Alternative parameterizations of Metric Dimension

Gregory Gutin[1], M. S. Ramanujan[2], Felix Reidl[1], and Magnus Wahlström [1]

[1]Royal Holloway, University of London, TW20 0EX, UK
[2]University of Warwick, CV4 7AL, UK

Abstract

A set of vertices W in a graph G is called *resolving* if for any two distinct $x, y \in V(G)$, there is $v \in W$ such that $\mathrm{d}_G(v, x) \neq \mathrm{d}_G(v, y)$, where $\mathrm{d}_G(u, v)$ denotes the length of a shortest path between u and v in the graph G. The metric dimension $\mathrm{md}(G)$ of G is the minimum cardinality of a resolving set. The Metric Dimension problem, i.e. deciding whether $\mathrm{md}(G) \leqslant k$, is NP-complete even for interval graphs (Foucaud *et al.*, 2017). We study Metric Dimension (for arbitrary graphs) from the lens of parameterized complexity. The problem parameterized by k was proved to be $\mathsf{W}[2]$-hard by Hartung and Nichterlein (2013) and we study the dual parameterization, i.e., the problem of whether $\mathrm{md}(G) \leqslant n - k$, where n is the order of G. We prove that the dual parameterization admits (a) a kernel with at most $6(k+1)$ vertices and (b) a randomized algorithm of runtime $O^*(4^{k+o(k)})$. Hartung and Nichterlein (2013) also observed that Metric Dimension is fixed-parameter tractable when parameterized by the vertex cover number $vc(G)$ of the input graph. We complement this observation by showing that it does not admit a polynomial kernel even when parameterized by $vc(G) + k$, unless NP $\subseteq$ coNP/poly. Our reduction also gives evidence for non-existence of polynomial Turing kernels. We also prove that Metric Dimension parameterized by bandwidth or cutwidth does not admit a polynomial kernel, unless NP $\subseteq$ coNP/poly. Finally, using Eppstein's results (2015) we show that Metric Dimension parameterized by max-leaf number does admit a polynomial kernel.

1 Introduction

A set of vertices W of a graph G is a *resolving* set for G if for any two distinct $x, y \in V(G)$, there is $v \in W$ such that $\mathrm{d}_G(v, x) \neq \mathrm{d}_G(v, y)$, where $\mathrm{d}_G(u, v)$ denotes the length of a shortest path between u and v in the graph G. The *metric dimension* $\mathrm{md}(G)$ of G is the minimum cardinality of a resolving set for G. The metric dimension of graphs was introduced independently by Slater [39] and Harary and Melter [27]. Metric Dimension as a computational problem was first mentioned in the literature by Garey and Johnson [22] and its decision version is defined as follows.

Metric Dimension

Input:	A graph G and an integer k.
Problem:	Does G have a resolving set of size at most k?

Garey and Johnson [22] proved this problem to be NP-complete in general. Their proof was never published, a reduction from 3SAT was provided by Khuller *et al.* [33]. Diaz *et al.* [12] showed that the problem is NP-complete even when restricted to planar graphs of bounded degree but that it is solvable in polynomial time on the class of outer-planar graphs.

Prior to this, not much was known about the computational complexity of this problem except that it is polynomial-time solvable on trees (see [39, 33]), although there are several results proving combinatorial bounds on the metric dimension of various graph classes [7]. Subsequently, Epstein et al. [16] showed that this problem is NP-complete on split graphs, bipartite and co-bipartite graphs. They also showed that the *weighted version* of Metric Dimension can be solved in polynomial time on paths, trees, cycles, co-graphs and trees augmented with k edges for a fixed k. Hoffmann and Wanke [30] extended the tractability results to a subclass of unit disk graphs, while Foucaud *et al.* [20] showed that this problem is NP-complete on interval graphs.

The parameterized complexity of Metric Dimension under the standard parameterization—the metric dimension of the input graph—was open until 2012, when Hartung and Nichterlein [28] proved that it is W[2]-hard. Foucaud *et al.* [20] showed the problem becomes fixed-parameter tractable when restricted to interval graphs. The parameterized complexity of Metric Dimension on graphs of bounded treewidth is currently unresolved (the question of whether it is polynomial-time solvable on graphs of treewidth 2 is still open), however, Belmonte *et al.* [3] proved that it is FPT when parameterized by the tree*length*[1] plus the solution size. In a different line of work, Eppstein [15] showed that Metric Dimension is FPT when parameterized by the max-leaf number of the input graph alone.

In this paper we initiate the study of the parametric dual of Metric Dimension. To avoid confusion, we will use k to denote the (standard) parameter and phrase the parameterized dual as follows:

Saving Landmarks

Input: A graph G and an integer k.
Problem: Does G have a resolving set of size at most $n - k$?

We call a set T of vertices of G a *co-resolving set* if $V(G) \setminus T$ is a resolving set of G. Clearly, an instance of Saving Landmarks is positive if and only if there is a co-resolving set T of size at least k.

This choice of parameterization is informed by previous studies of the parametric dual (see e.g. [2, 9, 25, 26, 37]): problems that are hard with respect to the standard parameter often admit FPT-algorithms or even polynomial kernels under the dual parameter. A classic example is the Independent Set problem which is W[1]-hard while its dual, the Vertex Cover problem is among the earliest problems shown to be in FPT and even admits a linear vertex kernel.

We add yet another entry to the list of hard problems with tractable duals by showing that Saving Landmarks admits a linear vertex kernel and a single-exponential FPT algorithm. Concretely, we prove the following two results.

Theorem 1. Saving Landmarks *admits a kernel with at most* $6(k+1)$ *vertices.*

Theorem 2. Saving Landmarks *can be solved by an* $O^*(4^{k+o(k)})$*-time algorithm.*

We will also consider two variants of metric dimension studied in [6, 17, 18, 31, 40]. A vertex set R in a graph G is an *adjacency-resolving set* if all vertices in $V(G) \setminus R$ have a distinct neighborhood within R. A *locating-dominating set* in G is an adjacency-resolving set, which is also a dominating set of G. Observe that a adjacency-resolving set and locating-dominating set are resolving sets. The problems Adjacency Dimension (Adjacency Domination Dimension, respectively) consist of deciding whether a connected graph G has an adjacency-resolving set (a locating-dominating set) of size at most k. Similar to Metric Dimension, Adjacency Dimension and Adjacency Domination Dimension are NP-complete [6, 18]. We also study the Metric Dimension problem from the kernelization perspective when parameterized by the vertex cover number of the input graph. As Hartung and Nichterlein observed [28], the parameterization of Metric Dimension by the vertex cover number of the input graph (denoted Metric Dimension[VC]) can be easily seen to be in FPT. It is therefore natural to ask whether this structural parameterization allows a polynomial kernel in general graphs, a question we answer in the negative. In fact, we show that not only is the problem unlikely to admit a polynomial kernel with the vertex cover as the parameter, even adding the size of the solution (the metric dimension of the graph) to the parameter is unlikely to be helpful in this regard. Specifically, we prove the following result:

Theorem 3. Metric Dimension[$\mathrm{VC}+k$], Adjacency Dimension[$\mathrm{VC}+k$] *and* Adjacency Domination Dimension[$\mathrm{VC}+k$] *do not admit polynomial kernels unless* $\mathrm{NP} \subseteq \mathrm{coNP/poly}$.

The reduction used in the proof of Theorem 3 also gives evidence for non-existence of polynomial Turing kernels, generalisations of (ordinary) kernels, informally introduced in the end of Section 4.1.

We further consider parameterizations by the bandwidth and cutwidth of the input graph. Bandwidth and cutwidth are classic graph layout parameters with numerous practical applications. The bandwidth of a graph is defined as the minimum possible *stretch* of a linear ordering of its vertices where the stretch of an ordering is the maximum, taken over all edges of G, of the distance between the endpoints of an

[1]The *length* of a tree decomposition is the maximum diameter of the bags in this tree-decomposition and the *treelength* of a graph is the minimum length over all tree decompositions.

edge in the ordering. The cutwidth of a graph on the other hand is the minimum possible *width* of a linear ordering of its vertices where the width of an ordering is the maximum, taken over all prefixes of this ordering, of the number of edges with exactly one endpoint in a prefix. For survey papers on these and related parameters, see, e.g., [8, 11, 38]. We obtain a negative result regarding the existence of polynomial kernels for Metric Dimension parameterized by either of these two parameters.

Theorem 4. *Unless* NP $\subseteq$ coNP/poly*, neither* Metric Dimension[**bw**] *nor* Metric Dimension[**cw**] *has a polynomial kernel, even when restricted to connected instances.*

To the best of our knowledge, the parameterized complexities of Metric Dimension[**bw**] and Metric Dimension[**cw**] are not known yet. It would be interesting to determine them.

Finally, we show that Eppstein's [15] result on the parameterized tractabiliy of Metric Dimension parameterized by the max-leaf number implies the existence of a polynomial kernel for the same parameter.

Theorem 5. Metric Dimension *parameterized by the max-leaf number has a polynomial kernel.*

2 Preliminaries

For a graph G we denote by d_G the standard distance-metric where $\mathrm{d}_G(u, v)$ is the length of a shortest path between vertices $u, v \in V(G)$. We denote by $N_G(v)$ and $N_G[v]$ the open and closed neighbourhood of a vertex. We omit the subscript G if clear from the context in all these notations. As customary, the number of vertices of a graph G under consideration will be denoted by n.

Two vertices u, v are *true twins* if $N_G[u] = N_G[v]$ (implying that $uv \in E(G)$) and they are *false twins* if $N_G(u) = N_G(v)$ (implying that $uv \notin E(G)$). A *twin class* is a maximal vertex set in G in which all vertices are pairwise true twins or in which all vertices are pairwise false twins.

A vertex set $S \subseteq V(G)$ *resolves* a set $T \subseteq V(G)$ if for every pair of distinct vertices $u, v \in T$ there exists at least one vertex $w \in S$ such that $\mathrm{d}_G(u, w) \neq \mathrm{d}_G(v, w)$. We will also say that a pair u, v is *resolved* by S if the above holds and further that sets A, B are *distinguished* by S if every pair $u \in A$, $v \in B$ is resolved by S. A vertex subset $S \subseteq V(G)$ is a *resolving set* of G if S resolves $V(G)$. We call the members of such a set S *landmarks*.

A vertex set $S \subseteq V(G)$ *resolves* a set $T \subseteq V(G)$ *via adjacencies* if for every pair of distinct vertices $u, v \in T$ there exists at least one vertex $w \in S$ such that $uw \in E(G)$ but $vw \notin E(G)$ or vice versa. In other words, all vertices in T have a distinct neighbourhood in S. We call such a set S a *adjacency-resolving* set of T. If a set S dominates *and* adjacency-resolves $V(G)$ then we call S a *locating-dominating set* of G.

Parameterized complexity is a two dimensional framework for studying the computational complexity of a problem. One dimension is the input size n and the other is a parameter k. A problem is said to be *fixed parameter tractable* (FPT) or in the class FPT, if it can be solved in time $f(k) \cdot n^{O(1)}$ for some computable function f. We refer to the books of Cygan *et al.* [10] and Downey and Fellows [14] for detailed introductions to parameterized complexity.

Kernelization offers a mathematically rigorous way of analysing and comparing preprocessing algorithms for NP-hard problems in general and for parameterized problems in particular. A *generalized kernel*[2] *of size* $g(k)$ from a parameterized problem Π_1 to a parameterized problem Π_2 is a polynomial time algorithm that takes as input an instance (I, k) of Π_1 (where k is the parameter for Π_1) and outputs an instance (I', k') of Π_2 (where k is the parameter for Π_2) such that (I, k) is a yes-instance of Π_1 if and only if (I', k') is a yes-instance of Π_2 and $|I'| + k' \leqslant g(k)$. The notion of "effective" preprocessing is captured by requiring the function g to be polynomially bounded, in which case the generalized kernel is called a *polynomial generalized kernel.* When Π_1 and Π_2 coincide, a generalized kernel from Π_1 to Π_2 is called a *kernel* of Π_1. The reader is referred to Cygan *et al.* [10], Downey and Fellows [14], Fomin *et al.* [19] and the surveys [34, 35] for a comprehensive introduction to the topic of kernelization.

Definition 6 (Pruned graph). *For a graph G we define the* pruned graph $\widetilde{G}$ *as the graph obtained (up to isomorphism) from G by iteratively removing vertices from twin-classes of size three or larger. We say that a graph is* pruned *if $G = \widetilde{G}$.*

[2] It is called a bikernel (as it involves two problems) in Alon et al. [1] and some other papers.

The following observation simply follows from the fact that among a twin class U in G, all but one vertex of U must be contained in any resolving set. Thus, we may assume that all vertices of $V(G) \setminus V(\tilde{G})$ are in a resolving set and obtain the following:

Observation 7. *A graph G has a resolving set of size k if and only if the pruned graph $\tilde{G}$ has a resolving set of size $k - (|V(G)| - |V(\tilde{G})|)$.*

Consequently, we call an instance (G, k) of Metric Dimension or Saving Landmarks *reduced* if G is pruned.

3 Standard Parameterization for Saving Landmarks

We present two positive results in this section, namely, that Saving Landmarks admits a linear vertex kernel and a single-exponential FPT algorithm.

We begin by describing the kernel. Assume in the following that the input instance (G, k) is pruned as per Observation 7. This will be the only reduction rule. Now we will prove the first result of this section.

Theorem 1. Saving Landmarks *admits a kernel with at most $6(k+1)$ vertices.*

Proof. Recall the instance (G, k) is pruned, i.e. every twin class has at most two vertices. We may assume that $k \geqslant 1$ and G is of order at least 12. Let H be a graph obtained from G by deleting a vertex in each twin class of size 2.

Recall that a locating-dominating set is a resolving set. Garijo et al. [23] proved that every graph of order $p \geqslant 4$ without twin vertices has a locating-dominating set of size at most $\lfloor \frac{2}{3}p \rfloor + 1$, which can be computed in polynomial time. Thus, $\mathrm{md}(H) \leqslant \lfloor \frac{2}{3}t \rfloor + 1$, where t is the order of H. Observe that adding a twin vertex to H may increase the metric dimension of H by at most 1. To obtain G from H, we add $n - t$ vertices and thus

$$\mathrm{md}(G) \leqslant \frac{2t}{3} + 1 + n - t \leqslant \frac{5n}{6} + 1$$

since $t \geqslant n/2$. If $n - k \geqslant \frac{5n}{6} + 1$, (G, k) is a positive instance. Otherwise, $n - k < \frac{5n}{6} + 1$ and $\frac{n}{6} \leqslant k + 1$ implying that $n \leqslant 6(k+1)$. □

Let us now move on to the second result, the single-exponential FPT algorithm. To better describe the algorithm, let us introduce a definition. For a set $X \subseteq V(G)$, we say that two vertices u and v are *X-equidistant* if $\mathrm{d}(u, w) = \mathrm{d}(v, w)$ for every $w \in X$, i.e., if X fails to resolve u and v. Note that this induces an equivalence relation over $V(G)$.

The main ingredient will be the fact that a solution to Saving Landmarks is witnessed already by a small resolving set.

Lemma 8. *Let T be a co-resolving set of a graph G. Then there exists a set $S \subseteq V(G) \setminus T$ of size at most $|T|$ that resolves T.*

Proof. We construct S iteratively as follows. Begin with $S = \emptyset$ and pick a pair u, v of S-equidistant vertices in T. Since $V(G) \setminus T$ resolves T, there exists a vertex $w \in V(G) \setminus T$ that distinguishes u and v. Add w to S and partition T into equivalence classes of S-equidistant vertices. Pick a new pair of S-equidistant vertices from one of the classes and repeat. Observe that the number of equivalence classes increases with every addition to S, hence after at most $|T|$ steps the set S resolves every pair in T. □

We are now ready to complete the proof of Theorem 2.

Theorem 2. Saving Landmarks *can be solved by an $O^*(4^{k+o(k)})$-time algorithm.*

Proof. We may assume that $n \geqslant 2k$. Let us first show the following claim: there exists a co-resolving set T of G of size at least k if and only if there is a partition $V(G) = R \uplus B$ of $V(G)$ such that R contains at least k equivalence classes of B-equidistant vertices. Suppose that there exists a co-resolving set T of G of size at least k. Then by Lemma 8, there is a set $S \subseteq V(G) \setminus T$ of size at most $|T|$ that resolves T. Let $T \subseteq R$ and $S \subseteq B$ for a partition $V(G) = R \uplus B$. Then B resolves T and hence R has at least $|T| \geqslant k$ equivalence classes of B-equidistant vertices. Suppose now that there is a partition $R \uplus B$ of $V(G)$ such that R has at least k equivalence classes of B-equidistant vertices. Choose a vertex from each equivalence class to form a set T. Then T is a co-resolving set of G.

The above claim leads to the following randomised algorithm. Choose a natural number N defined later on. Repeat N times the following: uniformly at random partition the vertices of G into B and R, and derive equivalence classes of B-equidistant vertices in R. If the number of classes is at least k, then conclude that (G, k) is a yes-instance and stop. If after all repetitions we do not conclude that (G, k) is a yes-instance, then we conclude that (G, k) is a no-instance.

Let us argue about the success probability of the randomised algorithm and how to choose N. The probability that for a random partition the vertices of G as $V(G) = R \uplus B$, R has at least k equivalence classes of B-equidistant vertices is at least the probability that $T \subseteq R$ and $S \subseteq B$, where sets T, S are as in Lemma 8 and $|T| = k$. Consequently, the probability of this event is at least $2^{-|T|-|S|} \geqslant 4^{-k}$. Thus, $N = 4^k$ is enough to achieve a constant success probability, see e.g. Chapter 5 of [10].

Observe that every loop in the randomised algorithm can be executed in polynomial time. Thus, the running time of the randomised algorithm is $O^*(4^k)$. The randomised algorithm can be derandomised using the standard (n, k)-universal set technique (see e.g. Section 5.6 of [10]), which brings an additional $o(k)$ to the exponent of the running time. $\square$

4 Structural parameterizations for Metric Dimension

As Hartung and Nichterlein observed [28], Metric Dimension[VC] is trivially FPT by virtue of Observation 7: After reducing the size of each twin class to at most two, any instance with a vertex cover X of size t will have at most $t + 2^{t+1}$ vertices. In sparse graph classes, the twin reduction even results in a polynomial-size kernel: in classes of bounded expansion (e.g. planar graphs or graphs excluding a topological minor), the number of twin classes in $V(G) \setminus X$ is bounded linearly in t and in nowhere dense classes by $t^{1+o(1)}$ (*cf.* Lemma 4.3 and Corollary 4.4 in [21]). Furthermore, if the input graphs stem from a d-degenerate class, the number of twin-classes and thus the number of vertices in the kernel is bounded by $O(t^{d+1})$; a fact that follows easily from the observation that in such a class at most dt vertices in the independent set can have degree more than d.

It is therefore natural to ask whether this structural parameterization allows a polynomial kernel in general graphs, a question we answer in the negative. In fact, in Section 4.1 we will show that adding to VC the size of the solution k (the metric dimension of the graph) is unlikely to be helpful in this regard. (In what follows, Metric Dimension[VC+k] is Metric Dimension parameterized by VC+k.)

Note that for any parameter w which is bounded by the vertex cover number of the graph, such as *treedepth*, *pathwidth* or *treewidth*, Theorem 3 and Corollary 9 proved in Subsection 4.1 provide evidence against polynomial kernels and polynomial Turing kernels, parameterized by w as well. However, there remain more restrictive structural parameters that are not bounded by the vertex cover number, such as *bandwidth* or *cutwidth*. In Section 4.2, we show that unless NP $\subseteq$ coNP/poly, Metric Dimension admits no polynomial kernel even under these parameters.

In Section 4.3, using results of Eppstein [15] we show that Metric Dimension admits a polynomial kernel when parameterized by maximum leaf number.

4.1 Parameterization by VC + k

In the Hitting Set problem, the input is a set system $\mathcal{F}$ over a universe U and a non-negative integer ℓ and the goal is to decide whether there is a set $X \subseteq U$ of size at most ℓ such that $X \cap R \neq \emptyset$ for every $R \in \mathcal{F}$. We refer to the sets in $\mathcal{F}$ as hyperedges and always assume without loss of generality that $\mathcal{F}$ has no copies of hyperedges as we may delete them without changing the output.

Theorem 3. Metric Dimension[VC + k], Adjacency Dimension[VC + k] *and* Adjacency Domination Dimension[VC + k] *do not admit polynomial kernels unless* NP $\subseteq$ coNP/poly.

Proof. First we will prove the theorem for Metric Dimension[VC + k] and then observe that the proof implies the theorem for the other two problems. We do this by using the notion of Polynomial Parameter Transformation (PPT) which is a reduction that can be used to propagate polynomial kernelization hardness results. A PPT from a problem Π_1 to a problem Π_2 is a polynomial time algorithm that takes as input an instance (x_1, k_1) of Π_1 and outputs an equivalent instance (x_2, k_2) of Π_2 such that $k_2 = k_1^{O(1)}$. Consequently, if Π_1 is an NP-hard language and Π_2 is in NP, then a PPT from Π_1 to Π_2

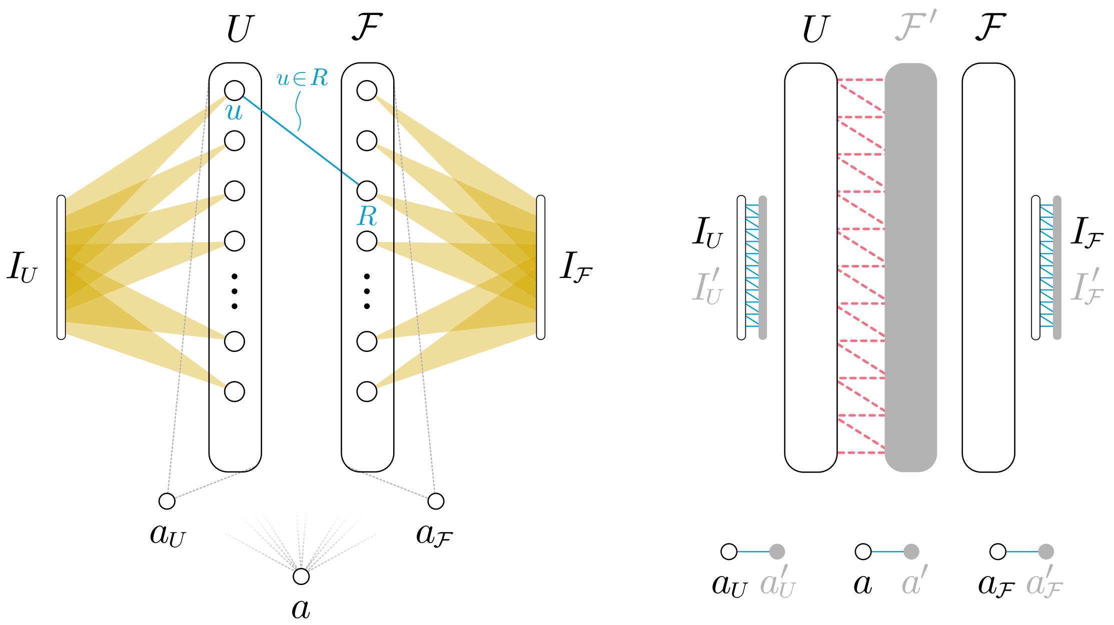

Figure 1: A schematic of the reduction from a Hitting Set[$|U| + \ell$] instance $(U, \mathcal{F}, \ell)$ to a Metric Dimension[VC $+ k$] instance. The drawing on the left shows the basic construction and the drawing on the right shows the addition of false and true twins (an edge between a white set and its grey counterpart indicates that they are true twins, the absence of an edge that they are false twins). Note that the construction removes edges between the set U and $\mathcal{F}'$ (indicated by the dashed red lines).

and a polynomial kernel for Π_2 together imply a polynomial kernel for Π_1. Since Hitting Set[$|U| + \ell$], i.e. parameterized by the size of the universe and the solution size does not admit a polynomial kernel unless NP $\subseteq$ coNP/poly [13], it is sufficient for us to provide a polynomial parameter transformation from Hitting Set[$|U| + \ell$] to Metric Dimension[VC $+ k$].

Let $(U, \mathcal{F}, \ell)$ be a Hitting Set instance with $n = |U|$ and $m = |\mathcal{F}|$. We construct a graph G as follows (*cf.* Figure 1):

1. Begin with the usual bipartite representation of $U, \mathcal{F}$, i.e., create a bipartite graph $G = (U \cup \mathcal{F}, E)$ where for vertices $u \in U$ and $R \in \mathcal{F}$ we have $uR \in E$ if and only if $u \in R$;
2. add $t_U := 2\lceil \log_2 n \rceil$ vertices I_U to the graph and edges between U, I_U so that every vertex in U has a unique neighbourhood in I_U of size $t_U/2$;
3. add $t_\mathcal{F} := 2\lceil \log_2 m \rceil$ vertices $I_\mathcal{F}$ to the graph and edges between $\mathcal{F}$ and $I_\mathcal{F}$ such that every vertex in $\mathcal{F}$ has a unique neighbourhood in $I_\mathcal{F}$ of size $t_\mathcal{F}/2$;
4. add three vertices $a_U, a, a_\mathcal{F}$ where $N(a_U) = U$, $N(a_\mathcal{F}) = \mathcal{F}$, and
   $N(a) = U \cup \mathcal{F} \cup I_U \cup I_\mathcal{F} \cup \{a_U, a_\mathcal{F}\}$, i.e., a is an apex vertex;
5. create true twin copies $I'_U, I'_\mathcal{F}, a'_U, a', a'_\mathcal{F}$ of $I_U, I_\mathcal{F}, a_U, a, a_\mathcal{F}$, and finally
6. create false twin copies $\mathcal{F}'$ of $\mathcal{F}$ but remove all edges from $\mathcal{F}'$ to U afterwards. For simplicity, we will label the copy of any vertex $R \in \mathcal{F}$ by $R' \in \mathcal{F}'$.

In summary, except for being adjacent to the apex vertices a, a', vertices in $I_U \cup I'_U$ neighbour only their twin copy and vertices in U; vertices in $I_\mathcal{F} \cup I'_\mathcal{F}$ neighbour only their twin copy and vertices in $\mathcal{F} \cup \mathcal{F}'$; the edges between $U, \mathcal{F}$ encode the hitting set instance; and the pairs $\{a_U, a'_U\}$ and $\{a_\mathcal{F}, a'_\mathcal{F}\}$ are apices for the sets U and $\mathcal{F} \cup \mathcal{F}'$, respectively. Our construction concludes with (G, X, k) as the Metric Dimension[VC $+ k$] instance with the vertex cover $X := V(G) \setminus (\mathcal{F} \cup \mathcal{F}')$ and solution size $k := \ell + t_U + t_\mathcal{F} + 3$.

Let us check that the parameter $|X| + k$ is polynomial in n and ℓ:

$$\begin{aligned} |X| + k &= (2t_U + 2t_{\mathcal{F}} + n + 6) + (\ell + t_U + t_{\mathcal{F}} + 3) \\ &= 3t_U + 3t_{\mathcal{F}} + n + \ell + 9 \\ &= O(\log m + \log n + n + \ell) = O(n + \ell), \end{aligned}$$

where we used that $m \leqslant 2^n$ (there are no copies of hyperedges).

Since the construction is clearly a polynomial time construction, it remains to prove that $(U, \mathcal{F}, \ell)$ is a yes-instance if and only if so is (G, X, k). Let us first show that if $(U, \mathcal{F}, \ell)$ is a yes-instance then so is (G, X, k). Suppose that $H \subseteq U$ is a hitting set for $\mathcal{F}$ of size ℓ. We construct a landmark set S for G by setting $S = H \cup I_U \cup I_{\mathcal{F}} \cup \{a_U, a, a_{\mathcal{F}}\}$. Let us now argue that is indeed a resolving set. It suffices to show that every pair $u, v \in U \cup I'_U \cup \mathcal{F} \cup \mathcal{F}' \cup I'_{\mathcal{F}} \cup \{a'_U, a', a'_{\mathcal{F}}\}$ of distinct vertices not in S is distinguished by a vertex in S. Observe that U is distinguished from $V(G) \setminus U$ by $a_U \cup I_U$ and $u, v \in U$ are distinguished by a vertex in I_U belonging to $N(u)$ but not to $N(v)$. Similarly, $\mathcal{F} \cup \mathcal{F}'$ is distinguished from $V(G) \setminus (\mathcal{F} \cup \mathcal{F}')$ by $a_{\mathcal{F}} \cup I_{\mathcal{F}}$, $R, Q \in \mathcal{F}$ (resp. $R', Q' \in \mathcal{F}'$, resp. $R \in \mathcal{F}, Q' \in \mathcal{F}'$ $(R \neq Q)$) are distinguished by an appropriate vertex in $I_{\mathcal{F}}$, and $R \in \mathcal{F}, R' \in \mathcal{F}'$ are distinguished by any vertex in H adjacent to R. It remains to observe that pairs of vertices in $I'_U \cup I'_{\mathcal{F}} \cup \{a'_U, a'_{\mathcal{F}}\}$ are distinguished by their true twins, and a' is distinguished from $I'_U \cup I'_{\mathcal{F}} \cup \{a'_U, a'_{\mathcal{F}}\}$ by S since it is the only vertex except a to neighbour both I_U and $I_{\mathcal{F}}$. Thus, we conclude that S is a resolving set.

In the other direction, assume that S is a resolving set of size k for G. Observe that the diameter of G equals 2 due to apex a. Since for each pair of twins at least one vertex has to be in any resolving set, we may assume, without loss of generality, that $I_U \cup I_{\mathcal{F}} \cup \{a, a_U, a_{\mathcal{F}}\} \subseteq S$. Let us call this collection of $k - \ell$ vertices[3] $S' \subseteq S$ and let us see what it resolves in G. As argued above, every pair except those of the form $R \in \mathcal{F}$, $R' \in \mathcal{F}'$ is certainly resolved by S'.

We also note that S' indeed does not resolve such pairs R, R'; indeed, since the diameter of G is 2, any pair of vertices can only be resolved by a landmark in the neighbourhood of at least one of these vertices, and R and R' have identical neighbourhoods inside S' (recall that $\mathcal{F}$ and $\mathcal{F}'$ were originally created as false twin sets). Hence S' cannot resolve any pair $R, R' \in \mathcal{F} \cup \mathcal{F}'$ and these pairs must then be resolved by the remaining ℓ vertices in $S \setminus S'$. All vertices outside of $U \cup \mathcal{F} \cup \mathcal{F}'$ are either selected or twins to selected vertices, hence we may assume that $S \setminus S' \subseteq U \cup \mathcal{F} \cup \mathcal{F}'$.

First, consider a potential landmark R or R' in $\mathcal{F} \cup \mathcal{F}'$. Since $\mathcal{F} \cup \mathcal{F}'$ is an independent set in G and G is of diameter 2, the only pairs (Q, Q') resolved by this landmark but not by S' has $Q = R$. Thus, for any pair (R, R') such that $R \in S$ or $R' \in S$, we could replace $S \cap \{R, R'\}$ by an arbitrary neighbour of R in U for a new resolving set of at most the same cardinality as S.

Thus, let S be a resolving set and assume $S \cap (\mathcal{F} \cup \mathcal{F}') = \emptyset$. Let $H = S \cap U$. It then follows from the above discussion that for every $R \in \mathcal{F}$, there is a vertex in $H \cap N(R)$; thus H is a hitting set for the set system $\mathcal{F}$. We also have $|H| \leqslant k - |S'| = \ell$. We conclude that $(U, \mathcal{F}, \ell)$ is a yes-instance.

Recall that the graph G is of diameter 2. The fact that for graphs of diameter 2 Metric Dimension[VC+k] is equivalent to Adjacency Dimension[VC+k], implies the theorem for Adjacency Dimension[VC+k]. Since every resolving set of G must contain at least one of the apex vertices a or a', for G Adjacency Dimension[VC + k] is equivalent to Adjacency Domination Dimension[VC + k]. Thus, the theorem holds for Adjacency Domination Dimension[VC + k] as well. □

We note that this reduction also gives evidence against a more general form of kernelization. Where a standard kernel can be understood as a many-one reduction from a problem to itself, with output size bounded by a function of the parameter, a *Turing kernel* is the corresponding Turing reduction notion. In other words, informally, a Turing kernel is a polynomial-time procedure that solves a parameterized problem, with access to an oracle for the problem but with a bound $f(k)$ on the maximum length of the questions it may ask of the oracle. A *polynomial Turing kernel* is a Turing kernel with a bound $f(k) = k^{O(1)}$ on the question size. For a more formal definition, see [29, 10]. It is known that there are parameterized problems that do not allow a polynomial kernel (assuming NP $\not\subseteq$ coNP/poly) but which do allow polynomial Turing kernels; cf. [29, 42, 32].

Although we do not have a framework for excluding polynomial Turing kernels that is as powerful as that for excluding standard polynomial kernels, Hermelin *et al.* [29] defined a hierarchy of complexity classes, conjectured to represent problems that do not allow polynomial Turing kernels. The most basic

[3] Recall that by definition, $k := \ell + t_U + t_{\mathcal{F}} + 3$.

and most common of these hardness classes is WK[1], which is in turn contained in a larger class MK[2]. It is conjectured in [29] that no WK[1]-hard problem has a polynomial Turing kernel. Since Hitting Set[n] is known to be MK[2]-hard [29], the above reduction gives the following.

Corollary 9. Metric Dimension[VC + k] *is MK[2]-hard (hence also WK[1]-hard) under polynomial parameter transformations, and does not allow a polynomial Turing kernel unless* CNF-SAT[n] *and every other problem in MK[2] does. The same assertion holds for* Adjacency Dimension[VC + k] *and* Adjacency Domination Dimension[VC + k].

4.2 Parameterizations by Bandwidth and Cutwidth

For a graph G, a bijection $\pi\colon V(G) \to \{1, 2, \ldots, |V(G)|\}$ is called a *linear ordering of* $V(G)$. The *bandwitdh* $\mathbf{bw}(\pi)$ of π is $\max\{|\pi(u) - \pi(v)| : uv \in E(G)\}$. The *bandwidth* $\mathbf{bw}(G)$ of G is the minimum of $\mathbf{bw}(\pi)$ over all linear orderings π of $V(G)$. The *cutwidth* $\mathbf{cw}(\pi)$ of π is $\max_{1 \leqslant i < |V(G)|} |\{uv \in E(G) : \pi(u) \leqslant i < \pi(v)\}|$. The *cutwitdh* $\mathbf{cw}(G)$ of G is the minimum of $\mathbf{cw}(\pi)$ over all linear orderings π of $V(G)$.

The problem of deciding whether $\mathbf{bw}(G) \leqslant k$ ($\mathbf{cw}(G) \leqslant k$, respectively) is NP-complete [36] ([24], respectively). The NP-completeness of the problems was proved for special classes of graphs; see lists of such results in [11]. When k is the parameter, the problem of deciding whether $\mathbf{cw}(G) \leqslant k$ is fixed-parameter tractable [41], while the problem of deciding whether $\mathbf{bw}(G) \leqslant k$ is W[k]-hard [4].

We will make use of the following relationship between the two measures:

Proposition 10. *For every graph G, $\mathbf{cw}(G) \leqslant (\mathbf{bw}(G))^2$.*

Proof. Let $\pi : V(G) \to [|V(G)|]$ be an ordering such that $\mathbf{bw}(\pi) = \mathbf{bw}(G) = b$. We argue that $\mathbf{cw}(\pi) \leqslant b^2$. Recall that $\mathbf{cw}(\pi) = \max_{1 \leqslant i < |V(G)|} |\{uv \in E(G) : \pi(u) \leqslant i < \pi(v)\}|$, where $1 \leqslant i < |V(G)|$. Fix some i such that $1 \leqslant i < |V(G)|$. Observe that the number of edges uv such that $\pi(u) \leqslant i < \pi(v)$ and $|\pi(u) - \pi(v)| = j$ ($1 \leqslant j \leqslant b$) is at most j. Thus, the total number of such edges for $1 \leqslant j \leqslant b$ is at most $b(b + 1)/2 \leqslant b^2$. Thus, $\mathbf{cw}(\pi) \leqslant b^2 = (\mathbf{bw}(G))^2$. □

Let us now recall the framework of *cross-composition*, due to Bodlaender, Jansen and Kratsch [5] as we use it in the main proof of this subsection.

Definition 11 (polynomial equivalence relation [5]). *An equivalence relation $\mathcal{R}$ on Σ^* is called a* polynomial equivalence relation *if the following two conditions hold.*

1. *There is an algorithm that given two strings x, $y \in \Sigma^*$ decides whether x and y belongs to the same equivalence class in time polynomial in $|x| + |y|$.*

2. *For any finite set $S \subset \Sigma^*$ the equivalence relation $\mathcal{R}$ partitions the elements of S into a number of classes that is polynomially bounded in the maximum size of an element of S.*

Then *OR-cross-compositions* are then defined as follows.

Definition 12 (OR-cross-composition [5]). *Let $L \subseteq \Sigma^*$ be a language, $\mathcal{R}$ a polynomial equivalence relation on Σ^*, and let $\mathcal{Q} \subseteq \Sigma^* \times \mathbb{N}$ be a parameterized problem. An* OR-cross-composition of L into $\mathcal{Q}$ (with respect to $\mathcal{R}$) *is an algorithm that, given t instances $x_1, \ldots, x_t \in \Sigma^*$ of L belonging to the same equivalence class of $\mathcal{R}$, takes time polynomial in $\sum_{i=1}^{t} |x_i|$ and outputs an instance $(y, k) \in \Sigma^* \times \mathbb{N}$, such that the following hold:*

1. *The parameter value k is polynomially bounded in $\max_{1 \leqslant i \leqslant t} |x_i| + \log t$,*

2. *The instance (y, k) is positive for $\mathcal{Q}$ if and only if at least one instance x_i is positive for L.*

We say that L OR-cross-composes into $\mathcal{Q}$ *if there exists such an OR-cross-composition algorithm with respect to some polynomial equivalence relation $\mathcal{R}$.*

OR-cross-compositions provide evidence against polynomial kernelizations in the following sense:

Theorem 13 ([5]). *If an NP-hard language L OR-cross-composes into the parameterized problem $\mathcal{Q}$, then $\mathcal{Q}$ does not admit a polynomial kernelization unless* NP $\subseteq$ coNP/poly.

To facilitate the lower bounds, we will give an OR-cross-composition from the *improvement version* of the problem, which we call IMPROVEMENT METRIC DIMENSION. In this version, the input consists of a graph G and a resolving set $S \subseteq V(G)$, and the question is whether G has a resolving set of cardinality at most $|S|-1$. We show that this version of the problem is still NP-hard.

Lemma 14. *The improvement version of* METRIC DIMENSION *is NP-hard, even for instances with two apex vertices.*

Proof. We use the reduction from HITTING SET given in Theorem 3. We begin by showing that the improvement version of HITTING SET, called IMPROVEMENT HITTING SET, is itself NP-hard. Indeed, let $(U, \mathcal{F}, k)$ be an instance of HITTING SET over a ground set U, and define an output instance as follows. Let $U' = U \cup \{x_1, \ldots, x_{k+1}\}$ where x_i, $i \in [k+1]$ is a new element for each index i, and define a set system

$$\mathcal{F}' = \{C \cup \{x_i\} \mid C \in \mathcal{F}, i \in [k+1]\}.$$

Let $S = \{x_1, \ldots, x_{k+1}\}$. Then $(U', \mathcal{F}', S)$ is a positive instance of IMPROVEMENT HITTING SET if and only if $(U, \mathcal{F}, k)$ is a positive instance of HITTING SET. Indeed, S is a hitting set for $\mathcal{F}'$, and it is clear that there exists a hitting set for $\mathcal{F}'$ of size smaller than $|S|$ if and only if there exists a hitting set for $\mathcal{F}$ of cardinality k.

To complete the proof, we only need to observe that Theorem 3 describes a polynomial-time reduction from IMPROVEMENT HITTING SET to IMPROVEMENT METRIC DIMENSION. Note that the planted solution S for $(U', \mathcal{F}', S)$ produces a resolving set S' for the output graph G, with $|S'| = k+1$ (where k is the parameter computed in Theorem 3), and the instance (G, S') of IMPROVEMENT METRIC DIMENSION is equivalent to the input instance $(U', \mathcal{F}', S)$ of IMPROVEMENT HITTING SET. It is also clear that the reduction runs in polynomial time. Finally, since the output of Theorem 3 has a pair of (true twin) apex vertices, the last part of the present lemma holds as well. ☐

We can now show the lower bound against a polynomial kernel for METRIC DIMENSION parameterized by bandwidth, METRIC DIMENSION[**bw**]. First we will properly define this problem. Let METRIC DIMENSION[**bw**] be the problem with input $((G, \pi, k), w)$ where π is a linear ordering of $V(G)$, w is the bandwidth of π, and the question is whether G has a resolving set of cardinality at most k.[4] Using Lemma 14 and Theorem 13, it essentially suffices for this result to consider the disjoint union of input instances (of IMPROVEMENT METRIC DIMENSION), but this results in an output instance that is disconnected. We show that the result also applies to connected instances of the problem.

Theorem 4. *Unless* NP $\subseteq$ coNP/poly*, neither* METRIC DIMENSION[**bw**] *nor* METRIC DIMENSION[**cw**] *has a polynomial kernel, even when restricted to connected instances.*

Proof. We will show an OR-cross-composition of IMPROVEMENT METRIC DIMENSION into connected instances of METRIC DIMENSION[**bw**] and argue about METRIC DIMENSION[**cw**] afterwards.

By Lemma 14, we may restrict our attention to input instances with a pair of apex vertices. Let $\mathcal{R}$ be an equivalence relation on instances of IMPROVEMENT METRIC DIMENSION where two instances $x = (G, S)$, $y = (G', S')$ are equivalent if and only if $|V(G)| = |V(G')|$ and $|S| = |S'|$; clearly $\mathcal{R}$ is a polynomial equivalence relation. Let an input of t instances $x_1, \ldots, x_t$ of IMPROVEMENT METRIC DIMENSION be given, all belonging to the same equivalence class. Assume $t \geqslant 2$ (or else create an output simply replicating the input with an arbitrary vertex ordering), and for $i \in [t]$ write $x_i = (G_i, S_i)$ and let u_i and v_i denote the two apex vertices of G_i. For ease of presentation, we denote by V_i the set $V(G_i)$ for each $i \in [t]$. Let $n = |V_1|$ and $k = |S_1|$; by definition n and k are invariant across all instances x_i. We define a vertex ordering π_i of V_i by placing u_i first and all other vertices in arbitrary order. We also assume that $u_i \in S_i$; indeed, S_i must contain either u_i or v_i, and if $S_i \cap \{u_i, v_i\} = \{v_i\}$ then we may simply replace v_i by u_i in S_i for an equivalent resolving set.

Finally, we define an output instance $X = ((G, \pi, tk-1), n)$ of METRIC DIMENSION[**bw**] as follows. We begin by letting G be the disjoint union of all graphs G_i. Then, for each $2 \leqslant i \leqslant t$ we connect vertex u_i to vertex u_{i-1}. We define an ordering π by using the orderings $\pi_1, \ldots, \pi_t$ in order. We first note that π has bandwidth n: every edge $u_i u_{i+1}$ has length n in π and no longer edges exist. It is also clear that G is connected. Thus X is a valid connected instance of METRIC DIMENSION[**bw**], with parameter value $n \leqslant \max_i |x_i|$. To show that the above is an OR-cross-composition it remains to show that X is positive if and only if at least one input instance x_i is positive.

[4]The nested parentheses in the problem definition are in order to comply with the notation of Def. 12.

On the one hand, assume that instance x_i is positive, $i \in [t]$. Let S'_i be a resolving set of x_i of cardinality $k-1$. We claim that $S' := S'_i \cup \bigcup_{j\in[t],j\neq i} S_j$ is a resolving set for G. Specifically, we note that any pair of vertices $p, q \in V_j$ are resolved, by assumption, either by S_j or by S'_i, and if $p \in V_a$ and $q \in V_b$ for some $a \neq b$, then p and q are resolved by the distance to u_1 or to u_t (note that at least one of u_1 and u_t is contained in S'). Thus S' is a resolving set for G. Also note that $|S'| = tk - 1$ as required.

On the other hand, let S' be a resolving set for G with $|S'| \leqslant tk - 1$, and let $i \in [t]$ be such that $|S' \cap V_i| < k$, which exists by a counting argument. Let $S'_i = S' \cap V_i$, by the same argument as above we can assume that the apex u_i is contained in S'_i. We claim that S'_i is a resolving set for G_i. Let $p, q \in V_i$ be an arbitrary pair of vertices. If either of p, q is in S'_i, then clearly S'_i resolves them. Hence assume that $p, q \notin S'_i$, in particular $p, q \neq u_i$. Consider any vertex $r \in S' \setminus S'_i$. By our construction, $\mathrm{d}(p, r) = \mathrm{d}(q, r) = 1 + \mathrm{d}(u_i, r)$, and p and q are not resolved by r. Hence p and q must be resolved by a vertex of S'_i. This completes the proof and we arrive at an OR-cross-composition from an NP-hard language into connected instances of Metric Dimension[**bw**]. Thus by Theorem 13 the polynomial kernel lower bound for parameterization by bandwidth follows.

Finally, by Proposition 10, if Metric Dimension parameterized by cutwidth had a polynomial kernel then the same would hold for Metric Dimension[**bw**]. We conclude that neither parameterization admits a polynomial kernel under the stated complexity-theoretic assumptions. □

We note that with a slight modification it is possible to provide evidence against kernelization of c-connected instances for any constant c, by replacing the twin classes $\{u_i, v_i\}$ by a collection $\{u_i^1, \ldots, u_i^c\}$ of true twin apex vertices, and by connecting u_i^a with u_{i+1}^b for all $a, b \in [c]$ in the construction. Other variations are also possible, to prevent these vertices from being twins in G; we omit the details.

4.3 Parameterization by Max-leaf Number

Finally, we consider a simple positive case. Recall that the *max-leaf number* of a graph G is the maximum number of leaves in a spanning tree of G. Eppstein [15] showed that Metric Dimension is FPT parameterized by the max-leaf number; we observe that his algorithm implies a polynomial kernel. We note that no evidence needs to be given for the max-leaf number of the input; the theorem provides a polynomial-time reduction algorithm such that for any input (G, k) where G is connected, the size of the output is polynomially bounded by the max-leaf number of G.

Theorem 5. Metric Dimension *parameterized by the max-leaf number has a polynomial kernel.*

Proof. Let (G, k) be an instance of Metric Dimension and let p be the max-leaf number of G. Note that we do not assume that p is provided explicitly. Eppstein [15] defines a *branch* in G as a maximal path or cycle subgraph of G where every internal vertex has degree 2 in G. In other words, a branch is either a single edge uv, or a path P between two vertices u, v in G, or a cycle C attached to a single vertex v in G, where every vertex except u and v has degree 2. Note that the branches of G are easy to compute, since every vertex of degree 2 belongs to a unique branch in G and every other branch is a single edge where neither vertex is of degree 2. Eppstein shows the following:

Proposition 15 (Lemma 1 of [15]). *A connected graph with max-leaf number p has $O(p^2)$ branches.*

Thus, let X be the set of endpoints of branches in G; we have an upper bound $|X| = O(p^2)$. Also note that every edge of G is contained in a branch; that every vertex of $V(G) \setminus X$ is degree 2 in G; and that every pair of branches are internally vertex-disjoint. Hence we get a complete description of G by marking those vertices that are part of X and writing down for every branch B the type of B (edge, cycle, or path), its endpoint or endpoints, and the number of internal vertices of B coded in binary. This is a complete description of G in $O(p^2 \log n)$ bits, where $n = |V(G)|$. Let us call this encoding of a graph its *branch-encoding* in the following. Our first step is to reduce the encoding size to a polynomial-length description by bounding $\log n$. For this, we need the main result of [15]:

Proposition 16 (Theorem 1 of [15]). *The metric dimension of a connected graph G with b branches can be computed in time $O(n) + 2^{O(b^3 \log b)} \log n$.*

We consider two cases. Let b be the number of branches of G as constructed above. If $n > 2^{b^3 \log b}$, then the algorithm of Eppstein runs in total time $n^{O(1)}$ and we can afford to exhaustively solve G

and produce an equivalent dummy output (a constant-sized positive or negative instance). Otherwise, $\log n \leqslant b^3 \log b = O(p^6 \log p)$, and the branch-encoding of G uses $O(p^8 \log p)$ bits. We can then finish the generalised kernel by producing an output $(\tilde{G}, k)$ where $\tilde{G}$ is the branch-encoding of G.

We now define the (artificial) problem Branch-Encoded Metric Dimension (BEMD) as taking an input $(\tilde{G}, k)$, where $\tilde{G}$ is a branch-encoded graph, and the instance is positive if and only if the graph G encoded by $\tilde{G}$ has a resolving set of size at most k. Thus, the procedure described above is a polynomial generalised kernelization from Metric Dimension parameterized by the max-leaf number into BEMD. To complete the proof, it suffices to show that BEMD, as a decision problem, lies in NP. Indeed, in this case there exists a polynomial-time reduction to Metric Dimension by the NP-completeness of the latter, and given that the input instance to this reduction has total coding length $p^{O(1)}$ the same will hold for the output.

Hence, it remains to show that BEMD has a polynomial-time verifiable witness, i.e., a witness that can be verified in time polynomial in $|X| + b + \log n$, where n is the number of vertices of the graph G defined by the branch-encoding $\tilde{G}$. The existence of such a witness follows from Eppstein's algorithm: in broad terms, after guessing for each branch how many solution vertices it contains and their approximated position (one of $O(b^2)$ many intervals on the branch called 'stems'), he shows that the remaining problem of finding a solution that obeys theses guessed constraints can be encoded in an ILP with $O(b)$ variables and $O(b^3)$ linear constraints using numbers encodable with $O(\log n)$ bits. But then verifying whether a set S is a solution to BEMD is equivalent to checking whether the variable assignment corresponding to S in Eppstein's ILP satisfies all $O(b^3)$ constraints, a task clearly possible in time polynomial in $|X| + b + \log n$. It follows that BEMD is contained in NP and finally that Metric Dimension parameterized by the max-leaf number admits a polynomial kernel. □

5 Conclusion

We initiated the study of the parameterized complexity of the dual of the classic Metric Dimension problem and obtained a polynomial kernel as well as a single-exponential FPT algorithm. To the best of our knowledge, this is the first non-trivial parameterization for Metric Dimension which leads to a polynomial kernel. Since our focus in this article was on obtaining new classification results, we leave the improvement of the kernel size or a potential proof of a lower bound on the bitsize of our kernel, to future work. Our hardness results provide more evidence to support the view that Metric Dimension is a notoriously hard problem from the computational point of view.

As we wrote in Section 1, to the best of our knowledge, the parameterized complexities of Metric Dimension parameterized by the bandwidth and cutwidth of the input graph are not known yet. It would be interesting to determine them. Finally, we note that it remains open whether Metric Dimension is polynomial time solvable even on series-parallel graphs [3, 12]. Since series-parallel graphs are precisely the graphs of treewidth 2, a negative answer would also imply that there is no XP algorithm for Metric Dimension parameterized by the treewidth.

The *feedback edge set* of a graph $G = (V, E)$ is the minimum number of edges whose deletion makes G acyclic. Clearly, this number equals $|E| - |V| + c$, where c is the number of connected components in G. Epstein et al. [16] proved that Metric Dimension parameterized by feedback edge set is in XP and Eppstein [15] asked whether the problem is FPT. It would be interesting to know whether the problem admits a polynomial-size kernel.

Acknowledgement We are thankful to the referees for their very helpful comments and suggestions. Gutin was partially supported by Royal Society Wolfson Research Merit Award.

References

[1] N. Alon, G. Z. Gutin, E. J. Kim, S. Szeider, and A. Yeo. Solving MAX-r-SAT Above a Tight Lower Bound. *Algorithmica*, 61(3):638–655, 2011.

[2] M. Basavaraju, M. C. Francis, M. S. Ramanujan, and S. Saurabh. Partially polynomial kernels for Set Cover and Test Cover. *SIAM J. Discrete Math.*, 30(3):1401–1423, 2016.

[3] R. Belmonte, F. V. Fomin, P. A. Golovach, and M. S. Ramanujan. Metric dimension of bounded tree-length graphs. *SIAM J. Discrete Math.*, 31(2):1217–1243, 2017.

[4] H. L. Bodlaender, M. R. Fellows, and M. T. Hallett. Beyond NP-completeness for problems of bounded width: hardness for the W hierarchy. In F. T. Leighton and M. T. Goodrich, editors, *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada*, pages 449–458. ACM, 1994.

[5] H. L. Bodlaender, B. M. P. Jansen, and S. Kratsch. Kernelization lower bounds by cross-composition. *SIAM J. Discrete Math.*, 28(1):277–305, 2014.

[6] I. Charon, O. Hudry, and A. Lobstein. Minimizing the size of an identifying or locating-dominating code in a graph is NP-hard. *Theor. Comput. Sci.*, 290(3):2109–2120, 2003.

[7] G. Chartrand, L. Eroh, M. A. Johnson, and O. Oellermann. Resolvability in graphs and the metric dimension of a graph. *Discrete Applied Mathematics*, 105(1-3):99–113, 2000.

[8] F. R. K. Chung. Labelings of graphs. In L. W. Beineke and R. J. Wilson, editors, *Selected Topics in Graph Theory*, pages 151–168. Academic Press, 1988.

[9] R. Crowston, G. Gutin, M. Jones, S. Saurabh, and A. Yeo. Parameterized study of the Test Cover Problem. In *Mathematical Foundations of Computer Science 2012 - 37th International Symposium, MFCS 2012, Bratislava, Slovakia, August 27-31, 2012. Proceedings*, pages 283–295, 2012.

[10] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms.* Springer, 2015.

[11] J. Díaz, J. Petit, and M. J. Serna. A survey of graph layout problems. *ACM Comput. Surv.*, 34(3):313–356, 2002.

[12] J. Díaz, O. Pottonen, M. J. Serna, and E. J. van Leeuwen. Complexity of metric dimension on planar graphs. *Journal of Computer and System Sciences*, 83(1):132–158, 2017.

[13] M. Dom, D. Lokshtanov, and S. Saurabh. Kernelization lower bounds through colors and IDs. *ACM Trans. Algorithms*, 11(2):13:1–13:20, 2014.

[14] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity.* Texts in Computer Science. Springer, 2013.

[15] D. Eppstein. Metric dimension parameterized by max leaf number. *J. Graph Algorithms Appl.*, 19(1):313–323, 2015.

[16] L. Epstein, A. Levin, and G. J. Woeginger. The (weighted) metric dimension of graphs: Hard and easy cases. *Algorithmica*, 72(4):1130–1171, 2015.

[17] H. Fernau, P. Heggernes, P. van 't Hof, D. Meister, and R. Saei. Computing the metric dimension for chain graphs. *Inform. Process. Lett.*, 115(9):671–676, 2015.

[18] H. Fernau and J. A. Rodríguez-Velázquez. Notions of metric dimension of corona products: Combinatorial and computational results. In E. A. Hirsch, S. O. Kuznetsov, J. Pin, and N. K. Vereshchagin, editors, *Computer Science - Theory and Applications - 9th International Computer Science Symposium in Russia, CSR 2014, Moscow, Russia, June 7-11, 2014. Proceedings*, volume 8476 of *Lecture Notes in Computer Science*, pages 153–166. Springer, 2014.

[19] F. V. Fomin, D. Lokshtanov, S. Saurabh, and M. Zehavi. *Kernelization: Theory of Parameterized Preprocessing.* Springer, in preparation.

[20] F. Foucaud, G. B. Mertzios, R. Naserasr, A. Parreau, and P. Valicov. Identification, location-domination and metric dimension on interval and permutation graphs. II. algorithms and complexity. *Algorithmica*, 78(3):914–944, 2017.

[21] J. Gajarský, P. Hlinený, J. Obdrzálek, S. Ordyniak, F. Reidl, P. Rossmanith, F. S. Villaamil, and S. Sikdar. Kernelization using structural parameters on sparse graph classes. *J. Comput. Syst. Sci.*, 84:219–242, 2017.

[22] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

[23] D. Garijo, A. G. Herrera, and A. Márquez. The difference between the metric dimension and the determining number of a graph. *Applied Mathematics and Computation*, 249:487–501, 2014.

[24] F. Gavril. Some NP-complete problems on graphs. In *Proceedings of the 11th Conference on Information Sciences and Systems, John Hopkins University, Baltimore*, pages 91–95, 1977.

[25] G. Gutin, M. Jones, and A. Yeo. Kernels for below-upper-bound parameterizations of the hitting set and directed dominating set problems. *Theor. Comput. Sci.*, 412(41):5744–5751, 2011.

[26] G. Z. Gutin, G. Muciaccia, and A. Yeo. (non-)existence of polynomial kernels for the test cover problem. *Inf. Process. Lett.*, 113(4):123–126, 2013.

[27] F. Harary and R. A. Melter. On the metric dimension of a graph. *Ars Combinatoria*, 2:191–195, 1976.

[28] S. Hartung and A. Nichterlein. On the parameterized and approximation hardness of metric dimension. In *Proceedings of the 28th Conference on Computational Complexity, CCC 2013, K.lo Alto, California, USA, 5-7 June, 2013*, pages 266–276, 2013.

[29] D. Hermelin, S. Kratsch, K. Soltys, M. Wahlström, and X. Wu. A completeness theory for polynomial (Turing) kernelization. *Algorithmica*, 71(3):702–730, 2015.

[30] S. Hoffmann and E. Wanke. Metric dimension for Gabriel unit disk graphs is NP-complete. In *ALGOSENSORS 2012*, volume 7718 of *Lecture Notes in Computer Science*, pages 90–92. Springer, 2012.

[31] M. Jannesari and B. Omoomi. The metric dimension of the lexicographic product of graphs. *Discrete Mathematics*, 312(22):3349–3356, 2012.

[32] B. M. P. Jansen and D. Marx. Characterizing the easy-to-find subgraphs from the viewpoint of polynomial-time algorithms, kernels, and Turing kernels. In *SODA*, pages 616–629. SIAM, 2015.

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

[34] S. Kratsch. Recent developments in kernelization: A survey. *Bulletin of the EATCS*, 113, 2014.

[35] D. Lokshtanov, N. Misra, and S. Saurabh. Kernelization - preprocessing with a guarantee. In *The Multivariate Algorithmic Revolution and Beyond - Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday*, pages 129–161, 2012.

[36] C. Papadimitriou. The NP-completeness of the bandwidth minimisation problem. *Computing*, 16:263–270, 1976.

[37] I. Razgon and B. O'Sullivan. Almost 2-SAT is fixed-parameter tractable. *J. Comput. Syst. Sci.*, 75(8):435–450, 2009.

[38] M. J. Serna and D. M. Thilikos. Parameterized complexity for graph layout problems. *Bulletin of the EATCS*, 86:41–65, 2005.

[39] 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)*, pages 549–559. Congressus Numerantium, No. XIV. Utilitas Math., Winnipeg, Man., 1975.

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

[41] D. M. Thilikos, M. J. Serna, and H. L. Bodlaender. Constructive linear time algorithms for small cutwidth and carving-width. In D. T. Lee and S. Teng, editors, *Algorithms and Computation, 11th International Conference, ISAAC 2000, Taipei, Taiwan, December 18-20, 2000, Proceedings*, volume 1969 of *Lecture Notes in Computer Science*, pages 192–203. Springer, 2000.

[42] S. Thomassé, N. Trotignon, and K. Vuskovic. A polynomial Turing-kernel for weighted independent set in bull-free graphs. *Algorithmica*, 77(3):619–641, 2017.