree was not the largest tree possible with that threshold-k metric dimension.
Once we have done this and reach the optimal 'skeleton' we can calculate the largest number of vertices this skeleton can have, and then we use this to calculate the lower bound of the threshold-k metric dimension. We then further develop this lower bound for the threshold-2 metric dimension to take into account the structure of less optimal trees by decomposing the tree into a skeleton with dangling ends. These dangling ends need additional sensors. We calculate the number of additional sensors needed, which allows us to find this lower bound by simply adding the number of additional sensors to the number of sensors the skeleton needs, which follows the original lower bound.

2 Preliminaries

In this section, we will mathematically define most of the terms used in the rest of this thesis.
A graph G is a set of points called vertices combined with a set of edges, denoted by $G = (V, E)$. Each element in the set of edges consists of two vertices that the edge connects. A trail is a route along vertices by way of distinct edges. The length of a trail is the number of edges in the trail. A path is a trail in which every vertex is distinct. A cycle is a trail in which the only repeating vertices are the first and the last vertex. The distance between two vertices v and w, denoted by $d(v, w)$ is the number of edges in the shortest path between the vertices. The degree of a vertex is the number of edges containing that vertex. A vertex of degree one is called a leaf. A leaf path is a path of at least two vertices of which the last vertex is a leaf, and every other vertex except the first one is of degree two. We denote the vertex set V combined with an extra vertex v by $V \cup \{v\}$ and $V \backslash \{v\}$ corresponds to the set V without the vertex v. Similarly, adding a set of vertices S to V is indicated by $V \cup S$ and removing S from V by $V \backslash S$. The addition of edge (v, w) to E is denoted by $E \cup (v, w)$ and the removal of (v, w) from E by $E \backslash (v, w)$. A connected graph is a graph where for any two vertices there is at least one path between them. A tree is a connected graph that does not have any cycles. We can now define the main topic of this thesis:

Definition 2.1. *Let $G = (V, E)$ be an arbitrary graph. A vertex z resolves a pair of vertices x,y in V if $d(x, z) \neq d(y, z)$ and if at least one of the conditions $d(x, z) \leq k$ and $d(y, z) \leq k$ are satisfied. A subset S of V such that for every pair of vertices x,y in V there is some vertex z in S such that z resolves x, y is called a threshold-k resolving set for G. The threshold-k metric dimension of G,* *denoted by $Tmd_k(G)$,is the smallest integer n such that there exists a threshold-k resolving S for G with $|S| = n$.*

Lastly, a vertex s is called a sensor if $s \in S$ where S is a threshold-k resolving set, and we say that the sensor s measures the vertex x if $d(x, s) \leq k$. This allows us to make the last definitions of this section.

Definition 2.2. *In a tree, the path between two sensors is called a sensor path if it does not contain any other sensors. It is called a strong sensor path if the distance between the sensors is less than $k + 1$.*

Definition 2.3. *In a tree, the vertex v is said to be identified by the sensor path between sensors w and y if:*
1. The path between v and y does not contain w.
2. The path between v and w does not contain y.
3. v is measured by both w and y.
5. At least one of the conditions $d(v, w) \neq d(x, w)$, $d(v, y) \neq d(x, y)$ are satisfied for any $x \in V$.

3 Lower bound for the threshold-k metric dimension

In the last section the threshold-k metric dimension was defined. In this section we will find the largest possible tree with a certain threshold-k metric dimension and use this to get a sharp lower bound for the threshold-k metric dimension of a tree on n vertices. Let $\mathcal{G}_p$ be the set of trees such that $G_p \in \mathcal{G}_p$ if $Tmd_k(G_p) = p$. Let $\mathcal{G}_p^*$ be the set of trees such that $G_p^* \in \mathcal{G}_p^*$ if $G_p^* \in \mathcal{G}_p$ and $|G_P^*| = \max_{G \in \mathcal{G}_p}(|G|)$. We start by stating the following theorem about the lower bound of the threshold-1 metric dimension, proven by Slater [18].

Theorem 3.1 (Lower bound of the threshold-1 metric dimension)**.** *Let T be a tree on n vertices, then a lower bound for its threshold-1 metric dimension is $(n+1)/3$. This lower bound is attainable if it is an integer.*

We will now state the main theorem of this section, which we will prove later. This is followed by Corollary 3.3 where we expand Theorem 3.1 to work for the threshold-k metric dimension for arbitrary k, which we will prove immediately.

Theorem 3.2. *Let $G \in \mathcal{G}_p^*$, then $|G| = (k + 1)p + (p - 1)(k^2 + k + 1)/3$ if k mod $3 = 1$ and $|G| = (k + 1)p + (p - 1)(k^2 + k)/3$ otherwise.*

Corollary 3.3 (Lower bound of the threshold-k metric dimension for arbitrary k)**.** *Let T be a tree on n vertices. A lower bound for its threshold-k metric dimension is $(3n+k^2+k+1)/(k^2+4k+4)$ if k mod $3 = 1$ and $(3n + k^2 + k)/(k^2 + 4k + 3)$ otherwise. This lower bound is attainable if it is an integer.*

Proof. Let $Tmd_k(T) = p$. Take $G' \in \mathcal{G}_p^*$. G' attains the lower bound, and T cannot have more vertices than G' by definition of G'. As such the lower bound must also hold for T □

Remark 3.4. *The ceiling of this lower bound is sharp and attainable.*

To prove Theorem 3.2 we first introduce two definitions and prove a lemma.

Definition 3.5. *Let G be a graph with threshold-k resolving set S. The subset of sensors $S' \subseteq S$ with $S' = \{s_1, \ldots, s_n\}$ is said to uniquely measure a vertex x if $d(s_i, x) \leq k$, for $i = 1, \ldots, n$, $d(x, s_j) \neq d(x, s_i) + d(s_i, s_j)$ for $i, j = 1, \ldots, n$, $i \neq j$ and for every y measured by the same set of sensors there is some s_i that resolves x and y.*

Definition 3.6. *Let G be a graph with threshold-k resolving set S. The 1-attraction of a sensor $s \in S$, denoted by $A^1(s)$, is the set of vertices uniquely measured by s, excluding s itself.*

Lemma 3.7. *Let $T = (V, E)$ be a tree. S is a threshold-k resolving set for T if every vertex in V can be uniquely measured by a subset of S.*

Proof. Let x,y be an arbitrary pair of vertices in V and let x be uniquely measured by $S_1 \subseteq S$. If y is also uniquely measured by S_1 then there is some $s_i \in S_1$ that resolves x and y by definition. If y is not uniquely measured by S_1 then there are two possibilities. The first possibility is that there is some $s \in S_1$ such that $d(y, s) > k$, so s resolves x and y as $d(x, s) \leq k$ by definition. The second possibility is that there exist $s, w \in S_1$ such that $d(w, y) = d(w, s)+d(s, y)$ which tells us that either s or w resolves x and y as otherwise we have $d(w, x) = d(w, y) = d(w, s)+d(s, y) = d(w, s)+d(s, x)$ which contradicts that $s, w \in S_1$. This completes the proof of the Lemma. □

To prove Theorem 3.2, we first show in Lemma 3.8 that for every tree there exists another tree with the same vertex set and threshold-k resolving set, such that every sensor has its 1-attraction contained in a leaf path starting from itself. We then show in Lemma 3.9 that every sensor path in any $G^*_p \in \mathcal{G}^*_p$ is disjoint, using the fact that if there are sensor paths that share vertices, we can transform the graph in certain ways, resulting in a graph larger than G^*_p that can still be measured by the same set of sensors, which is a contradiction. In Lemma 3.10 we find the number of sensor paths G^*_p has, and in Lemma 3.11 we determine the maximal number of vertices that can be identified by a sensor path, which allows us to prove Theorem 3.2.

Lemma 3.8. *Let $G = (V, E)$ be a tree on n vertices with threshold-k resolving set S. Let $s \in S$ and let $A^1(s) = \{s_1, \ldots, s_u\}$. Then there exists a tree $G' = (V, E')$, that also has threshold-k resolving set S, such that $A^1(s)$ is contained in a leaf path from s to s_u.*

Proof. We first define the edge set $E_1 \subseteq E$, and the new edge sets E_2 and E_3. Let E_1 be a set of edges with $(x, y) \in E_1$ if $(x, y) \in E$ and at least one of x,$y \in A^1(s)$. Let $E_2 = \{(s, s_1) \cup (s_1, s_2) \cup \cdots \cup (s_{u-1}, s_u)\}$. Lastly, let E_3 be a set of edges with $(x, y) \in E_3$ if x,$y \notin A^1(s)$, the path from s to x is contained in the path from s to y and all vertices on the path from x to y excluding x and y itself are contained in $A^1(s)$. Note that it is possible for (x, y) to be in E_3 with $x = s$, as can be seen in Figure 2. Let $G' = (V, E')$ with $E' = (E \backslash E_1) \cup E_2 \cup E_3$. We claim that if S is a threshold-k resolving set for G, then S is also a threshold-k resolving set for G' . Figure 2 shows an example of the difference between G and G'.

To prove this we can already see that every vertex in $A^1(s)$ is still uniquely measured by s, since these vertices are the only vertices uniquely measured by s, and they all have a different distance to s. We prove the rest by contradiction. Assume there is a pair $x, y \in V \backslash A^1(s)$ that is not resolved by any sensors in G' but that is resolved by some sensor in G. We know that the moved vertices were only measured by s, which allows us to show that $d(s^*, v)$ is the same in both G and G' or $d(s^*, v) > k$ in both G and G' for any sensor $s^* \neq s$ and any vertex $v \in V$. This can easily be shown as if $d(s^*, v) \leq k$ in G, s^* measures every vertex on the path between s^* and v too, meaning that none of these vertices move so $d(s^*, v)$ is the same in G', and if $d(s^*, v) > k$ there are at least k vertices between s^* and v, of which the first k are measured by s^* and as such do not move, so we have $d(s^*, v) > k$ in G' too. As a result of this we know that if s^* cannot resolve x and y in G' then s^* cannot resolve x and y in G either as if $d(x, s^*) = d(y, s^*)$ in G', then $d(x, s^*) = d(y, s^*)$ in G too and if both $d(x, s^*) > k$ and $d(y, s^*) > k$ in G' then $d(x, s^*) > k$ and $d(y, s^*) > k$ in G too.

Now all that is left is to show that s cannot resolve x and y in G, if there is no sensor that can resolve x and y in G'. Since s cannot resolve x and y in G', $d(x, s) = d(y, s)$ in G'. If $d(x, s) \neq d(y, s)$ in G, then some vertex s_i was moved during the transformation to G' that is on the path from s to x and not on the path from s to y or on the path from s to y and not on the path from s to x. As G is a tree, this means that s_i is on the path from x to y. As $x \notin A^1(s)$ we know there is at least one other sensor s' that also measures x. As s_i is on the path from x to y, and s_i is only measured by s, y cannot be measured by s' meaning that s' resolves x and y, which is a contradiction. As such we know that $d(x, s) = d(y, s)$ in G. We have now shown that if there is no sensor $s \in S$ that resolves x and y in G' then neither is there a sensor $s \in S$ that resolves

x and y in G. This completes our proof, as this contradicts our initial assumption that there is a sensor that resolves x and y in G but no sensor that resolves x and y in G'. As such we know that since S is a threshold-k resolving set of G, it is also a threshold-k resolving set of G', proving the Lemma. ☐

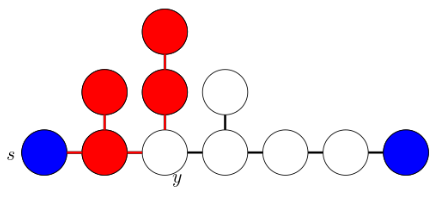

Figure 2: An example of Lemma 3.8 with $k = 4$. The red vertices are in $A^1(s)$ and as such the red edges are in E_1 and the green edges are in E_2. As $s, y \notin A^1(s)$, the path from s to s is contained in the path from s to y and every vertex on the path from s to y except for s and y are in $A^1(s)$, the purple edge (s, y) is in E_3. The blue vertices are sensors and are able to uniquely measure every vertex in both graphs.

Lemma 3.9. *Let G be a graph on n vertices with threshold-k resolving set S. If $G \in \mathcal{G}_p^*$ then, after applying Lemma 3.8 on every sensor in S, every sensor path is disjoint except for their sensors.*

Proof. We prove this by contradiction. We show that if G has sensor paths that are not disjoint, we can change the graph in such a way that we end up with a different graph on more than n vertices that still has threshold-k resolving set S using two kinds of transformations, which we will now show.

For transformation A we require that there is at least one pair of strong sensor paths that starts at the same sensor and share vertices. Take the shortest of these sensor paths and the second shortest sensor path that starts at either of the sensors of the first sensor path. Denote the sensors in these sensor paths by s, v and w such that $d(s, v) \leq d(s, w) \leq d(w, v)$ and let the sensor paths be $V = (s, v_1, \ldots, v_p, v)$ and $W = (s, w_1, \ldots, w_u, w)$. Because G is a tree, there is some $i \in \mathbb{N}$ such that $V \cap W = (s, w_1, \ldots, w_i)$. We define $X = \{x_1, \ldots, x_n\}$ to be the set of vertices such that $x \in X$ if for any sensor $s_i \in S$ that measures x the path from s_i to x goes through the vertex w_i. We now define the new edge sets $E^- = ((x, y)|y$ is on the path from w_i to $x, x \in X$ and $(x, y) \in E)$ and $E^+ = \{(w_i, x_1), (x_1, x_2), \ldots, (x_{n-1}, x_n)\}$. We now transform G into $G_1 = (V_1, E_1)$ with $V_1 = V \cup \{t\}$ where t is a new vertex, and $E_1 = E \backslash E^- \cup (w_{i+1}, t) \cup (t, s) \cup E_2 \backslash (w_i, w_{i+1})$. An example of this transformation can be seen in Figure 3.

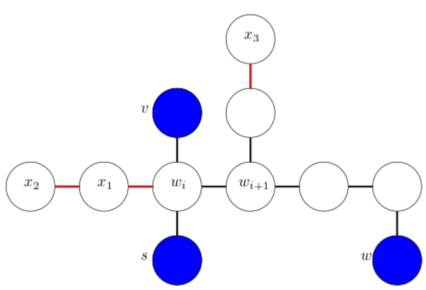

Figure 3: An example of transformation A with $k = 4$. As you can see x_1, x_2 and x_3 are only measured by s and v and the path from s and v to these vertices crosses w_i, hence these vertices are in X. Because of this the red edges are in E^- and the green edges in E^+. The blue vertices are sensors, and they are able to uniquely measure each vertex in the graph both before and after the transformation.

For every pair $x, y \in V$ we have some $s_i \in S$ that resolves them in G, meaning that $d(s_i, x) \neq d(s_i, y)$ and at least one of the conditions $d(s_i, x) < k + 1$ and $d(s_i, y) < k + 1$ hold true. We can then also find a sensor in G_1 that resolves x and y. We will now go through all the possible cases. If either x or y are in X then the pair is resolved as each vertex in X is uniquely measured. This is because each sensor that measured vertices in X in G is still able to measure this set of vertices, because the path of the sensors to the sensors still goes through w_i in G_1. The pair x and y is also resolved if both vertices are in X as each vertex in X is at a different distance from each sensor that measures them. As such we know that the pair x and y is resolved if one or both is in X, and we can assume that neither x nor y is a vertex in X for the other cases.
If neither path from s_i to x and y crosses the edge (w_i, w_{i+1}) in G then $d(s_i, x) \neq d(s_i, y)$ in G_1, so s_i resolves x and y.
If the paths from s_i to x and y both cross the edge (w_i, w_{i+1}) from w_i in G then $d(w_{i+1}, x) \neq d(w_{i+1}, y)$ in G_1 so $d(s, x) = d(s, w_{i+1}) + d(w_{i+1}, x) \neq d(s, w_{i+1}) + d(w_{i+1}, y) = d(s, y)$ holds in G_1. $d(s, w_i) = 2$ in G_1, $d(s_i, w_i) \geq 2$ in G and $d(w_i, x)$ and $d(w_i, y)$ are the same in both G and G_1. This tells us that s is at least as close to x and y in G_1 as s_i is in G. This proves that $d(s, x) \leq k$ and $d(s, y) \leq k$ which means that s measures both x and y in G_1 which allows us to conclude that s resolves x, y.
If the paths from s_i to x and y both cross the edge (w_i, w_{i+1}) from w_{i+1} in G then $d(w_i, x) \neq d(w_i, y)$ in G_1 then at least one of $d(s, x) = d(s, w_i) + d(w_i, x) \neq d(s, w_i) + d(w_i, y) = d(s, y)$ and $d(v, x) = d(v, w_i) + d(w_i, x) \neq d(v, w_i) + d(w_i, y) = d(v, y)$ holds. s and v are the closest sensors to w_i and since s_i crosses from w_{i+1} $s_i \neq v$ and $s_i \neq s$. Because s and v are the closest sensors to w_i we know that s and v are closer to w_i than s_i, and as such closer to x and y. Because of this we know that $d(s, x) \leq k$, $d(s, y) \leq k$, $d(v, x) \leq k$ and $d(v, y) \leq k$ in G, which since the paths from s and v to x and y are the same in G_1 as they are in G tells us that s and v both measure x and y allowing us to conclude that at least one of s and v resolves x and y.
If the path from s_i to x crosses the edge (w_i, w_{i+1}) from w_i in G but the path from s_i to y does not cross this edge then we know that $x \notin A^1(s)$ as the path from s to x does not cross the edge (w_i, w_{i+1}). This means that x is measured by another sensor that is not s, and as we have earlier proven that we can assume that $x \notin X$, there is some sensor s_j that measures x in G_1. As the path from s_j to y crosses the edge (s, t) in G_1, which tells us that $d(s_j, y) = d(s_j, s) + d(s, y)$. As such we require that either $d(s, x) \neq d(s, y)$ or $d(s_j, x) \neq d(s_j, y)$ because if neither holds then $d(s_j, x) = d(s_i, y) = d(s_j, s) + d(s, y) = d(s_j, s) + d(s, x)$, which means that the path from s_j to x crosses the vertex s, which also means that the path from s_j to x crosses the edge (w_i, w_{i+1}) in G which is a contradiction. We know that s_j measures x in G_1 and that s can measure y, hence this shows that either s or s_i resolves x and y.
If the path from s_i to x crosses the edge (w_i, w_{i+1}) from w_{i+1} in G but the path from s_i to y does not cross this edge. As x crosses the edge (w_i, w_{i+1}) from w_{i+1} in G it crosses the edge (s, t) in G_1, which tells us that $d(s_i, x) = d(s_i, s) + d(s, x)$. As such we require that either $d(s, x) \neq d(s, y)$ or $d(s_i, x) \neq d(s_i, y)$ because if neither holds then $d(s_i, y) = d(s_i, x) = d(s_i, s) + d(s, x) = d(s_i, s) + d(s, y)$, which means that the path from s_i to y crosses the vertex s, which also means that the path from s_i to y crosses the edge (w_i, w_{i+1}) in G which is a contradiction. We know that s_i measures y in G_1 and we have shown previously that s can measure x, hence this shows that either s or s_i resolves x and y. As such we have proven that S also resolves every pair of vertices in G_1, finishing the proof that transformation A works.
For transformation B we require that the assumptions for transformation A do not hold, meaning that there is no pair of strong sensor paths that starts at the same sensor and share vertices. Let $S_1 = \{s_1, s_2, \ldots, s_u\}$ be a set of sensors that all have sensor paths between them, w.l.o.g. let $d(s_1, s_2) = \max(d(s_i, s_j))$. Let t_1 be the vertex on the sensor path between s_1 and s_2 such that $(s_1, t_1) \in E$, and let $T = \{t_1, \ldots, t_u\}$ be the set of vertices that are uniquely measured by the same set $S_2 = \{s_1, s^*\}$ that uniquely measures t_1. There are only two sensors in this set as each sensor in this set has to be able to measure vertex t_1, which means that every sensor in the set except s_1 has a strong sensor path with s_1, which means that if there were more than two sensors in the set, transformation A would be possible. The t_i's are all connected, as if vertex t_i and vertex t_j are not connected, then there is some vertex v between t_i and

t_j that is also uniquely measured by some set S_3 such that there is some sensor $s_r \in S_3$ but $s_r \notin S_2$. This gives us that either $d(s_r, s_2) = d(s_r, t_i) + d(t_i, s_2) > d(s_1, t_i) + d(t_i, s_2) = d(s_1, s_2)$ or $d(s_r, s_2) = d(s_r, t_j) + d(t_j, s_2) > d(s_1, t_j) + d(t_j, s_2) = d(s_1, s_2)$, depending on which side s_2 is on. In both cases, this contradicts $d(s_1, s_2) = \max(d(s_i, s_j))$, which proves that the t_i's are all connected. We also know that s^* is connected to some t_i as otherwise the second vertex in the path from s^* to s_1 would have to be measured by some sensor s_r which would mean that the sensor paths between s^* and s_r and between s^* and s_1 would both be strong sensor paths, giving the contradiction that transformation A would be possible. We now transform $G = (V, E)$ to $G_1 = (V, (E \backslash E^-) \cup E^+))$ with $E^- = ((v, w)|(v, w) \in E$ and $v \in T$ and $w \notin S_2 \cup T)$ and $E^+ = ((v, s_1)|(v, t) \in E^-$ for some $t \in T)$. After this transformation S still resolves G. Let x be any vertex in V and let $S_4 \in S$ be the set of sensors that uniquely measures x in G. Then x is uniquely measured in G_1 by either S_4 or $S_4 \cup (s_1) \backslash (s^*)$. This is because if x is one of the t_i then the distance of every t_i to s_1 and s^* is unchanged, meaning that x is still uniquely measured by S_2, and if $x \neq t_i$ then every vertex measured by the same set of sensors as x in G_1 has the same distance to every sensor in this set except if s_1 is in this set, whose distance has decreased a constant number when compared to $d(s_1, x)$ or $d(s*, x)$ in G, because the changes of the edges means that the distance is unchanged except that the path from s_1 to x no longer contains the t_i vertices that the path from s_1 or s_2 to x used to have.

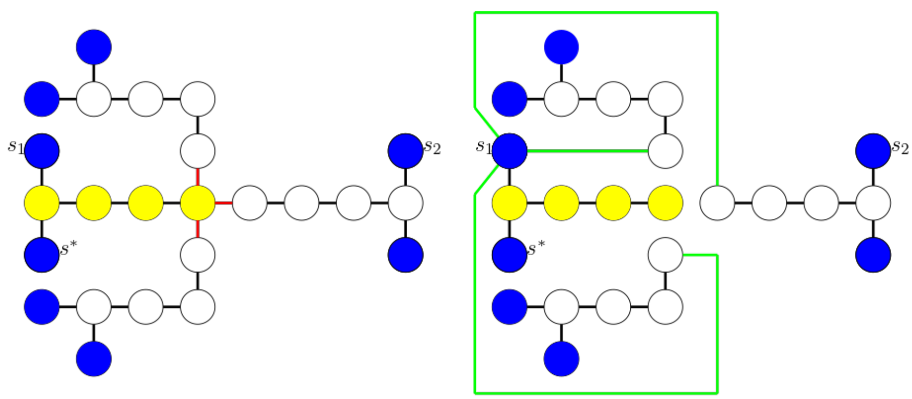

Figure 4: An example of transformation B with $k = 4$. The yellow vertices are in T, the red edges are in E^- and the green edges in E^+. The blue vertices are sensor and are able to measure the entire graph both before and after the transformation

As a result of using transformation B, the distance between s_1 and s_2 has decreased. Let u be the first vertex in the path from s_2 to s_1, u has to be measured by some sensor $s \in S_1 \backslash S_2$ because Lemma 3.8 implies u has to be uniquely measured by at least two sensors, and this second sensor cannot be in S_2 as otherwise this sensor would have a strong sensor path with both s_1 and s_2 which would contradict with the requirement to do transformation B. As such, after doing transformation B we still have three sensors that have non-disjoint sensor paths. This means that we are now either able to do transformation A, or we can repeat transformation B. Every time we do transformation B, the distance between the two sensors that have the longest sensor path between them decreases, so we will eventually be able to do transformation A if we repeat transformation B enough times. As transformation A increases the number of vertices in the graph, this proves the Lemma. □

Lemma 3.10. *Let $G \in \mathcal{G}_p^*$ and let S be a threshold-k resolving set of G. Then G has $p-1$ sensor paths.*

Proof. We renormalise the tree: we contract every sensor path to be a single edge and delete all vertices that are not sensors. This gives us $H = (V_2, E_2)$ with $V_2 = S$ and $(s_1, s_2) \in E_2$ if

$s_1, s_2 \in S$ and there is a sensor path between s_1 and s_2 in G.
We now prove that H is a tree. H is connected, as if there were any s_1, s_2 in H with no path between them, then there would also not be a path between s_1, s_2 in G_p^*, which contradicts with G being a tree. Then, assume H has cycle $(s_1, s_2, \dots s_n, s_1)$. Then the union of the sensor paths between these s_i's would form a cycle in G, as these sensor paths were proven to be disjoint in Lemma 3.9. But G having a cycle contradicts with G being a tree, so H cannot have a cycle. As such H is a tree as it is a connected graph without cycles. As H is a tree on n vertices, it has $p-1$ edges, which by the way edges in H were defined, means that there are $p-1$ sensor paths in G. □

We have now proven that any graph $G \in \mathcal{G}_p^*$ has $p-1$ sensor paths, so now we need to optimize the number of vertices that can be identified by a sensor path.

Lemma 3.11. *In any tree $G \in \mathcal{G}_p^*$, the maximal number of vertices that can be identified by a sensor path is $(k^2+k+1)/3$ if k mod $3 = 1$ and $(k^2+k)/3$ otherwise.*

Proof. By Lemma 3.8 and Lemma 3.9 we know that all sensor paths in G are disjoint Each vertex on a sensor path can be of at most degree 3, and the other vertices connected to the path can be at most of degree 2, because otherwise two of the vertices connected to the vertex for which this does not hold will have the same distance to every sensor, as illustrated by Figure 5.

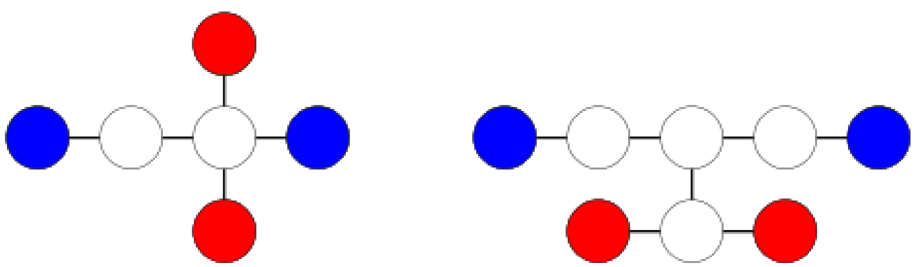

Figure 5: The blue vertices are the sensors and they are unable to differentiate between the red vertices in both graphs.

Denote the number of vertices between the sensors of the sensor path by m, so the distance between the sensors is $m+1$. Let one of the vertices on this sensor path have distance i to one sensor, then it has distance $m-i+1$ to the other sensor, so a leaf path of length $\min(k-i, k-(m-i+1))$ can be connected to this vertex, and all of the vertices on it will be identified by the sensor path. This holds for every vertex between the sensors, so the maximum number of vertices identified by the sensor path is $m + \sum_{i=1}^{m} min(k-i, k-(m-i+1))$ vertices. This can be simplified, namely if m is even we have:

$$\begin{aligned} m + \sum_{i=1}^{m} \min(k-i, k-(m-i+1)) &= m + 2\sum_{i=1}^{m/2} k-(m-i+1) \\ &= \frac{-3m^2}{4} + mk + \frac{m}{2} \end{aligned} \tag{1}$$

And if m is odd we have:

$$\begin{aligned} m + \sum_{i=1}^{m} \min(k-i, k-(m-i+1)) &= m + (k - \frac{m+1}{2}) + 2\sum_{i=1}^{(m-1)/2} (k-(m-i+1)) \\ &= \frac{-3m^2}{4} + mk + \frac{m}{2} + \frac{1}{4} \end{aligned} \tag{2}$$

Because these functions of m are both concave parabolas, to find the maximal value we simply differentiate with respect to m and set the result equal to 0. In both cases this results in

$m = (2k+1)/3$. Because the formulas are quadratic, the maximal integer value of the formula is found by rounding $(2k+1)/3$ to the closest integer and plugging this into the formula.
We can now calculate the maximal number of vertices that can be identified by a single sensor path by looking at the three cases separately.

1. If $k \bmod 3 = 0$, then the closest integer to $(2k+1)/3$ is $2k/3$. This value is always even, so we plug $m = 2k/3$ into $-3m^2/4 + mk + m/2$, which gives $(k^2+k)/3$.

2. If $k \bmod 3 = 1$, then the closest integer to $(2k+1)/3$ is $(2k+1)/3$. This value is always odd, so we plug $m = (2k+1)/3$ into $-3m^2/4+mk+m/2+1/4$, which gives $(k^2+k+1)/3$.

3. If $k \bmod 3 = 2$, then the closest integer to $(2k+1)/3$ is $(2k+2)/3$. This value is always even, so we plug $m = (2k+2)/3$ into $(-3m^2)/4 + mk + m/2$, which gives $(k^2+k)/3$.

□

We can now prove Theorem 3.2.

Proof of Theorem 3.2. any $G \in G_p^*$ has p sensors that can each uniquely measure $k+1$ vertices, namely their 1-attraction and the sensor itself. G also has $p-1$ sensor paths, each with $(k^2+k+1)/3$ vertices being identified by them if $k \bmod 3 = 1$ and $(k^2+k)/3$ otherwise. In total this means $|G| = (k+1)p + (p-1)(k^2+k+1)/3$ if $k \bmod 3 = 1$ and $|G| = (k+1)p + (p-1)(k^2+k)/3$ otherwise.

□

4 Improved lower bound for $k = 2$

From the previous section we know that a lower bound for the threshold-2 metric dimension of a tree on n vertices is $(n+2)/5$, which is a sharp lower bound if $(n+2)/5$ is an integer. In this section we will further develop this lower bound that also takes the structure of the tree into account. First, we find parts of trees that requires sensors at places that are not fully optimal (in other words, are used to measure less than 5 vertices). Then we determine how many vertices are measured by these additional sensors, leading to a lower bound that more accurately reflects the threshold-2 metric dimension for trees that are less optimal. For this section we exclude line graphs of length less than 7, which have a threshold-2 metric dimension of one if their length is less than 3 and of two otherwise.
Let $T = (V, E)$ be a tree with threshold-2 resolving set S, let L be the set of leaves in T and let $\text{Supp}(T)$ be the set of vertices that have a distance of less than four to two or more leaves that they are connected to by leaf paths. For readability, when we mention "a short leaf path" we are referring to one of these leaf paths with a length of less than four that starts in some $s \in \text{Supp}(T)$. Let $\text{Supp}_i(T)$, $i = 2, 3$ be subsets of $\text{Supp}(T)$ such that $v \in \text{Supp}_i(T)$ if and only if $v \in \text{Supp}(T)$ and the minimal length of the short leaf paths of v is of length i. Now, let $\text{Supp}_1(T)$ be the subset of $\text{Supp}(T)$ such that $s \in \text{Supp}_1(T)$ if s is adjacent to multiple leaves or adjacent to one leaf and connected by at least two short leaf paths to other leaves. In Figure 6 you can see examples of the vertices in $\text{Supp}_i(T)$. $\text{Supp}_1(T)$ does not include the vertices in $\text{Supp}(T)$ that are adjacent to only one leaf and connected by a single short leaf path to a different leaf, because in that case the tree requires one additional sensor that can be used to measure five vertices. This means this sensor measures the same number of vertices that a sensor in the optimal tree can measure, meaning that this does not affect the lower bound, hence this case is not included. Recall that L is the set of leaves in T. Let L_i, $i = 1, 2, 3$ be subsets of L such that $l \in L_i$ if $l \in L$ and there is a $s \in \text{Supp}_j(T), j = 1, 2, 3$ such that there is a short leaf path from s to l of length i.

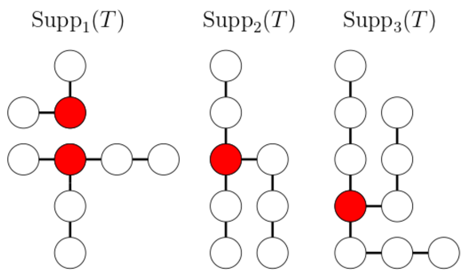

Figure 6: The red vertices are examples of the vertices in $\text{Supp}_i(T)$, $i = 1, 2, 3$. Note that there are two examples of $\text{Supp}_1(T)$ as there are two different requirements that vertices in $\text{Supp}_1(T)$ can have.

Using the following lemmas we prove that graphs with these short leaf paths graphs require extra sensors, followed by where we can optimally place these additional sensors, and lastly we will see how many vertices are measured by these sensors which gives us an improved lower bound for the threshold-2 metric dimension of an arbitrary tree.

Lemma 4.1. *Let T be a tree and let $s \in \text{Supp}(T)$ then for any threshold-2 resolving set S of T either all or all but one of the short leaf paths that start at s require a sensor.*

Proof. We argue by contradiction. Suppose that more than one short leaf path that starts at s has no sensor. Let v and w be the first vertices on two of the short leaf paths without a sensor starting in s. Since neither path has a sensor, for any sensor x we have $d(x, v) = d(x, s) + 1 = d(x, w)$ which contradicts with S being a threshold-2 resolving set of T. □

Lemma 4.2. *Let T be a tree and let $s \in \text{Supp}_i(T)$, then there exists a threshold-2 resolving set of T such that every short leaf path of s has a sensor except possibly one of length i.*

Proof. By Lemma 4.1 we know that for any threshold-2 resolving set S for each $s \in \text{Supp}_i(T)$ at least all but one of the short leaf paths starting in s need a sensor. Let $P_1, P_2, \ldots$ be the sets of vertices in the leaf paths starting in s. As can be seen in Figure 7, if there is a sensor in $P_i \backslash s$, then $P_i \backslash s$ is uniquely measured as long as $(T \backslash (P_1 \cup P_2 \cup \ldots)) \cup (s)$ is uniquely measured too. As such placing the sensors in the longest of these short leaf paths that s has measures the most possible vertices per sensor. As such we can say that all short leaf paths of s have a sensor except possibly one of length i, the length of the shortest leaf path that starts in s, depending on how the rest of the graph is uniquely measured. □

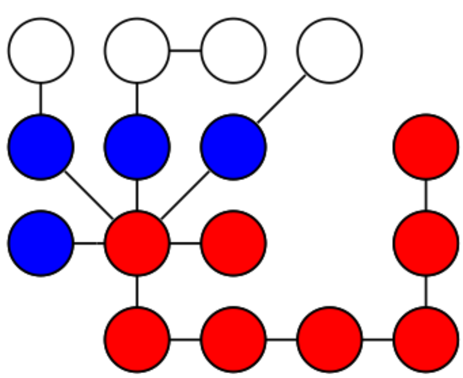

Figure 7: The sensors in the short leaf paths measure the entire short leaf path as long as the red vertices are uniquely measured.

Theorem 4.3. *A lower bound for the threshold-2 metric dimension of a tree on n vertices is*

$$Tmd_2(T) \geq \frac{n + 2 + 2(|L_3| - |\mathrm{Supp}_3(T)|) + 3(|L_2| - |\mathrm{Supp}_2(T)|) + 4|L_1| - 7|\mathrm{Supp}_1(T)|}{5}.$$

This lower bound is attainable if it is an integer.

Proof. We know by Lemma 4.2 that for every $s \in \mathrm{Supp}_i(T)$ with $i = 1, 2, 3$ all but possibly one short leaf path of length i needs a sensor. This means that every short leaf path ending in a vertex in L_3 needs a sensor except for one starting at every $\mathrm{Supp}_3(T)$, which means there is are $|L_3| - |\mathrm{Supp}_2(T)|$ sensors needed for these paths that uniquely measure three vertices each. For the same reason we need $|L_2| - |\mathrm{Supp}_2(T)|$ sensors that uniquely measure two vertices each and $|L_1| - |\mathrm{Supp}_1(T)|$ sensors that uniquely measure one vertex each. However, because the sensors in the short leaf paths of each $s \in \mathrm{Supp}_1(T)$ are able to measure the vertex in the short leaf path without a sensor, as the this short leaf path is a leaf adjacent to s. As such, these sensors are able to measure more vertices. This is not possible for any $s \in \mathrm{Supp}_2(T)$ or $s \in \mathrm{Supp}_3(T)$, because the sensors in the short leaf paths cannot reach every vertex in the short leaf path without the sensor. As such for each $s \in \mathrm{Supp}_1(T)$ the sensors in the other short leaf paths can measure additional vertices, which allows us to remove s, the vertex in the short leaf path, and one more vertex from the number the rest of the sensors need to measure by themselves, as these vertices can be uniquely measured with help from an extra sensor outside these short leaf paths. We remove this last vertex, because the neighbours of s that remain if we remove all short leaf paths of s could be uniquely measured as well without requiring additional sensors, as illustrated by Figure 8. Though it is possible that no additional vertex is saved by doing this, if for example s only has neighbours that are also in $\mathrm{Supp}_1(T)$, we remove this vertex from the total as it is possible to save this vertex as can be seen in Figure 8.

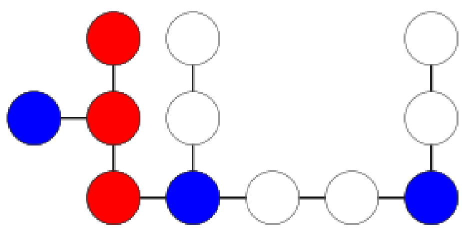

Figure 8: An example of why we remove three vertices per vertex in $\mathrm{Supp}_1(T)$. The blue vertices are sensors, and the red vertices are uniquely measured thanks to the left-most sensor in the short leaf path.

As such we can remove three from the number of vertices that needs to be measured for each vertex in $\mathrm{Supp}_1(T)$. We now know that the graph requires $|L_3| - |\mathrm{Supp}_3(T)| + |L_2| - |\mathrm{Supp}_2(T)| + |L_1| - |\mathrm{Supp}_1(T)|$ additional sensors that are able to measure up to $3|| - 3|\mathrm{Supp}_3(T)| + 2|L_2| - 2|\mathrm{Supp}_2(T)| + 1|L_1| + 2|\mathrm{Supp}_1(T)|$ vertices uniquely, as long as the rest of the graph is measured uniquely. For the remaining $n - (3|L_3| - 3|\mathrm{Supp}_3(T)| + 2|L_2| - 2|\mathrm{Supp}_2(T)| + |L_1| - 2|\mathrm{Supp}_1(T)|)$ vertices the original lower bound still applies, which allows us to calculate the final lower bound. An example of this lower bound can be seen in Figure 9.

$$\begin{aligned} Tmd_2(T) \geq\ & \frac{n - (3|L_3| - 3|\mathrm{Supp}_3(T)| + 2|L_2| - 2|\mathrm{Supp}_2(T)| + |L_1| + 2|\mathrm{Supp}_1(T)|) + 2}{5} \\ & + |L_3| - |\mathrm{Supp}_3(T)| + |L_2| - |\mathrm{Supp}_2(T)| + |L_1| - |\mathrm{Supp}_1(T)| \\ =\ & \frac{n + 2(|L_3| - |\mathrm{Supp}_3(T)|) + 3(|L_2| - |\mathrm{Supp}_2(T)|) + 4|L_1| - 7|\mathrm{Supp}_1(T)|}{5} \end{aligned} \tag{3}$$

□

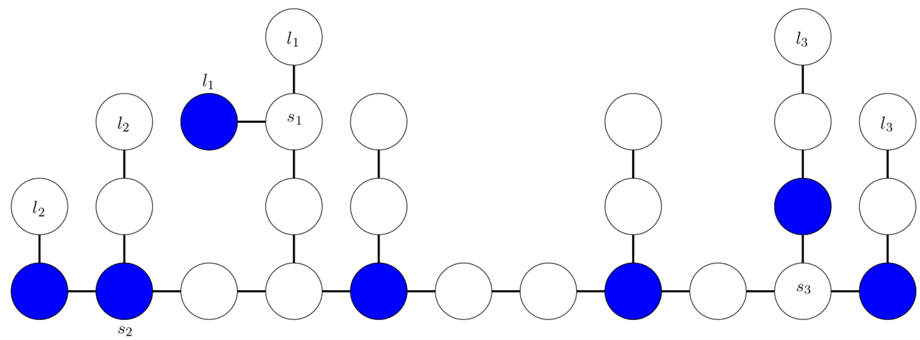

Figure 9: An example where the lower bound of Theorem 4.3 is sharp and each term in the formula for the lower bound is non-zero. Each $l_i \in L_i$ and each $s_i \in \text{Supp}_i(T)$.

References

[1] M. Blidia, M. Chellali, F. Maffray, J. Moncel, and A. Semri. Locating-domination and identifying codes in trees. *The Australasian Journal of Combinatorics*, 39:219–232, 2007.

[2] P. Buczkowski, G. Chartrand, C. Poisson, and P. Zhang. On k-dimensional graphs and their bases. *Periodica Mathematica Hungarica*, 46(1):9–15, 2003.

[3] J. Cáceres, C. Hernando, M. Mora, I. M. Pelayo, M. L. Puertas, C. Seara, and D. R. Wood. On the metric dimension of cartesian products of graphs. *SIAM journal on discrete mathematics*, 21(2):423–441, 2007.

[4] F. Harary and R. A. Melter. On the metric dimension of a graph. *Ars combin*, 2(191-195):1, 1976.

[5] M. Hauptmann, R. Schmied, and C. Viehmann. Approximation complexity of metric dimension problem. *Journal of Discrete Algorithms*, 14:214–222, 2012.

[6] B. Hulme, A. Shiver, and P. Slater. A boolean algebraic analysis of fire protection. In *North-Holland mathematics studies*, volume 95, pages 215–227. Elsevier, 1984.

[7] M. Imran, A. Q. Baig, S. A. U. H. Bokhary, and I. Javaid. On the metric dimension of circulant graphs. *Applied mathematics letters*, 25(3):320–325, 2012.

[8] M. Johnson. Structure-activity maps for visualizing the graph variables arising in drug design. *Journal of Biopharmaceutical Statistics*, 3(2):203–236, 1993.

[9] S. Khuller, B. Raghavachari, and A. Rosenfeld. Localization in graphs, technical report cs-tr- 3326. *University of Maryland at College Park*, 1994.

[10] S. Khuller, B. Raghavachari, and A. Rosenfeld. Landmarks in graphs. *Discrete applied mathematics*, 70(3):217–229, 1996.

[11] J. Komjáthy and G. Ódor. The metric dimension of critical galton-watson trees and linear preferential attachment trees. *arXiv preprint arXiv:2002.08503*, 2020.

[12] S. Mashkaria, G. Ódor, and P. Thiran. On the robustness of the metric dimension to adding a single edge. *arXiv:2010.11023*, 2020.

[13] D. Mitsche and J. Rué. On the limiting distribution of the metric dimension for random forests. *European Journal of Combinatorics*, 49:68–89, 2015.

[14] O. R. Oellermann and J. Peters-Fransen. The strong metric dimension of graphs and digraphs. *Discrete Applied Mathematics*, 155(3):356–364, 2007.

[15] F. Okamoto, B. Phinezy, and P. Zhang. The local metric dimension of a graph. *Mathematica Bohemica*, 135(3):239–255, 2010.

[16] B. Shanmukha, B. Sooryanarayana, and K. Harinath. Metric dimension of wheels. *Far East J. Appl. Math*, 8(3):217–229, 2002.

[17] P. J. Slater. Leaves of trees. *Congr. Numer*, 14(549-559):37, 1975.

[18] P. J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987.

[19] P. J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci*, 22(4):445–455, 1988.

[20] P. J. Slater and J. Louis Sewell. A sharp lower bound for locating-dominating sets in trees. *Australasian Journal of Combinatorics*, 60(2):136–149, 2014.

[21] B. Sooryanarayana, S. Kunikullaya, and N. Swamy. k-metric dimension of a graph. *International Journal of Mathematical Combinatorics*, 4, 2016.

[22] B. Sooryanarayana, K. Shreedhar, and N. Narahari. On the metric dimension of the total graph of a graph. *Notes on Number Theory and Discrete Mathematics*, 22(4):82–95, 2016.

[23] S. Zejnilović, J. Gomes, and B. Sinopoli. Network observability and localization of the source of diffusion based on a subset of nodes. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 847–852. IEEE, 2013.