On the Parameterized Complexity of
[1, j]-Domination Problems

Mohsen Alambardar Meybodi

Department of Computer Science, Yazd University, Yazd, Iran
m.alambardar@stu.yazd.ac.ir

Fedor Fomin

Department of Informatics, University of Bergen, Norway
fomin@ii.uib.no

Amer E. Mouawad

Computer Science Department, American University of Beirut, Beirut, Lebanon
aa368@aub.edu.lb

Fahad Panolan

Department of Informatics, University of Bergen, Norway
fahad.panolan@ii.uib.no

Abstract

For a graph G, a set $D \subseteq V(G)$ is called a $[1, j]$-dominating set if every vertex in $V(G) \setminus D$ has at least one and at most j neighbors in D. A set $D \subseteq V(G)$ is called a $[1, j]$-total dominating set if every vertex in $V(G)$ has at least one and at most j neighbors in D. In the $[1, j]$-(TOTAL) DOMINATING SET problem we are given a graph G and a positive integer k. The objective is to test whether there exists a $[1, j]$-(total) dominating set of size at most k. The $[1, j]$-DOMINATING SET problem is known to be NP-complete, even for restricted classes of graphs such as chordal and planar graphs, but polynomial-time solvable on split graphs. The $[1, 2]$-TOTAL DOMINATING SET problem is known to be NP-complete, even for bipartite graphs. As both problems generalize the DOMINATING SET problem, both are W[1]-hard when parameterized by solution size. In this work, we study $[1, j]$-DOMINATING SET on sparse graph classes from the perspective of parameterized complexity and prove the following results when the problem is parameterized by solution size:

- $[1, j]$-DOMINATING SET is W[1]-hard on d-degenerate graphs for $d = j + 1$;
- $[1, j]$-DOMINATING SET is FPT on nowhere dense graphs.

We also prove that the known algorithm for $[1, j]$-DOMINATING SET on split graphs is optimal under the Strong Exponential Time Hypothesis (SETH). Finally, assuming SETH, we provide a lower bound for the running time of any algorithm solving the $[1, 2]$-TOTAL DOMINATING SET problem parameterized by pathwidth.

1 Introduction

A dominating set of a graph G is a subset D of vertices such that each vertex in $V(G) \setminus D$ is adjacent to at least one vertex in D. Various extensions of domination, such as independent, total, efficient, and perfect domination, have been introduced and widely studied both combinatorially and algorithmically. A discussion of these extensions can be found in [14]. A

$[1, j]$-dominating set, as first defined in [3], is a set that dominates the vertices of the graph but every vertex outside of the set must have at most j neighbors in it. In [3], it was shown that a minimum $[1, 2]$-dominating set and a minimum dominating set are of the same size in several classes of graphs such as claw-free graphs, P_4-free graphs, and caterpillars. It was also shown that the problem is NP-complete, even for bipartite graphs. In the $[1, j]$-Dominating Set problem the input is a graph G and a positive integer k. The objective is to test whether there is a $[1, j]$-dominating set of size at most k. The authors in [3] raised several open problems, including whether restricting to specific classes of graphs leads to strictly better upper bounds for the size of $[1, j]$-dominating sets and whether $[1, j]$-Dominating Set is efficiently solvable on trees. In [29] the first question was answered negatively for the classes of planar, bipartite, and triangle-free graphs in which the smallest $[1, 2]$-dominating set is the entire set of vertices. In [12] the second question was answered positively via a linear-time algorithm. In [2], the $[1, j]$-Dominating Set problem was shown to be NP-hard even for chordal and planar graphs. However, for a constant j, a polynomial-time algorithm running in time $O(n^j p(\lg n))$ where p is a polynomial function, was obtained for n-vertex split graphs [2]. This is in contrast to the classic Dominating Set problem which is NP-hard for this class of graphs.

The Dominating Set problem has been widely studied in the realm of parameterized complexity. In general, finding a dominating set of size k is a canonical W[2]-complete problem and therefore unlikely to admit FPT algorithms [7]. Moreover, the problem remains W[2]-complete for split and bipartite graphs [23]. Nevertheless, there are interesting classes of sparse graphs for which the Dominating Set problem admits FPT algorithms. For example, there is an $O^*(3^{\mathsf{tw}})$-time algorithm for graphs of treewidth at most tw [28, 15], and FPT algorithms for nowhere dense graphs [6] and d-degenarate graphs [1]. Also, an FPT algorithm was reported in [27] for t-biclique-free graphs, i.e., graphs that do not contain $K_{t,t}$ as a subgraph. To the best of our knowledge, this is the largest class of graphs for which the Dominating Set problem is known to be fixed-parameter tractable; d-degenerate and nowhere dense graphs are subclasses of t-biclique-free graphs.

Another variant of dominating sets and $[1, j]$-dominating sets is $[1, j]$-total dominating sets. For a graph G, a subset $D \subseteq V(G)$ is called a $[1, j]$-total dominating set if $1 \leq |N(v) \cap D| \leq j$ for all $v \in V(G)$, where $N(v)$ denotes the open neighbourhood of v in G. In the $[1, j]$-Total Dominating Set problem we are given a graph G and a positive integer k. The objective is to check whether G admits a $[1, j]$-total dominating set of size at most k. $[1, 2]$-Total Dominating Set is NP-complete even for bipartite graphs [24]. Sharp upper bounds on the $[1, 2]$-total domination number of a graph are investigated in [16, 29]. Using a result of Rooij et al. [28], we can show that $[1, j]$-Dominating Set and $[1, j]$-Total Dominating Set are solvable in time $O^*((j + 2)^{\mathsf{tw}})$ and $O^*((2j + 2)^{\mathsf{tw}})$, respectively, on graphs of treewidth at most tw.

Our Contribution.   In this paper, we study $[1, j]$-Dominating Set and $[1, j]$-Total Dominating Set from the parameterized complexity perspective. We prove the following results.

1. For any $\epsilon > 0$, there is no algorithm with running time $O(n^{j-\epsilon})$ for $[1, j]$-Dominating Set on split graphs assuming the Strong Exponential Time Hypothesis (SETH).

2. For the case of d-degenerate graphs, we tighten the complexity results and show that the $[1, j]$-Dominating Set problem is W[1]-hard for $d = j + 1$.

3. $[1, j]$-Dominating Set is FPT on classes of nowhere dense graphs.

4. There is no algorithm for $[1, 2]$-Total Dominating Set running in time $O^*((4 - \epsilon)^{\mathsf{pw}})$ assuming SETH, where pw is the pathwidth of the input graph.

We begin by some basic terminology and notation in Section 2. Then the next sections each address one of the results mentioned above.

2   Preliminaries

Graphs.   We assume G is a simple graph with vertex set $V(G)$ and edge set $E(G)$. For brevity, we often denote these sets by V and E. We let $n = |V(G)|$ denote the order of G. For a vertex $v \in V$, the open neighborhood of v, denoted by $N(v)$, is defined as $\{u : \{u, v\} \in E\}$ and the closed neighborhood $N[v]$ is defined as $N(v) \cup \{v\}$. For a set $S \subseteq V$, we use $N(S)$ and $N[S]$ to denote the open and closed neighborhood S, respectively. That is, $N[S] = \bigcup_{v \in S} N[v]$ and $N(S) = N[S] \setminus S$. For a set $U \subseteq V$, we use $G[U]$ to denote the subgraph of G induced on U. A tree decomposition of a graph G is a tree T in which each vertex $x \in T$ has an assigned set of vertices $B_x \subseteq V(G)$ (called a bag) which satisfies the following properties: (i) $\bigcup_{x \in T} B_x = V(G)$; (ii) For any $\{u, v\} \in E$, there exists $x \in V(T)$ such that $u, v \in B_x$; (iii) For any $v \in V(G)$, the subtree of T induced on $\{x \in V(T) : v \in B_x\}$ is connected. The *width* of a tree decomposition T is $\max_{x \in V(T)}(|B_x| - 1)$. The treewidth of G, denoted by $tw(G)$, is the minimum width over all tree decompositions of G. The pathwidth of G, denoted by $pw(G)$, is the minimum width over all tree decompositions T of G, where T is a path.

Parameterized complexity.   We now review some necessary concepts from parameterized complexity. For more details we refer the reader to [4, 8]. Given a finite alphabet Σ, a parameterization of Σ^* is a function $p : \Sigma^* \to \mathbb{N}$. A parameterized language L is a subset of $\{(x, k) \mid x \in \Sigma^* \land k = p(x)\}$. Here k is called the parameter. A parameterized language $L \subseteq \Sigma^* \times \mathbb{N}$ is called fixed-parameter tractable (FPT) if there exist an algorithm A (called a *FPT algorithm*) and a computable function $f : \mathbb{N} \to \mathbb{N}$ such that given $(x, k) \in \Sigma^* \times \mathbb{N}$, the algorithm A correctly decides whether $(x, k) \in L$ in time $f(k) \cdot |(x, k)|^{O(1)}$. The class of all fixed-parameter tractable problems is denoted by FPT.

▶ **Definition 1.** Let L and L' be two parameterized languages with parameterization functions p and p'. An FPT reduction from L to L' is a mapping $\rho : \Sigma^* \to \Sigma^*$ such that the following holds.

- For all $x \in \Sigma^*$, $(x, p(x)) \in L$ if and only if $(\rho(x), p'(\rho(x))) \in L'$
- There exists a computable function $g : \mathbb{N} \to \mathbb{N}$ such that for all $x \in \Sigma^*$, $p'(\rho(x)) \leq g(p(x))$.
- ρ is computable in FPT time, i.e., there exists a computable function f such that $\rho(x)$ is computable in time $O(f(p(x)) \cdot |x|^{O(1)})$.

To classify problems that are not FPT, Downey and Fellows [8] introduced the W-hierarchy. The hierarchy consists of complexity class W[t] for every integer $t \in \mathbb{N}$ such that $W[t] \subseteq W[t + 1]$, for all t. More generally, FPT $\subseteq$ W[1] $\subseteq$ W[2] $\subseteq \cdots \subseteq$ W[t]. For our purposes, it is sufficient to note that these classes are closed under FPT reduction.

ETH and SETH.   For $q \geq 3$, let δ_q be the infimum of the set of constants c for which there exists an algorithm solving q-SAT with n variables and m clauses in time $2^{cn} \cdot m^{O(1)}$. The *Exponential-Time Hypothesis (ETH)* and *Strong Exponential-Time Hypothesis (SETH)* are then formally defined as follows. ETH conjectures that $\delta_3 > 0$ and SETH that $\lim_{q \to \infty} \delta_q = 1$. In other words, SETH conjectures that for all $0 < \epsilon < 1$, there exist a (large) $q = q(\epsilon)$ such that q-SAT cannot be solved in time $O(2^{(1-\epsilon)n})$, where n is the number of variables in the input formula.
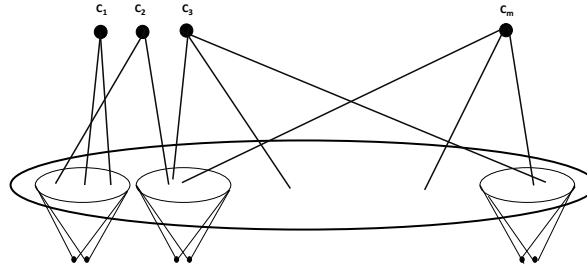
■ **Figure 1** Constructed Split Graph.

3 Split Graphs

A graph G is called a split graph if its vertices can be partitioned into $V_1 \uplus V_2$ such that $G[V_1]$ is a complete graph and $G[V_2]$ is an empty graph (i.e., V_2 is an independent set in G). Several domination-like problems such as domination, total domination, and k-tuple domination are known to be NP-complete even on split graphs. On the other hand, there is a $n^j \cdot (\lg n)^{O(1)}$ time algorithm for $[1, j]$-DOMINATING SET on split graphs [2]. In this section, we prove that this is optimal, in the sense that one cannot obtain an $O(n^{j-\epsilon})$ algorithm for $[1, j]$-DOMINATING SET on split graphs unless SETH fails.

Reduction.   Given an instance $I = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ of q-SAT over the set $X = \{x_1, \ldots, x_n\}$ of variables, we construct a graph G_I as follows.
- We partition the set X into j subsets $X_1, \ldots, X_j$ of size at most $\lceil n/j \rceil$.
- For each X_i we add a set S_i of $2^{|X_i|}$ vertices to the graph, each corresponding to one possible valuation of the variables in X_i. We also add two distinguished vertices u_i and v_i and connect them to all of S_i.
- We connect all the vertices in $\bigcup_{i=1}^{j} S_i$. That is $\bigcup_{i=1}^{j} S_i$ forms a clique in G_I.
- For every clause C_i, we add a vertex c_i and connect it to every vertex w of any S_j where the valuation corresponding to w satisfies C_i.

This completes the construction of G_I. See Figure 1 for an illustration. Our reduction algorithm will output an instance (G_I, j) of $[1, j]$-DOMINATING SET. We now proceed to proving the correctness of the reduction.

▶ **Lemma 2.** *If I is satisfiable then G_I has a $[1, j]$-dominating set of size j.*

Proof. Take one satisfying valuation of I and let $s_1 \in S_1, s_2 \in S_2, \ldots, s_j \in S_j$ be the vertices of G_I that correspond to this valuation. We claim that $S = \{s_1, s_2, \ldots, s_j\}$ is a $[1, j]$-dominating set. Every vertex in any of the S_i's is dominated by all of S, every $\{u_i, v_i\}$ pair is dominated only by the corresponding s_i and every c_i is dominated by a non-empty subset of S, i.e., the vertices whose corresponding valuation forces satisfaction of C_i. Given that $|S| = j$, there can be at most j such vertices. ◀

▶ **Lemma 3.** *If G_I has a $[1, j]$-dominating set of size j then I is satisfiable.*

Proof. First note that any dominating set of G_I of size j is also a $[1, j]$-dominating set. Let S be a dominating set of size at most j (and hence a $[1, j]$-dominating set) in G_I. Since $\{N[u_i] : i \in \{1, \ldots, j\}\}$ are pairwise vertex disjoint we have that $|S| = j$ and $|S \cap N[u_i]| = 1$

for all $i \in \{1, \ldots, j\}$. Moreover, since $N(u_i) = N(v_i)$, we conclude that $S \subseteq (\bigcup_{i=1}^{j} S_i)$. Consider the valuation of variables in X that corresponds to the vertices in S. This valuation satisfies every C_i, because S dominates every c_i. ◀

▶ **Theorem 4.** *For any $\varepsilon < 1$ and constant j, there is no $O(n^{j-\varepsilon})$ time algorithm for* $[1, j]$-DOMINATING SET *on split graphs unless SETH fails.*

Proof. For the sake of contradiction assume that there is an $O(n^{j-\varepsilon})$ time algorithm $\mathcal{A}$ for $[1, j]$-DOMINATING SET on split graphs. Then we claim that SETH is false. Let $\epsilon = (1 - \frac{\varepsilon}{j})$. Let $q = q(\epsilon)$ be the constant defined in the SETH conjecture. Given a q-SAT instance I we construct an instance (G_I, j) as mentioned in the reduction. By Lemmas 2 and 3 we know that I is satisfiable if and only if (G_I, j) is a yes-instance. Therefore, we use algorithm $\mathcal{A}$ to solve q-SAT. Now consider the running time for solving q-SAT using $\mathcal{A}$. The construction of (G_I, j) takes time $2^{\lceil n/j \rceil} \cdot n^{O(1)}$ and the number of vertices in G_I is at most $2j + m + j2^{\lceil n/j \rceil} = 2^{\lceil n/j \rceil} \cdot n^{O(1)}$. Thus the algorithm $\mathcal{A}$ on instance (G_I, j) takes time $2^{\lceil n/j \rceil (j-\varepsilon)} \cdot n^{O(1)} = 2^{(n/j + (j-1))(j-\epsilon)} \cdot n^{O(1)} = 2^{(n/j)(j-\varepsilon)} \cdot n^{O(1)} = 2^{n(1-\epsilon/j)} \cdot n^{O(1)} = 2^{\epsilon n} \cdot n^{O(1)}$. This refutes SETH and the proof of the theorem is complete. ◀

4 Degenerate Graphs

A graph G is d-degenerate if every subgraph of G contains a vertex of degree at most d. Equivalently, a graph G is d-degenerate if and only if there exists an elimination ordering on its vertices such that every vertex has at most d neighbors appearing later in the ordering. In this section we prove the following result.

▶ **Theorem 5.** $[1, j]$-DOMINATING SET *parameterized by solution size is W[1]-hard on graphs of degeneracy $j + 1$.*

The following parameterized problem, proved to be W[1]-hard in [11], is used in our proof. In the MULTICOLORED INDEPENDENT SET problem, we are given a graph G and a proper vertex coloring of $V(G)$ with k colors. The parameter k is equal to the number of colors and the goal is to find a k-sized independent set in G containing exactly one vertex from each color class (such independents sets are called *k-colored independent sets*). We shall reduce the MULTICOLORED INDEPENDENT SET problem (with parameter k) to the problem of finding a $[1, j]$-dominating set of size at most $2k + j - 1$ in a graph of degeneracy $j + 1$.

The reduction. Let k be an integer and G be a proper k-vertex colored graph such that its vertices are partitioned into k groups $V_1, V_2, \ldots, V_k$, where each group corresponds to an independent set of the same color. Now we construct an instance $(G', 2k + j - 1)$ of $[1, j]$-DOMINATING SET as follows. For every edge $e = \{u, v\} \in E(G)$, we replace it by a path $u v_e v$, where v_e is a new vertex corresponding to the edge e. Let $S_E = \{v_e : e \in E(G)\}$. For each group V_i, $1 \le i \le k$, we build a $K_{1,2k+j}$ graph centered at a new vertex u_i. We also add new vertices x_1^i and x_2^i and connect the vertices u_i, x_1^i and x_2^i to the vertices of V_i. Moreover, we build $j - 1$ star graphs $K_{1,2k+j}$ centered at vertices $r_1, \ldots, r_{j-1}$ and make $r_1, \ldots, r_{j-1}$ adjacent to the vertices in S_E. This concludes the construction of G'. See Figure 2 for an illustration. Now we output $(G', 2k + j - 1)$ as an instance of $[1, j]$-DOMINATING SET. Clearly our reduction takes time polynomial in $|V(G)|$ and k.

▶ **Lemma 6.** *The constructed graph G' has degeneracy $j + 1$.*
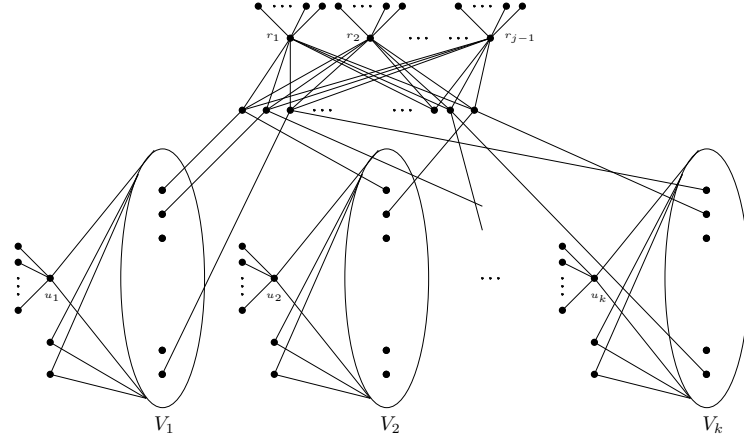
Figure 2 Reducing the MULTICOLORED INDEPENDENT SET problem to the $[1, j]$-DOMINATING SET problem.

Proof. The proof is by constructing a degeneracy ordering, which is an ordering on the vertices that we get from repeatedly removing a vertex of minimum degree in the remaining subgraph. First, we put all of the degree-one vertices in the degeneracy ordering and delete them. In the remaining subgraph, we select all the vertices in S_E and put them in ordering, because every vertex in S_E has degree $j + 1$ in G'. After removing all vertices of S_E from the graph, each vertex in any block V_i, for $1 \leq i \leq k$, has degree three because after removing S_E such vertices are only connected to x_1^i, x_2^i, and u_i. So, next we can put the vertices $\bigcup_{i=1}^{k} V_i$ in the ordering. Finally we add all the remaining vertices. ◀

Lemmas 6, 7, and 8 below, imply Theorem 5.

▶ **Lemma 7.** *If there exists a k-colored independent set in G then there exists a $[1, j]$-dominating set of size $2k + j - 1$ in G'.*

Proof. Suppose that S is a k-colored independent set in G. We claim that $D = S \cup \{u_i : i \in \{1, \ldots, k\}\} \cup \{r_i : i \in \{1, \ldots, j - 1\}\}$ is a $[1, j]$-dominating set of G'. Clearly, the size of D is $2k + j - 1$. Each vertex $v_j \in V_i$ is dominated only by u_i in D. Moreover, each pair of x_1^i, x_2^i vertices is dominated by the single vertex in $V_i \cap S$. All the vertices in S_E are dominated by $\{r_i : i \in \{1, \ldots, j - 1\}$ and by at most one vertex from S (since S is an independent set). Moreover all the degree one vertices in G' are dominated exactly once by D. ◀

▶ **Lemma 8.** *If there exists a $[1, j]$-dominating set of size $2k + j - 1$ in G' then there exists a k-colored independent set in G.*

Proof. Let D be a $[1, j]$-dominating set of size at most $2k + j - 1$. First, note that since $|D| \leq 2k + j - 1$, we have that $\{r_1, \ldots, r_{j-1}\} \cup \{u_i, \ldots, u_k\} \subseteq D$ (because each u_i and r_i is connected to $2k + j$ degree one vertices). We claim that $S = D \setminus (\{r_1, \ldots, r_{j-1}\} \cup \{u_i, \ldots, u_k\})$ is a k-colored independent set in G. Clearly $|S| \leq k$. Also, to dominate all the vertices x_1^i, x_i^2 for $1 \leq i \leq k$, we should have exactly one vertex from each V_i. Therefore $S \subseteq V(G)$ and $|S \cap V_i| = 1$ for all $1 \leq i \leq k$. Suppose $u, v \in S$ is adjacent in G. Then the vertex v_e, where $e = \{u, v\}$ is dominated $j + 1$ times by D (because v_e is adjacent to $\{u, v\}$ and $\{r_1, \ldots, r_{j-1}\}$). Therefore since D is a $[1, j]$-dominating set in G', S is a k-colored independent set in G This completes the proof of the lemma. ◀

5   Nowhere Dense Graphs

The notion of nowhere denseness was introduced by Nešetřil and Ossona de Mendez [20, 21] as a general model of uniform sparseness of graphs. Many familiar classes of sparse graphs, like planar graphs, graphs of bounded tree-width, graphs of bounded degree, and all classes that exclude a fixed (topological) minor, are nowhere dense. An important and related concept is the notion of a graph class of bounded expansion, which was also introduced by Nešetřil and Ossona de Mendez [17, 18, 19].

▶ **Definition 9.** Let H be a graph and let $r \in \mathbb{N}$. An r-subdivision of H is obtained by replacing all edges of H by internally vertex disjoint paths of length at most r.

▶ **Definition 10.** A class $\mathcal{C}$ of graphs is nowhere dense if there exists a function $t\colon \mathbb{N} \to \mathbb{N}$ such that for all $q \in \mathbb{N}$ and for all $G \in \mathcal{C}$ we do not find an q-subdivision of the complete graph $K_{t(q)}$ as a subgraph of G. Otherwise, $\mathcal{C}$ is called somewhere dense.

▶ **Definition 11.** A class $\mathcal{C}$ of graphs has bounded expansion if there exists a function $d\colon \mathbb{N} \to \mathbb{N}$ such that for all $r \in \mathbb{N}$ and all graphs H, such that an r-subdivision of H is a subgraph of G for some $G \in \mathcal{C}$, satisfy $|E(H)|/|V(H)| \leq d(r)$.

Every class of bounded expansion is nowhere dense, which in turn excludes some biclique as a subgraph and hence is biclique-free. For extensive background on bounded expansion and nowhere dense graphs we refer to the textbook of Nešetřil and Ossona de Mendez [22].

Before we state our result, we quickly recall the necessary definitions from logic. For our purpose, it suffices to consider first-order logic over the vocabulary of graphs. We refer to the textbook [9] for extensive background on logic. A (relational) vocabulary is a finite set of relation symbols, each with a prescribed arity. We let σ be a vocabulary. A σ-structure A consist of a (not necessarily finite) set $V(A)$, called the universe or vertex set of A, and for each k-ary relation symbol $R \in \sigma$ a k-ary relation $R(A) \subseteq V(A)^k$. A structure A is finite if its universe is. For example, graphs may be viewed as $\{E\}$-structures, where E is a binary relation symbol. First-order formulas of vocabulary σ are formed from atomic formulas $x = y$ and $R(x_1, \ldots, x_k)$, where $R \in \sigma$ is a k-ary relation symbol and $x, y, x_1, \ldots, x_k$ are variables (we assume that we have an infinite supply of variables) by the usual Boolean connectives $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), and existential and universal quantifications $\exists x$ and $\forall x$, respectively. The set of all first-order formulas of vocabulary σ is denoted by $\mathrm{FO}[\sigma]$, and the set of all first-order formulas by FO. The free variables of a formula are those not in the scope of a quantifier, and we write $\phi(x_1, \ldots, x_k)$ to indicate that the free variables of the formula ϕ are among $x_1, \ldots, x_k$. A sentence is a formula without free variables. To define the semantics, we inductively define a satisfaction relation $\models$, where for a σ-structure A, a formula $\phi(x_1, \ldots, x_k)$, and elements $a_1, \ldots, a_k \in V(A)$, $A \models \phi(a_1, \ldots, a_k)$ means that A satisfies ϕ if the free variables $x_1, \ldots, x_k$ are interpreted by $a_1, \ldots, a_k$, respectively. If $\phi(x_1, \ldots, x_k) = R(x_1, \ldots, x_k)$ is atomic, then $A \models \phi(a_1, \ldots, a_k)$ if $(a_1, \ldots, a_k) \in R(A)$. The meaning of the equality symbol, the Boolean connectives, and the quantifiers is the usual one. For example, consider the formula $\phi(x_1, x_2) = \forall y(x_1 = y \vee x_2 = y \vee E(x_1, y) \vee E(x_2, y))$ in the vocabulary $\{E\}$ of graphs. For every graph G and vertices $v_1, v_2 \in V(G)$ we have $G \models \phi(v_1, v_2)$ if any only if $\{v_1, v_2\}$ is a dominating set of G. Thus G satisfies the sentence $\exists x_1 \exists x_2 \phi(x_1, x_2)$ if and only if it has a (nonempty) dominating set of size at most 2.

▶ **Theorem 12.** *The* $[1, j]$-Dominating Set *problem parameterized by solution size k is fixed-parameter tractable on nowhere dense classes of graphs.*

Proof. Our proof is based on a result of Grohe, Kreutzer, and Siebertz [13], which states that for every first-order sentence ψ (or formula without free variables), every nowhere dense class $\mathcal{C}$ of graphs and every real $\epsilon > 0$, there exists a constant $f(|\psi|, \epsilon)$, such that given an n-vertex graph $G \in \mathcal{C}$, one can decide in time $f(|\psi|, \epsilon) \cdot n^{1+\epsilon}$ whether ψ holds in G.

It is easy to verify that the $[1, j]$-DOMINATING SET problem is expressible in FO. We let ψ be the following sentence.

$$\exists v_1, v_2, \ldots, v_k \forall u \Big((u = v_1 \vee u = v_2, \ldots, u = v_k) \vee ((\phi_1(u, v_1, v_2, \ldots, v_k) \vee \phi_2(u, v_1, v_2, \ldots, v_k)$$

$$\cdots \vee \phi_j(u, v_1, v_2, \ldots, v_k)) \Big),$$

where the function $\phi_i(u, v_1, v_2, \ldots, v_k)$ is true where the vertex u is adjacent to exactly i vertices of $v_1, v_2, \ldots, v_k$, which can be represented using a formula of length bounded by a function of i and k. Note that the length of ψ is bounded by a function depending only on k (and on j, though only as a fixed constant). Then, by fixing any $\epsilon > 0$ and using the result of [13], we conclude that $[1, j]$-DOMINATING SET is fixed-parameter tractable parameterized by solution size k on every nowhere dense class $\mathcal{C}$. ◀

6 Bounded Treewidth Graphs

It is well-known [4] that the DOMINATING SET problem can be solved in $O^*(3^t)$ time on graphs of treewidth at most t. Lokshtanov et al. [15] showed that this is essentialy optimal, i.e., they showed that the problem cannot be solved in $O^*((3 - \epsilon)^t)$ time unless SETH fails. The situation is almost identical for the CONNECTED DOMINATING SET problem. The problem can be solved in $O^*(4^t)$ time, but cannot be solved in time $O^*(4 - \epsilon)^t$ unless SETH fails [5]. The $[1, j]$-DOMINATING SET problem is a special case of the (σ, ρ)-domination problem. The concept of (σ, ρ)-domination was introduced by Telle [26]. Let σ, ρ be a pair of non-empty sets of non-negative integers. A set S of vertices of a graph G is called (σ, ρ)-dominating if for every vertex $v \in S, |S \cap N(v)| \in \sigma$, and for every $v \notin S, |S \cap N(v)| \in \rho$. $[1, j]$-DOMINATING SET and $[1, 2]$-TOTAL DOMINATING SET are special cases of the (σ, ρ)-DOMINATING SET problem. For $[1, j]$-DOMINATING SET, we set $\sigma = \{0, 1, \ldots\}$ and $\rho = \{1, \ldots, j\}$. On the other hand for $[1, 2]$-TOTAL DOMINATING SET we set $\sigma = \rho = \{1, \ldots, j\}$. The following result is due to Rooij et al. [28].

▶ **Proposition 13.** *Let $\sigma, \rho \subseteq \mathcal{N}$ be finite or cofinite. There exists an algorithm with running time $O^*(s^t)$ for finding a (σ, ρ)-dominating set in graphs of treewidth at most t where s is the number of states per vertex used in the representations of the solution.*

▶ **Corollary 14.** $[1, j]$-DOMINATING SET *and* $[1, j]$-TOTAL DOMINATING SET *are solvable in time* $O^*((j + 2)^{\text{tw}})$ *and* $O^*((2j + 2)^{\text{tw}})$, *respectively, on graphs of treewidth at most* tw.

Proof. As mentioned before, $[1, j]$-DOMINATING SET is a special case of (σ, ρ)-DOMINATING SET, where $\sigma = \{0, 1, \ldots\}$ and $\rho = \{1, \ldots, j\}$. The states needed in the representation of the solution in this special case are $\{\rho_0, \rho_1, \ldots, \rho_j\}$ and $*$, where $\rho_i, 0 \leq i \leq j$ means the vertex in the bag is not in the dominating set and dominated i times and $*$ means that the vertex is in the dominating set. Then by Proposition 13, $[1, j]$-DOMINATING SET is solvable in time $O^*((j + 2)^{\text{tw}})$.

$[1, j]$-TOTAL DOMINATING SET is the special case of (σ, ρ)-DOMINATING SET, with $\sigma = \rho = \{1, \ldots, j\}$. In this case the states needed in the representation of the solution are $\{\sigma_0, \rho_0, \sigma_1, \rho_1, \ldots, \sigma_j, \rho_j\}$, where $\rho_i, 0 \leq i \leq j$ means the vertex in the bag is not in
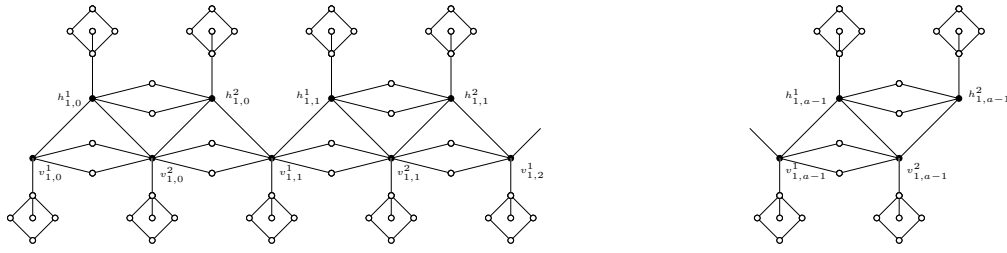
Figure 3 Parts of the construction.

the dominating set and dominated i times and σ_i, $0 \le i \le j$ means the vertex in the bag is in the dominating set and i of its neighbours are also in the dominating set. Then by Proposition 13, $[1, j]$-Total Dominating Set is solvable in time $O^*((2j + 2)^{\mathsf{tw}})$. ◀

In what follows, we prove that the $[1, 2]$-Total Dominating Set problem cannot be solved in time $O^*(4 - \epsilon)^{\mathsf{pw}}$ (unless SETH fails) and which is also a lower bound in terms of the treewidth of the input graph. It remains open whether a similar result holds for $[1, 2]$-Dominating Set. Our proof closely follow the work of Cygan et al. [5] and we use the notions of path decompositions, pathwidth, tree decompositions, treewidth, and mixed search games. We give a reduction from CNF-SAT to the $[1, 2]$-Total Dominating Set problem and prove that the reduced graph has pathwidth at most $\frac{n}{2} + O(1)$, where n is the number of variables in the input CNF-SAT formula.

The reduction. Given $\epsilon > 0$ and an instance Φ of CNF-SAT with n variable and m clauses, we construct a graph G as follows. We assume that the number of variables n is even, otherwise we add a single dummy variable. We partition the variables of Φ into groups $F_1, F_2, \ldots, F_{n'}$, each of size two, where $n' = n/2$. We let $a = m(n + 1)$.

For each $1 \le t \le n'$ we create a path P_t of length $4a = 4m(n + 1)$ consisting of vertices $v_{t,q}^\alpha$ and $h_{t,q}^\alpha$, where $1 \le \alpha \le 2$ and $0 \le q < a$. The vertices are arranged on the path in the following order: $v_{t,0}^1, h_{t,0}^1, v_{t,0}^2, h_{t,0}^2, v_{t,1}^1, h_{t,1}^1, v_{t,1}^2, h_{t,1}^2, \ldots, v_{t,a-1}^1, h_{t,a-1}^1, v_{t,a-1}^2, h_{t,a-1}^2$. We let $\mathcal{V}$ and $\mathcal{H}$ denote the sets of all $v_{t,q}^\alpha$ vertices and $h_{t,q}^\alpha$ vertices, respectively.

For each vertex $v_{t,q}^\alpha$, we add a forcing gadget consisting of a 4-cycle with one additional pendant vertex connected to a vertex which we denote as the root vertex $r_{t,q}^\alpha$. We add an edge between $v_{t,q}^\alpha$ and the root of the cycle $r_{t,q}^\alpha$. Similarly, for each vertex $h_{t,q}^\alpha$, we add a forcing gadget consisting of a 4-cycle rooted at vertex $s_{t,q}^\alpha$ (i.e. the pendant vertex is connected to $s_{t,q}^\alpha$). We add an edge between $h_{t,q}^\alpha$ and $s_{t,q}^\alpha$ (see Figure 3). Note that any dominating set in the graph will contain at least two vertex from a forcing gadget and if a $[1, 2]$-total dominating set contains only two vertex from a forcing gadget, then it include the root vertex and one of its neighbours on the 4-cycle (by the definition of $[1, 2]$-total dominating set).

Next, we add three pairs of guard vertices $p_{t,q}^1$, $p_{t,q}^2$, and $p_{t,q}^3$, for each $1 \le t \le n'$ and $0 \le q < a$. Each of the vertices in these pairs are of degree two and are connected to other vertices as follows: (i) vertices in $p_{t,q}^1$ are adjacent to $v_{t,q}^1$ and $v_{t,q}^2$; (ii) vertices in $p_{t,q}^2$ are adjacent to $v_{t,q}^2$ and $v_{t,q+1}^2$; and (iii) vertices in $p_{t,q}^3$ are adjacent to $h_{t,q}^1$ and $h_{t,q}^2$. This structure forces a dominating set to either contain the vertices from the guard sets or one of their two neighbors. For instance, to dominate the pair $p_{t,q}^1$, either both vertices in $p_{t,q}^1$ must be in the dominating set or one of $v_{t,q}^1$ or $v_{t,q}^2$. We use $\mathcal{G}$ to denote the set of all guard vertices.

The intuition of the construction made so far is as follows. For each two-variable block F_t we encode any assignment of the variables in F_t as a choice of whether to take $v_{t,q}^1$ or $v_{t,q}^2$ and $h_{t,q}^1$ or $h_{t,q}^2$ into the dominating set. This concludes the construction of the "variable gadgets" which are required for encoding an assignment. The forcing gadgets attached to vertices in $\mathcal{V}$ and $\mathcal{H}$ will guarantee that those vertices are all dominated at least once.

We now add "clause gadgets" required for checking the satisfiability of Φ. For each clause C_i, we build $(n+1)$ vertices $c_{i,j}$, one for each $0 \leq j \leq n$. Consider a clause C_i and a group of variables $F_t = \{x_t^1, x_t^2\}$. If x_t^1 occurs positively as the ℓth literal in C_i, then connect $v_{t,mj+i}^1$ to $c_{i,j}$ via a path of length five by adding four vertices $w_{i,j}^\ell$, $x_{i,j}^\ell$, $y_{i,j}^\ell$ and $z_{i,j}^\ell$ and edges $\{c_{i,j}, w_{i,j}^\ell\}$, $\{w_{i,j}^\ell, x_{i,j}^\ell\}$, $\{x_{i,j}^\ell, y_{i,j}^\ell\}$, $\{y_{i,j}^\ell, z_{i,j}^\ell\}$, and $\{z_{i,j}^\ell, v_{t,mj+i}^1\}$. If x_t^1 occurs negatively as the ℓth literal in C_i, we connect $v_{t,mj+i}^2$ to $c_{i,j}$ again via a path of length five by adding four vertices $w_{i,j}^\ell$, $x_{i,j}^\ell$, $y_{i,j}^\ell$ and $z_{i,j}^\ell$ and edges $\{c_{i,j}, w_{i,j}^\ell\}$, $\{w_{i,j}^\ell, x_{i,j}^\ell\}$, $\{x_{i,j}^\ell, y_{i,j}^\ell\}$, $\{y_{i,j}^\ell, z_{i,j}^\ell\}$, and $\{z_{i,j}^\ell, v_{t,mj+i}^2\}$. Similarly, if x_t^2 occurs positively as the ℓth literal in C_i, we connect $h_{t,mj+i}^1$ to $c_{i,j}$ via a path of length five and if it occurs negatively, we connect $h_{t,mj+i}^2$ to $c_{i,j}$ via a path of length five. We let $\mathcal{W}$, $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ denote the sets of all w, x, y, and z vertices added between clause vertices and vertices in $\mathcal{V} \cup \mathcal{H}$, respectively. This concludes the construction of the graph G.

▶ **Lemma 15.** *If Φ is satisfiable then G has a $[1, 2]$-total dominating set D of size $k = 10an' + 2(n+1)\Sigma_{i=1}^m |C_i|$.*

Proof. Consider a satisfying assignment ϕ for Φ. We construct a $[1, 2]$-total dominating set D of size k in G as follows. We first add the root vertex and one of its neighbors on the 4-cycle of each forcing gadget to D. Then, for each block $F_t = \{x_t^1, x_t^2\}$ and each $0 \leq q < a$, we add the vertex $v_{t,q}^1$ to D if the value of x_t^1 is true, otherwise we add $v_{t,q}^2$ to D. Similarly, if the value of x_t^2 is true, we add $h_{t,q}^1$ to D, otherwise we add $h_{t,q}^2$. Since the clause C_i is satisfied, at least one of the vertices $v_{t,q}^\alpha$ or $h_{t,q}^\alpha$ (for some t, α) will be in the dominating set D. Hence, the corresponding $z_{i,j}^\ell$ vertex will be dominated, for some $1 \leq \ell \leq |C_i|$. We can therefore add vertices $x_{i,j}^\ell$ and $w_{i,j}^\ell$ to D as to dominate $c_{i,j}$ and $y_{i,j}^\ell$ (and maintain the $[1, 2]$-total dominating set property). To dominate the remaining vertices in the clause gadget for $c_{i,j}$, we add all $y_{i,j}^{\ell'}$ and $x_{i,j}^{\ell'}$ vertices, where $1 \leq \ell' \leq |C_i|$ and $\ell' \neq \ell$.

It is clear that for each clause C_i, we add $2(n+1)|C_i|$ vertices to D, for a total of $2(n+1)\Sigma_{i=1}^m |C_i|$ vertices. Moreover, for each path P_t, we add $10a$ vertices to D. Therefore, accounting for the fact that we have n' paths in total, the total number of vertices in D is $10an' + 2(n+1)\Sigma_{i=1}^m |C_i| = k$. It remains to show that D is in fact a $[1, 2]$-total dominating set. Every vertex in $\mathcal{V} \cup \mathcal{H}$ is dominated at most twice; once by a vertex in its corresponding forcing gadget and possibly once by a neighbor in $\mathcal{V} \cup \mathcal{H}$. Each vertex in the guard sets is dominated exactly once and each vertex in $\mathcal{W}$, $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ is dominated at most twice (as they have degree two). Finally, each vertex $c_{i,j}$ is dominated exactly once (by a vertex in $\mathcal{W}$), as needed. This concludes the proof of the lemma.   ◀

▶ **Lemma 16.** *If G has a $[1, 2]$-total dominating set of size at most $k = 10an' + 2(n+1)\Sigma_{i=1}^m |C_i|$ then Φ is satisfiable.*

Proof. Consider a $[1, 2]$-total dominating set D of size at most k. The set D must contain at least 2 vertex from each forcing gadget. This constitute at least $8an'$ vertices from forcing gadgets. Since $2(n+1)|C_i|$ vertices are required to total dominate the vertices in the clause gadget for C_i, we know that $|D \cap (\mathcal{W} \cup \mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z})| \geq 2(n+1)\Sigma_{i=1}^m |C_i|$ and, therefore, $|D \cap (\mathcal{V} \cup \mathcal{H} \cup \mathcal{G})| \leq 2an'$. Moreover, G contains n' paths each of length $4a$ and the guard vertices in $\mathcal{G}$ ensure that at least $2a$ vertices are required to dominate the vertices of each path.

Finally, since we have $2an'$ pairs of guards with disjoint neighbors, exactly one vertex from $\{v_{t,q}^1, v_{t,q}^2\}$ and one vertex from $\{h_{t,q}^1, h_{t,q}^2\}$ must be in D, for each $1 \leq t \leq n'$ and $0 \leq q < a$. This implies that each forcing gadget will contribute two vertices to D– the root vertex and one of its neighbour in the 4-cycle. Moreover $|D \cap (\mathcal{W} \cup \mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z})| = 2(n+1)\Sigma_{i=1}^m |C_i|$. This implies that exactly two vertex from a path from $c_{i,j}$ to a vertex in $\mathcal{V} \cup \mathcal{H}$ is included in D. Moreover, this in turn implies that $c_{i,j} \notin D$ for any i, j.

Now, for each $0 \leq q < a$, we construct an assignment ϕ_q as follows. For each block $F_t = \{x_t^1, x_t^2\}$, we define $\phi_q(x_t^1)$ as true if $v_{t,q}^1 \in D$ and we define $\phi_q(x_t^1)$ to be false if $v_{t,q}^2 \in D$. Similarly, for x_t^2, we define $\phi_q(x_t^2)$ to be true if $h_{t,q}^1 \in D$ and we define $\phi_q(x_t^2)$ to be false if $h_{t,q}^2 \in D$. Note that for each block $F_t = \{x_t^1, x_t^2\}$ and each $0 \leq q < a$, we have:

- If $\phi_q(x_t^1)$ is true then $\phi_{q+1}(x_t^1)$ is true. Otherwise, both $v_{t,q}^2$ and $v_{t,q+1}^1$ are not in the dominating set and vertices in $p_{t,q}^2$ is not dominated by D.
- If $\phi_q(x_t^2)$ is false then $\phi_{q+1}(x_t^2)$ is false. Otherwise, both $h_{t,q}^2$ and $h_{t,q+1}^1$ are in the dominating set and vertex $v_{t,q+1}^1$ is dominated three times in D; by $h_{t,q}^2$, $h_{t,q+1}^1$, and the root of the forcing gadget corresponding to $v_{t,q+1}^1$.

For each variable x, we define a sequence $\hat{\phi}_x = \phi_0(x)\phi_1(x) \ldots \phi_{a-1}(x)$. By the discussion above we infer that for each variable x, the sequence $\hat{\phi}_x$ can change its value at most once. Hence, as $a = m(n+1)$, we conclude that there exists $0 \leq j < n+1$ such that for all $0 \leq i < m$, the assignment $\phi_{mj+i}(x)$ are equal. We claim that the assignment $\phi = \phi_{mj}(x)$ satisfies ϕ. Consider a clause C_i and focus on the vertex $c_{i,j}$. We know that $c_{i,j} \notin D$. Thus one of its neighbors from $\mathcal{W}$ (say $w_{i,j}^{\ell'}$) is contained in D. Therefore, by the definition of a $[1,2]$-total dominating set, $x_{i,j}^{\ell'}$ is in D. We have already argued that any path from $c_{i,j}$ to any vertex in $\mathcal{V} \cup \mathcal{H}$ can contain at most 2 vertices in D. This implies that $z_{i,j}^{\ell'}$ is dominated by a vertex in $\mathcal{V} \cup \mathcal{H}$ and the literal corresponding to it will be set to true by ϕ. This completes the proof of the lemma.   ◄

▶ **Lemma 17.** $pw(G) \leq n' + O(1)$.

Proof. Let us first recall the definition of a mixed search game. In a mixed search game, the graph G represents a "system of tunnels". Initially, all edges are contaminated by a gas. An edge is cleared by placing searchers at both its endpoints simultaneously or by sliding a searcher along the edge. A cleared edge is re-contaminated if there is a path from an uncleared edge to the cleared edge without any searchers on its vertices or edges. A search is a sequence of operations that can be of the following types: (i) placement of a new searcher on a vertex; (ii) removal of a searcher from a vertex; (iii) sliding a searcher on a vertex along an incident edge and placing the searcher on the other end. A search strategy is winning if after its termination all edges are cleared. The mixed search number of a graph G, denoted by $ms(G)$, is the minimum number of searchers required for a winning strategy of mixed searching on G. Takahashi et al. [25] obtained the following relationship between $pw(G)$ and $ms(G)$: $pw(G) \leq ms(G) \leq pw(G) + 1$.

We give a mixed strategy to clean the graph with $n' + O(1)$ searchers. The cleaning process will be done in a rounds. At each round $0 \leq q < a$, we put a searcher on clause variable $c_{i,j}$ and keep this searcher there until the cleaning round is completed. It is clear we could clean each forcing gadget by four searcher and keep searcher on the root vertex after cleaning the edges in the force gadget. For each $1 \leq t \leq n'$ and $0 \leq q < a$, after cleaning four connected forcing gadgets we slide the searchers from the root of forcing gadgets to the vertices $v_{t,q}^1, h_{t,q}^1, v_{t,q}^2, h_{t,q}^2$. Then we put searchers on the the (at most two) z vertices connected to them and the guard vertices in sets $p_{t,q}^1, p_{t,q}^2$ and $p_{t,q}^3$. As we have a searcher
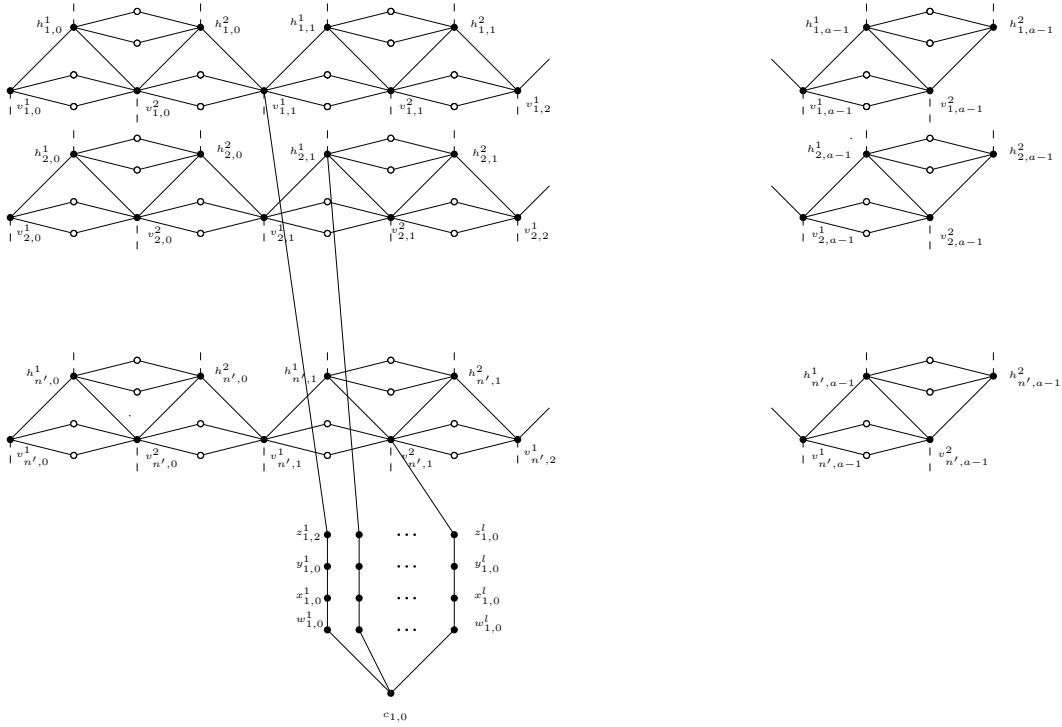
Figure 4 Parts of the construction. Dashed edges are connecting vertices with a forcing gadget.

on $c_{i,j}$ unil the end of this round we could clean the paths between vertex $c_{i,j}$ and those z vertices using one more searcher. The last step of the round is removing the searchers from vertices $v_{t,q}^1, h_{t,q}^1, v_{t,q}^2, h_{t,q}^2$ and searchers on the gaurd sets $p_{t,q}^1, p_{t,q}^2$ and $p_{t,q}^3$. We just keep the searcher on $v_{t,q+1}^1$. Also, we remove searchers in path between $c_{i,j}$ and $\mathcal{V} \cup \mathcal{H}$ except for the one standing on the $c_{i,j}$. To commence the next round, the searcher in $c_{i,j}$ is deleted and a new searcher is put on $c_{i,j+1}$. After the last round the whole graph G is cleaned. Since at any point in time we need at most 14 searchers (they are on $c_{i,j}$, two z vertices, $v_{t,q}^1, h_{t,q}^1$, $v_{t,q}^2, h_{t,q}^2$, guard vertices in sets $p_{t,q}^1, p_{t,q}^2$ and $p_{t,q}^3$, one searcher to clean the paths between two z vertices and $c_{i,j}$) and we reuse 14 searchers in the cleaning process, $n' + 14$ searchers suffice to clean the graph. ◀

▶ **Theorem 18.** *Assuming SETH, for any $\epsilon > 0$ there is no algorithm running in time $(4 - \epsilon)^{\mathsf{pw}} |V(G)|^{O(1)}$ for $[1,2]$-TOTAL DOMINATING SET on a graph G with pathwidth* pw.

Proof. Suppose that $[1,2]$-TOTAL DOMINATING SET can be solved in time $(4-\epsilon)^{\mathsf{pw}} |V(G)|^{O(1)}$. Given an instance of CNF-SAT, we can construct an instance of $[1,2]$-TOTAL DOMINATING SET using the above construction and solve it with a $(4-\epsilon)^{\mathsf{pw}} |V(G)|^{O(1)}$ time algorithm. The correctness of the algorithm follows from Lemmata 15 and 16. Lemma 17 implies that the running time of the algorithm is $(4-\epsilon)^{\frac{n}{2}} |V(G)|^{O(1)}$. However, we have $(4-\epsilon)^{\frac{n}{2}} = (\sqrt{4-\epsilon})^n$ and $\sqrt{4-\epsilon} < 2$ which refutes SETH. This concludes the proof. ◀

7   Conclusion

We have shown that the $[1, j]$-Dominating Set problem parameterized by solution size is W[1]-hard on graphs of degeneracy $(j + 1)$. It is thus natural to ask whether the problem becomes fixed-parameter tractable when the degeneracy of the graph is smaller or equal to j. In particular, is the $[1, j]$-Dominating Set problem fixed-parameter tractable on j-degenerate graphs? There is a very rich literature [6, 1, 10] on kernelization for the Dominating Set problem on sparse graphs and it would be interesteing to see where (if anywhere) the known techniques for the Dominating Set problem become applicable to the $[1, j]$-Dominating Set problem. Finally, we have shown a lower bound of $O^*((4 - \epsilon)^{\mathsf{tw}})$ for $[1, 2]$-Total Dominating Set assuming SETH, while the known upper bound is $O^*(6^{\mathsf{tw}})$. Closing this gap is an interesting open problem. On the other hand $[1, 2]$-Dominating Set can be solved in time $O^*(4^{\mathsf{tw}})$ and getting a matching lower bound is another open problem. Moreover, can we get lower bounds for more general problems such as $[1, j]$-Dominating Set and $[1, j]$-Total Dominating Set?

───── **References** ─────

1   Noga Alon and Shai Gutner. Linear time algorithms for finding a dominating set of fixed size in degenerated graphs. *Algorithmica*, 54(4):544, 2009.

2   Arijit Bishnu, Kunal Dutta, Arijit Ghosh, and Subhabrata Paul. (1, j)-set problem in graphs. *Discrete Mathematics*, 339(10):2515–2525, 2016.

3   Mustapha Chellali, Teresa W Haynes, Stephen T Hedetniemi, and Alice McRae. [1, 2]-sets in graphs. *Discrete Applied Mathematics*, 161(18):2885–2893, 2013.

4   Marek Cygan, Fedor V Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*, volume 3. Springer, 2015.

5   Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michal Pilipczuk, Joham MM van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 150–159. IEEE, 2011.

6   Anuj Dawar and Stephan Kreutzer. Domination Problems in Nowhere-Dense Classes. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2009, December 15-17, 2009, IIT Kanpur, India*, volume 4 of *LIPIcs*, pages 157–168. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2009. `doi:10.4230/LIPIcs.FSTTCS.2009.2315`.

7   Rod G Downey and Michael R Fellows. Fixed-parameter tractability and completeness I: Basic results. *SIAM Journal on Computing*, 24(4):873–921, 1995.

8   Rodney G Downey and Michael Ralph Fellows. *Parameterized complexity*. Springer Science & Business Media, 2012.

9   Heinz-Dieter Ebbinghaus, Jörg Flum, and Wolfgang Thomas. *Mathematical logic (2. ed.)*. Undergraduate texts in mathematics. Springer, 1994.

10   Eduard Eiben, Mithilesh Kumar, Amer E. Mouawad, Fahad Panolan, and Sebastian Siebertz. Lossy Kernels for Connected Dominating Set on Sparse Graphs. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, volume 96 of *LIPIcs*, pages 29:1–29:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018. `doi:10.4230/LIPIcs.STACS.2018.29`.

11   Michael R Fellows, Danny Hermelin, Frances Rosamond, and Stéphane Vialette. On the parameterized complexity of multiple-interval graph problems. *Theoretical Computer Science*, 410(1):53–61, 2009.

12 Amir Kafshdar Goharshady, Mohammad Reza Hooshmandasl, and M Alambardar Meybodi. [1, 2]-sets and [1, 2]-total sets in trees with algorithms. *Discrete Applied Mathematics*, 198:136–146, 2016.

13 Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. In *STOC 2014*, pages 89–98. ACM, 2014.

14 Teresa W Haynes, Stephen Hedetniemi, and Peter Slater. *Fundamentals of domination in graphs*. CRC Press, 1998.

15 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Known Algorithms on Graphs of Bounded Treewidth Are Probably Optimal. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '11, pages 777–789, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics. URL: `http://dl.acm.org/citation.cfm?id=2133036.2133097`.

16 Xuezheng Lv and Baoyindureng Wu. Total [1, 2]-domination in graphs. *arXiv preprint arXiv:1503.04939*, 2015.

17 Jaroslav Nešetřil and Patrice Ossona de Mendez. Grad and classes with bounded expansion I. Decompositions. *European Journal of Combinatorics*, 29(3):760–776, 2008.

18 Jaroslav Nešetřil and Patrice Ossona de Mendez. Grad and classes with bounded expansion II. Algorithmic aspects. *European Journal of Combinatorics*, 29(3):777–791, 2008.

19 Jaroslav Nešetřil and Patrice Ossona de Mendez. Grad and classes with bounded expansion III. Restricted graph homomorphism dualities. *European Journal of Combinatorics*, 29(4):1012–1024, 2008.

20 Jaroslav Nešetřil and Patrice Ossona de Mendez. First order properties on nowhere dense structures. *The Journal of Symbolic Logic*, 75(03):868–887, 2010.

21 Jaroslav Nešetřil and Patrice Ossona de Mendez. On nowhere dense graphs. *European Journal of Combinatorics*, 32(4):600–617, 2011.

22 Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity — Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012.

23 Venkatesh Raman and Saket Saurabh. Short cycles make W-hard problems hard: FPT algorithms for W-hard problems in graphs with no short cycles. *Algorithmica*, 52(2):203–225, 2008.

24 P Sharifani and MR Hooshmandasl. Some Results on [1, k]-sets of Lexicographic Products of Graphs. *arXiv preprint arXiv:1708.00219*, 2017.

25 Atsushi Takahashi, Shuichi Ueno, and Yoji Kajitani. Mixed searching and proper-pathwidth. *Theoretical Computer Science*, 137(2):253–268, 1995.

26 Jan Arne Telle. Complexity of domination-type problems in graphs. *Nord. J. Comput.*, 1(1):157–171, 1994.

27 Jan Arne Telle and Yngve Villanger. FPT algorithms for domination in biclique-free graphs. In *ESA*, pages 802–812. Springer, 2012.

28 Johan MM Van Rooij, Hans L Bodlaender, and Peter Rossmanith. Dynamic Programming on Tree Decompositions Using Generalised Fast Subset Convolution. In *ESA*, volume 9, pages 566–577. Springer, 2009.

29 Xiaojing Yang and Baoyindureng Wu. [1, 2]-domination in graphs. *Discrete Applied Mathematics*, 175:79–86, 2014.