Structural Parameterizations of Geodetic Set on Directed (Acyclic) Graphs

Laurent Beaudou
Université Clermont-Auvergne, CNRS, Mines de Saint-Étienne, Clermont-Auvergne-INP, LIMOS, 63000 Clermont-Ferrand, France

Florent Foucaud
Université Clermont-Auvergne, CNRS, Mines de Saint-Étienne, Clermont-Auvergne-INP, LIMOS, 63000 Clermont-Ferrand, France

Lucas Lorieau
Université Clermont-Auvergne, CNRS, Mines de Saint-Étienne, Clermont-Auvergne-INP, LIMOS, 63000 Clermont-Ferrand, France

Prafullkumar Tale
Indian Institute of Science Education and Research Pune, India

Abstract

In Directed Geodetic Set, we are given a (directed) graph and seek a small solution set $S \subseteq V(G)$ such that every vertex lies on a shortest directed path between two vertices in S. While most prior work on Directed Geodetic Set has focused on undirected graphs, in this article we study the problem on directed graphs from the perspective of parameterized complexity.

It is known that the problem is W[2]-hard when parameterized by the solution size k, even on directed acyclic graphs (DAGs). We investigate structural parameterizations of the problem.

Our first result is a kernel of size $2^{O(\mathsf{vcn})}$ for Directed Geodetic Set on general digraphs, where vcn denotes the vertex cover number of the underlying (undirected) graph. This implies an algorithm running in time $2^{O(\mathsf{vcn}^2)} \cdot n^{O(1)}$. Furthermore, we prove that, assuming the ETH, the problem does not admit an algorithm running in time $2^{o(\mathsf{vcn}^2)} \cdot n^{O(1)}$. Such a tight quadratic exponential lower bound in the parameter is relatively uncommon in parameterized complexity. These results generalize earlier work on undirected graphs by Foucaud et al. [STACS 2025], and complements a recent result on directed graph by Foucaud et al. [CALDAM 2026], that showed that the problem is para-NP-hard for the pathwidth and feedback vertex set number of the underlying graph.

Next, we show that on general digraphs, Directed Geodetic Set admits a natural kernel of size $(k\Delta)^{O(\mathsf{rdiam})}$, where Δ is the maximum degree and rdiam denotes the reachability diameter of the digraph (a natural analogue of diameter of undirected graphs). This yields an algorithm running in time $(k\Delta)^{O(\mathsf{rdiam}\cdot k)} \cdot n^{O(1)}$. We further prove that, assuming the ETH, the problem does not admit an algorithm running in time $(k\Delta)^{o(\mathsf{rdiam}\cdot k)} \cdot n^{O(1)}$. Finally, we justify the necessity of combining parameters by establishing the following hardness results for Directed Geodetic Set:

1. It is W[2]-hard parameterized by k, even on digraphs of maximum degree 3.
2. It is para-NP-hard parameterized by maximum degree and reachability diameter.

One can infer that the problem remains W[2]-hard when parameterized by k, even on graphs of reachability diameter 3 from Araújo and Arraes [DAM 2022].

All our conditional lower bounds and hardness results hold even when the input digraph is restricted to be a DAG.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Geodetic Set, Directed Graphs, NP-hardness, Parameterized Complexity

Digital Object Identifier 10.4230/LIPIcs.MFCS.2026.88

Related Version *Full Version*: `https://arxiv.org/abs/2606.26414` [5]

Funding *Florent Foucaud*: This author received support from the ANR project GRALMECO (ANR-21-CE48-0004), the French government IDEX-ISITE initiative 16-IDEX-0001 (CAP 20-25), the International Research Center "Innovation Transportation and Production Systems" of the I-SITE CAP 20-25, and the CNRS IRL ReLaX.
Lucas Lorieau: This author has received financial support from the CNRS through the MITI interdisciplinary programs.
Prafullkumar Tale: ARG-MATRICS Grant titled MaPLE (ARGM/2026/0089).

1 Introduction

We study GEODETIC SET on directed graphs (digraphs for short), which we thus call DIRECTED GEODETIC SET. GEODETIC SET is a classic network design problem introduced in [25] based on the notion of convexity for the shortest path metric. See the books [3, 31] for more on graph convexity, for which GEODETIC SET is a central representative problem. In (DIRECTED) GEODETIC SET, one asks for a minimum-size set S of vertices such that each vertex in the input (di)graph is part of some shortest directed path between two vertices of S. (DIRECTED) GEODETIC SET models natural network design problems such as selecting hubs in a transportation network [9, 34]. It is extensively studied in the literature, and serves as one of the key problems in the area of metric-based algorithmic graph theory, together with its close relative METRIC DIMENSION [6, 20]. DIRECTED GEODETIC SET is formally stated as follows.

DIRECTED GEODETIC SET
Input: A directed graph D and an integer k.
Task: Determine whether there exists a subset $S \subseteq V(D)$ of size at most k such that every vertex in $V(D)$ lies on some directed shortest path between two vertices in S.

GEODETIC SET was extensively studied on undirected graphs, on which its algorithmic complexity was almost exhaustively determined for standard graph classes [1, 8, 9, 10, 7, 17, 18, 30] and under the lenses of parameterized complexity [8, 20, 21, 27, 32], approximation algorithms [9, 15] and enumeration complexity [6].

On digraphs, several studies on DIRECTED GEODETIC SET exist regarding upper and lower bounds [11, 12, 13, 16, 19, 29], see also the book [31, Chapter 6]. However, DIRECTED GEODETIC SET was only recently studied on directed graphs from an algorithmic perspective. In [2], DIRECTED GEODETIC SET was shown to be NP-hard on DAGs (even when the underlying graph is either split, bipartite or co-bipartite) but polynomial-time solvable on oriented cactus graphs (without 2-cycles). (For a digraph, its *underlying graph* is the undirected graph obtained by forgetting the orientations of arcs, possibly deleting multiple edges.) In a very recent paper [22], DIRECTED GEODETIC SET was shown to be NP-hard even on DAGs whose underlying graph has small feedback vertex set number, but it is polynomial-time solvable on digraphs whose underlying graph is a tree, and it is FPT on DAGs for the feedback edge set number of the underlying graph.

Our goal is to continue the study of DIRECTED GEODETIC SET and exhibit parameters for which we can obtain FPT algorithms, and prove their optimality under standard complexity assumptions. One natural parameter is the *vertex cover number* vcn of the underlying graph of the input digraph. Note that there exists an FPT algorithm for DIRECTED GEODETIC SET with respect to vcn on undirected graphs [21]. Another relevant parameter is the *reachability diameter*: the maximum directed distance between any pair of vertices in the digraph that are reachable. This parameter was recently introduced in [24] (see also [4]) and earlier under the name *maximum finite distance* [23] (unlike the classical diameter of a digraph, it is always finite). This metric-based parameter is an analogue of the diameter of a connected undirected graph, and greatly restricts the possible distances in the input digraph.

Our results. We first extend a result of [21] (for GEODETIC SET on undirected graphs) to general directed graphs, by giving a kernel of size $2^{O(\mathsf{vcn})}$, and deriving from it an FPT algorithm. We also prove that, under the Exponential-Time Hypothesis (ETH), this algorithm is essentially optimal.

▶ **Theorem 1.** *DIRECTED GEODETIC SET admits an algorithm running in time $2^{O(\mathsf{vcn}^2)} \cdot n^{O(1)}$. Moreover, assuming the ETH, it admits no algorithm running in time $2^{o(\mathsf{vcn}^2)} \cdot n^{O(1)}$, even on DAGs.*

We then give another positive result by showing that if we combine as parameters the solution size, reachability diameter, and maximum degree of the input digraph, we obtain a kernel and an FPT algorithm. Moreover, we show that its running time is essentially optimal under the ETH.

▶ **Theorem 2.** *DIRECTED GEODETIC SET admits a kernel of size $(k\Delta)^{O(\mathsf{rdiam})}$, and an algorithm running in time $(k\Delta)^{O(\mathsf{rdiam}\cdot k)} \cdot n^{O(1)}$. Moreover, assuming the ETH, it admits no algorithm running in time $(k\Delta)^{o(\mathsf{rdiam}\cdot k)} \cdot n^{O(1)}$, even on DAGs with $\mathsf{rdiam} = 2$.*

Unfortunately, there is little hope to generalize the above result by restricting the parameter combination. Indeed, we show in the two next theorems that one cannot omit neither the reachability diameter, nor the solution size, under standard complexity theory assumptions.

▶ **Theorem 3.** *DIRECTED GEODETIC SET remains W[2]-hard for solution size k, even on DAGs with maximum degree 3.*

Note that DIRECTED GEODETIC SET is trivially in XP for the solution size k, since one can check (in polynomial time), for each vertex subset of size at most k, whether it is a valid solution.

▶ **Theorem 4.** *DIRECTED GEODETIC SET remains NP-complete on DAGs with maximum degree 3 and reachability diameter 14.*

Finally, it is known that DIRECTED GEODETIC SET is NP-complete even on DAGs with reachability diameter 3, and the proof actually implies W[2]-hardness for solution size on such DAGs [2]. However, the produced inputs have large maximum degree.

Our results show that DIRECTED GEODETIC SET remains hard even on very restrictive classes of digraphs. In that sense, it differs from other metric-based parameters such as BROADCAST DOMINATION, which is NP-hard but FPT for solution size on DAGs and also FPT for solution size and maximum degree on general digraphs [23]. Note that, interestingly, DIRECTED GEODETIC SET also differs from the related METRIC DIMENSION problem, which is FPT for solution size plus reachability diameter.[1]

Due to space constraints, some proofs are omitted for results marked with ($\star$). A complete version of the paper can be found in [5]. Some standard notations are also omitted; we recall the main ones here. For a subset S of V, the *geodetic interval* of S, denoted by $I(S)$, is the

[1] Indeed, in this problem, every vertex needs a distinct distance-vector to the solution vertex set of size k (where the distance between two vertices may be infinite), and there can be at most $k + (\mathsf{rdiam} + 1)^k$ such distinct vectors in a YES-instance), yielding a trivial kernel of this size.

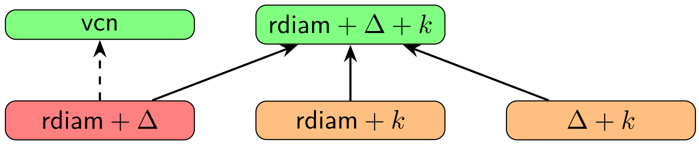


Figure 1 The parameters studied in this paper, and their relations, where vcn and Δ denote the vertex cover number and maximum degree of the underlying undirected graph, rdiam is the reachability diameter, and k is the solution size. An arc indicates that the lower parameter is upper-bounded by a function of the upper parameter. The dashed arc denotes that only rdiam is directly upper bounded by a function of vcn, but Δ is not. However, with a simple pre-processing on the Directed Geodetic Set instance, one can obtain that $\Delta \leq 2^{O(\mathsf{vcn})}$. For the top two parameters, Directed Geodetic Set is FPT (and our algorithms are asymptotically optimal under the ETH). For the bottom left parameter, Directed Geodetic Set is para-NP-hard; for the two bottom right parameters, it is W[2]-hard and in XP.

set of all vertices which lie in some shortest path between two vertices of S. We say vertices in $I(S)$ *are covered* by vertices in S. A vertex that is a source or a sink is called *extremal*. A vertex v is *transitive* if, for every in-neighbor u_1 and out-neighbor u_2 of v, either the arc u_1u_2 exists, or $u_1 = u_2$. Note that in any digraph D, every source, sink and transitive vertex of D is in every geodetic set of D.

Outline of the paper. In Section 2, we first study the tractability of Directed Geodetic Set with respect to the parameter vertex cover number. Sections 3–5 each deal with combinations of solution size, maximum degree and reachability diameter as parameters. They contain the proofs of Theorems 2–4, respectively. We end the paper by presenting some open questions in Section 6.

2 Vertex cover number parameterization

We restate the main theorem of this section.

▶ **Theorem 1.** *Directed Geodetic Set admits an algorithm running in time $2^{O(\mathsf{vcn}^2)} \cdot n^{O(1)}$. Moreover, assuming the* ETH*, it admits no algorithm running in time $2^{o(\mathsf{vcn}^2)} \cdot n^{O(1)}$, even on DAGs.*

2.1 A kernel and an algorithm

Two vertices of a digraph D are said to be *false twins* if they share the same incoming and outgoing open neighborhoods. Note that if two false twins share an arc, then one of them has to admit a loop, which is not possible in the considered setting. A set S of vertices is called a *false twin set*, or a *set of false twins*, if all vertices of S are pairwise false twins. This notion of twins is central in the kernalization algorithm we present in the following. Indeed, since twins share exactly the same neighborhood, they behave exactly in the same manner with respect to the rest of the graph. We also need to define the following notion.

▶ **Definition 5.** *The neighborhood of a set S of vertices is said to be* shortcutting *if, for all vertices $v \in S$, v is transitive in $D[(V \setminus (S - v)]$.*

In particular, if S is an independent set and its neighborhood is shortcutting, then any vertex in S is transitive. The following lemma will be useful.

▶ **Lemma 6** (⋆). *If a digraph D contains a set T of false twins whose neighborhood is not shortcutting, then any optimal geodetic set of D contains at most four vertices of T.*

The two following reduction rules remove some vertices of false twin sets. Proposition 9 shows that this is enough to get a kernel for the parameter vcn.

▶ Reduction rule 7. Let (D, k) be an instance of Directed Geodetic Set and T be a set of at least three false twins in D. If their neighborhood is shortcutting, then delete one vertex of T and decrease k by one.

▶ Reduction rule 8. Let (D, k) be an instance of Directed Geodetic Set and T be a set of at least six false twins in D. If their neighborhood is not shortcutting, then delete one vertex of T without decreasing k.

▶ **Proposition 9** (⋆). *The algorithm applying exhaustively Rules 7 and 8 in a sequential manner on an instance (D, k) returns an equivalent instance (D', k') with $|V(D')| = 2^{O(\mathsf{vcn})}$.*

The next proposition states the existence of the algorithm to be applied on the above kernel. It is the last element necessary for the proof of Theorem 1.

▶ **Proposition 10** (⋆). *Directed Geodetic Set admits an algorithm running in time $n^{O(\mathsf{vcn})}$.*

This implies the proof of the first part of Theorem 1, which we state below.

Proof of Theorem 1 (algorithm). Let (D, k) be an instance of Directed Geodetic Set and denote by vcn the vertex cover number of D. First, we apply the kernelization algorithm presented before to obtain an equivalent instance (D', k') of size $|V(D')| = 2^{O(\mathsf{vcn})}$ in polynomial time. Then, we apply the algorithm presented in the proof of Proposition 10 on instance (D', k') and return the answer obtained. Correctness of the algorithm is ensured by Propositions 9 and 10. This last step is executed in time $(2^{O(\mathsf{vcn})})^{O(\mathsf{vcn})} = 2^{O(\mathsf{vcn}^2)}$, thus ending the proof. ◀

2.2 ETH-based lower bound

In this subsection, we prove the second part of Theorem 1, *i.e.* that the algorithm obtained in Proposition 10 is optimal under ETH [26]. To this end, we present a reduction from the following variant of 3-SAT defined in [28].

3-Partitioned 3-SAT
Input: A formula φ in 3-CNF form, together with a partition of the set of its variables $\mathcal{X}$ into three disjoint sets X, Y, Z, with $|X| = |Y| = |Z| = N$, such that no clause contains more than one variable from each X, Y and Z.
Task: Determine whether φ is satisfiable.

▶ **Proposition 11** ([28]). *Unless* ETH *fails, 3-Partitioned 3-SAT does not admit an algorithm running in time $2^{o(N)}$.*

Reduction description. Consider an instance $\mathcal{I} = (X, Y, Z, \mathcal{C})$ of 3-SAT where $\mathcal{X} = X \cup Y \cup Z$ is the variable set, with $|X| = |Y| = |Z| = N$. By adding dummy variables, if necessary, we can assume that $N = n^2$ for some integer n. Let $\mathcal{C} = \{C_1, \ldots, C_m\}$ be the set of clauses of $\mathcal{I}$. We create a digraph $\mathcal{D} = (V, A)$ through the following procedure. The construction is highly symmetric with respect to the partition of $\mathcal{X}$, so we describe it only for the set X, for the sake of clarity.

- Partition the variables of X into buckets $X^1, \ldots, X^n$, such that each bucket contains n variables. Without loss of generality, we assume an arbitrary but fixed ordering of buckets and variables inside each bucket. We use the notation X^i_j to denote j-th variable of the i-th bucket. Throughout the construction, we follow the notation that superscript will corresponds the index of the bucket while subscript denotes the vertex index itself.
- For every bucket X^i, add to $V(\mathcal{D})$ a set A^i of 2^n vertices corresponding to each possible assignment of the variables of the bucket. Denote those vertices by a^i_p for $p \in [2^n]$.
- For each $i \in [n]$, add two new vertices α_{A^i} and β_{A^i}. For each $i \in [n]$, add the arcs $\alpha_{A^i} a^i_p$ and $a^i_p \beta_{A^i}$ for each $p \in [2^n]$.
- Add vertex sets V, R, T and F on n vertices each. Denote by v_i (respectively r_i, t_i, f_i) the vertices of V (respectively R, T, F) for $i \in [n]$. Add the arcs $a^i_p v_i$ for each $i \in [n]$ and $p \in [2^n]$. Add also the arcs $v_i r_j$ for all $i, j \in [n]$ such that $i \neq j$. Finally, add the arcs $\alpha_{A_i} v_j$ and $\alpha^i_p r_j$ for all $i, j \in [n]$ and $p \in [2^n]$.
- For every $i \in [n]$ and every $p \in [2^n]$, connect a^i_p with vertices in $T \cup F$ such that the connection represents the values of variables set X^i by the assignment corresponding to a^i_p. Formally, the reduction adds the following arcs.
  - If the partial assignment associated to a^i_p is setting variable X^i_j to `true`, then we add the arc $a^i_p t_j$. See the green arcs in Figure 2.
  - If the partial assignment associated to a^i_p is setting variable X^i_j to `false`, then we add the arc $a_{i,p} f_j$. See the red arcs in Figure 2.

  Note that a^i_p is connected to exactly n vertices in $T \cup F$. More precisely for every $j \in [n]$, a^i_p is connected to t_j if and only if it is *not* connected to f_j, and vice versa.
- Add vertex sets U and W, both on n vertices. Denote by u_i (respectively w_i) the vertices of the set U (respectively W) for $i \in [n]$. Add the arcs $u_i w_i$ and $r_i w_i$ for all $i \in [n]$.
- For all sets S in $\{T, F\}$, add two vertices α_S and β_S, as well as a vertex α_U. For $S \in \{T, F, U\}$, add the arcs $\alpha_S s_i$ and $s_i \beta_S$ (when β_S exists) for all $i \in [n]$, where s_i designates the vertex of S with index i.

The previous construction is duplicated twice, to encode variables of Y and Z as well. When the naming of vertices of the previously described gadgets is ambiguous, we add a superscript to indicate the variable set it corresponds to. For instance, the vertex v^Y_i corresponds to the vertex v_i of the above construction for the variable set Y. For sets A^i and vertices a^i_p, instead of using a single index as superscript, we give directly the bucket to which it is associated. For instance, vertex $a^{Z^3}_5$ corresponds to the fifth assignment of bucket Z^3. Additionally, call B the set of all vertices α_S and β_S mentioned during the construction.

The end of the construction focuses on encoding the clauses as follows. Recall that $\mathcal{I}$ is an instance of 3-Partitioned 3-SAT, each clause of $\mathcal{C}$ contains exactly one variable from X, one from Y and one from Z.

- Consider a clause $C_\ell \in \mathcal{C}$, and denote literals in C_ℓ (when they exist) by $X^{i_x}_{j_x}$, $Y^{i_y}_{j_y}$ and $Z^{i_z}_{j_z}$ respectively. Add a vertex c_ℓ in the graph and add the following arcs.
  - If assigning `true` (respectively `false`) to variable $X^{i_x}_{j_x}$ (when it exists) satisfies the clause C_ℓ, then add the arc $t_{j_x} c_\ell$ (respectively $f_{j_x} c_\ell$) and the arc $c_\ell u_{i_x}$.
  - When they exist, perform a similar operation with variables $Y^{i_y}_{j_y}$ and $Z^{i_z}_{j_z}$.

Let C be the set of all the vertices added with respect to some clause. Note that for every vertex in C, the number of its in-neighbors is equal to the number of its out-neighbors, which is equal to number of literals in it. We will argue that a vertex c_ℓ is in a shortest path from a vertex a^i_p , corresponding to a (partial) assignment, and some sink vertex if and only if the (partial) assignment corresponding to a^i_p satisfies the clause. We conclude the construction by adding some arcs to prevent “unwanted coverage”.

- Add for all $i \in [n]$ and all $S, S' \in \{X, Y, Z\}$ the arcs $\alpha_{A^{S^i}} v_i^{S'}$, $\alpha_{T^S} u_i^{S'}$ and $\alpha_{F^S} u_i^{S'}$. These arcs will prevent any coverage from a source vertex associated with the variable set S to cover clauses with a path going to a sink associated with variable set S'.
- Add for each $S, S' \in \{X, Y, Z\}$ such that $S \neq S'$ the arcs $a_p^{S^i} u_j^{S'}$ for all $i, j \in [n]$ and all $p \in [2^n]$. These arcs will prevent any coverage from a selected vertex associated with the variable set S to cover clauses with a path going to a sink associated with variable set S'.

This ends the construction of $\mathcal{D}$. The instance returned by this reduction is $(\mathcal{D}, k)$ with $k = 12n + 15$. Figure 2 is a partial representation of the obtained instance.

Before justifying the correctness of the reduction, we recall a useful lemma.

▶ **Lemma 12** ([2]). *In any digraph D, every source, sink and transitive vertex of D (in particular, every leaf of D) belongs to every geodetic set of D.*

Correctness of the reduction. In Lemma 13, we prove that if $\mathcal{I}$ is satisfiable, then $\mathcal{D}$ admits a geodetic set of size k. In this section, we present an outline of the proof in the reverse direction.

We first present an informal overview. It is easy to see that all sources and sinks must belong to any geodetic set of $\mathcal{D}$ (Lemma 12). We argue that the shortest paths between the sinks and sources cover all the vertices in $V(\mathcal{D})$, except the vertices in C and $\bigcup_{S\in\{X,Y,Z\}} V^S$. For example, consider vertex c_ℓ and set V in Figure 2. We use the simplified notation from the figure to complete the overview.

To cover vertices v_1, v_2, v_3, v_4, and vertex c_ℓ, any geodetic set must include *at least one* vertex from the sets A^1, A^2, A^3, and A^4, respectively. The cardinality constraint on the geodetic set ensures that it contains *exactly* one vertex from each set. This selection naturally corresponds to a valid assignment of the variables in the instance $\mathcal{I}$. We will argue that this must indeed be a satisfying assignment.

For vertices corresponding to assignments (such as a_p^1 and a_q^3 in Figure 2), their variable assignments are evident from their adjacencies to vertices in T and F. These two sets relay the assignment information to the clause vertices. Note that both a_p^1 and a_q^3 set the fourth variable in their respective buckets to `true`. Consider the clause c_ℓ, which contains the variable X_4^3. To obtain a valid assignment, we must ensure that the shortest path between a_q^3 and some sink vertex covers the vertex c_ℓ. Moreover, the shortest path between a_p^1 and the sink vertices must *not* cover c_ℓ. The relevant sink vertices correspond to the set W.

These conditions are enforced by the vertices in sets V and R, along with their interconnecting arcs. Consider the two paths $a_p^1 - t_4 - c_\ell - u_3 - w_3$ and $a_q^3 - t_4 - c_\ell - u_3 - w_3$. Note that the first path is *not* a shortest path because there is a shorter alternative: $a_p^1 - v_1 - r_3 - w_3$. However, there is no analogous shortcut path $a_q^3 - v_3 - r_3 - w_3$ because the arc from v_3 to r_3 does not exist. This distinction forms the crux of the proof of correctness.

In the remainder of this section, we formalize this intuition. Before proceeding, we remark that this missing arc and the reliance on these connections to bypass V for shortest paths critically depends on the fact that clause c_ℓ contains at most one literal from the set X.

In the considered construction, we define set $\mathcal{S}_0$ as the collection of sinks and sources, i.e., $S_0 := B \cup \bigcup_{S\in\{X,Y,Z\}} W^S$. In the next two observations, we argue that vertices of $\mathcal{S}^0$ cannot cover vertices in C and that vertices of $\bigcup_{S\in\{X,Y,Z\}} V^S$ can only be covered by paths starting at specific vertices not belonging to $\mathcal{S}^0$.

▶ **Lemma 13** ($\star$). *If $\mathcal{I}$ is satisfiable, then $\mathcal{D}$ admits a geodetic set of size k.*

▶ **Observation 14** ($\star$). *Let $S \in \{X, Y, Z\}$. No shortest path between vertices of $\mathcal{S}^0$ covers vertices of C.*

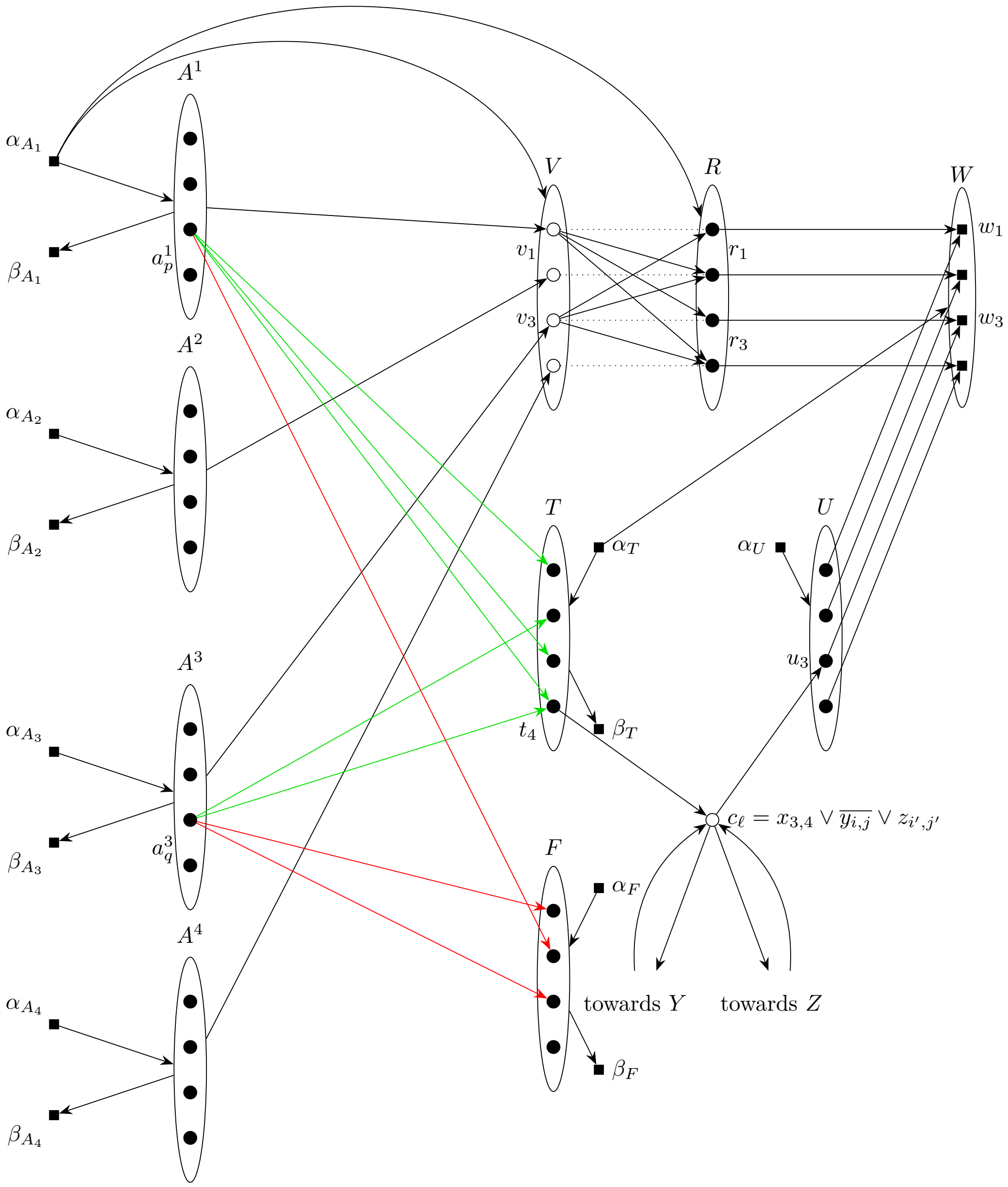


Figure 2 Partial representation of the instance of Directed Geodetic Set obtained from 3-Partitioned 3-SAT by the reduction described in Section 2.2 for $n = 4$. If an arc goes from one vertex v to a vertex set S, it represents all arcs vv' with $v' \in S$ (similarly for an arc from one vertex set to a set). Sources and sinks are represented with squares, while empty circles represent vertices that are *not* covered by shortest paths from sources to sinks. For the sake of clarity, we only show representative arcs. For example, only vertices associated to variables of X and a single clause vertex c_ℓ are represented. The arc from α_{A_1} represent the arcs of the form α_{A_i} to V and α_{A_i} to R. The arcs added while encoding clauses from vertices associated to variables X to vertices associated to other variable sets (namely Y and Z) are *not* represented. Arcs between V and R are represented only for v_1 and v_3 as starting vertices and dotted arcs represent missing arcs.

▶ **Observation 15** ($\star$)**.** *Let $S \in \{X, Y, Z\}$. The vertex v_i^S can only be covered by a path that starts from a vertex of $A^{S^i} \cup \{v_i^S\}$.*

We also argue that $\mathcal{D}$ is an acyclic digraph by using another simple observation.

▶ **Observation 16** ($\star$)**.** *Digraph $\mathcal{D}$ can be decomposed into layers $L_1, \ldots, L_5$ such that all arcs of $A(\mathcal{D})$ are either internal to a layer and adjacent to a leaf vertex, or going from layer L_i to layer L_j with $i < j$. In particular, $\mathcal{D}$ is acyclic.*

▶ **Lemma 17** ($\star$)**.** *If $\mathcal{D}$ admits a geodetic set of size k, then $\mathcal{I}$ is satisfiable.*

Proof of Theorem 1 (optimality). The reduction presented above takes an instance $\mathcal{I}$ of 3-Partitioned 3-SAT and by Lemmas 13 and 17 returns an equivalent instance $(\mathcal{D}, k)$ of Directed Geodetic Set in time $2^{O(n)}$ with $|V(\mathcal{D})| = 2^{O(n)}$. Furthermore, by Observation 16, $\mathcal{D}$ is a DAG. Note that taking all the vertices in layers L_2, L_4 and L_5 from the proof of Observation 16 as well as vertices of B results in a vertex cover of the underlying graph of $\mathcal{D}$. This means that

$$\mathsf{vcn}(\mathcal{D}) \leq |B| + \sum_{S \in \{X,Y,Z\}} \left(|V^S| + |T^S| + |F^S| + |U^S| + |W^S|\right) = 21n + 15.$$

We thus get that $\mathsf{vcn}(\mathcal{D}) + k = O(n)$. Now, suppose for a contradiction that ETH holds and that there exists an algorithm to solve Directed Geodetic Set in time $2^{o(\mathsf{vcn}^2)} \cdot n^{O(1)}$. Consider the following algorithm to solve 3-Partitioned 3-SAT on the instance $\mathcal{I}$ on $3N$ variables. First, use the reduction described above to construct an equivalent instance $(\mathcal{D}, k)$ of Directed Geodetic Set with $\mathsf{vcn}(\mathcal{D}) + k = O(n)$ and $|V(\mathcal{D})| = 2^{O(n)} = 2^{o(n^2)} = 2^{o(N)}$. Then, solve Directed Geodetic Set on this new instance with the assumed algorithm in time

$$2^{o(\mathsf{vcn}^2)} \cdot |V(\mathcal{D})|^{O(1)} = 2^{o(N)} \cdot 2^{o(N)} = 2^{o(N)}$$

and return the same answer for instance $\mathcal{I}$. This algorithm is correct by our assumption and by Lemmas 13 and 17, and its total running time is $2^{o(N)}$, thus it contradicts Proposition 11. ◀

3 Solution size, maximum degree and reachability diameter

We restate the result proved in this section.

▶ **Theorem 2.** *Directed Geodetic Set admits a kernel of size $(k\Delta)^{O(\mathsf{rdiam})}$, and an algorithm running in time $(k\Delta)^{O(\mathsf{rdiam} \cdot k)} \cdot n^{O(1)}$. Moreover, assuming the* ETH*, it admits no algorithm running in time $(k\Delta)^{o(\mathsf{rdiam} \cdot k)} \cdot n^{O(1)}$, even on DAGs with $\mathsf{rdiam} = 2$.*

3.1 A kernel and an algorithm

We now prove the first part of Theorem 2. Let D be an input digraph to Directed Geodetic Set with maximum degree Δ and reachability diameter rdiam. Consider the partition of D into strongly connected components $C_1, \ldots, C_t$. We define the digraph H on vertex set $C_1, \ldots, C_t$, with an arc from C_i to C_j ($i \neq j$) if there exists an arc from a vertex in C_i to a vertex in C_j in D. Note that H is a DAG, for otherwise some strongly connected component from the partition would not be maximal. Assume without loss of generality that $C_1, \ldots, C_p$ ($1 \leq p \leq t$) are the sources of H.

Let v be any vertex of D, with $v \in C_i$ fir some $i \in [t]$. Then, there exists C_j with $1 \leq j \leq p$ (that is, C_j is a source in H), such that C_i is reachable from C_j in H. For any i with $1 \leq i \leq p$, let x_i be an arbitrary vertex of the source component C_i and let $X = \{x_i \mid 1 \leq i \leq p\}$. Then, every vertex in D is reachable from at least one vertex in X.

Note that the maximum number of vertices reachable from any vertex of D is at most $\sum_{i=0}^{\mathsf{rdiam}} \Delta^i \leq \Delta^{\mathsf{rdiam}+1}$. In any yes-instance of Directed Geodetic Set, we must have $p \leq k$, as any geodetic set must include some vertex in each strongly connected component that is a source of H. Hence, in a yes-instance, the number of vertices must be at most $k\Delta^{\mathsf{rdiam}+1}$.

The kernelization algorithm is hence straightforward: if D has more than $k\Delta^{\mathsf{rdiam}+1}$ vertices, we return a trivial no-instance (such as the digraph with $k+1$ isolated vertices, k being unchanged); otherwise, we return (D, k) itself.

The algorithm is a simple brute-force algorithm over such a kernel, testing all the possible subsets of size at most k as a potential geodetic set, and checking in polynomial time whether each of them is indeed a geodetic set or not. This runs in time at most $\binom{k\Delta^{\mathsf{rdiam}+1}}{k}(k\Delta^{\mathsf{rdiam}+1})^{O(1)}$ which is in $(k\Delta)^{O(\mathsf{rdiam}\cdot k)}$.

3.2 ETH-based lower bound

In this subsection, we prove the second part of Theorem 2. In particular, we show that the simple algorithm obtained by applying brute force to the reduced instance is optimal under the ETH. To this end, we present a reduction from $\ell \times \ell$-Independent Set, which we define formally below and state the corresponding conditional lower bound.

> $\ell \times \ell$-Independent Set
> **Input:** A graph H together with a partition of its vertex set $\langle V_1, V_2, \ldots, V_\ell \rangle$, where each V_i is an independent set of cardinality ℓ, for all $i \in [\ell]$.
> **Task:** Determine whether there exists an independent set $X \subseteq V(H)$ such that $|X \cap V_i| = 1$ for every $i \in [\ell]$.

▶ **Proposition 18** (Theorem 14.12 in [14]). *Unless the ETH fails, $\ell \times \ell$-Independent Set does not admit an algorithm running in time $\ell^{o(\ell)}$.*

The reduction takes as input an instance $(H, \langle V_1, V_2, \ldots, V_\ell \rangle)$ of $\ell \times \ell$-Independent Set, runs in polynomial time, and outputs an instance (D, k) of Geodetic Set such that $k = 3\ell + 1$, the maximum degree of D satisfies $\Delta \leq \ell^4$, and $\mathsf{rdiam} = 2$.

- Initially, set $V(D) := V(H)$ and $A(D) := \emptyset$.
- For each part V_i, introduce three new vertices a_i, b_i, and c_i, and add the arcs $a_i u$, ub_i, and uc_i for every $u \in V_i$.
- For every pair $i \neq j \in [\ell]$, construct a set $F_{i,j}$ as follows. For every edge $uv \in E(H)$ with $u \in V_i$ and $v \in V_j$, add a corresponding vertex f_{uv} to $F_{i,j}$.
- For every $i \in [\ell]$ and every vertex $w \in V_i$, add an arc wf_{uv} for every vertex $f_{uv} \in F_{i,j}$ with $j \in [\ell] \setminus \{i\}$, provided that $w \notin \{u, v\}$. Equivalently, the arc wf_{uv} is added if and only if the edge uv is not incident to w in H.
- Introduce a special vertex d. For every $i \neq j \in [\ell]$, add the arc $f_{uv}d$ for every vertex $f_{uv} \in F_{i,j}$. Moreover, for every $i \in [\ell]$, add the arcs $a_i d$ and $c_i d$.

This completes the construction of the digraph D. The reduction outputs the instance (D, k) with $k = 3\ell + 1$ of Geodetic Set. In the following lemma, we prove the correctness of the reduction.

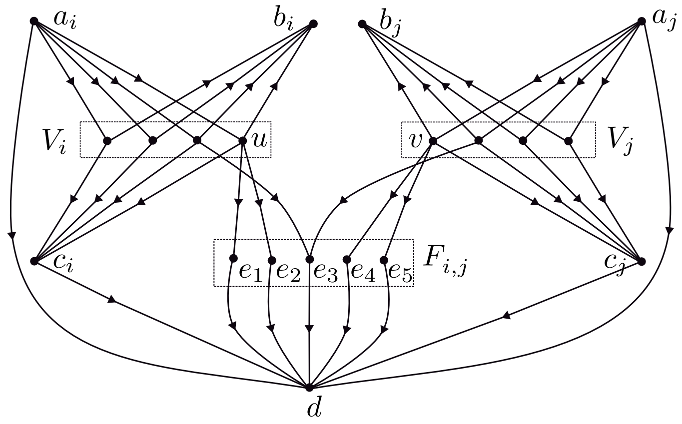


Figure 3 Overview of the reduction from $\ell \times \ell$-INDEPENDENT SET to GEODETIC SET. Edges e_3, e_4, e_5 are incident on u, and edges e_1, e_2, e_3 are incident on v. Note that for the sake of clarity, we do not show all the arcs.

▶ **Lemma 19.** *$(H, \langle V_1, V_2, \ldots, V_\ell \rangle)$ is a* `yes`*-instance of $\ell \times \ell$-INDEPENDENT SET if and only if (D, k) is a* `yes`*-instance of GEODETIC SET.*

Proof. ($\Rightarrow$) Let $X = \{u_1, \ldots, u_\ell\}$, where $u_i \in V_i$ for each $i \in [\ell]$, be an independent set in H. Consider the set $S := X \cup \{a_1, \ldots, a_\ell\} \cup \{b_1, \ldots, b_\ell\} \cup \{d\}$. Clearly, $|S| = 3\ell + 1$. We show that S is a geodetic set of D.

Every vertex $u_i \in V(H) \subseteq V(D)$ lies on a shortest path from a_i to b_i by construction. Consider any edge $vw \in E(H)$. Since X is an independent set, there exists a vertex $u_i \in X$ that is not incident to the edge vw. By the construction of D, there is a shortest path (u_i, f_{vw}, d), which covers the vertex f_{vw}. Finally, for every $i \in [\ell]$, the vertex c_i lies on a shortest path from u_i to d. Thus, every vertex of D lies on a shortest path between some pair of vertices in S, and hence S is a geodetic set of D.

($\Leftarrow$) Suppose that S is a geodetic set of D of size $3\ell + 1$. It must contain the set $S_0 := \{a_1, \ldots, a_\ell\} \cup \{b_1, \ldots, b_\ell\} \cup \{d\}$, since these vertices are sources or sinks in the digraph D and must therefore belong to any geodetic set.

As shown in the forward direction, the vertices in S_0 together cover all vertices in V_i for every $i \in [\ell]$. Hence, the remaining vertices of S are used to cover vertices in the sets $\{c_i \mid i \in [\ell]\}$ and in the sets $F_{i,j}$ for $i \neq j \in [\ell]$.

By construction, no shortest path between any pair of vertices in S_0 covers the vertex c_i for any $i \in [\ell]$. Therefore, S must contain at least one vertex from $\{c_i\} \cup V_i$ for each $i \in [\ell]$. Suppose that for some $i \in [\ell]$, the vertex $c_i \in S$. Let u_i be any vertex in V_i, and define $S' := (S \cup \{u_i\}) \setminus \{c_i\}$. Observe that no shortest path starting or ending at c_i covers any vertex in $F_{i,j}$ for $j \neq i$. Moreover, the shortest path (u_i, c_i, d) covers c_i, and thus replacing c_i by u_i preserves the geodetic property. Hence, S' is also a geodetic set of D. Applying this argument for every $i \in [\ell]$, we obtain a geodetic set S' such that $S' \cap V_i \neq \emptyset$ for all $i \in [\ell]$. Since $S' \setminus S_0$ contains exactly ℓ vertices, it follows that $S' \cap V_i$ contains exactly one vertex for each $i \in [\ell]$.

Let X be the set of vertices in S' that belong to $\bigcup_{i=1}^{\ell} V_i$. We claim that X is an independent set in H. Suppose, for the sake of contradiction, that X is not independent. Then there exists an edge $uv \in E(H)$ such that $u \in V_i$ and $v \in V_j$ for some distinct $i, j \in [\ell]$. Consider the vertex f_{uv} in D corresponding to the edge uv. By construction, this vertex is not covered by any shortest path between vertices in S_0. Furthermore, neither the shortest path from u to d nor the shortest path from v to d covers f_{uv}. Additionally, f_{uv} is not reachable from any vertex in $V_{i'}$ for $i' \in [\ell] \setminus \{i, j\}$. Thus, f_{uv} cannot lie on any shortest path between a pair of vertices in S', contradicting the assumption that S' is a geodetic set of D. Therefore, X is an independent set in H containing exactly one vertex from each V_i, completing the proof. ◀

The conditional lower bound mentioned in Theorem 2 follows from Proposition 18 and Lemma 19.

4 W[2]-hardness for solution size on subcubic DAGs

We restate the result proved in this section.

▶ **Theorem 3.** *Directed Geodetic Set remains* W*[2]-hard for solution size k, even on DAGs with maximum degree 3.*

The proof is adapted from the reduction from Set Cover to Directed Geodetic Set in [2], that implies its NP-hardness and W[2]-hardness on DAGs (of unbounded maximum degree). In fact, we find it notationally cleaner to first revisit the proof from Set Cover to Directed Geodetic Set mentioned in [2]. Then, we modify the reduced instance to ensure that it has bounded maximum degree. Given an instance $(U, \mathcal{F}, k)$ of Set Cover with $U = \{1, \ldots, n\}$, and $\mathcal{F} = \{F_1, \ldots, F_m\}$, we construct a directed acyclic graph D as follows. The vertex set of D is $V(D) = X \cup Y \cup \{s, t_1, t_2\}$, where $X = \{f_1, \ldots, f_m\}$ and $Y = \{u_1, \ldots, u_n\}$. The arc set $A(D)$ contains the following arcs: (i) For every set $F_i \in \mathcal{F}$ and element $j \in U$, if $j \in F_i$ then add the arc $(f_i, u_j) \in A(D)$. (ii) For each set vertex $f_i \in X$, add arcs sf_i, and f_it_1. (iii) For each vertex $u_j \in Y$, add the arc u_jt_2. (iv) Finally add a single arc $st_2 \in A(D)$. It is easy to see that D is a DAG, since we orient all arcs consistently from $\{s\}$ towards the sinks $\{t_1, t_2\}$.

We construct a new digraph D' from the digraph D obtained in the last reduction by locally replacing high-degree vertices with directed binary trees. The transformation ensures: $V(D) \subseteq V(D')$, D' is a DAG, the maximum degree in D' is 3, and finally $(D, k+3)$ is a `yes`-instance of Geodetic Set if and only if $(D', k+3)$ is.

Let $\Delta = \Delta(D)$ denote the maximum total degree of D. We define a *binary tree of height exactly* $\lceil \log \Delta \rceil$ as a rooted tree in which every internal vertex has at most two children, every root-to-leaf path has length exactly $\lceil \log \Delta \rceil$, and the tree has exactly $L \leq \Delta$ leaves. All internal vertices have total degree at most 3.

- Delete the vertex t_1 from D (it will be reintroduced later on).
- Construct an outgoing binary tree T_s^{out} with $|N^+(s)|$ leaves, whose arcs are oriented away from its root s^+. For every original arc sf_i where $f_i \in X$, delete it and instead connect the corresponding leaf of T_s^{out} to f_i. Add the arc ss^+.
- For each $f_i \in X$, construct an outgoing binary tree $T_{f_i}^{\text{out}}$ with $|N^+(f_i)|$ leaves, oriented away from its root f_i^+. For every original arc f_iu_j, delete it. Similarly, for each $u_j \in Y$, construct an incoming binary tree $T_{u_j}^{\text{in}}$ with $|N^-(u_j)|$ leaves, whose arcs are oriented toward its root u_j^-.

  For every original arc f_iu_j of D, connect one leaf of $T_{f_i}^{\text{out}}$ to one distinct leaf of $T_{u_j}^{\text{in}}$. This replaces all arcs from X to Y.
- Construct an incoming binary tree $T_{t_2}^{\text{in}}$ whose arcs are oriented toward its root t_2^-. Attach every leaf of $T_{u_j}^{\text{out}}$ corresponding to an original arc u_jt_2 to a distinct leaf of $T_{t_2}^{\text{in}}$. Add the arc $t_2^-t_2$.

- Reintroduce vertex t_1 by adding it to the graph. For every $u_j \in Y$, add the arc $u_j^- t_1$. Construct an incoming binary tree $T_{t_1}^{\text{in}}$ with leaves corresponding to all arcs (u_j^-, t_1), and orient all arcs toward its root t_1^-. Replace each arc $u_j^- t_1$ by an arc from u_j^- to a distinct leaf of $T_{t_1}^{\text{in}}$, and add the arc $t_1^- t_1$.

Let D' be the resulting digraph. We remark that arc st_2 has not been disturbed by the above modifications and it is present in D'. All added gadgets are directed trees with consistent orientation. Since the original graph is a DAG, replacing arcs by tree paths cannot create a directed cycle. Hence D' is a DAG. Finally, every internal node of every tree has total degree at most 3, and each original vertex is incident to at most one incoming and two outgoing arcs and hence the maximum degree of D is at most 3. Due to space constraints, we omit the formal proof of the reduction.

5 NP-hardness for subcubic DAGs with small reachability diameter

We restate the result proved in this section.

▶ **Theorem 4.** *Directed Geodetic Set remains* NP*-complete on DAGs with maximum degree 3 and reachability diameter 14.*

We prove the theorem by reducing from a variant of 3-SAT where the number of occurrences of a variable in clauses is at most three.

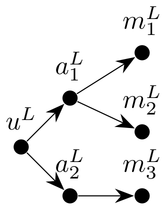
3-Occ 3-SAT
Input: A set X of n variables and a set C of m clauses such that any variable appears at most three times and each clause contains either two or three variables.
Task: Find a truth assignment φ of the variables of X such that it satisfies all clauses of C.

▶ **Theorem 20** ([33], Theorem 2.1)**.** *3-Occ 3-SAT is* NP*-complete.*

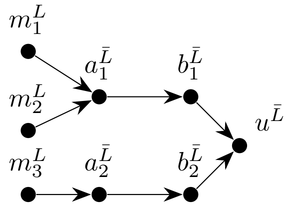
We consider an instance $\mathcal{I} = (X, C)$ of 3-Occ 3-SAT and construct a digraph $D = (V, A)$ based on $\mathcal{I}$. Without loss of generality, we order the set of variables X such that $X = \{x_1, \ldots, x_n\}$. We also order the set of clauses C such that $C = \{c_1, \ldots, c_m\}$.

Before describing the structure of D, we introduce *link gadgets* that will be used throughout the construction of D to ensure the maximum degree of D remains low. Those gadgets are similar to the binary trees used in Section 4. A *simple link gadget* L is a binary tree of height exactly 2 with three leaves. Denote its vertex set as $V^L = \{m_1^L, m_2^L, m_3^L\} \cup \{a_1^L, a_2^L\} \cup \{u^L\}$, where the vertex u^L is the root, the three vertices m_i^L for $1 \le i \le 3$ the leaves, and vertices a_1^L and a_2^L are the inner vertices of said binary tree. Furthermore, the orientation of the arcs is either from u^L to vertices m_i^L (in this case, the linking gadget is said to be *dividing*) or from vertices m_i^L to u^L (and the linking gadget is said to be *grouping*).

Finally, we also define an *extended* version of both types of gadgets. To do so, subdivide the arcs respectively from a_1^L and a_2^L to u^L by respectively adding the vertices b_1^L and b_2^L and orient the new arcs in the direction of the subdivided arcs. To distinguish the extended version of the gadgets from the original one, we will denote the latter with a superscript bar, so that the notation L corresponds to a simple linking gadget, while $\bar{L}$ corresponds to an extended one. Figure 4 depicts one simple and one extended link gadget, and introduces a compact representation for both types of link gadgets.



(a) A simple dividing link gadget L.



(b) An extended grouping link gadget $\bar{L}$.

(c) The corresponding compact representations of L and $\bar{L}$. Note that the labeling of the boundary vertices of each gadget may be omitted in further representations.

Figure 4 Representations of link gadgets.

Variable gadgets. For each variable $x \in X$, add in V a set V^x of vertices such that $V^x = \{s^x, t_1^x, t_2^x, a^x\} \cup \{\mu_i^x \mid \mu \in \{\nu, \pi\},\ i \in [7]\}$. We also add:

- three arcs from s^x respectively to π_0^x, ν_0^x and t_2^x;
- for all $\mu \in \{\pi, \nu\}$ and all $i \in [0, 6]$, an arc from μ_i^x to μ_{i+1}^x;
- for all $\mu \in \{\pi, \nu\}$, two arcs from μ_7^x respectively to t_1^x and a^x;
- an arc from a^x to t_2^x.

Clause gadgets. For any clause $c \in C$, add one extended grouping gadget $\bar{L}^c$ and a dividing gadget L^c, along with two vertices t_1^c and t_2^c. Call V^c the set of vertices of a clause gadget, such that $V^c = V^{L^c} \cup V^{\bar{L}^c} \cup \{t_1^c, t_2^c\}$. Add also the following arcs:

- two arcs respectively from $b_1^{\bar{L}^c}$ and $b_2^{\bar{L}^c}$ to t_1^c.
- an arc from $u^{\bar{L}^c}$ to u^{L^c}.
- three arcs respectively from $m_1^{L^c}$, $m_2^{L^c}$ and $m_3^{L^c}$ to t_2^c.

Linking variables and clauses. Consider a variable $x \in X$ and denote by $C(x)$ the set of clauses $\{c_i, c_j, c_k\}$ such that x occurs in c_i, c_j and c_k, and $i \leq j \leq k$. Note that the set $C(x)$ can also be of size 1 or 2, in which case we respectively consider that $i = j = k$ and $j = k$. Define also $C_0(x) = c_i$, $C_1(x) = c_j$ and $C_2(x) = c_k$. We consider similar notions for a clause $c \in C$, so that the set of variables appearing in c is denoted by $X(c) = \{x_i, x_j, x_k\}$ with $i \leq j \leq k$. We also define $X_0(c) = x_i$, $X_1(c) = x_j$ and $X_2(c) = x_k$. We will add to A some arcs by following the next procedure:

- for $1 \leq p \leq |C(x)|$:
  - Let $c = C_p(x)$ and q be the smallest integer such that $X_q(c) = x$,
  - if x appears as a positive literal in c, then add the arcs from π_p^x to $m_q^{L^c}$ and from π_{3+p}^x to $m_q^{\bar{L}^c}$, otherwise add the arc from ν_p^x to $m_q^{L^c}$ and from ν_{3+p}^x to $m_q^{\bar{L}^c}$.

This concludes the construction of D. A partial representation of D displaying the gadgets corresponding to a variable x appearing in a clause c can be found in Figure 5. Due to space constraints, we omit the formal proof of the reduction.

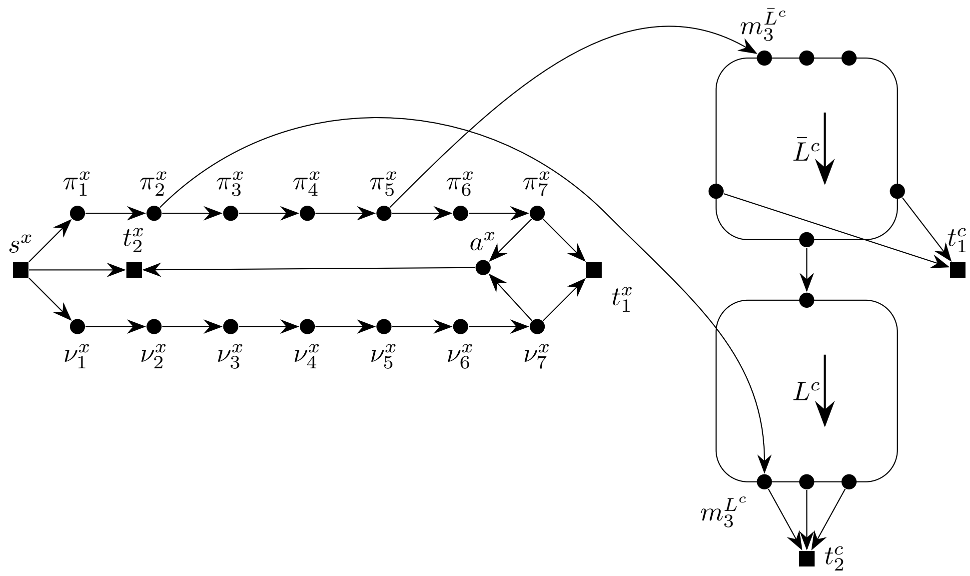


Figure 5 The gadgets of D corresponding to a variable $x \in X$ appearing as a positive literal in clause $c \in C$ such that $C_2(x) = c$ and $X_3(c) = x$. The sources and sinks of D are represented with squares, and link gadgets are represented using their compact representation.

6 Conclusion and further work

In this work, we studied some structural parameters that, combined together, give an optimal (under ETH) FPT algorithm to solve Directed Geodetic Set. In particular, we considered the influence of the maximum degree of a digraph, which can be seen as some measure of sparsity. The parameterized complexity of Directed Geodetic Set has been studied recently for other sparse digraphs, namely "tree-like" directed (acyclic) graphs [22]. A natural extension of this work would be to consider the parameterized complexity of Directed Geodetic Set with respect to other parameters capturing some notion of sparsity, or other classes of digraphs that are sparse. One could also study the existence of other kernels for Directed Geodetic Set on directed graphs, especially for cases where the problem is known to be FPT. In particular, to the best of our knowledge, no polynomial size kernel for Directed Geodetic Set on digraphs is known.

References

1 J. Ahn, L. Jaffke, O. Kwon, and P. T. Lima. Well-partitioned chordal graphs. *Discrete Mathematics*, 345(10):112985, 2022. doi:10.1016/J.DISC.2022.112985.

2 J. Araújo and P. S. M. Arraes. Hull and geodetic numbers for some classes of oriented graphs. *Discrete Applied Mathematics*, 323:14–27, 2022. doi:10.1016/J.DAM.2021.03.016.

3 J. Araújo, M. C. Dourado, F. Protti, and R. M. Sampaio. *Introduction to Graph Convexity: An Algorithmic Approach*. Springer, 2025.

4 B. Bals, J. Blikstad, D. Dadush, Y. Nazari, and J. Schmidt. Revisiting diameter in directed graphs. *arXiv eprint*, 2026. arXiv:2606.08217.

5 L. Beaudou, F. Foucaud, L. Lorieau, and P. Tale. Structural parameterizations of geodetic set on directed (acyclic) graphs. *arXiv eprint*, 2026. arXiv:2606.26414.

6 B. Bergougnoux, O. Defrain, and F. Mc Inerney. Enumerating minimal solution sets for metric graph problems. In *WG 2024*, volume 14760 of *Lecture Notes in Computer Science*, pages 50–64, 2024. doi:10.1007/978-3-031-75409-8_4.

7 D. Chakraborty, S. Das, F. Foucaud, H. Gahlawat, and D. Lajou. Algorithms and complexity for geodetic sets on interval and chordal graphs. *Information and Computation*, 311:105456, 2026. doi:10.1016/J.IC.2026.105456.

8 D. Chakraborty, S. Das, F. Foucaud, H. Gahlawat, D. Lajou, and B. Roy. Algorithms and complexity for geodetic sets on planar and chordal graphs. In *ISAAC 2020*, volume 181 of *LIPIcs*, pages 7:1–7:15, 2020. doi:10.4230/LIPIcs.ISAAC.2020.7.

9 D. Chakraborty, F. Foucaud, H. Gahlawat, S. K. Ghosh, and B. Roy. Hardness and approximation for the geodetic set problem in some graph classes. In *CALDAM 2020*, volume 12016 of *Lecture Notes in Computer Science*, pages 102–115, 2020. doi:10.1007/978-3-030-39219-2_9.

10 D. Chakraborty, H. Gahlawat, and B. Roy. Algorithms and complexity for geodetic sets on partial grids. *Theoretical Computer Science*, 979:114217, 2023. doi:10.1016/J.TCS.2023.114217.

11 G. J. Chang, L. D. Tong, and H. T. Wang. Geodetic spectra of graphs. *European Journal of Combinatorics*, 25(3):383–391, 2004. doi:10.1016/J.EJC.2003.09.010.

12 G. Chartrand, J. F. Fink, and P. Zhang. The hull number of an oriented graph. *International Journal of Mathematics and Mathematical Sciences*, 2003(36):2265–2275, 2003.

13 G. Chartrand and P. Zhang. The geodetic number of an oriented graph. *European Journal of Combinatorics*, 21(2):181–189, 2000. doi:10.1006/EUJC.1999.0301.

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

15 T. Davot, L. Isenmann, and J. Thiebaut. On the approximation hardness of geodetic set and its variants. In *COCOON 2021*, volume 13025 of *Lecture Notes in Computer Science*, pages 76–88, 2021. doi:10.1007/978-3-030-89543-3_7.

16 L. Dong, C. Lu, and X. Wang. The upper and lower geodetic numbers of graphs. *Ars Combinatoria*, 91:401–409, 2009.

17 M. C. Dourado, F. Protti, D. Rautenbach, and J. L. Szwarcfiter. Some remarks on the geodetic number of a graph. *Discrete Mathematics*, 310(4):832–837, 2010. doi:10.1016/J.DISC.2009.09.018.

18 T. Ekim, A. Erey, P. Heggernes, P. van't Hof, and D. Meister. Computing minimum geodetic sets of proper interval graphs. In *LATIN 2012*, volume 7256 of *Lecture Notes in Computer Science*, pages 279–290, 2012. doi:10.1007/978-3-642-29344-3_24.

19 A. Farrugia. Orientable convexity, geodetic and hull numbers in graphs. *Discrete Applied Mathematics*, 148(3):256–262, 2005. doi:10.1016/J.DAM.2005.03.002.

20 F. Foucaud, E. Galby, L. Khazaliya, S. Li, F. Mc Inerney, R. Sharma, and P. Tale. Problems in np can admit double-exponential lower bounds when parameterized by treewidth or vertex cover. In *ICALP 2024*, volume 297 of *LIPIcs*, pages 66:1–66:19, 2024. doi:10.4230/LIPIcs.ICALP.2024.66.

21 F. Foucaud, E. Galby, L. Khazaliya, S. Li, F. Mc Inerney, R. Sharma, and P. Tale. Metric dimension and geodetic set parameterized by vertex cover. In *STACS 2025*, volume 327 of *LIPIcs*, pages 33:1–33:20, 2025. doi:10.4230/LIPIcs.STACS.2025.33.

22 F. Foucaud, N. Ghareghani, L. Lorieau, M. Mohammad Noori, R. Parvini Oskuei, and P. Tale. Algorithms and hardness for geodetic set on tree-like digraphs. In N. Misra and A. Pandey, editors, *CALDAM 2026*, volume 16445 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 2026. doi:10.1007/978-3-032-17156-6_14.

23 F. Foucaud, B. Gras, A. Perez, and F. Sikora. On the complexity of broadcast domination and multipacking in digraphs. *Algorithmica*, 83(9):2651–2677, 2021. doi:10.1007/S00453-021-00828-5.

24 B. Haeupler, Y. Jiang, and T. Saranurak. Reducing shortcut and hopset constructions to shallow graphs. In S. Assadi and E. Rotenberg, editors, *SOSA 2026*, pages 385–393. SIAM, 2026. doi:10.1137/1.9781611978964.30.

25 F. Harary, E. Loukakis, and C. Tsouros. The geodetic number of a graph. *Mathematical and Computer Modelling*, 17(11):89–95, 1993.

26 R. Impagliazzo and R. Paturi. Complexity of k-sat. In *Proceedings of the Fourteenth Annual IEEE Conference on Computational Complexity*, pages 237–240, 1999.

27 L. Kellerhals and T. Koana. Parameterized complexity of geodetic set. *Journal of Graph Algorithms and Applications*, 26(4):401–419, 2022. doi:10.7155/JGAA.00601.

28 M. Lampis, N. Melissinos, and M. Vasilakis. Parameterized max min feedback vertex set. *SIAM Journal on Discrete Mathematics*, 39(3):1587–1620, 2025. doi:10.1137/23M1605247.

29 C. Lu. The geodetic numbers of graphs and digraphs. *Science in China Series A: Mathematics*, 50(8):1163–1172, 2007.

30 M. Mezzini. Polynomial time algorithm for computing a minimum geodetic set in outerplanar graphs. *Theoretical Computer Science*, 745:63–74, 2018. doi:10.1016/J.TCS.2018.05.032.

31 I. M. Pelayo. *Geodesic Convexity in Graphs*. Springer, 2013.

32 P. Tale. Geodetic set on graphs of constant pathwidth and feedback vertex set number. In A. Agrawal and E. Jan van Leeuwen, editors, *IPEC 2025*, volume 358 of *LIPIcs*, pages 28:1–28:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.IPEC.2025.28.

33 C. A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984. doi:10.1016/0166-218X(84)90081-7.

34 C. Wang, Y. Song, G. Fan, H. Jin, L. Su, F. Zhang, and X. Wang. Optimizing cross-line dispatching for minimum electric bus fleet. *IEEE Transactions on Mobile Computing*, 22(4):2307–2322, 2023. doi:10.1109/TMC.2021.3119421.