THE COMPUTATIONAL COMPLEXITY OF POSITIVE NON-CLASHING TEACHING IN GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the classical and parameterized complexity of computing the positive non-clashing teaching dimension of a set of concepts, that is, the smallest number of examples per concept required to successfully teach an intelligent learner under the considered, previously established model. For any class of concepts, it is known that this problem can be effortlessly transferred to the setting of balls in a graph G. We establish (1) the NP-hardness of the problem even when restricted to instances with positive non-clashing teaching dimension $k = 2$ and where all balls in the graph are present, (2) near-tight running time upper and lower bounds for the problem on general graphs, (3) fixed-parameter tractability when parameterized by the vertex integrity of G, and (4) a lower bound excluding fixed-parameter tractability when parameterized by the feedback vertex number and pathwidth of G, even when combined with k. Our results provide a nearly complete understanding of the complexity landscape of computing the positive non-clashing teaching dimension and answer open questions from the literature.

1 INTRODUCTION

While typical machine learning models task a learner with finding a concept C from a concept class $\mathcal{C}$ based on an—often randomly drawn—sample, in *machine teaching* (specifically in its commonly considered batch variant) the learner is provided a set of examples by a teacher; crucially, here the examples can be selected in a way which allows the concept to be reconstructed from as few examples as possible. Machine teaching is a core topic in computational learning theory and has found applications in a variety of areas including robotics (Thomaz & Cakmak, 2009; Akgun et al., 2012), trustworthy AI (Mei & Zhu, 2015; Zhang et al., 2018), inverse reinforcement learning (Ho et al., 2016; Brown & Niekum, 2019), and education (Zhu, 2015; Chen et al., 2018; Zhu et al., 2018). While numerous models of machine teaching have been investigated to date (Shinohara & Miyano, 1991; Goldman & Kearns, 1995; Goldman & Mathias, 1996; Zilles et al., 2011; Gao et al., 2017; Mansouri et al., 2019; Telle et al., 2019), in this article we focus on the recently developed *positive non-clashing teaching model* (Kirkpatrick et al., 2019; Fallat et al., 2023).

In non-clashing teaching, given a finite binary concept class $\mathcal{C}$, for each pair C_1, C_2 of distinct concepts in $\mathcal{C}$, at least one example provided for at least one of C_1 or C_2 must not be consistent with the other concept. A key feature of non-clashing teaching is that it is the most efficient model (in terms of the number of required examples) satisfying the Goldman-Mathias collusion-avoidance criterion (Goldman & Mathias, 1996)—the "gold standard" for ensuring that the learner cannot cheat, *e.g.*, via a hidden communication channel with the teacher. Moreover, a teaching model is *positive* if the examples provided for each concept C are required to be positively labeled for C. The restriction of teaching models to the positive setting is common and well-motivated from applications in, *e.g.*, recommendation systems (Schwab et al., 2000), computational biology (Wang et al., 2006; Yousef et al., 2008), and grammatical inference (Stolcke & Omohundro, 1994; Denis, 2001); see also the early works of Angluin (Angluin, 1980a;b). Non-clashing teaching has been proposed and studied in the positive setting not only within the initial papers introducing the concept (Kirkpatrick et al., 2019; Fallat et al., 2023), but also in subsequent works (*e.g.*, (Simon, 2023; Chalopin et al., 2024)).

While positive non-clashing teaching has the potential to be highly efficient, realizing this potential requires us to solve the computational task of actually constructing a small set of examples for the given concepts. More precisely, one aims at computing (a witness for) the teaching dimension of a

given concept class, *i.e.*, the minimum integer k such that each concept is provided with at most k examples that satisfy the conditions of the model. On the positive side, instead of considering various different types of concepts and examples, we can restrict our attention to the setting where each concept is a ball in some input-specified graph G, and the possible examples are vertices of G. Indeed, it is known that any finite binary concept class $\mathcal{C} \subseteq 2^V$ can be represented by a set $\mathcal{B}$ of balls in a graph G as follows: $V(G) = V \cup \{x_C \mid C \in \mathcal{C}\}$, x_C is adjacent to $x_{C'}$ for all $C, C' \in \mathcal{C}$, x_C is adjacent to $v \in V$ if and only if $v \in C$, and $\mathcal{B} = \{B_1(x_C) \mid C \in \mathcal{C}\}$ (Chalopin et al., 2023; 2024).

On the negative side, the problem is computationally intractable, and remains so even in the restricted setting where *every* possible concept (*i.e.*, every ball in G) is present; to distinguish this case from the general one where not all concepts need to be present, we refer to it as *strict*. In particular, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) recently carried out an initial complexity-theoretic investigation of computing the positive non-clashing teaching dimension in the strict setting. There, they established the NP-hardness of the problem for instances with large teaching dimension (even when restricted to the highly restricted class of split graphs), obtained runtime upper and lower bounds under the *Exponential Time Hypothesis* (Impagliazzo et al., 2001), and designed a so-called *fixed-parameter* algorithm for the problem when parameterized by the size of the vertex cover of G.

In this article, we significantly improve over each of these results, answer two open questions posed by the authors of the aforementioned work (Chalopin et al., 2024), and obtain a nearly complete understanding of the computational complexity of computing the positive non-clashing teaching dimension (in both the strict and non-strict settings).

Contributions. Let us refer to the problems of computing the positive non-clashing teaching dimension in the strict and non-strict settings as STRICT NON-CLASH and NON-CLASH, respectively. Formal definitions complementing the informal descriptions given above are provided in Section 2.

Our first result concerns the complexity of STRICT NON-CLASH on instances with constant positive non-clashing teaching dimension. The reductions of Chalopin, Chepoi, Mc Inerney, and Ratel (2024) only establish the NP-hardness of the problem for instances with large (*i.e.*, input-dependent) positive non-clashing teaching dimension k. In fact, as their first open question, the authors ask whether STRICT NON-CLASH is NP-hard or polynomial-time solvable when the sought-after dimension k is a fixed constant; the question is not only theoretically interesting, but also highly relevant as instances of small teaching dimension are precisely the candidates for efficient teaching. We settle this by a highly non-trivial reduction (Theorem 1) which establishes that determining whether the positive non-clashing teaching dimension is at most 2 is NP-hard—and remains so even when restricted to the same class of *split graphs* where STRICT NON-CLASH was previously shown to be NP-hard (for large k) (Chalopin et al., 2024). We note that this result is, in a sense, best possible: determining whether an instance of STRICT NON-CLASH has a positive non-clashing teaching dimension of 1 is trivial as it is equivalent to testing whether G is edgeless (Chalopin et al., 2024).

Next, we proceed to the running time bounds for solving the problem. Typically, the running time upper bounds are given by an exact algorithm, while the lower bounds are obtained from a suitable "tight" reduction under the Exponential Time Hypothesis (Impagliazzo et al., 2001). In the preceding work, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) obtained algorithmic lower and upper bounds of $2^{o(n \cdot d)}$ and $2^{\mathcal{O}(n^2 \cdot d)}$, where n and d are the number of vertices and the diameter of G, respectively. From our reduction and a more careful algorithmic analysis, we obtain a lower bound of $2^{o(n \cdot d \cdot k)}$ and an upper bound of $2^{\mathcal{O}(n \cdot d \cdot k \cdot \log n)}$ (Theorem 4 and Proposition 5)—making the bounds almost tight, with just a logarithmic factor in the exponent separating the two.

While the aforementioned bounds apply to the problem in general, often one may only need to solve the problem on "well-structured" graphs. The more refined *parameterized complexity* paradigm (Downey & Fellows, 2013; Cygan et al., 2015) offers the perfect tools to analyze and identify precisely which structural properties of input graphs—usually captured by a suitable integer *parameter* k—allow us to circumvent its general intractability. The analog to the complexity class P in the parameterized setting is FPT ("*fixed-parameter tractable*"), which characterizes parameterized problems solvable in $f(k) \cdot n^{\mathcal{O}(1)}$ time; intuitively, this means that the problem is solvable in uniformly polynomial time for each constant value of k. Parameterized complexity is well-established and has been successfully applied for non-clashing teaching (Chalopin et al., 2024) as well as in a variety of related subfields of learning theory (Downey et al., 1993; Li & Liang, 2018; Ganian & Korchemna, 2021; Ordyniak & Szeider, 2021; Brand et al., 2023; Eiben et al., 2023a;b).

In their previous work, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) established the fixed-parameter tractability of STRICT NON-CLASH when parameterized by the *vertex cover number* of the input graph G—or, equivalently, the vertex deletion distance to a graph consisting only of isolated vertices. The drawback of that result is that the vertex cover number is a highly "restrictive" parameter, in the sense that it achieves low values only on rather simple graphs. This is also reflected in the open question posed in that article, which asked about the problem's complexity under other parameterizations. As our third contribution, we establish (in Theorem 17) the fixed-parameter tractability of NON-CLASH—*i.e.*, the more general task of computing the non-clashing teaching dimension when not all concepts (*i.e.*, balls) need to be present—parameterized by the *vertex integrity* of G. Vertex integrity is a well-studied graph parameter (Drange et al., 2016; Gima et al., 2022; Gima & Otachi, 2024; Hanaka et al., 2024) that essentially captures the vertex deletion distance to a graph consisting only of small connected components; it is known (and easily observed) to be a less restrictive parameterization than the vertex cover number (cf. Section 2), meaning that our result significantly pushes the boundaries of tractability even for the simpler strict variant of the problem.

The proof of Theorem 17 is highly non-trivial. In particular, while the algorithm itself is simple and merely uses a data reduction technique ("*kernelization*" (Cygan et al., 2015)) that iteratively removes certain parts of the instance, the crucial correctness proof underlying the result is very involved and relies on identifying a carefully defined set of "canonical" examples for our instances. The algorithm is also constructive, meaning that it can output a set of examples for the concepts as a witness.

As our final contribution, in Theorem 18 we complement Theorem 17 with a complexity-theoretic lower bound excluding fixed-parameter tractability under many other graph parameters previously considered in the literature, including *pathwidth*, *treewidth*, and the *feedback vertex number*—the latter two of which were explicitly mentioned in the aforementioned open question (Chalopin et al., 2024). We do so through a complex $\mathsf{W}[1]$-*hardness reduction* (which can be seen as a parameterized analog to classical reductions used to establish NP-hardness) that excludes, under well-established complexity assumptions, NON-CLASH from being in FPT even when combining all the parameterizations mentioned in the previous sentence with the positive non-clashing teaching dimension k.

Related Work. It is known that NON-CLASH is significantly more challenging than the special case captured by STRICT NON-CLASH. For instance, the reduction of Kirkpatrick, Simon, and Zilles (2019, Subsection 7.1) establishes that NON-CLASH is NP-hard even if the task is to determine whether the positive non-clashing teaching dimension of the instance is 1; on the other hand, the analogous question for STRICT NON-CLASH is trivial as it simply requires determining whether the input graph is edgeless or not (Chalopin et al., 2024). In fact, unless $\mathsf{P} = \mathsf{NP}$, that reduction also rules out a polynomial-time 1.999-approximation algorithm. For clarity, note that while that reduction does not consider concepts that are balls in a graph, as mentioned earlier *every* finite binary concept class can be easily transformed into a class of balls in a graph (Chalopin et al., 2023; 2024).

Apart from the computational questions resolved in this work, another prominent open question is whether the non-clashing teaching dimension is upper-bounded by the VC-dimension (Kirkpatrick et al., 2019; Fallat et al., 2023; Simon, 2023). It is known that the non-clashing teaching dimension (where one allows negative examples) can be significantly smaller than the positive variant, *e.g.*, balls in cycles have a non-clashing teaching dimension of 2, but their positive non-clashing teaching dimension is not bounded by any constant (Chalopin et al., 2024). It was also pointed out that balls in cacti or planar graphs could be good candidates for concept classes negatively answering this question (Chalopin et al., 2024). Further, Simon (2023) recently explored the relationship between non-clashing teaching and *recursive teaching* and identified the precise gap between the two notions.

Concept classes consisting of balls in a graph are a discrete analog of the geometric concept classes of balls in a Euclidean space which have been investigated in PAC-learning, *e.g.*, as part of the more general Dudley concept classes (Floyd, 1989; Ben-David & Litman, 1998). Apart from non-clashing teaching, they have also been explored for the closely related and well-studied sample compression schemes (Chalopin et al., 2023) introduced by Littlestone and Warmuth (1986). As discussed in prior works (Kirkpatrick et al., 2019; Fallat et al., 2023; Chalopin et al., 2024), non-clashing teaching maps can be viewed as signed variants of representation maps for concept classes, a notion introduced to design unlabeled sample compression schemes for maximum concept classes (Kuzmin & Warmuth, 2007) (and subsequently the more general ample concept classes (Chalopin et al., 2022)).

Due to space constraints, formal proofs are deferred to the full version of the paper in the appendix.

2 PRELIMINARIES

We assume familiarity with graph terminology (Diestel, 2024). We only consider simple, finite, and undirected graphs. For an integer $n \geq 1$, we set $[n] := \{1, \ldots, n\}$. As we only consider finite binary concept classes which can be represented as balls in graphs (Chalopin et al., 2023; 2024), we introduce the terminology for positive non-clashing teaching directly in the setting of graphs.

POSITIVE NON-CLASHING TEACHING IN GRAPHS. Let G be a graph. For an integer $r \geq 0$ and a vertex $v \in V(G)$, the *ball* $B_r(v)$ is the set of all vertices at distance at most r from its *center* v. Let $\mathcal{B}$ be a set of balls of G. A *positive teaching map* T for $\mathcal{B}$ is a mapping which assigns to each ball $B \in \mathcal{B}$ a *teaching set* $T(B) \subseteq B$, *i.e.*, a subset of the vertices of B. The *dimension* of T is $\max_{B \in \mathcal{B}} |T(B)|$—in other words, the largest image of T. A positive teaching map T is called *non-clashing* for $\mathcal{B}$ if for each pair of distinct balls $B_1, B_2 \in \mathcal{B}$, there exists a vertex $w \in T(B_1) \cup T(B_2)$ such that $w \notin B_1 \cap B_2$. Note that w must lie in $B_1 \cup B_2$ by definition, and hence, this condition ensures that one of the balls has a teaching set which is not contained in the other ball. We say that w *distinguishes* B_1 and B_2, or distinguishes B_1 from B_2 (or vice versa). If a teaching map is not non-clashing, we say that there is a *conflict* between any two balls for which there is no element distinguishing them. We now define our problems of interest:[1]

STRICT NON-CLASH	
Input:	A graph G and an integer k.
Question:	Is there a positive non-clashing teaching map for the set of all balls of G with dimension at most k?

NON-CLASH	
Input:	A graph G, a set $\mathcal{B}$ of balls of G, and an integer k.
Question:	Is there a positive non-clashing teaching map for $\mathcal{B}$ with dimension at most k?

We call a teaching map satisfying the conditions of the respective problem statement a *solution*. To avoid any confusion, we remark that the above definitions—as well as every result obtained in this article—concerns non-clashing teaching in the previously studied *positive* setting.

PARAMETERIZED COMPLEXITY. In parameterized complexity (Downey & Fellows, 2013; Cygan et al., 2015), the running-times of algorithms are studied with respect to a parameter $p \in \mathbb{N}$ and input size n. It is normally used for NP-hard problems, with the aim of finding a parameter describing a feature of the instance such that the combinatorial explosion is confined to this parameter. A parameterized problem is *fixed-parameter tractable* (FPT) if it can be solved by an algorithm running in time $f(p) \cdot n^{\mathcal{O}(1)}$, where f is a computable function; these are *fixed-parameter algorithms*. Proving that a problem is W[1]-hard via a *parameterized reduction* from a W[1]-hard problem rules out the existence of a fixed-parameter algorithm under the well-established hypothesis that W[1] $\neq$ FPT.

STRICT NON-CLASH is known to be fixed-parameter tractable when parameterized by the *vertex cover number* of G, *i.e.*, the minimum integer a such that there is a subset $X \subset V(G)$ of at most a vertices where $G - X$ is an edgeless graph. In this article, we consider three parameters which are upper-bounded by the vertex cover number (or, more precisely, the vertex cover number plus one):

- the *vertex integrity* of G, which is the minimum integer b such that there is a vertex subset $X \subset V(G)$ where for every connected component H of $G - X$, $|V(H) \cup X| \leq b$;
- the *feedback vertex number* of G (denoted by $\texttt{fvs}(G)$), which is the minimum integer c such that there is a vertex subset $X \subset V(G)$ where $G - X$ is acyclic;
- the *pathwidth* of G (denoted by $\texttt{pw}(G)$), which has a more involved definition based on the notion of *path decompositions*. However, for the purposes of this article it is sufficient to note the well-known facts (Downey & Fellows, 2013; Cygan et al., 2015) that deleting one vertex from each connected component of G will decrease the pathwidth by at most one, and that a graph consisting of a disjoint union of paths and *subdivided caterpillars* (*i.e.*, graphs consisting of a central path with pendent paths attached to it) has pathwidth 2.

[1] To provide more concise complexity-theoretic statements, our problems are formalized in their decision variants. All algorithms obtained in this article are constructive and can also output a teaching map as a witness.

3 INTRACTABILITY AND RUNNING TIME LOWER BOUNDS

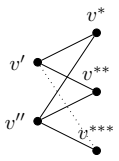
In this section, we establish that STRICT NON-CLASH is NP-hard even when $k = 2$, and thus, that it cannot be 1.499-approximated in polynomial time unless $\mathsf{P} = \mathsf{NP}$. Recall that the former result is tight in the sense that STRICT NON-CLASH is trivial when $k = 1$ (Chalopin et al., 2024).

Theorem 1. STRICT NON-CLASH *is* NP*-hard even when restricted to split graphs with* $k = 2$.

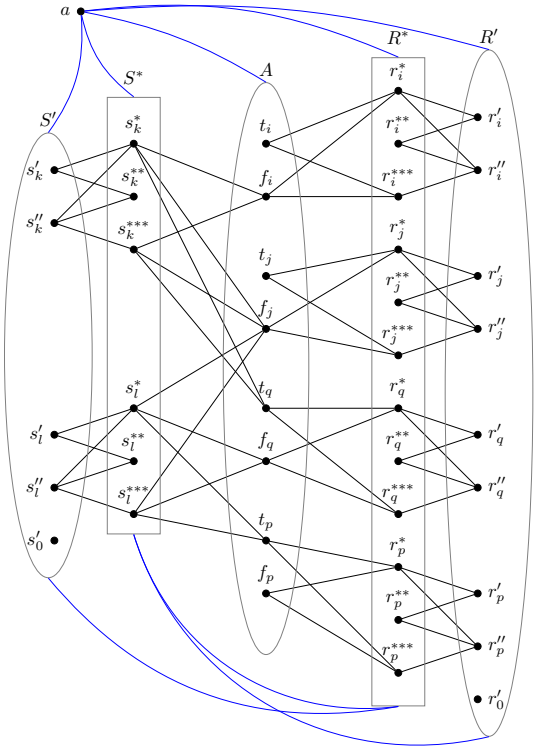
We prove Theorem 1 via a polynomial-time reduction that, given an instance of 3-SAT, constructs an equivalent instance (G, k) of STRICT NON-CLASH, where G is a split graph and $k = 2$.

> 3-SAT
> **Input:** A CNF formula over a set of clauses $\mathcal{C} = \{c_1, \ldots, c_m\}$ containing variables from $\mathcal{X} = \{x_1, \ldots, x_n\}$, where each clause has exactly 3 literals.
> **Question:** Is there a variable assignment $\tau : \mathcal{X} \to \{\texttt{True}, \texttt{False}\}$ satisfying each clause in $\mathcal{C}$?

The Reduction. Given an instance $\phi = (\mathcal{C}, \mathcal{X})$ of 3-SAT, we construct the graph G as follows; a schematic illustration is provided in Figure 1b.



(a) The force-gadget. The dotted edge corresponds to the absence of that edge.



(b) An example of the graph G obtained by applying our reduction on the 3-SAT instance with $\mathcal{X} = \{x_i, x_j, x_q, x_p\}$ and $\mathcal{C} = \{(x_i \vee x_j \vee \overline{x_q}), (x_j \vee x_q \vee \overline{x_p})\}$. Vertices in ovals form independent sets, while cliques are depicted by rectangles. Blue edges denote the existence of all possible edges between the two sets.

- First, for each $i \in [n]$, we create a pair of vertices $\{t_i, f_i\}$. We set $A := \{t_i, f_i\}_{i \in [n]}$.
- For each $i \in [n]$, we introduce a *variable force-gadget*, which consists of a set of vertices $\{r^*_i, r^{**}_i, r^{***}_i, r'_i, r''_i\}$ and edges as depicted in Figure 1a.
- For each $i \in [n]$, we attach the variable force-gadget to the pair $\{t_i, f_i\}$ as shown in Figure 1b, by making both r^*_i and r^{***}_i adjacent to both t_i and f_i. This gadget will guarantee that the corresponding assignment of the ith variable is well-defined. We set $R^* := \{r^*_i, r^{**}_i, r^{***}_i\}_{i \in [n]}$ and $R' := \{r'_0\} \cup \{r'_i, r''_i\}_{i \in [n]}$, where r'_0 is a new vertex.

- Similarly, for each $k \in [m]$, we introduce a *clause force-gadget* on the set of new vertices $\{s_k^*, s_k^{**}, s_k^{***}, s_k', s_k''\}$ (as depicted in Figure 1a). This gadget corresponds to the clause c_k of the instance ϕ. We add adjacencies according to the appearance of literals in c_k, *i.e.*, if $x_i \in c_k$ for some $i \in [n]$, then we connect both s_k^* and s_k^{***} to f_i; and if $\overline{x_i} \in c_k$, then we connect both s_k^* and s_j^{***} to t_i. Intuitively, we connect the gadget to those vertices whose underlying assignments (`True` or `False` for t_i and f_i, resp.) **do not** satisfy c_j, while the opposite assignments would (see Figure 1b). We set $S^* := \{s_k^*, s_k^{**}, s_k^{***}\}_{k \in [m]}$ and $S' := \{s_0'\} \cup \{s_k', s_k''\}_{k \in [m]}$, where s_0' is a new vertex.
- We add all possible edges between (a) S^* and R^*; (b) R^* and S'; (c) S^* and R'.
- We add all possible edges within S^*, and within R^*.
- Lastly, we add a special vertex a and make it adjacent to all the other vertices of the graph.

This concludes the construction of G; given the aforementioned instance $\phi = (\mathcal{C}, \mathcal{X})$ of 3-SAT, the reduction outputs the STRICT NON-CLASH instance $(G, 2)$. It now remains to establish correctness, which is handled by the following two lemmas.

Lemma 2. *If ϕ is a YES-instance of 3-*SAT*, then $(G, 2)$ is a YES-instance of* STRICT NON-CLASH.

Lemma 3. *If $(G, 2)$ is a YES-instance of* STRICT NON-CLASH*, then ϕ is a YES-instance of 3-*SAT.

The proof of Theorem 1 then follows from Lemma 2 and Lemma 3. In particular, they prove that there is a polynomial-time reduction which transforms any instance of 3-SAT with n variables and m clauses into an equivalent instance $(G, 2)$ of STRICT NON-CLASH where $|V(G)| = \mathcal{O}(n + m)$ and G is a split graph of diameter 2. The properties of this reduction also allow us to infer more precise algorithmic lower bounds. In particular, since an algorithm solving STRICT NON-CLASH in $2^{o(|V(G)| \cdot d \cdot k)}$ time would allow us to solve 3-SAT in $2^{o(n+m)}$ time:

Theorem 4. *Unless the Exponential Time Hypothesis fails, there is no algorithm solving* STRICT NON-CLASH *in time $2^{o(|V(G)| \cdot d \cdot k)}$, where d and k are the diameter of G and the target positive non-clashing teaching dimension of the instance, respectively.*

We complement this lower bound with a refined upper bound for the more general NON-CLASH:

Proposition 5. NON-CLASH *can be solved in $2^{\mathcal{O}(|V(G)| \cdot d \cdot k \cdot \log |V(G)|)}$ time.*

4 FIXED-PARAMETER TRACTABILITY VIA VERTEX INTEGRITY

Given that NON-CLASH is NP-hard, it is natural to ask whether the problem can be solved efficiently on inputs exhibiting some well-defined structural properties. In this section, we establish the fixed-parameter tractability of NON-CLASH when parameterized by the vertex integrity of the input graph.

Consider an instance $(G, \mathcal{B}, k)$ of NON-CLASH and let p be the vertex integrity of G. As the first step, we invoke the known algorithm to compute a "witness" for the vertex integrity in time $p^{\mathcal{O}(p)}|V(G)|$ (Fellows & Stueckle, 1989), *i.e.*, a set $X \subset V(G)$ such that $|V(H) \cup X| \leq p$ for each connected component H of $G - X$. Let $\mathcal{H}$ denote the set of connected components of $G - X$. To make use of the vertex integrity of G, we will partition the components of $\mathcal{H}$ into a parameter-bounded number of equivalence classes such that the elements belonging to the same class share some structural properties that will allow us to consider them, to some extent, interchangeable.

Definition 1. *Two subgraphs $H, H' \in \mathcal{H}$ are* twin-blocks *with respect to $\mathcal{B}$, denoted $H \sim_{\mathcal{B}} H'$, if there exists an isomorphism $\alpha_{H,H'}$ from H to H' with the following properties:*

- *for each $u \in V(H)$ and $v \in X$, $uv \in E(G)$ if and only if $\alpha_{H,H'}(u)v \in E(G)$, and*
- *for each $u \in V(H)$ and $r \in \mathbb{N}$, $B_r(u) \in \mathcal{B}$ if and only if $B_r(\alpha_{H,H'}(u)) \in \mathcal{B}$.*

Intuitively, $H \sim_{\mathcal{B}} H'$ if and only if there is a bijection $\alpha_{H,H'}$ between the vertices of the two subgraphs which preserves (1) edges inside H and H', (2) edges to X, and (3) the existence of balls in $\mathcal{B}$ centered at the vertices of H and H'. We refer to $\alpha_{H,H'}$ as the *canonical isomorphism* between the two twin-blocks at hand, and if multiple choices of α exist, we choose and fix one arbitrarily; we drop the indices of α when the subgraphs are clear from the context. Observe that for any choice of H and H', $H \sim_{\mathcal{B}} H'$ can be tested in time at most $p^{\mathcal{O}(p)}$ by enumerating all possible choices of α.

Clearly, $\sim_{\mathcal{B}}$ is an equivalence relation and we denote by $[H]_{\sim_{\mathcal{B}}}$ the equivalence class containing H. For $u \in V(H)$, we further define $[u]_{\sim_{\mathcal{B}}} = \{\alpha_{H,H'}(u) \mid H' \in [H]_{\sim_{\mathcal{B}}}\}$, and similarly for $B_r(u) \in \mathcal{B}$, $[B_r(u)]_{\sim_{\mathcal{B}}} = \{B_r(u') \in \mathcal{B} \mid u' \in [u]_{\sim_{\mathcal{B}}}\}$; intuitively, these refer to the sets of counterparts of u and $B_r(u)$ in the equivalence class, respectively. For brevity, we overload the notation $\sim_{\mathcal{B}}$ and use $v \sim_{\mathcal{B}} w$ (or $B_r(v) \sim_{\mathcal{B}} B_r(w)$) to express that $v \in [w]_{\sim_{\mathcal{B}}}$ (or $B_r(v) \in [B_r(w)]_{\sim_{\mathcal{B}}}$, respectively).

Observation 6. *The number of equivalence classes on $\mathcal{H}$ defined by $\sim_{\mathcal{B}}$ is at most $2^{\mathcal{O}(p^3)}$.*

While the equivalence relation $\sim_{\mathcal{B}}$ is defined based on the input (in particular, G and $\mathcal{B}$), our proof requires also considering a more refined equivalence relation based on the structure of a hypothetical positive non-clashing teaching map. Toward this, we use the following notion to capture how a hypothetical teaching set interacts with the balls centered in the components of $\mathcal{H}$.

Definition 2. *The* blueprint *S of a teaching set $T(B)$ for a ball $B = B_r(u)$ centered in $H \in \mathcal{H}$ is a tuple (S_X, S_H, S_f) composed of:*

1. *the set $S_X = T(B) \cap X$,*
2. *the set $S_H = T(B) \cap V(H)$,*
3. *the set $S_f = \{f_{H_0} \mid H_0 \in \mathcal{H}\}$ of functions, where for each H_0 the function $f_{H_0} : V(H_0) \to \{0, 1, 2\}$ specifies whether for a vertex $v \in V(H_0)$, the set $([v]_{\sim_{\mathcal{B}}} \cap T(B)) \setminus V(H)$ of counterparts of v* outside of *H has size 0, 1 or at least 2.*

Intuitively, the blueprint specifies how the teaching set for B interacts with (1) the set X and (2) the vertices inside H itself; for the rest of the graph, the blueprint also counts how many "equivalent" vertices it contains from each equivalence class of $\mathcal{H}$, *but only up to* 2. At this point, it may not be clear why we do not differentiate between any size greater than 2; the reason is that if the actual size is 3 or more, there are superfluous elements in the teaching set, as we prove below. In fact, we prove a more general statement which holds regardless of whether vertices in H are counted or not.

Lemma 7. *Let $u \in V(G)$, $B = B_r(u) \in \mathcal{B}$, and T any positive non-clashing teaching map for $\mathcal{B}$. Suppose there exist $H_0 \in \mathcal{H}$ and $v \in V(H_0)$ such that $|[v]_{\sim_{\mathcal{B}}} \cap T(B)| \geq 3$. Then, there exists z in $[v]_{\sim_{\mathcal{B}}} \cap T(B)$ such that removing z from $T(B)$ yields a positive non-clashing teaching map for $\mathcal{B}$.*

As a corollary of Lemma 7, we can obtain an upper bound on the positive non-clashing teaching dimension of our instance. This bound will, in fact, be useful later in this section.

Corollary 8. *Let G be a graph with vertex integrity p and $\mathcal{B}$ an arbitrary set of balls of G. Then, the positive non-clashing teaching dimension of $\mathcal{B}$ is upper-bounded by a function $s(p) = 2^{\mathcal{O}(p^3)}$.*

Returning to the notion of blueprints defined earlier, we can now formalize a refined notion of equivalence that also takes into account the behavior of a hypothetical solution.

Definition 3. *Given $B = B_r(u)$ centered in H, and $H' \sim_{\mathcal{B}} H$, we say that B and $B' = B(\alpha(u), r)$* share the same blueprint *if, for $(S_X, S_H, (f_{H_0})_{H_0 \in \mathcal{H}})$ and $(S'_X, S'_H, (f'_{H_0})_{H_0 \in \mathcal{H}})$ the respective blueprints of B and B', the following hold: (1) $S_X = S'_X$, (2) $\forall v \in V(H), v \in S_H \Leftrightarrow \alpha(v) \in S'_H$, and (3) $\forall H_0 \in \mathcal{H}, f_{H_0} = f'_{H_0}$.*

Definition 4. *Given a positive non-clashing teaching map T for $\mathcal{B}$, we say that two components $H, H' \in \mathcal{H}$ are* perfectly-equivalent twin-blocks *(or simply* perfectly equivalent*) with respect to T on $\mathcal{B}$, denoted $H \equiv^T_{\mathcal{B}} H'$, if $H \sim_{\mathcal{B}} H'$ and for all $u \in V(H)$, $r \in \mathbb{N}$, and $B_r(u) \in \mathcal{B}$, it holds that $B_r(u)$ and $B_r(\alpha(u))$ share the same blueprint. Note that $\equiv^T_{\mathcal{B}}$ is an equivalence relation.*

Crucially, we can also bound the number of equivalence classes for this refined equivalence defined w.r.t. a hypothetical teaching map. We denote the equivalence class of H by $[H]_{\equiv^T_{\mathcal{B}}}$.

Observation 9. *The number of equivalence classes on $\mathcal{H}$ defined by $\equiv^T_{\mathcal{B}}$ is at most $2^{2^{\mathcal{O}(p^3)}}$.*

With this refined equivalence in hand, we proceed to the second milestone required for our algorithm: a structural result establishing the existence of "well-behaved teaching maps". Roughly speaking, by "well-behaved" we mean teaching maps which only use examples (*i.e.*, vertices) from X and a bounded number of "core" components in $\mathcal{H}$, with some controlled exceptions. Toward formalizing this, let us fix an arbitrary total ordering $\prec$ on the components in $\mathcal{H}$.

Definition 5. *For $B \in \mathcal{B}$ and $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$, the (B, x)-*core *$\mathcal{K}_{(B,x)}$ contains the first $s(p) + 1$ elements w.r.t. $\prec$ of $\{H' \in [H]_{\sim_{\mathcal{B}}} \mid \exists B' \in [B]_{\equiv^T_{\mathcal{B}}}, \alpha(x) \in T(B') \cap V(H')\}$. The B-*core *is the union of the (B, x)-cores for all such x, and the* core *$\mathcal{K}$ is the union for all $B \in \mathcal{B}$ of the B-cores.*

Definition 6. *A component $H \in \mathcal{H}$ is in the* 1-extended core *$\mathcal{K}_1$ if $H \in \mathcal{K}$ or there exists $B \in \mathcal{B}$ and $x \in V(H)$ such that (1) B is centered in $\mathcal{K}$ or in X, and (2) $x \in T(B)$. Going one step further, a component $H \in \mathcal{H}$ is in the* 2-extended core *$\mathcal{K}_2$ if $H \in \mathcal{K}_1$ or there exists $B \in \mathcal{B}$ and $x \in V(H)$ such that (1) B is centered in $\mathcal{K}_1$ or in X, and (2) $x \in T(B)$.*

For $A \in \{\mathcal{K}, \mathcal{K}_1, \mathcal{K}_2\}$, note that A is a subset of $\mathcal{H}$, but by slight abuse of notation we will write $x \in V(A)$ for $x \in V(G)$ to denote $\exists H \in A$ with $x \in V(H)$ when not leading to any ambiguity. It is not difficult to see that the definitions yield the following.

Observation 10. *The number of components in $\mathcal{K}_2$ is upper-bounded by $c(p) = 2^{2^{\mathcal{O}(p^3)}}$.*

We can now formalize the notion of "well-behaved" teaching maps via the property of compactness.

Definition 7. *A teaching map T is* compact *if for any $H \in \mathcal{H} \setminus \mathcal{K}_2$ and any $x \in V(H)$, each ball whose teaching set contains x is centered in H.*

Our next aim is to prove in Lemma 13 that we can safely restrict our attention to computing compact teaching maps only. Before we can establish that result, we first prove two auxiliary statements.

Lemma 11. *Let T be a positive non-clashing teaching map for $\mathcal{B}$. Let $B \in \mathcal{B}$ be centered in $H_B \in \mathcal{H} \setminus \mathcal{K}_1$ with $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$ and $H_i \in \mathcal{K}_{(B,x)}$. By definition, there exists $x_i \in [x]_{\sim_{\mathcal{B}}} \cap V(H_i)$. If replacing x by x_i in $T(B)$ creates a conflict, then there exists $z_i \in T(B) \cap V(H_i)$.*

Lemma 12. *Let T be an optimal positive non-clashing teaching map for $\mathcal{B}$. Let $B \in \mathcal{B}$ be centered in $H_B \in \mathcal{H} \setminus \mathcal{K}_1$. For $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$, there exists $H_i \in \mathcal{K}_{(B,x)}$ and $x_i \in [x]_{\sim_{\mathcal{B}}} \cap V(H_i)$ such that replacing x by x_i in $T(B)$ does not create a conflict.*

Lemma 13. *If $\mathcal{B}$ has positive non-clashing teaching dimension k, then it also admits a compact positive non-clashing teaching map of dimension k.*

We now proceed to the crux of our algorithm: the proof that one can reduce the size of the instance $(G, \mathcal{B})$ without changing its positive non-clashing teaching dimension (formalized in Lemma 15). Toward this, it will be useful to focus on how the balls in $\mathcal{B}$ interact with certain subgraphs of G.

Definition 8. *Let G' be an induced subgraph of G. Then, $\mathcal{B}$* induces *the set $\mathcal{B}'$ of vertex sets w.r.t. G' where $\mathcal{B}' = \{B_r(u) \cap V(G') \mid B_r(u) \in \mathcal{B} \wedge u \in V(G')\}$.*

We note that the set of $\mathcal{B}'$ need not necessarily be a set of balls in G' itself: for instance, it may well happen that for some $B \in \mathcal{B}'$ the vertex set $B \cap V(G')$ is not even connected. However, under some conditions on G' that we will be able to guarantee, $\mathcal{B}'$ is, in fact, a set of balls of G'.

Lemma 14. *Let G be a graph, $\mathcal{B}$ a set of balls on G, and G' an induced subgraph of G. If $V(G')$ contains X and, for every $H \in \mathcal{H}$, either (1) $V(H) \subseteq V(G')$ or (2) $V(H) \cap V(G') = \emptyset \wedge \exists H', H'' \in [H]_{\sim_{\mathcal{B}}}, H \neq H'', V(H') \cup V(H'') \subseteq V(G')$ holds, then the set $\mathcal{B}'$ induced by $\mathcal{B}$ w.r.t. G' is a set of balls on G'. Moreover, there is a bijection between balls in $\mathcal{B}'$ and balls in $\mathcal{B}$ centered in G'.*

Let us now define the *reduced graph G' of G* as the graph obtained from G by removing all but $f(p) = c(p) + \binom{p \cdot c(p) + p}{s(p)}^{b(p)} + 1$ twin-blocks from each large class of $\sim_{\mathcal{B}}$. Additionally, we set $\mathcal{B}'$ to be the set induced by $\mathcal{B}$ on G' according to Definition 8, and note that Lemma 14 implies that it is a set of balls on G'. The size of G' is bounded by $g(p) = p + f(p) \cdot 2^{\mathcal{O}(p^3)} = 2^{2^{\mathcal{O}(p^3)}}$, and $\mathcal{B}'$ is hence also of bounded size: in particular, an optimal positive non-clashing teaching map for $\mathcal{B}'$ can be computed, *e.g.*, by brute force in time that depends only on p. We now aim at proving that $\mathcal{B}'$ is equivalent to $\mathcal{B}$, *i.e.*, that the optimal positive non-clashing teaching dimension is the same for both.

Lemma 15. *Suppose that there is a solution for $(G, \mathcal{B}, k)$. Then, there exists a solution T' for the reduced instance $(G', \mathcal{B}', k)$.*

The following lemma establishes that a solution for the reduced instance can be lifted to one for the original instance; note this also ensures that our algorithm will be constructive.

Lemma 16. *Let $k \in \mathbb{N}$ and T' be a compact solution for the instance $(G', \mathcal{B}', k)$. We can construct a compact solution T for $(G, \mathcal{B}, k)$ in time $2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$.*

Finally, we obtain our main parameterized tractability result by combining the previous ingredients. In particular, in the proof we construct $(G', \mathcal{B}', k)$, solve the problem there and argue correctness.

Theorem 17. NON-CLASH *is fixed-parameter tractable when parameterized by the vertex integrity of the input graph G.*

5 HARDNESS FOR CLASSICAL STRUCTURAL PARAMETERIZATIONS

In this section, we prove that NON-CLASH is intractable for many of the structural graph parameters that have been studied in the literature. We prove the result for the feedback vertex number and pathwidth; by extension, intractability then follows also for well-known generalizations of these measures such as treewidth and clique-width.

Theorem 18. NON-CLASH *is* W[1]*-hard parameterized by* $\texttt{fvs}(G) + \texttt{pw}(G) + k$.

We establish Theorem 18 by describing a parameterized reduction below and proving its correctness in Lemma 19. We reduce from a problem called NAE-INTEGER-3-SAT, which is W[1]-hard parameterized by the number of variables (Bringmann et al., 2016), and is defined as follows.

NAE-INTEGER-3-SAT
Input: A set of clauses $\mathcal{C}$ over variables $\mathcal{X}$, and an integer d. Any clause $c \in \mathcal{C}$ has the form $x \leq c_x$, $y \leq c_y$, and $z \leq c_z$, where $c_x, c_y, c_z \in \{1, \ldots, d\}$.
Question: Is there a variable assignment $\mathcal{X} \to \{1, \ldots, d\}$ such that for each clause, either 1 or 2 of its three inequalities are satisfied?

The Reduction. Given an instance ϕ of NAE-INTEGER-3-SAT, we construct an instance $(G, \mathcal{B}, k)$ of NON-CLASH in polynomial time as follows, starting with the graph G.

- For each variable $x \in \mathcal{X}$, make a (variable) path $P^x := (x_1, \ldots, x_d)$ of order d.
- For each clause $c \in \mathcal{C}$, make two vertices c and c' and, w.l.o.g., suppose that the clause c contains the variables x, y, and z, and that $c_x \leq c_y \leq c_z$. Connect the vertex c to x_d, y_d, and z_d by three distinct paths of lengths $4d$, $4d + c_y - c_x$, and $4d + c_z - c_x$, respectively.[2] These long paths ensure that there are no unwanted shortcuts in G and that, for $r_c = 5d - c_x - 1$, the ball $B_{r_c}(c)$ contains all of P^x, P^y, and P^z except for their first c_x, c_y, and c_z vertices (whose respective indices correspond to the respective variable values that satisfy the clause c), respectively. Similarly, connect the vertex c' to x_1, y_1, and z_1 by three distinct paths of lengths $4d + c_z - c_x$, $4d + c_z - c_y$, and $4d$, respectively. Analogously, this ensures that, for $r'_{c'} = 4d + c_z - 1$, the ball $B_{r'_{c'}}(c')$ contains the opposite vertices to $B_{r_c}(c)$ in P^x, P^y, and P^z (whose respective indices correspond to the respective variable values that do not satisfy the clause c), while ensuring that no unwanted shortcuts exist in G.
- For each clause $c \in \mathcal{C}$ and each variable $q \in \mathcal{X}$ such that c does not contain q, in G, connect the vertices c and c' to q_1 by distinct paths of length $3d$. These paths ensure that the balls $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$ described above contain every other variable path completely, while ensuring that no unwanted shortcuts exist in G.
- Let S be the set of all the vertices that currently exist in G. For each $x \in \mathcal{X}$, in G, make a vertex f_x and connect it to each vertex in S except those in P^x via a distinct path of length $6d$. Finally, for all $x, y \in \mathcal{X}$, make f_x adjacent to f_y. This ensures that, for each $x \in \mathcal{X}$, the ball $B_{6d}(f_x)$ contains every vertex in G except for those in P^x, while ensuring that no unwanted shortcuts exist in G. This completes the construction of G (see Figure 2).

Set $k := |\mathcal{X}|$ and $\mathcal{B}$ to contain $V(G)$, $B_{6d}(f_x)$ for all $x \in \mathcal{X}$, and $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$ for all $c \in \mathcal{C}$.

Correctness of the Reduction. Suppose, given an instance ϕ of NAE-INTEGER-3-SAT, that the reduction from the subsection above returns $(G, \mathcal{B}, k)$ as an instance of NON-CLASH.

[2] "Connect two vertices u, v by a path of length p" means to make u and v the endpoints of a path of length p.

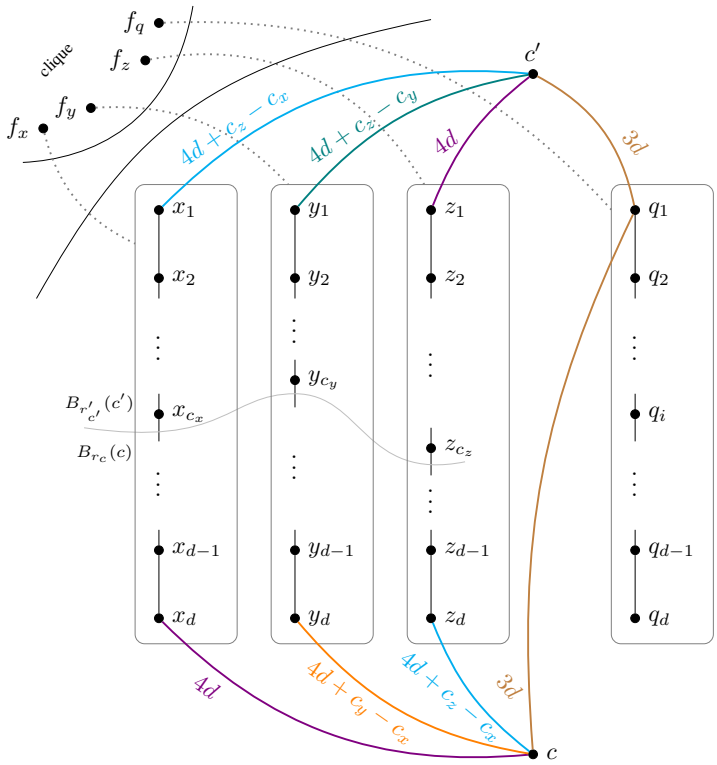


Figure 2: Colorful edges denote paths of the depicted lengths. Two black curves separate the clique from the rest of the graph. Each dotted edge shows that a vertex of the clique is adjacent to all the vertices of the graph below the separating curves except those in the adjacent rectangle. The gray curve gives an intuition of which vertices of P_x, P_y, and P_z are contained in $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$.

Lemma 19. *ϕ is a YES-instance of* NAE-INTEGER-3-SAT *if and only if $(G, \mathcal{B}, k)$ is a YES-instance of* NON-CLASH.

Proof of Theorem 18. Lemma 19 establishes the correctness of the polynomial-time reduction from the beginning of Section 5. To complete the proof, it remains to show that $\texttt{fvs}(G) + \texttt{pw}(G) + k$ is bounded above by a function of $|\mathcal{X}|$. This clearly holds for k which is, by definition, $|\mathcal{X}|$. Deleting from G the vertices x_1, x_d, and f_x for all $x \in \mathcal{X}$ results in an acyclic graph G'; in particular G has a feedback vertex set of size $3|\mathcal{X}|$. To establish a bound on $\texttt{pw}(G)$, it now suffices to show that G' also has bounded pathwidth. Note that G' consists of a set of connected components, each of which is either a subdivided caterpillar (this is what remains of each component containing a variable path) or a vertex (of the form c or c') with multiple pendent subdivided caterpillars and (simple) paths. Since deleting one further vertex from each connected component may only reduce the pathwidth by 1 and we need a single such deletion operation to reach a graph class of constant pathwidth (see Section 2), we also obtain that $\texttt{pw}(G)$ is bounded by a function of $|\mathcal{X}|$. □

6 CONCLUDING REMARKS

Our computational upper and lower bounds provide a near-comprehensive understanding of the complexity of computing the positive non-clashing teaching dimension. Apart from our contributions to the previously studied strict setting, we consider it notable that our work is the first to also tackle the complexity of non-clashing teaching in the non-strict setting—*i.e.*, the more general (and arguably more natural) case where not all possible concepts are present. One open question highlighted by our work concerns the tiny remaining gap between the algorithmic lower and upper bounds obtained in Theorem 4 and Proposition 5. More general directions for future work are to perform a similar complexity analysis in the non-positive setting and to consider approximation algorithms.

REFERENCES

Baris Akgun, Maya Cakmak, Jae W. Yoo, and Andrea L. Thomaz. Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective. In *Proc. of the 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI 2012)*, pp. 391–398, 2012.

Dana Angluin. Inductive inference of formal languages from positive data. *Information and Control*, 45:117–135, 1980a.

Dana Angluin. Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, 21:46–62, 1980b.

Shai Ben-David and Ami Litman. Combinatorial variability of Vapnik-Chervonenkis classes with applications to sample compression schemes. *Discrete Applied Mathematics*, 86:3–25, 1998.

Cornelius Brand, Robert Ganian, and Kirill Simonov. A parameterized theory of PAC learning. In *Proc. of the 37th AAAI Conference on Artificial Intelligence (AAAI 2023)*, pp. 6834–6841, 2023.

Karl Bringmann, Danny Hermelin, Matthias Mnich, and Erik Jan van Leeuwen. Parameterized complexity dichotomy for steiner multicut. *Journal of Computer and System Sciences*, 82(6): 1020–1043, 2016.

Daniel S. Brown and Scott Niekum. Machine teaching for inverse reinforcement learning: Algorithms and applications. In *Proc. of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)*, volume 33, pp. 7749–7758, 2019.

Jérémie Chalopin, Victor Chepoi, Shay Moran, and Manfred K. Warmuth. Unlabeled sample compression schemes and corner peelings for ample and maximum classes. *Journal of Computer and System Sciences*, 127:1–28, 2022.

Jérémie Chalopin, Victor Chepoi, Fionn Mc Inerney, Sébastien Ratel, and Yann Vaxès. Sample compression schemes for balls in graphs. *SIAM Journal on Discrete Mathematics*, 37(4):2585–2616, 2023.

Jérémie Chalopin, Victor Chepoi, Fionn Mc Inerney, and Sébastien Ratel. Non-clashing teaching maps for balls in graphs. In *Proc. of the 37th Annual Conference on Learning Theory (COLT 2024)*, volume 247 of *Proceedings of Machine Learning Research*, pp. 840–875, 2024.

Yuxin Chen, Adish Singla, Oisin Mac Aodha, Pietro Perona, and Yisong Yue. Understanding the role of adaptivity in machine teaching: The case of version space learners. In *Advances in Neural Information Processing Systems (NeurIPS 2018)*, volume 31, pp. 1483–1493, 2018.

Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

François Denis. Learning regular languages from simple positive examples. *Machine Learning*, 44: 37–66, 2001.

Reinhard Diestel. *Graph Theory, 6th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2024.

Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

Rodney G. Downey, Patricia A. Evans, and Michael R. Fellows. Parameterized learning complexity. In *Proc. of the 6th Annual ACM Conference on Computational Learning Theory (COLT 1993)*, pp. 51–57, 1993.

Pål Grønås Drange, Markus S. Dregi, and Pim van 't Hof. On the computational complexity of vertex integrity and component order connectivity. *Algorithmica*, 76(4):1181–1202, 2016.

Eduard Eiben, Robert Ganian, Iyad A. Kanj, Sebastian Ordyniak, and Stefan Szeider. The computational complexity of concise hypersphere classification. In *Proc. of the International Conference on Machine Learning (ICML 2023)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 9060–9070, 2023a.

Eduard Eiben, Sebastian Ordyniak, Giacomo Paesani, and Stefan Szeider. Learning small decision trees with large domain. In *Proc. of the 32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*, pp. 3184–3192, 2023b.

Shaun Fallat, David Kirkpatrick, Hans U. Simon, Abolghasem Soltani, and Sandra Zilles. On batch teaching without collusion. *Journal of Machine Learning Research*, 24:40:1–40:33, 2023.

Michael R Fellows and Sam Stueckle. The immersion order, forbidden subgraphs and the complexity of network integrity. *J. Combin. Math. Combin. Comput*, 6(1):23–32, 1989.

S. Floyd. *Space-bounded learning and the Vapnik-Chervonenkis dimension*. PhD thesis, U.C. Berkeley, 1989.

Robert Ganian and Viktoriia Korchemna. The complexity of Bayesian network learning: Revisiting the superstructure. In *Advances in Neural Information Processing Systems (NeurIPS 2021)*, volume 34, pp. 430–442, 2021.

Ziyuan Gao, Christoph Ries, Hans U. Simon, and Sandra Zilles. Preference-based teaching. *Journal of Machine Learning Research*, 18:1–32, 2017.

Tatsuya Gima and Yota Otachi. Extended MSO model checking via small vertex integrity. *Algorithmica*, 86(1):147–170, 2024.

Tatsuya Gima, Tesshu Hanaka, Masashi Kiyomi, Yasuaki Kobayashi, and Yota Otachi. Exploring the gap between treedepth and vertex cover through vertex integrity. *Theoretical Computer Science*, 918:60–76, 2022.

Sally A. Goldman and Michael J. Kearns. On the complexity of teaching. *Journal of Computer and System Sciences*, 50(1):20–31, 1995.

Sally A. Goldman and H. David Mathias. Teaching a smarter learner. *Journal of Computer and System Sciences*, 52(2):255–267, 1996.

Tesshu Hanaka, Michael Lampis, Manolis Vasilakis, and Kanae Yoshiwatari. Parameterized vertex integrity revisited. In *Proc. of the 49th International Symposium on Mathematical Foundations of Computer Science (MFCS 2024)*, volume 306 of *LIPIcs*, pp. 58:1–58:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

Mark K. Ho, Michael Littman, James MacGlashan, Fiery Cushman, and Joseph L. Austerweil. Showing versus doing: Teaching by demonstration. In *Advances in Neural Information Processing Systems (NeurIPS 2016)*, volume 33, pp. 3027–3035, 2016.

Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001.

David Kirkpatrick, Hans U. Simon, and Sandra Zilles. Optimal collusion-free teaching. In *Proc. of the 30th International Conference on Algorithmic Learning Theory (ALT 2019)*, volume 98 of *Proceedings of Machine Learning Research*, pp. 506–528, 2019.

Dima Kuzmin and Manfred K. Warmuth. Unlabelled compression schemes for maximum classes. *Journal of Machine Learning Research*, 8:2047–2081, 2007.

Yuanzhi Li and Yingyu Liang. Learning mixtures of linear regressions with nearly optimal complexity. In *Proc. of the 31st Conference on Learning Theory (COLT 2018)*, volume 75 of *Proceedings of Machine Learning Research*, pp. 1125–1144, 2018.

N. Littlestone and M. K. Warmuth. Relating data compression and learnability. *Unpublished*, 1986.

Farnam Mansouri, Yuxin Chen, Ara Vartanian, Jerry Zhu, and Adish Singla. Preference-based batch and sequential teaching: Towards a unified view of models. In *Advances in Neural Information Processing Systems (NeurIPS 2019)*, volume 32, pp. 9195–9205, 2019.

Shike Mei and Xiaojin Zhu. Using machine teaching to identify optimal training-set attacks on machine learners. In *Proc. of the 29th AAAI Conference on Artificial Intelligence (AAAI 2015)*, volume 29, pp. 2871–2877, 2015.

Sebastian Ordyniak and Stefan Szeider. Parameterized complexity of small decision tree learning. In *Proc. of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*, pp. 6454–6462, 2021.

Ingo Schwab, Wolfgang Pohl, and Ivan Koychev. Learning to recommend from positive evidence. In *Proc. of the 5th International Conference on Intelligent User Interfaces (IUI 2000)*, pp. 241–247, 2000.

Ayumi Shinohara and Satoru Miyano. Teachability in computational learning. *New Generation Computing*, 8:337–347, 1991.

Hans U. Simon. Tournaments, johnson graphs, and NC-teaching. In *Proc. of the 34th International Conference on Algorithmic Learning Theory (ALT 2023)*, volume 201 of *Proceedings of Machine Learning Research*, pp. 1411–1428, 2023.

Andreas Stolcke and Stephen Omohundro. Inducing probabilistic grammars by Bayesian model merging. In *Proc. of the 2nd International Colloquium on Grammatical Inference (ICGI 1994)*, pp. 106–118, 1994.

Jan Arne Telle, José Hernández-Orallo, and Cèsar Ferri. The teaching size: computable teachers and learners for universal languages. *Machine Learning*, 108(8-9):1653–1675, 2019.

Andrea L. Thomaz and Maya Cakmak. Learning about objects with human teachers. In *Proc. of the 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI 2009)*, pp. 15–22, 2009.

Chunlin Wang, Chris Ding, Richard F. Meraz, and Stephen R. Holbrook. PSoL: a positive sample only learning algorithm for finding non-coding RNA genes. *Bioinformatics*, 22(21):2590–2596, 2006.

Malik Yousef, Segun Jung, Louise C. Showe, and Michael K. Showe. Learning from positive examples when the negative class is undetermined - microRNA gene identification. *Algorithms for Molecular Biology*, 3(2), 2008.

Xuezhou Zhang, Xaiojin Zhu, and Stephen Wright. Training set debugging using trusted items. In *Proc. of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*, pp. 4482–4489, 2018.

Xiaojin Zhu. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *Proc. of the 29th AAAI Conference on Artificial Intelligence (AAAI 2015)*, volume 29, pp. 4083–4087, 2015.

Xiaojin Zhu, Adish Singla, Sandra Zilles, and Anna N. Rafferty. An overview of machine teaching. *Arxiv:1801.05927*, 2018. URL `https://arxiv.org/abs/1801.05927`.

Sandra Zilles, Steffen Lange, Robert Holte, and Martin Zinkevich. Models of cooperative teaching and learning. *Journal of Machine Learning Research*, 12:349–384, 2011.

THE COMPUTATIONAL COMPLEXITY OF POSITIVE NON-CLASHING TEACHING IN GRAPHS (APPENDIX: FULL VERSION)

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the classical and parameterized complexity of computing the positive non-clashing teaching dimension of a set of concepts, that is, the smallest number of examples per concept required to successfully teach an intelligent learner under the considered, previously established model. For any class of concepts, it is known that this problem can be effortlessly transferred to the setting of balls in a graph G. We establish (1) the NP-hardness of the problem even when restricted to instances with positive non-clashing teaching dimension $k = 2$ and where all balls in the graph are present, (2) near-tight running time upper and lower bounds for the problem on general graphs, (3) fixed-parameter tractability when parameterized by the vertex integrity of G, and (4) a lower bound excluding fixed-parameter tractability when parameterized by the feedback vertex number and pathwidth of G, even when combined with k. Our results provide a nearly complete understanding of the complexity landscape of computing the positive non-clashing teaching dimension and answer open questions from the literature.

1 INTRODUCTION

While typical machine learning models task a learner with finding a concept C from a concept class $\mathcal{C}$ based on an—often randomly drawn—sample, in *machine teaching* (specifically in its commonly considered batch variant) the learner is provided a set of examples by a teacher; crucially, here the examples can be selected in a way which allows the concept to be reconstructed from as few examples as possible. Machine teaching is a core topic in computational learning theory and has found applications in a variety of areas including robotics (Thomaz & Cakmak, 2009; Akgun et al., 2012), trustworthy AI (Mei & Zhu, 2015; Zhang et al., 2018), inverse reinforcement learning (Ho et al., 2016; Brown & Niekum, 2019), and education (Zhu, 2015; Chen et al., 2018; Zhu et al., 2018). While numerous models of machine teaching have been investigated to date (Shinohara & Miyano, 1991; Goldman & Kearns, 1995; Goldman & Mathias, 1996; Zilles et al., 2011; Gao et al., 2017; Mansouri et al., 2019; Telle et al., 2019), in this article we focus on the recently developed *positive non-clashing teaching model* (Kirkpatrick et al., 2019; Fallat et al., 2023).

In non-clashing teaching, given a finite binary concept class $\mathcal{C}$, for each pair C_1, C_2 of distinct concepts in $\mathcal{C}$, at least one example provided for at least one of C_1 or C_2 must not be consistent with the other concept. A key feature of non-clashing teaching is that it is the most efficient model (in terms of the number of required examples) satisfying the Goldman-Mathias collusion-avoidance criterion (Goldman & Mathias, 1996)—the "gold standard" for ensuring that the learner cannot cheat, *e.g.*, via a hidden communication channel with the teacher. Moreover, a teaching model is *positive* if the examples provided for each concept C are required to be positively labeled for C. The restriction of teaching models to the positive setting is common and well-motivated from applications in, *e.g.*, recommendation systems (Schwab et al., 2000), computational biology (Wang et al., 2006; Yousef et al., 2008), and grammatical inference (Stolcke & Omohundro, 1994; Denis, 2001); see also the early works of Angluin (Angluin, 1980a;b). Non-clashing teaching has been proposed and studied in the positive setting not only within the initial papers introducing the concept (Kirkpatrick et al., 2019; Fallat et al., 2023), but also in subsequent works (*e.g.*, (Simon, 2023; Chalopin et al., 2024)).

While positive non-clashing teaching has the potential to be highly efficient, realizing this potential requires us to solve the computational task of actually constructing a small set of examples for the given concepts. More precisely, one aims at computing (a witness for) the teaching dimension of a given concept class, *i.e.*, the minimum integer k such that each concept is provided with at most k examples that satisfy the conditions of the model. On the positive side, instead of considering various different types of concepts and examples, we can restrict our attention to the setting where each concept is a ball in some input-specified graph G, and the possible examples are vertices of G. Indeed, it is known that any finite binary concept class $\mathcal{C} \subseteq 2^V$ can be represented by a set $\mathcal{B}$ of balls in a graph G as follows: $V(G) = V \cup \{x_C \mid C \in \mathcal{C}\}$, x_C is adjacent to $x_{C'}$ for all $C, C' \in \mathcal{C}$, x_C is adjacent to $v \in V$ if and only if $v \in C$, and $\mathcal{B} = \{B_1(x_C) \mid C \in \mathcal{C}\}$ (Chalopin et al., 2023; 2024).

On the negative side, the problem is computationally intractable, and remains so even in the restricted setting where *every* possible concept (*i.e.*, every ball in G) is present; to distinguish this case from the general one where not all concepts need to be present, we refer to it as *strict*. In particular, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) recently carried out an initial complexity-theoretic investigation of computing the positive non-clashing teaching dimension in the strict setting. There, they established the NP-hardness of the problem for instances with large teaching dimension (even when restricted to the highly restricted class of split graphs), obtained runtime upper and lower bounds under the *Exponential Time Hypothesis* (Impagliazzo et al., 2001), and designed a so-called *fixed-parameter* algorithm for the problem when parameterized by the size of the vertex cover of G.

In this article, we significantly improve over each of these results, answer two open questions posed by the authors of the aforementioned work (Chalopin et al., 2024), and obtain a nearly complete understanding of the computational complexity of computing the positive non-clashing teaching dimension (in both the strict and non-strict settings).

Contributions. Let us refer to the problems of computing the positive non-clashing teaching dimension in the strict and non-strict settings as STRICT NON-CLASH and NON-CLASH, respectively. Formal definitions complementing the informal descriptions given above are provided in Section 2.

Our first result concerns the complexity of STRICT NON-CLASH on instances with constant positive non-clashing teaching dimension. The reductions of Chalopin, Chepoi, Mc Inerney, and Ratel (2024) only establish the NP-hardness of the problem for instances with large (*i.e.*, input-dependent) positive non-clashing teaching dimension k. In fact, as their first open question, the authors ask whether STRICT NON-CLASH is NP-hard or polynomial-time solvable when the sought-after dimension k is a fixed constant; the question is not only theoretically interesting, but also highly relevant as instances of small teaching dimension are precisely the candidates for efficient teaching. We settle this by a highly non-trivial reduction (Theorem 1) which establishes that determining whether the positive non-clashing teaching dimension is at most 2 is NP-hard—and remains so even when restricted to the same class of *split graphs* where STRICT NON-CLASH was previously shown to be NP-hard (for large k) (Chalopin et al., 2024). We note that this result is, in a sense, best possible: determining whether an instance of STRICT NON-CLASH has a positive non-clashing teaching dimension of 1 is trivial as it is equivalent to testing whether G is edgeless (Chalopin et al., 2024).

Next, we proceed to the running time bounds for solving the problem. Typically, the running time upper bounds are given by an exact algorithm, while the lower bounds are obtained from a suitable "tight" reduction under the Exponential Time Hypothesis (Impagliazzo et al., 2001). In the preceding work, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) obtained algorithmic lower and upper bounds of $2^{o(n \cdot d)}$ and $2^{\mathcal{O}(n^2 \cdot d)}$, where n and d are the number of vertices and the diameter of G, respectively. From our reduction and a more careful algorithmic analysis, we obtain a lower bound of $2^{o(n \cdot d \cdot k)}$ and an upper bound of $2^{\mathcal{O}(n \cdot d \cdot k \cdot \log n)}$ (Theorem 4 and Proposition 5)—making the bounds almost tight, with just a logarithmic factor in the exponent separating the two.

While the aforementioned bounds apply to the problem in general, often one may only need to solve the problem on "well-structured" graphs. The more refined *parameterized complexity* paradigm (Downey & Fellows, 2013; Cygan et al., 2015) offers the perfect tools to analyze and identify precisely which structural properties of input graphs—usually captured by a suitable integer *parameter* k—allow us to circumvent its general intractability. The analog to the complexity class P in the parameterized setting is FPT ("*fixed-parameter tractable*"), which characterizes parameterized problems solvable in $f(k) \cdot n^{\mathcal{O}(1)}$ time; intuitively, this means that the problem is solvable in uniformly polynomial time for each constant value of k. Parameterized complexity is

well-established and has been successfully applied for non-clashing teaching (Chalopin et al., 2024) as well as in a variety of related subfields of learning theory (Downey et al., 1993; Li & Liang, 2018; Ganian & Korchemna, 2021; Ordyniak & Szeider, 2021; Brand et al., 2023; Eiben et al., 2023a;b).

In their previous work, Chalopin, Chepoi, Mc Inerney, and Ratel (2024) established the fixed-parameter tractability of STRICT NON-CLASH when parameterized by the *vertex cover number* of the input graph G—or, equivalently, the vertex deletion distance to a graph consisting only of isolated vertices. The drawback of that result is that the vertex cover number is a highly "restrictive" parameter, in the sense that it achieves low values only on rather simple graphs. This is also reflected in the open question posed in that article, which asked about the problem's complexity under other parameterizations. As our third contribution, we establish (in Theorem 18) the fixed-parameter tractability of NON-CLASH—*i.e.*, the more general task of computing the non-clashing teaching dimension when not all concepts (*i.e.*, balls) need to be present—parameterized by the *vertex integrity* of G. Vertex integrity is a well-studied graph parameter (Drange et al., 2016; Gima et al., 2022; Gima & Otachi, 2024; Hanaka et al., 2024) that essentially captures the vertex deletion distance to a graph consisting only of small connected components; it is known (and easily observed) to be a less restrictive parameterization than the vertex cover number (cf. Section 2), meaning that our result significantly pushes the boundaries of tractability even for the simpler strict variant of the problem.

The proof of Theorem 18 is highly non-trivial. In particular, while the algorithm itself is simple and merely uses a data reduction technique ("*kernelization*" (Cygan et al., 2015)) that iteratively removes certain parts of the instance, the crucial correctness proof underlying the result is very involved and relies on identifying a carefully defined set of "canonical" examples for our instances. The algorithm is also constructive, meaning that it can output a set of examples for the concepts as a witness.

As our final contribution, in Theorem 19 we complement Theorem 18 with a complexity-theoretic lower bound excluding fixed-parameter tractability under many other graph parameters previously considered in the literature, including *pathwidth*, *treewidth*, and the *feedback vertex number*—the latter two of which were explicitly mentioned in the aforementioned open question (Chalopin et al., 2024). We do so through a complex W[1]-*hardness reduction* (which can be seen as a parameterized analog to classical reductions used to establish NP-hardness) that excludes, under well-established complexity assumptions, NON-CLASH from being in FPT even when combining all the parameterizations mentioned in the previous sentence with the positive non-clashing teaching dimension k.

Related Work. It is known that NON-CLASH is significantly more challenging than the special case captured by STRICT NON-CLASH. For instance, the reduction of Kirkpatrick, Simon, and Zilles (2019, Subsection 7.1) establishes that NON-CLASH is NP-hard even if the task is to determine whether the positive non-clashing teaching dimension of the instance is 1; on the other hand, the analogous question for STRICT NON-CLASH is trivial as it simply requires determining whether the input graph is edgeless or not (Chalopin et al., 2024). In fact, unless $\mathsf{P} = \mathsf{NP}$, that reduction also rules out a polynomial-time 1.999-approximation algorithm. For clarity, note that while that reduction does not consider concepts that are balls in a graph, as mentioned earlier *every* finite binary concept class can be easily transformed into a class of balls in a graph (Chalopin et al., 2023; 2024).

Apart from the computational questions resolved in this work, another prominent open question is whether the non-clashing teaching dimension is upper-bounded by the VC-dimension (Kirkpatrick et al., 2019; Fallat et al., 2023; Simon, 2023). It is known that the non-clashing teaching dimension (where one allows negative examples) can be significantly smaller than the positive variant, *e.g.*, balls in cycles have a non-clashing teaching dimension of 2, but their positive non-clashing teaching dimension is not bounded by any constant (Chalopin et al., 2024). It was also pointed out that balls in cacti or planar graphs could be good candidates for concept classes negatively answering this question (Chalopin et al., 2024). Further, Simon (2023) recently explored the relationship between non-clashing teaching and *recursive teaching* and identified the precise gap between the two notions.

Concept classes consisting of balls in a graph are a discrete analog of the geometric concept classes of balls in a Euclidean space which have been investigated in PAC-learning, *e.g.*, as part of the more general Dudley concept classes (Floyd, 1989; Ben-David & Litman, 1998). Apart from non-clashing teaching, they have also been explored for the closely related and well-studied sample compression schemes (Chalopin et al., 2023) introduced by Littlestone and Warmuth (1986). As discussed in prior works (Kirkpatrick et al., 2019; Fallat et al., 2023; Chalopin et al., 2024), non-clashing teaching maps can be viewed as signed variants of representation maps for concept classes, a notion intro-

duced to design unlabeled sample compression schemes for maximum concept classes (Kuzmin & Warmuth, 2007) (and subsequently the more general ample concept classes (Chalopin et al., 2022)).

2 PRELIMINARIES

We assume familiarity with graph terminology (Diestel, 2024). We only consider simple, finite, and undirected graphs. For an integer $n \geq 1$, we set $[n] := \{1, \ldots, n\}$. As we only consider finite binary concept classes which can be represented as balls in graphs (Chalopin et al., 2023; 2024), we introduce the terminology for positive non-clashing teaching directly in the setting of graphs.

POSITIVE NON-CLASHING TEACHING IN GRAPHS. Let G be a graph. For an integer $r \geq 0$ and a vertex $v \in V(G)$, the *ball* $B_r(v)$ is the set of all vertices at distance at most r from its *center* v. Let $\mathcal{B}$ be a set of balls of G. A *positive teaching map* T for $\mathcal{B}$ is a mapping which assigns to each ball $B \in \mathcal{B}$ a *teaching set* $T(B) \subseteq B$, *i.e.*, a subset of the vertices of B. The *dimension* of T is $\max_{B \in \mathcal{B}} |T(B)|$—in other words, the largest image of T. A positive teaching map T is called *non-clashing* for $\mathcal{B}$ if for each pair of distinct balls $B_1, B_2 \in \mathcal{B}$, there exists a vertex $w \in T(B_1) \cup T(B_2)$ such that $w \notin B_1 \cap B_2$. Note that w must lie in $B_1 \cup B_2$ by definition, and hence, this condition ensures that one of the balls has a teaching set which is not contained in the other ball. We say that w *distinguishes* B_1 and B_2, or distinguishes B_1 from B_2 (or vice versa). If a teaching map is not non-clashing, we say that there is a *conflict* between any two balls for which there is no element distinguishing them. We now define our problems of interest:[1]

STRICT NON-CLASH	
Input:	A graph G and an integer k.
Question:	Is there a positive non-clashing teaching map for the set of all balls of G with dimension at most k?

NON-CLASH	
Input:	A graph G, a set $\mathcal{B}$ of balls of G, and an integer k.
Question:	Is there a positive non-clashing teaching map for $\mathcal{B}$ with dimension at most k?

We call a teaching map satisfying the conditions of the respective problem statement a *solution*. To avoid any confusion, we remark that the above definitions—as well as every result obtained in this article—concerns non-clashing teaching in the previously studied *positive* setting.

PARAMETERIZED COMPLEXITY. In parameterized complexity (Downey & Fellows, 2013; Cygan et al., 2015), the running-times of algorithms are studied with respect to a parameter $p \in \mathbb{N}$ and input size n. It is normally used for NP-hard problems, with the aim of finding a parameter describing a feature of the instance such that the combinatorial explosion is confined to this parameter. A parameterized problem is *fixed-parameter tractable* (FPT) if it can be solved by an algorithm running in time $f(p) \cdot n^{\mathcal{O}(1)}$, where f is a computable function; these are *fixed-parameter algorithms*.

Proving that a problem is W[1]-hard via a *parameterized reduction* from a W[1]-hard problem $\mathcal{P}$ rules out the existence of a fixed-parameter algorithm under the well-established hypothesis that W[1] $\neq$ FPT. A parameterized reduction from $\mathcal{P}$ to a parameterized problem $\mathcal{Q}$ is a function:

- which maps **YES**-instances to **YES**-instances and **NO**-instances to **NO**-instances,
- is computable in time $f(p) \cdot n^{\mathcal{O}(1)}$, where f is a computable function, and
- where the parameter of the output instance can be upper-bounded by some function of the parameter of the input instance.

STRICT NON-CLASH is known to be fixed-parameter tractable when parameterized by the *vertex cover number* of G, *i.e.*, the minimum integer a such that there is a subset $X \subset V(G)$ of at most a vertices where $G - X$ is an edgeless graph. In this article, we consider three parameters which are upper-bounded by the vertex cover number (or, more precisely, the vertex cover number plus one):

[1] To provide more concise complexity-theoretic statements, our problems are formalized in their decision variants. All algorithms obtained in this article are constructive and can also output a teaching map as a witness.

- the *vertex integrity* of G, which is the minimum integer b such that there is a vertex subset $X \subset V(G)$ where for every connected component H of $G - X$, $|V(H) \cup X| \leq b$;
- the *feedback vertex number* of G (denoted by $\texttt{fvs}(G)$), which is the minimum integer c such that there is a vertex subset $X \subset V(G)$ where $G - X$ is acyclic;
- the *pathwidth* of G (denoted by $\texttt{pw}(G)$), which has a more involved definition based on the notion of *path decompositions*. However, for the purposes of this article it is sufficient to note the well-known facts (Downey & Fellows, 2013; Cygan et al., 2015) that deleting one vertex from each connected component of G will decrease the pathwidth by at most one, and that a graph consisting of a disjoint union of paths and *subdivided caterpillars* (*i.e.*, graphs consisting of a central path with pendent paths attached to it) has pathwidth 2.

3 Intractability and Running Time Lower Bounds

In this section, we establish the NP-hardness of Strict Non-Clash when $k = 2$, and thus, that it cannot be 1.499-approximated in polynomial time unless $\mathsf{P} = \mathsf{NP}$; in fact, our results hold even when the graphs belong to the class of *split graphs*, *i.e.*, graphs which can be partitioned into an independent set and a clique. Recall that the former result is tight in the sense that Strict Non-Clash is trivial when $k = 1$ (Chalopin et al., 2024). We formalize the result below.

Theorem 1. Strict Non-Clash *is* NP*-hard even when restricted to split graphs with* $k = 2$.

We prove Theorem 1 via a polynomial-time reduction that, given an instance of 3-SAT, constructs an equivalent instance (G, k) of Strict Non-Clash, where G is a split graph and $k = 2$.

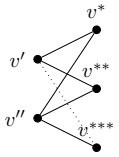
3-SAT
Input: A CNF formula over a set of clauses $\mathcal{C} = \{c_1, \ldots, c_m\}$ containing variables from $\mathcal{X} = \{x_1, \ldots, x_n\}$, where each clause has exactly 3 literals.
Question: Is there a variable assignment $\tau : \mathcal{X} \to \{\texttt{True}, \texttt{False}\}$ satisfying each clause in $\mathcal{C}$?

The Reduction. Given an instance $\phi = (\mathcal{C}, \mathcal{X})$ of 3-SAT, we construct the graph G as follows; a schematic illustration is provided in Figure 1b.

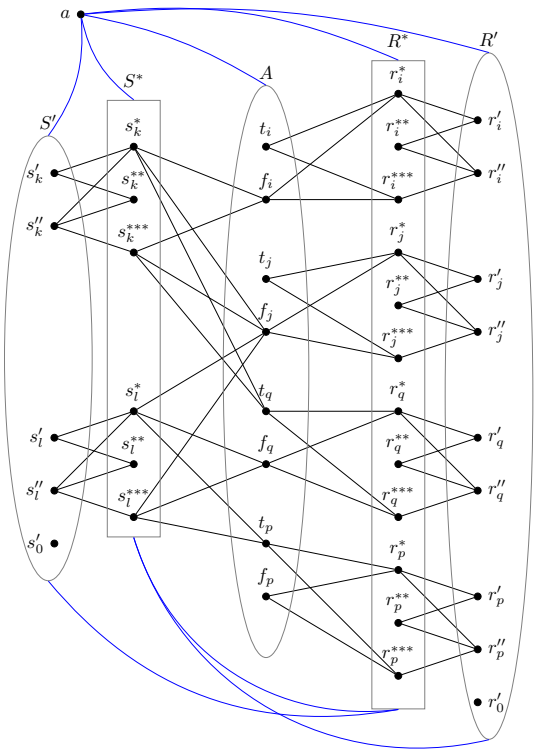
- First, for each $i \in [n]$, we create a pair of vertices $\{t_i, f_i\}$. We set $A := \{t_i, f_i\}_{i \in [n]}$.
- For each $i \in [n]$, we introduce a *variable force-gadget*, which consists of a set of vertices $\{r_i^*, r_i^{**}, r_i^{***}, r_i', r_i''\}$ and edges as depicted in Figure 1a.
- For each $i \in [n]$, we attach the variable force-gadget to the pair $\{t_i, f_i\}$ as shown in Figure 1b, by making both r_i^* and r_i^{***} adjacent to both t_i and f_i. This gadget will guarantee that the corresponding assignment of the ith variable is well-defined. We set $R^* := \{r_i^*, r_i^{**}, r_i^{***}\}_{i \in [n]}$ and $R' := \{r_0'\} \cup \{r_i', r_i''\}_{i \in [n]}$, where r_0' is a new vertex.
- Similarly, for each $k \in [m]$, we introduce a *clause force-gadget* on the set of new vertices $\{s_k^*, s_k^{**}, s_k^{***}, s_k', s_k''\}$ (as depicted in Figure 1a). This gadget corresponds to the clause c_k of the instance ϕ. We add adjacencies according to the appearance of literals in c_k, *i.e.*, if $x_i \in c_k$ for some $i \in [n]$, then we connect both s_k^* and s_k^{***} to f_i; and if $\overline{x_i} \in c_k$, then we connect both s_k^* and s_j^{***} to t_i. Intuitively, we connect the gadget to those vertices whose underlying assignments ($\texttt{True}$ or $\texttt{False}$ for t_i and f_i, resp.) **do not** satisfy c_j, while the opposite assignments would (see Figure 1b). We set $S^* := \{s_k^*, s_k^{**}, s_k^{***}\}_{k \in [m]}$ and $S' := \{s_0'\} \cup \{s_k', s_k''\}_{k \in [m]}$, where s_0' is a new vertex.
- We add all possible edges between (a) S^* and R^*; (b) R^* and S'; (c) S^* and R'.
- We add all possible edges within S^*, and within R^*.
- Lastly, we add a special vertex a and make it adjacent to all the other vertices of the graph.

This concludes the construction of G; given the aforementioned instance $\phi = (\mathcal{C}, \mathcal{X})$ of 3-SAT, the reduction outputs the Strict Non-Clash instance $(G, 2)$. It now remains to establish correctness, which is handled by the following two lemmas.

Lemma 2. *If* ϕ *is a YES-instance of 3-*SAT*, then* $(G, 2)$ *is a YES-instance of* Strict Non-Clash.



(a) The force-gadget. The dotted edge corresponds to the absence of that edge.



(b) An example of the graph G obtained by applying our reduction on the 3-SAT instance with $\mathcal{X} = \{x_i, x_j, x_q, x_p\}$ and $\mathcal{C} = \{(x_i \vee x_j \vee \overline{x_q}), (x_j \vee x_q \vee \overline{x_p})\}$. Vertices in ovals form independent sets, while cliques are depicted by rectangles. Blue edges denote the existence of all possible edges between the two sets.

for $k \in [m]$	$T(B_1(s^*_k)) = \{s'_k, v\}$ and c_k is satisfied by $\tau(x_i)$ for some $i \in [n]$, where
	if $\tau(x_i) = \texttt{True}$, then $v = f_i$ $\quad$ ($f_i \in B_1(s^*_k) \cap A$ since $x_i \in c_k$)
	if $\tau(x_i) = \texttt{False}$, then $v = t_i$ $\quad$ ($t_i \in B_1(s^*_k) \cap A$ since $\overline{x_i} \in c_k$)
	$T(B_1(s^{**}_k)) = \{s'_k, r'_0\}$ $\quad$ $T(B_1(s^{***}_k)) = \{s''_k, r'_0\}$
	$T(B_1(s'_k)) = \{s'_k, s^{**}_k\}$ $\quad$ $T(B_1(s''_k)) = \{s''_k, s^{***}_k\}$ $\quad$ $T(B_1(s'_0)) = \{s'_0, a\}$
for $i \in [n]$	$T(B_1(r^*_i)) = \{r'_i, v\}$, where
	if $\tau(x_i) = \texttt{True}$, then $v = t_i$
	if $\tau(f_i) = \texttt{False}$, then $v = f_i$
	$T(B_1(r^{**}_i)) = \{r'_i, s'_0\}$ $\quad$ $T(B_1(r^{***}_i)) = \{r''_i, s'_0\}$
	$T(B_1(r'_i)) = \{r'_i, r^{**}_i\}$ $\quad$ $T(B_1(r''_i)) = \{r''_i, r^{***}_i\}$ $\quad$ $T(B_1(r'_0)) = \{r'_0, a\}$
for $i \in [n]$	$T(B_1(t_i)) = \{t_i, a\}$ $\quad$ $T(B_1(f_i)) = \{f_i, a\}$
	$T(V(G)) = T(B_1(a)) = \{t_i, t_j\}$, for $i, j \in [n]$, $i \neq j$ such that there is no $k \in [m]$ where both x_i and x_j appear in c_k.[a]

Table 1: Positive teaching map for $\mathcal{B}$.

[a] We can assume the existence of such a pair i, j, because introducing an artificial variable and a unique clause in which it occurs gives an equivalent instance. Indeed, the artificial variable appears only once and its assignment can be chosen so that the added clause is satisfied without the rest of the formula being affected.

Proof. As G contains a universal vertex a, its diameter is 2, and thus, any ball of radius at least 2 contains $V(G)$. Hence, it is sufficient to consider balls of radius at most 2. Let $\tau : \mathcal{X} \to \{\texttt{True}, \texttt{False}\}$ be an assignment of the variables satisfying the given 3-SAT formula ϕ. Let us

define a positive teaching map T of dimension 2 for the set $\mathcal{B}$ of all the balls in G as shown in Table 1. It remains to prove that T is non-clashing for $\mathcal{B}$.

To this end, we will refer to Table 2 which intuitively describes which vertices are used to distinguish each pair of balls of radius 1. For any pair $u, z \in V(G)$, the corresponding entry in the table contains either v or a given vertex $w \in V(G)$. If it contains a vertex $w \in V(G)$, observe that $w \in T(B_1(u)) \cup T(B_1(z))$ by Table 1, and $w \notin B_1(u) \cap B_1(z)$. In the remaining cases, *i.e.*, *(1)* s_k^*, r_i^*; *(2)* s_k^*, s_k^{**}; *(3)* s_k^*, s_k'; *(4)* s_k^*, s_k'', we need to explicit which vertex v stands for. For pairs *(2)* s_k^*, s_k^{**}; *(3)* s_k^*, s_k'; *(4)* s_k^*, s_k'', v can be any vertex in $N(s_k^*) \cap A$ (by construction, there are 3 such vertices) since the neighborhoods of each of s_k^{**}, s_k', and s_k'' do not intersect A.

For the last type of pair *(1)*, we will use the fact that τ is a satisfying assignment. For $k \in [m]$ and $i \in [n]$, assume w.l.o.g. that c_k is satisfied by the assignment $\tau(x_i) = \texttt{True}$. Then, $T(B_1(s_k^*)) = \{s_k', f_i\}$ and $T(B_1(r_i^*)) = \{r_i', t_i\}$. With such an assignment of teaching sets, for any pair of type s_k^*, r_i^* we have that $t_i \in T(B_1(s_k^*)) \cup T(B_1(r_i^*)) = \{s_k', f_i\} \cup \{r_i', t_i\}$, and $t_i \notin B_1(s_k^*) \cap B_1(r_i^*)$.

	s_k^*	s_k^{**}	s_k^{***}	s_k'	s_k''	s_0'	t_i	f_i
r_i^*	v	r_0'	r_0'	r_i'	r_i'	r_i'	r_i'	r_i'
r_i^{**}	s_0'	r_0'	r_0'	s_0'	s_0'	r_i'	r_i'	r_i'
r_i^{***}	s_0'	r_0'	r_0'	s_0'	s_0'	r_i''	r_i''	r_i''
r_i'	s_k'	r_0'	r_0'	r_i'	r_i'	r_i'	r_i'	r_i'
r_i''	s_k'	r_0'	r_0'	r_i''	r_i''	r_i''	r_i''	r_i''
r_0'	s_k'	s_k'	s_k''	s_k'	s_k''	r_0'	r_0'	r_0'
t_j	s_k'	s_k'	s_k''	s_k'	s_k''	s_0'	t_i	f_i
f_j	s_k'	s_k'	s_k''	s_k'	s_k''	s_0'	t_i	f_i

(a) Here $i, j \in [n]$, $k \in [m]$, and $v \in \{t_i, f_i\} \setminus N(s_k^*)$. According to Table 1, $\{t_i, f_i\} \setminus N(s_k^*) = t_i$ if $x_i = \texttt{True}$ satisfies c_k; and $\{t_i, f_i\} \setminus N(s_k^*) = f_i$ if $x_i = \texttt{False}$ satisfies c_k.

	s_k^*	s_k^{**}	s_k^{***}	s_k'	s_k''
s_l^*	s_k'	v	s_k'	v	v
s_l^{**}	s_k'	s_k'	s_k'	r_0'	r_0'
s_l^{***}	s_k'	s_k'	s_k''	r_0'	r_0'
s_l'	s_k'	s_k'	s_k''	s_k'	s_k'
s_l''	s_k'	s_k'	s_k''	s_k'	s_k''
s_0'	s_0'	s_0'	s_0'	s_0'	s_0'

(b) Here, $k, l \in [m]$; filled cells correspond to the case $k = l$, and the others to $k \neq l$. The vertex v here is any vertex in $N(s_l^*) \cap A$. The table for vertices of the variable force-gadgets is defined similarly, interverting all s and r symbols.

Table 2: For each $u, z \in V(G)$, in a cell at the intersection of the corresponding row and column, we place a vertex $w \in V(G)$ such that $w \in T(B_1(u)) \cup T(B_1(z))$ and $w \notin B_1(u) \cup B_1(z)$ (according to the teaching map defined by Table 1).

Finally, we check that the teaching set for $T(V(G))$ distinguishes $V(G)$ from all the other balls in G. According to Table 1, for $i, j \in [n]$ and $i \neq j$, $T(V(G)) = \{t_i, t_j\}$ and there is no $k \in [m]$ such that both x_i and x_j appear in c_k. The last condition guarantees us that, for any $u \in V(G) \setminus \{a\}$, there is a vertex (either t_i or t_j) that is in $T(V(G))$ but not in $B_1(u)$ as there is no clause force-gadget that would be attached to both t_i and t_j.

Thus, we showed that for any pair of balls in $\mathcal{B}$, both necessary conditions for the defined teaching sets hold. Hence, the defined positive teaching map is non-clashing for $\mathcal{B}$. □

Lemma 3. *If* $(G, 2)$ *is a YES-instance of* STRICT NON-CLASH*, then* ϕ *is a YES-instance of 3-*SAT.

Proof. Let T be a positive non-clashing teaching map of dimension 2 for the set $\mathcal{B}$ of all balls in G. By definition, for each pair of distinct balls $B_1, B_2 \in \mathcal{B}$, there is $w \in T(B_1) \cup T(B_2)$ such that $w \notin B_1 \cap B_2$. For each $i \in [n]$, let us consider a pair of vertices r_i^* and r_i^{***}. By the construction, $B_1(r_i^{***}) \subset B_1(r_i^*)$ and r_i' is the unique vertex in $B_1(r_i^*)$ that is not in $B_1(r_i^{***}) \cap B_1(r_i^*)$. Thus, since T is a positive non-clashing teaching map for $\mathcal{B}$, r_i' has to be in $T(B_1(r_i^*))$. Now, let us consider a pair of vertices r_i^* and r_i^{**}. Similarly, $B_1(r_i^{**}) \subset B_1(r_i^*)$ and $\{t_i, f_i\} = B_1(r_i^*) \setminus B_1(r_i^{**})$. So, for T to distinguish the balls $B_1(r_i^{**})$ and $B_1(r_i^*)$, exactly one of t_i and f_i (since $T(B_1(r_i^*))$ already contains r_i') has to be in $T(B_1(r_i^*))$. The same arguments work for clause force-gadgets, by symmetry of the construction, and we obtain that, for each $k \in [m]$, there is $v \in B_1(s_k^*) \cap A$ such that $T(B_1(s_k^*)) = \{s_k', v\}$.

Now, let us use the fact that for $i \in [n]$ and $k \in [m]$, the balls $B_1(r_i^*)$ and $B_1(s_k^*)$ are distinguished by T. If $B_1(s_k^*) \cap \{t_i, f_i\} = \emptyset$, whichever of t_i, f_i that is in $B_1(r_i^*)$ distinguishes the two balls. However, as we have shown before, for each $k \in [m]$, $|T(B_1(s_k^*)) \cap A| = 1$. So, there exists $i \in [n]$ such that $B_1(s_k^*) \cap \{t_i, f_i\} \neq \emptyset$. W.l.o.g., let us assume that $B_1(s_k^*) \cap \{t_i, f_i\} = f_i$ (which means that $T(B_1(s_k^*)) = \{s_k', f_i\}$). As a result, the only valid option for $B_1(r_i^*)$ to be distinguished from $B_1(s_k^*)$ is that $T(B_1(r_i^*)) = \{r_i', t_i\}$. In the other symmetric case where $B_1(s_k^*) \cap \{t_i, f_i\} = t_i$, we obtain $T(B_1(r_i^*)) = \{r_i', f_i\}$.

Let us now define an assignment $\tau : \mathcal{X} \to \{\texttt{True}, \texttt{False}\}$ in the following way. For each $i \in [n]$, if $t_i \in T(B_1(r_i^*))$, we set $\tau(x_i) = \texttt{True}$; otherwise $f_i \in T(B_1(r_i^*))$ and we set $\tau(x_i) = \texttt{False}$. Let us show that τ indeed satisfies the 3-SAT instance ϕ. As we discussed above, for each $k \in [m]$, $T(B_1(s_k^*))$ has an intersection with A in exactly one vertex, w.l.o.g., let it again be that $T(B_1(s_k^*)) \cap A = \{f_i\}$. Then, $T(B_1(r_i^*)) \cap A = \{t_i\}$. So, $\tau(x_i) = \texttt{True}$. By our reduction, s_k^* is adjacent to f_i if assigning $\texttt{False}$ to x_i **does not** satisfy c_k, while assigning $\texttt{True}$ to x_i would. Thus, the assignment τ satisfies all the m clauses of the initial 3-SAT instance ϕ. □

The proof of Theorem 1 then follows from Lemma 2 and Lemma 3. In particular, they prove that there is a polynomial-time reduction which transforms any instance of 3-SAT with n variables and m clauses into an equivalent instance $(G, 2)$ of STRICT NON-CLASH where $|V(G)| = \mathcal{O}(n + m)$ and G is a split graph of diameter 2. The properties of this reduction also allow us to infer more precise algorithmic lower bounds. In particular, since an algorithm solving STRICT NON-CLASH in $2^{o(|V(G)| \cdot d \cdot k)}$ time would allow us to solve 3-SAT in $2^{o(n+m)}$ time:

Theorem 4. *Unless the Exponential Time Hypothesis fails, there is no algorithm solving* STRICT NON-CLASH *in time $2^{o(|V(G)| \cdot d \cdot k)}$, where d and k are the diameter of G and the target positive non-clashing teaching dimension of the instance, respectively.*

We complement this lower bound with a refined upper bound for the more general NON-CLASH:

Proposition 5. NON-CLASH *can be solved in $2^{\mathcal{O}(|V(G)| \cdot d \cdot k \cdot \log |V(G)|)}$ time.*

Proof. We can assume that G is connected, as otherwise we can solve NON-CLASH independently on each of the connected components of G. For any $v \in V(G)$ and $r \in \mathbb{N}$, there are at most $\binom{|V(G)|}{k} = \mathcal{O}(2^{k \cdot \log |V(G)|})$ possible choices for $T(B_r(v))$, and there are at most $\mathcal{O}(|V(G)| \cdot d)$ unique balls in G. Due to the latter, for each possible teaching map, we can check in polynomial time whether it is a positive teaching map that satisfies the non-clashing teaching property. Thus, there is a brute-force algorithm running in $2^{\mathcal{O}(|V(G)| \cdot d \cdot k \cdot \log |V(G)|)}$ time. □

4 FIXED-PARAMETER TRACTABILITY VIA VERTEX INTEGRITY

Given that NON-CLASH is NP-hard, it is natural to ask whether the problem can be solved efficiently on inputs exhibiting some well-defined structural properties. In this section, we establish the fixed-parameter tractability of NON-CLASH when parameterized by the vertex integrity of the input graph.

Consider an instance $(G, \mathcal{B}, k)$ of NON-CLASH and let p be the vertex integrity of G. As the first step, we invoke the known algorithm to compute a "witness" for the vertex integrity in time $p^{\mathcal{O}(p)}|V(G)|$ (Fellows & Stueckle, 1989), *i.e.*, a set $X \subset V(G)$ such that $|V(H) \cup X| \leq p$ for each connected component H of $G - X$. Let $\mathcal{H}$ denote the set of connected components of $G - X$. To make use of the vertex integrity of G, we will partition the components of $\mathcal{H}$ into a parameter-bounded number of equivalence classes such that the elements belonging to the same class share some structural properties that will allow us to consider them, to some extent, interchangeable.

Definition 1. *Two subgraphs $H, H' \in \mathcal{H}$ are* twin-blocks *with respect to $\mathcal{B}$, denoted $H \sim_{\mathcal{B}} H'$, if there exists an isomorphism $\alpha_{H,H'}$ from H to H' with the following properties:*

- *for each $u \in V(H)$ and $v \in X$, $uv \in E(G)$ if and only if $\alpha_{H,H'}(u)v \in E(G)$, and*
- *for each $u \in V(H)$ and $r \in \mathbb{N}$, $B_r(u) \in \mathcal{B}$ if and only if $B_r(\alpha_{H,H'}(u)) \in \mathcal{B}$.*

Intuitively, $H \sim_{\mathcal{B}} H'$ if and only if there is a bijection $\alpha_{H,H'}$ between the vertices of the two subgraphs which preserves (1) edges inside H and H', (2) edges to X, and (3) the existence of balls

in $\mathcal{B}$ centered at the vertices of H and H'. We refer to $\alpha_{H,H'}$ as the *canonical isomorphism* between the two twin-blocks at hand, and if multiple choices of α exist, we choose and fix one arbitrarily; we drop the indices of α when the subgraphs are clear from the context. Observe that for any choice of H and H', $H \sim_{\mathcal{B}} H'$ can be tested in time at most $p^{\mathcal{O}(p)}$ by enumerating all possible choices of α.

Clearly, $\sim_{\mathcal{B}}$ is an equivalence relation and we denote by $[H]_{\sim_{\mathcal{B}}}$ the equivalence class containing H. For $u \in V(H)$, we further define $[u]_{\sim_{\mathcal{B}}} = \{\alpha_{H,H'}(u) \mid H' \in [H]_{\sim_{\mathcal{B}}}\}$, and similarly for $B_r(u) \in \mathcal{B}$, $[B_r(u)]_{\sim_{\mathcal{B}}} = \{B_r(u') \in \mathcal{B} \mid u' \in [u]_{\sim_{\mathcal{B}}}\}$; intuitively, these refer to the sets of counterparts of u and $B_r(u)$ in the equivalence class, respectively. For brevity, we overload the notation $\sim_{\mathcal{B}}$ and use $v \sim_{\mathcal{B}} w$ (or $B_r(v) \sim_{\mathcal{B}} B_r(w)$) to express that $v \in [w]_{\sim_{\mathcal{B}}}$ (or $B_r(v) \in [B_r(w)]_{\sim_{\mathcal{B}}}$, respectively).

Observation 6. *The number of equivalence classes on $\mathcal{H}$ defined by $\sim_{\mathcal{B}}$ is at most $2^{\mathcal{O}(p^3)}$.*

Proof. All graphs in $\mathcal{H}$ have size at most p, which means that the number of non-isomorphic graphs in $\mathcal{H}$ can be trivially upper-bounded by $p \cdot 2^{p^2}$. Since $|X| < p$, there are also at most p^2 possible edges between X and any $H \in \mathcal{H}$ in G. Lastly, the number of balls centered in $H \in \mathcal{H}$ is bounded above by the number of vertices in H times the diameter of G which is $p \cdot \mathcal{O}(p^2) = \mathcal{O}(p^3)$. Indeed, the diameter of a connected graph with vertex integrity p is at most $\mathcal{O}(p^2)$, since the parameter does not increase by taking induced subgraphs and the vertex integrity of a path of length j is $\mathcal{O}(\sqrt{j})$. Combining these elements, we can upper-bound the total number of equivalence classes by $p \cdot 2^{p^2} \cdot 2^{p^2} \cdot 2^{\mathcal{O}(p^3)} = 2^{\mathcal{O}(p^3)}$. □

While the equivalence relation $\sim_{\mathcal{B}}$ is defined based on the input (in particular, G and $\mathcal{B}$), our proof requires also considering a more refined equivalence relation based on the structure of a hypothetical positive non-clashing teaching map. Toward this, we use the following notion to capture how a hypothetical teaching set interacts with the balls centered in the components of $\mathcal{H}$.

Definition 2. *The* blueprint *S of a teaching set $T(B)$ for a ball $B = B_r(u)$ centered in $H \in \mathcal{H}$ is a tuple (S_X, S_H, S_f) composed of:*

1. *the set $S_X = T(B) \cap X$,*
2. *the set $S_H = T(B) \cap V(H)$,*
3. *the set $S_f = \{f_{H_0} \mid H_0 \in \mathcal{H}\}$ of functions, where for each H_0 the function $f_{H_0} : V(H_0) \to \{0, 1, 2\}$ specifies whether for a vertex $v \in V(H_0)$, the set $([v]_{\sim_{\mathcal{B}}} \cap T(B)) \setminus V(H)$ of counterparts of v* outside of *H has size 0, 1 or at least 2.*

Intuitively, the blueprint specifies how the teaching set for B interacts with (1) the set X and (2) the vertices inside H itself; for the rest of the graph, the blueprint also counts how many "equivalent" vertices it contains from each equivalence class of $\mathcal{H}$, *but only up to* 2. At this point, it may not be clear why we do not differentiate between any size greater than 2; the reason is that if the actual size is 3 or more, there are superfluous elements in the teaching set, as we prove below. In fact, we prove a more general statement which holds regardless of whether vertices in H are counted or not.

Lemma 7. *Let $u \in V(G)$, $B = B_r(u) \in \mathcal{B}$, and T any positive non-clashing teaching map for $\mathcal{B}$. Suppose there exist $H_0 \in \mathcal{H}$ and $v \in V(H_0)$ such that $|[v]_{\sim_{\mathcal{B}}} \cap T(B)| \geq 3$. Then, there exists z in $[v]_{\sim_{\mathcal{B}}} \cap T(B)$ such that removing z from $T(B)$ yields a positive non-clashing teaching map for $\mathcal{B}$.*

Proof. Let v and H_0 satisfy the premise, and let z_1, z_2, z_3 be three distinct elements of $\{w \in T(B) \mid w \in V(H'), H' \sim_{\mathcal{B}} H_0, \alpha(w) = v\}$. We denote by H_1 (resp. H_2, H_3) the component of $\mathcal{H}$ containing z_1 (resp. z_2, z_3). These components are disjoint by definition since the vertices are $\sim_{\mathcal{B}}$-equivalent, and thus, u cannot be in more than one of H_1, H_2, H_3 (and possibly none, *i.e.*, u could be in some other component or in X). Without loss of generality, we assume that u is not in $V(H_3)$, and claim that removing z_3 from $T(B)$ results in a positive non-clashing teaching map T' for $\mathcal{B}$.

We prove this claim as follows. Toward a contradiction, suppose there is a ball $B' = B_{r'}(u')$ such that T' does not satisfy the non-clashing condition for B and B'. Then, z_3 was the only vertex in $T(B) \cup T(B')$ that was not contained in $B \cap B'$, and hence, $z_3 \notin B'$ and $z_1, z_2 \in B'$. Therefore, $d(u', z_1) < d(u', z_3)$, which implies that $u' \in V(H_1)$ since $z_1 \sim_{\mathcal{B}} z_3$. However, we also have that $d(u', z_2) < d(u', z_3)$, and so, $u' \in V(H_2)$, which is a contradiction since $V(H_1) \cap V(H_2) = \emptyset$.

Thus, such a ball B' does not exist, and we can safely remove z_3 from T, proving the lemma. □

As a corollary of Lemma 7, we can obtain an upper bound on the positive non-clashing teaching dimension of our instance. This bound will, in fact, be useful later in this section.

Corollary 8. *Let G be a graph with vertex integrity p and $\mathcal{B}$ an arbitrary set of balls of G. Then, the positive non-clashing teaching dimension of $\mathcal{B}$ is upper-bounded by a function $s(p) = 2^{\mathcal{O}(p^3)}$.*

Proof. For any ball $B \in \mathcal{B}$ and positive non-clashing teaching map T for $\mathcal{B}$, if Lemma 7 can be applied, then there is an unnecessary vertex in $T(B)$ that can be removed. Thus, without loss of generality, consider a positive non-clashing teaching map T for $\mathcal{B}$ for which Lemma 7 cannot be applied for any $B \in \mathcal{B}$. For each $B \in \mathcal{B}$ and each of the at most $2^{\mathcal{O}(p^3)}$ equivalence classes (by Observation 6) of $\mathcal{H}$ for $\sim_{\mathcal{B}}$, $T(B)$ can contain at most $2p$ vertices from that equivalence class, as otherwise it would contain 3 vertices that are equivalent according to $\sim_{\mathcal{B}}$. Thus, the total size of $T(B)$ is upper-bounded by $|X| + 2^{\mathcal{O}(p^3)} \cdot 2p = 2^{\mathcal{O}(p^3)}$. □

Returning to the notion of blueprints defined earlier, we can now formalize a refined notion of equivalence that also takes into account the behavior of a hypothetical solution.

Definition 3. *Given $B = B_r(u)$ centered in H, and $H' \sim_{\mathcal{B}} H$, we say that B and $B' = B(\alpha(u), r)$* share the same blueprint *if, for $(S_X, S_H, (f_{H_0})_{H_0 \in \mathcal{H}})$ and $(S'_X, S'_H, (f'_{H_0})_{H_0 \in \mathcal{H}})$ the respective blueprints of B and B', the following hold:*

- $S_X = S'_X$,
- $\forall v \in V(H), v \in S_H \Leftrightarrow \alpha(v) \in S'_H$, *and*
- $\forall H_0 \in \mathcal{H}, f_{H_0} = f'_{H_0}$.

Definition 4. *Given a positive non-clashing teaching map T for $\mathcal{B}$, we say that two components $H, H' \in \mathcal{H}$ are* perfectly-equivalent twin-blocks *(or simply* perfectly equivalent*) with respect to T on $\mathcal{B}$, denoted $H \equiv^T_{\mathcal{B}} H'$, if $H \sim_{\mathcal{B}} H'$ and for all $u \in V(H)$, $r \in \mathbb{N}$, and $B_r(u) \in \mathcal{B}$, it holds that $B_r(u)$ and $B_r(\alpha(u))$ share the same blueprint. Note that $\equiv^T_{\mathcal{B}}$ is an equivalence relation.*

Crucially, we can also bound the number of equivalence classes for this refined equivalence defined w.r.t. a hypothetical teaching map. We denote the equivalence class of H by $[H]_{\equiv^T_{\mathcal{B}}}$.

Observation 9. *The number of equivalence classes on $\mathcal{H}$ defined by $\equiv^T_{\mathcal{B}}$ is at most $2^{2^{\mathcal{O}(p^3)}}$.*

Proof. By Observation 6, there are $2^{\mathcal{O}(p^3)}$ possibilities for the equivalence class on $\mathcal{H}$ defined by $\sim_{\mathcal{B}}$. For any ball, there are then at most 2^p possibilities for the set S_X, as well as at most 2^p for the set S_H, and for a given H_0, at most 3^p for f_{H_0}. However, these choices for $(f_{H_0})_{H_0 \in \mathcal{H}}$ are not independent: $H' \sim_{\mathcal{B}} H'' \Rightarrow f_{H'} = f_{H''}$. Thus, there are in total $2^p \cdot 2^p \cdot (3^p)^{2^{\mathcal{O}(p^3)}} = 2^{2^{\mathcal{O}(p^3)}}$ possible blueprints for a given ball. There are at most $\mathcal{O}(p^3)$ balls centered in a given H (for details, see proof of Observation 6). Putting everything together gives us an upper bound for the number of equivalence classes for $\equiv^T_{\mathcal{B}}$ of $2^{\mathcal{O}(p^3)} \cdot (2^{2^{\mathcal{O}(p^3)}})^{\mathcal{O}(p^3)} = 2^{2^{\mathcal{O}(p^3)}}$. □

With this refined equivalence in hand, we proceed to the second milestone required for our algorithm: a structural result establishing the existence of "well-behaved teaching maps". Roughly speaking, by "well-behaved" we mean teaching maps which only use examples (*i.e.*, vertices) from X and a bounded number of "core" components in $\mathcal{H}$, with some controlled exceptions. Toward formalizing this, let us fix an arbitrary total ordering $\prec$ on the components in $\mathcal{H}$.

Definition 5. *For $B \in \mathcal{B}$ and $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$, the (B, x)-*core *$\mathcal{K}_{(B,x)}$ contains the first $s(p) + 1$ (see Corollary 8) elements w.r.t. $\prec$ of $\{H' \in [H]_{\sim_{\mathcal{B}}} \mid \exists B' \in [B]_{\equiv^T_{\mathcal{B}}}, \alpha(x) \in T(B') \cap V(H')\}$. The B-*core *is the union of the (B, x)-cores for all such x, and the* core *$\mathcal{K}$ is the union for all $B \in \mathcal{B}$ of the B-cores.*

Definition 6. *A component $H \in \mathcal{H}$ is in the* 1-extended core $\mathcal{K}_1$ *if $H \in \mathcal{K}$ or there exists $B \in \mathcal{B}$ and $x \in V(H)$ such that (1) B is centered in $\mathcal{K}$ or in X, and (2) $x \in T(B)$. Going one step further, a component $H \in \mathcal{H}$ is in the* 2-extended core $\mathcal{K}_2$ *if $H \in \mathcal{K}_1$ or there exists $B \in \mathcal{B}$ and $x \in V(H)$ such that (1) B is centered in $\mathcal{K}_1$ or in X, and (2) $x \in T(B)$.*

For $A \in \{\mathcal{K}, \mathcal{K}_1, \mathcal{K}_2\}$, note that A is a subset of $\mathcal{H}$, but by slight abuse of notation we will write $x \in V(A)$ for $x \in V(G)$ to denote $\exists H \in A$ with $x \in V(H)$ when not leading to any ambiguity. It is not difficult to see that the definitions yield the following.

Observation 10. *The number of components in $\mathcal{K}_2$ is upper-bounded by $c(p) = 2^{2^{\mathcal{O}(p^3)}}$.*

Proof. Any (B, x)-core $\mathcal{K}_{(B,x)}$ contains $s(p) + 1$ components by definition. Thus, any B-core contains at most $\mathcal{O}(s(p)^2)$ components (see Corollary 8). Since two $\sim_{\mathcal{B}}$-equivalent balls from perfectly-equivalent twin-blocks have the same B-core, we get that $\mathcal{K}$ has at most $\mathcal{O}(p^3) \cdot 2^{2^{\mathcal{O}(p^3)}} \cdot \mathcal{O}(s(p)^2) = 2^{2^{\mathcal{O}(p^3)}}$ components (cf. Observation 9). Adding p to this value before multiplying it by $\mathcal{O}(p^3)$ gives a bound on the number of balls centered in $\mathcal{K}$ or X, and with an additional factor $s(k)$ we get the maximum size of the union of their teaching sets, which is an upper bound for the number of components in $\mathcal{K}_1$. We can reiterate this to obtain an upper bound for $|\mathcal{K}_2|$ which is: $s(p) \cdot \mathcal{O}(p^3) \cdot \Big(k + s(p) \cdot \mathcal{O}(p^3) \cdot (k + 2^{2^{\mathcal{O}(p^3)}})\Big) = 2^{2^{\mathcal{O}(p^3)}}$. □

We can now formalize the notion of "well-behaved" teaching maps via the property of compactness.

Definition 7. *A teaching map T is* compact *if for any $H \in \mathcal{H} \setminus \mathcal{K}_2$ and any $x \in V(H)$, each ball whose teaching set contains x is centered in H.*

Our next aim is to prove that we can safely restrict our attention to computing compact teaching maps only; this is stated in Lemma 13 later on. Before we can establish that result, we first prove two auxiliary statements.

Lemma 11. *Let T be a positive non-clashing teaching map for $\mathcal{B}$. Let $B \in \mathcal{B}$ be centered in $H_B \in \mathcal{H} \setminus \mathcal{K}_1$ with $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$ and $H_i \in \mathcal{K}_{(B,x)}$. By definition, there exists $x_i \in [x]_{\sim_{\mathcal{B}}} \cap V(H_i)$. If replacing x by x_i in $T(B)$ creates a conflict, then there exists $z_i \in T(B) \cap V(H_i)$.*

Proof. Let B' be a ball creating a conflict with B when we replace x by x_i in $T(B)$. For a conflict to happen, it means that $x_i \in B'$ whereas $x \notin B'$. Since $x_i \in [x]_{\sim_{\mathcal{B}}}$, then B' has to be centered in H_i. Let us consider $B'_0 \in [B']_{\sim_{\mathcal{B}}}$ centered in H. Prior to the replacement, B'_0 has no conflict with B, and thus, there are two possibilities:

(1) Let us first consider the case where there exists $z \in T(B'_0)$ that distinguishes B'_0 from B. By definition, $z \notin B$. Since $B' \sim_{\mathcal{B}} B'_0$ and $H' \equiv^T_{\mathcal{B}} H$, there exists $z' \in T(B')$, such that $z' \in [z]_{\sim_{\mathcal{B}}}$. As z' does not distinguish B' from B, it holds that $z' \in B$. However, since $z' \sim_{\mathcal{B}} z$, this means that $z' \in H_B$, which leads to a contradiction since $H_B \notin \mathcal{K}_1$.

(2) Otherwise, there exists $z \in T(B)$ that distinguishes B from B'_0, which means that $z \notin B'_0$. However, $z \in B'$ since there is a conflict with B'. Since $B' \sim_{\mathcal{B}} B'_0$ and $H' \equiv^T_{\mathcal{B}} H$, then the facts that $z \notin B'_0$ and $z \in B'$ imply that $z \in H_i$. Hence, we found the z_i whose existence we claimed. □

Lemma 12. *Let T be an optimal positive non-clashing teaching map for $\mathcal{B}$. Let $B \in \mathcal{B}$ be centered in $H_B \in \mathcal{H} \setminus \mathcal{K}_1$. For $x \in T(B) \cap V(H)$ for some $H \in \mathcal{H}$, there exists $H_i \in \mathcal{K}_{(B,x)}$ and $x_i \in [x]_{\sim_{\mathcal{B}}} \cap V(H_i)$ such that replacing x by x_i in $T(B)$ does not create a conflict.*

Proof. We prove the lemma by contradiction. Suppose that for all $x_i \in [x]_{\sim_{\mathcal{B}}}$ of $\mathcal{K}_{(B,x)}$, a conflict is created. By Lemma 11, for all H_i in $\mathcal{K}_{(B,x)}$, there exists $z_i \in T(B) \cap V(H_i)$. However, these H_i's are all disjoint, and hence, $|T(B)| \geq |\mathcal{K}_{(B,x)}| = s(p) + 1$, which contradicts the maximum size of a teaching set established in Corollary 8. This proves that there exist such an H_i and x_i. □

Lemma 13. *If $\mathcal{B}$ has positive non-clashing teaching dimension k, then it also admits a compact positive non-clashing teaching map of dimension k.*

Proof. This directly follows from the repeated application of Lemma 12 to any positive non-clashing teaching map T for $\mathcal{B}$ of dimension k. Let us assume that we do not yet have a compact teaching map. By definition, there exists $H_v \in \mathcal{H} \setminus \mathcal{K}_2$, $v \in V(H_v)$ and $B_r(u)$ such that $v \in T(B_r(u))$ and $u \notin V(H_v)$, we know that $B_r(u)$ is not centered in $\mathcal{K}_1$. Indeed, otherwise $v \in T(B_r(u))$ would imply that $H_v \in \mathcal{K}_2$. We then can apply Lemma 12 to $B_r(u)$ and v, and replace v in $T(B_r(u))$ with some vertex in $\mathcal{K}$. This preserves the non-clashing quality of the teaching map, and cannot increase the size of the teaching sets. Moreover, since the number of such pairs $v, B_r(u)$ strictly decreases with each application, we eventually reach the point where there is no such occurence in the teaching map, meaning that the obtained non-clashing teaching map is compact. □

We now proceed to the crux of our algorithm: the proof that one can reduce the size of the instance $(G, \mathcal{B})$ without changing its positive non-clashing teaching dimension (formalized in Lemma 15). Toward this, it will be useful to focus on how the balls in $\mathcal{B}$ interact with certain subgraphs of G.

Definition 8. *Let G' be an induced subgraph of G. Then, $\mathcal{B}$* induces *the set $\mathcal{B}'$ of vertex sets w.r.t. G' where $\mathcal{B}' = \{B_r(u) \cap V(G') \mid B_r(u) \in \mathcal{B} \wedge u \in V(G')\}$.*

We note that the set of $\mathcal{B}'$ need not necessarily be a set of balls in G' itself: for instance, it may well happen that for some $B \in \mathcal{B}'$ the vertex set $B \cap V(G')$ is not even connected. However, under some conditions on G' that we will be able to guarantee, $\mathcal{B}'$ is, in fact, a set of balls of G'.

Lemma 14. *Let G be a graph, $\mathcal{B}$ a set of balls on G, and G' an induced subgraph of G. If $V(G')$ contains X and, for every $H \in \mathcal{H}$, either (1) $V(H) \subseteq V(G')$ or (2) $V(H) \cap V(G') = \emptyset \wedge \exists H', H'' \in [H]_{\sim_\mathcal{B}}, H \neq H'', V(H') \cup V(H'') \subseteq V(G')$ holds, then the set $\mathcal{B}'$ induced by $\mathcal{B}$ w.r.t. G' is a set of balls on G'. Moreover, there is a bijection between balls in $\mathcal{B}'$ and balls in $\mathcal{B}$ centered in G'.*

Proof. The key element of this proof is that in such a graph G', the distance $d_{G'}(u, v)$ between two vertices u and v is the same as the distance $d_G(u, v)$ between them in G. Indeed, for $u, v \in V(G')$, suppose that the shortest path between them in G uses some vertices not present in G'. For a consecutive set of such missing vertices on the path, since they're missing in G' and connected in G, they are not in X and all in the same component $H \in \mathcal{H}$. We know that all of H is missing in G', so the vertices directly before and after on the path are from X: these vertices exist because $u, v \in V(G')$. Since there is $H' \in [H]_{\sim_\mathcal{B}}$ such that $V(H') \subseteq V(G')$, we will replace the vertices of H by the $\sim_\mathcal{B}$-equivalent vertices of H'. The newly constructed path has the same length, is still a path by definition of $H \sim_\mathcal{B} H'$, and has strictly fewer vertices not in G'. Iterating this argument until we obtain a path with only vertices of G', we prove that the distance between u and v is the same in G' as it was in G. From now on, we will simply denote this distance $d(u, v)$, without referring to the graph.

We are now ready to prove that the vertex sets in $\mathcal{B}'$ are indeed balls on G'. Let $B' \in \mathcal{B}'$. There exist $u \in V(G')$ and $B_r(u) \in \mathcal{B}$, such that $B' = B_r(u) \cap V(G') = \{v \in V(G'), d(u, v) \leq r\}$. Hence, B' is indeed a ball on G'. It remains to prove that this construction is injective. Let $u, v \in V(G')$ and $B_r(u), B_{r'}(v) \in \mathcal{B}$, such that $\{B_r(u) \cap V(G')\} = \{B_{r'}(v) \cap V(G')\}$ and assume there is $w \in V(G) \setminus V(G')$ such that $w \in B_r(u)$ and $w \notin B_{r'}(v)$. Let $H \in \mathcal{H}$ such that $w \in V(H)$, and $H', H'' \in [H]_{\sim_\mathcal{B}}$ the two corresponding components whose existence is required by (2), and $w' \in V(H') \cap [w]_{\sim_\mathcal{B}}$ (resp. $w'' \in V(H'') \cap [w]_{\sim_\mathcal{B}}$). If $w' \notin B_r(u)$, it implies that $d(u, w) < d(u, w')$ and thus $u \in V(H)$, which is a contradiction since $V(G') \cap V(H) = \emptyset$. With the same argument for w'', we infer that $w', w'' \in B_r(u)$ and hence $w', w'' \in B_{r'}(v)$ since $w', w'' \in V(G')$. However, $w' \in B_{r'}(v)$ implies that $d(v, w') < d(v, w)$, which in turns imply $v \in V(H')$. We also obtain $v \in V(H'')$ by the same reasoning with w'', and this leads to a contradiction since $V(H') \cap V(H'') = \emptyset$. We proved that it was impossible to find two such balls, which proves the claimed bijection between balls in $\mathcal{B}'$ and balls centered in G' in $\mathcal{B}$. □

Let us now define the *reduced graph G' of G* as the graph obtained from G by removing all but $f(p) = c(p) + \binom{p \cdot c(p) + p}{s(p)}^{b(p)} + 1$ twin-blocks from each large class of $\sim_\mathcal{B}$. Additionally, we set $\mathcal{B}'$ to be the set induced by $\mathcal{B}$ on G' according to Definition 8, and note that Lemma 14 implies that it is a set of balls on G'. The size of G' is bounded by $g(p) = p + f(p) \cdot 2^{\mathcal{O}(p^3)} = 2^{2^{\mathcal{O}(p^3)}}$, and $\mathcal{B}'$ is hence also of bounded size: in particular, an optimal positive non-clashing teaching map for $\mathcal{B}'$ can be computed, *e.g.*, by brute force in time that depends only on p. We now aim at proving that $\mathcal{B}'$ is equivalent to $\mathcal{B}$, *i.e.*, that the optimal positive non-clashing teaching dimension is the same for both.

Lemma 15. *Suppose that there is a solution for $(G, \mathcal{B}, k)$. Then, there exists a solution T' for the reduced instance $(G', \mathcal{B}', k)$.*

Proof. Applying Lemma 13 gives us the existence of a compact solution T. Up to renaming, G' corresponds to G in which we remove only components **not** in $\mathcal{K}_2$ for T. Let us simply define T' as the restriction of T to $\mathcal{B}'$ –this is well-defined because each ball in $\mathcal{B}'$ corresponds to exactly one in $\mathcal{B}$ as per Lemma 14. It is easy to see that if two balls of $\mathcal{B}'$ were in conflict, the corresponding balls in $\mathcal{B}$ would be as well: the teaching sets are exactly the same, since we did not remove any component of $\mathcal{K}_2$ and T is compact, and the balls themselves can only be smaller in $\mathcal{B}'$, the element used by T to distinguish between the two balls of $\mathcal{B}$ can be used by T' to distinguish between the balls in $\mathcal{B}'$. □

The following lemma establishes that a solution for the reduced instance can be lifted to one for the original instance; note this also ensures that our algorithm will be constructive.

Lemma 16. *Let $k \in \mathbb{N}$ and T' be a compact solution for the instance $(G', \mathcal{B}', k)$. We can construct a compact solution T for $(G, \mathcal{B}, k)$ in time $2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$.*

Proof. We will iteratively add back the missing components $(H_0, \dots, H_x)$ of $\mathcal{H}$ and update the set of balls accordingly. We start by choosing an arbitrary ordering $\prec$ on $\mathcal{H}$ such that: $\forall H, H' \in \mathcal{H}, V(H) \subseteq V(G') \wedge V(H') \cap V(G') = \emptyset \Rightarrow H \prec H'$. We then compute $\mathcal{K}_2^{T'}$ for T' and $\prec$ on G', since it will be used at every step. The time necessary for it depends only on p and is $\mathcal{O}(|\mathcal{B}'| \cdot s(p))$, as it suffices to check for each ball which components its teaching set uses, aside from the component where the ball itself is potentially centered. Indeed, since T' is compact, H being in $\mathcal{K}_2^{T'}$ exactly means that there is a ball not centered in H using $x \in V(H)$ in its teaching set. In order to prove the lemma, we will use the following key claim.

Claim 17. *Let $0 \leq c < x$, $\mathcal{B}^*$ be the set of balls induced by $\mathcal{B}$ on $G^* = G[V(G') \bigcup_{0 \leq i < c} V(H_i)]$, and let T^* be a solution for $(G^*, \mathcal{B}^*, k)$ such that $\mathcal{K}_2^{T'}$ is the 2-extended core for T^*. We can construct in $2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$ time a solution T^+ for $(G^+, \mathcal{B}^+, k)$, where $G^+ = G[V(G^*) \cup V(H_c)]$, $\mathcal{B}^+$ is induced by $\mathcal{B}$ on G^+, and $\mathcal{K}_2^{T'}$ is the 2-extended core for T^+.*

Proof. It is easy to observe that $\mathcal{B}^*$ is induced by $\mathcal{B}^+$ on G^*. Thus, by Lemma 14, there is a bijection between the balls in $\mathcal{B}^*$ and those of $\mathcal{B}^+$ which are centered in G^*. We can compute $\mathcal{B}^+$ in time $|V(G)|^{\mathcal{O}(1)} \cdot p^2$, since for the at most $|V(G)| \cdot p^2$ balls $\mathcal{B}^+$, a breadth-first search on G^+ suffices to compute it. Note that for any ball $B^* \in \mathcal{B}^*$, the corresponding ball $B^+ \in \mathcal{B}^+$ is a superset of it, and we set $T^+(B^+) = T^*(B^*)$. Note that for any two balls whose teaching set we define in this way, there can be no conflict between them. Indeed, any element previously distinguishing their respective equivalent balls in $\mathcal{B}'$ in in $V(G')$ and thus will still be at the same distance from each center, meaning it distinguishes the two balls in $\mathcal{B}^+$ as well.

However, not all balls in $\mathcal{B}^+$ are centered in G^*: we now need to define T^+ for the balls centered in H_c. Toward this end, we first need to identify some components in G^* which will be useful to define the teaching map in a non-clashing manner.

We know that there are at least $f(p)$ components of $[H_c]_{\sim_{\mathcal{B}}}$ in G' by definition, and thus in G^*. Since $f(p) = c(p) + \binom{p \cdot c(p) + p}{s(p)}^{b(p)} + 1$, there are at least $\binom{p \cdot c(p) + p}{s(p)}^{b(p)} + 1$ components outside of $\mathcal{K}_2^{T'}$. Since all of these components are twin-blocks, they have the same balls (at most $b(p)$ of them), and by a pigeonhole principle, since each ball has a teaching set made of $s(p)$ vertices of the components of $\mathcal{K} \cup H \cup X$ (which are at most $p \cdot c(p) + p$), at least 2 of them have, for each pair of $\sim_{\mathcal{B}}$-equivalent balls, the exact same teaching sets in $\mathcal{K} \cup X$ as well as isomorphically identical teaching sets in their own components. Let us denote these two components as H', H''.

Let $u \in V(H_c)$, $r \in \mathbb{N}$ such that $B_r^+(u) \in \mathcal{B}^+$. We will simply "copy" the teaching set $T^*(B_r^*(v))$, where $v \in V(H') \cap [u]_{\sim_{\mathcal{B}}}$. Formally, for α the canonical isomorphism from H' to H_c, $T^+(B_r^+(u)) = \{\alpha(w) | w \in T^*(B_r^*(v)) \cap H'\} \cup \{w | w \in T^*(B_r^*(v)), w \notin H'\}$. Thus, from the point of view of any ball centered outside of H' and H_c, $B_r^+(u)$ and $T^+(B_r^+(u))$ will behave exactly as $B_r^*(v)$ and $T^*(B_r^*(v))$, and hence there are no conflicts with such balls, because they

were not in conflict with $B^*_r(v)$. Two balls centered in H_c are also now distinguished from each other, otherwise there would be indistinguishable pairs of balls centered in H'. The last to check is that balls centered in H_c are distinguished from balls centered in H'. And here, we use the fact that the balls in H'' have the same teaching sets (up to isomorphism between vertices of H' and H'') as those in H', and thus if there was a conflict between H' and H_c, there would also be one between balls centered in H' and H''.

Thus we managed to construct a teaching map T^+ for $\mathcal{B}^+$. Since the new teaching sets are imitations of pre-existing teaching sets in T^*, the size constraint is satisfied, and it is easy to check that the solution is still compact: the teaching sets of balls centered in H' contained only vertices of H' and $\mathcal{K}_2$, thus those of balls centered in H_c contain by construction only vertices of H_c and $\mathcal{K}_2$. H_c is not in $\mathcal{K}_2$ for T^+: indeed, $H_c \notin \mathcal{K}$ because there are already enough components $\sim_{\mathcal{B}}$ equivalent to H_c in G', and all of them are preceding H_c in the order $\prec$. Moreover, no vertex of H_c has been added to the teaching sets of the balls not centered in H_c: by definition, $H_c \notin \mathcal{K}_2$. Thus, $\mathcal{K}_2 = \mathcal{K}_2^{T'}$.

Note that we can find H' and H'' in time $f(p)^2 \cdot (c(p)+p)^{s(p)\cdot\mathcal{O}(p^3)}$ and then we construct teaching sets for at most $\mathcal{O}(p^3)$ balls, each of them being a simple copy of the equivalent teaching set in H', which can be constructed linearly in the set size: $\mathcal{O}(s(p))$. Thus the running time is $f(p)^2 \cdot (c(p) + p)^{s(p)\cdot\mathcal{O}(p^3)}\mathcal{O}(s(p)) \cdot |V(G)|^{\mathcal{O}(1)} = 2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$ and we have proven the claim. □

We can now finish proving the lemma by induction using Claim 17. Since the hypothesis trivially holds for $c = 0$ (by considering $G^* = G'$), we prove the claimed result, and the total running time follows from the fact that the number of components of $\mathcal{H}$ missing in G' are at most $|V(G)|$. □

Finally, we obtain our main parameterized tractability result by combining the previous ingredients. In particular, in the proof we construct $(G', \mathcal{B}', k)$, solve the problem there and argue correctness.

Theorem 18. NON-CLASH *is fixed-parameter tractable when parameterized by the vertex integrity of the input graph G.*

Proof. Let G be a graph with vertex integrity p, $\mathcal{B}$ a set of balls on G, and $k \in \mathbb{N}$. We provide an algorithm computing a positive non-clashing teaching map for $\mathcal{B}$ of dimension at most k, or correctly outputing that none exists, in time $q(p) \cdot |V(G)|^{\mathcal{O}(1)}$, where q is an elementary function. Before starting with the algorithm proper, it is useful to note that if $k \geq s(p)$, Corollary 8 gives that there is always a solution. Moreover, replacing the value of k by $s(p)$ for the rest of the algorithm does not hurt: the solution still exists and has dimension smaller than k. Hence, we assume in the rest of the proof that $k \leq s(p)$.

The first step of the actual algorithm is to compute the witness $X \subset V(G)$ for the vertex integrity and the corresponding set $\mathcal{H}$ of connected components. Next, we classify the elements of $\mathcal{H}$ with regard to the equivalence classes defined by $\sim_{\mathcal{B}}$, see Definition 1. Since there are at most $2^{\mathcal{O}(p^3)}$ equivalence classes and since the equivalence between two components can be tested in time $p^{\mathcal{O}(p)}$, we can compute the equivalence classes in time $|V(G)| \cdot 2^{\mathcal{O}(p^3)} \cdot p^{\mathcal{O}(p)}$ with brute force.

We are now ready to compute the reduced graph G' of G. We recall that to this end it suffices to remove some arbitrary components of $\mathcal{H}$ whose equivalence class are bigger than $f(p)$, which can be done in linear time. Recalling Definition 8, the set $\mathcal{B}'$ induced by $\mathcal{B}$ on G' can be computed in time $|V(G')| \cdot \mathcal{O}(p^2) \cdot |V(G')| \cdot s(p)$; indeed, it suffices to check for $u \in V(G')$ and $r \in \mathbb{N}$ at most the diameter of the graph whether $B_r(u)$ exists in $\mathcal{B}$, and then to compute the intersection of it with $V(G')$. The size of the instance $(G', \mathcal{B}', k)$ is upper-bounded by a function of p, meaning that we can compute a non-clashing teaching map of dimension at most k for $\mathcal{B}'$ in time depending only on p (or determine that none exists)—see Proposition 5.

Using Lemma 15, we know that if there is no such teaching map for $\mathcal{B}'$, there is also none of dimension k for $\mathcal{B}$: the algorithm can safely output that no solution exists. Conversely, if we obtain a teaching map for $\mathcal{B}'$, we can use Lemma 16 to construct a non-clashing teaching map of dimension at most k for $\mathcal{B}$ in time $2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$. This concludes the proof, and the total running time can be upper-bounded by the same asymptotic bound of $2^{2^{\mathcal{O}(p^3)}} \cdot |V(G)|^{\mathcal{O}(1)}$. □

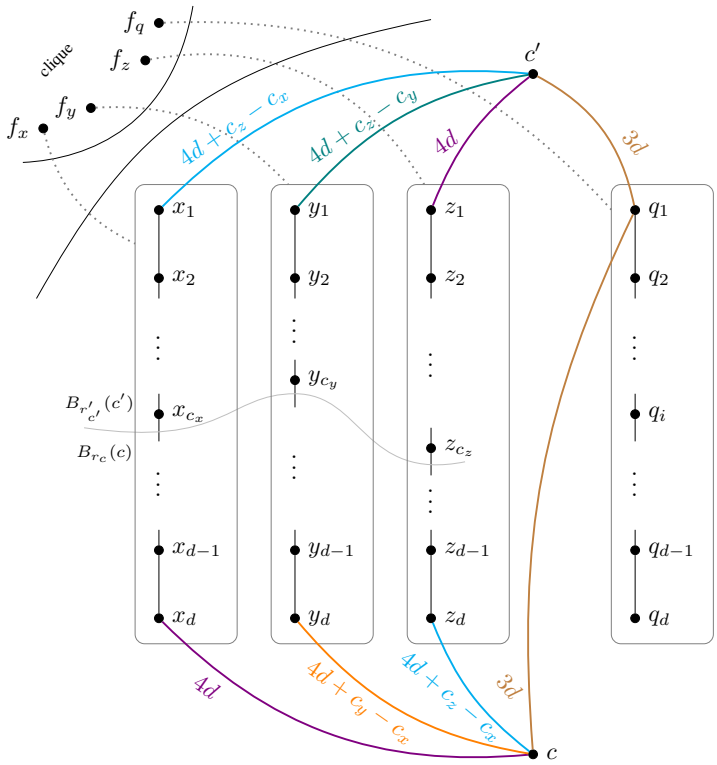


Figure 2: Colorful edges denote paths of the depicted lengths. Two black curves separate the clique from the rest of the graph. Each dotted edge shows that a vertex of the clique is adjacent to all the vertices of the graph below the separating curves except those in the adjacent rectangle. The gray curve gives an intuition of which vertices of P_x, P_y, and P_z are contained in $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$.

5 Hardness for Classical Structural Parameterizations

In this section, we prove that Non-Clash is intractable for many of the structural graph parameters that have been studied in the literature. We prove the result for the feedback vertex number and pathwidth; by extension, intractability then follows also for well-known generalizations of these measures such as treewidth and clique-width.

Theorem 19. Non-Clash *is* W[1]*-hard parameterized by* $\mathtt{fvs}(G) + \mathtt{pw}(G) + k$.

We establish Theorem 19 by describing a parameterized reduction below and proving its correctness in Lemma 20. We reduce from a problem called NAE-Integer-3-SAT, which is W[1]-hard parameterized by the number of variables (Bringmann et al., 2016), and is defined as follows.

NAE-Integer-3-SAT
Input: A set of clauses $\mathcal{C}$ over variables $\mathcal{X}$, and an integer d. Any clause $c \in \mathcal{C}$ has the form $x \leq c_x$, $y \leq c_y$, and $z \leq c_z$, where $c_x, c_y, c_z \in \{1, \ldots, d\}$.
Question: Is there a variable assignment $\mathcal{X} \to \{1, \ldots, d\}$ such that for each clause, either 1 or 2 of its three inequalities are satisfied?

The Reduction. Given an instance ϕ of NAE-Integer-3-SAT, we construct an instance $(G, \mathcal{B}, k)$ of Non-Clash in polynomial time as follows, starting with the graph G.

- For each variable $x \in \mathcal{X}$, make a (variable) path $P^x := (x_1, \ldots, x_d)$ of order d.
- For each clause $c \in \mathcal{C}$, make two vertices c and c' and, w.l.o.g., suppose that the clause c contains the variables x, y, and z, and that $c_x \leq c_y \leq c_z$. Connect the vertex c to x_d, y_d,

and z_d by three distinct paths of lengths $4d$, $4d + c_y - c_x$, and $4d + c_z - c_x$, respectively.[2] These long paths ensure that there are no unwanted shortcuts in G and that, for $r_c = 5d - c_x - 1$, the ball $B_{r_c}(c)$ contains all of P^x, P^y, and P^z except for their first c_x, c_y, and c_z vertices (whose respective indices correspond to the respective variable values that satisfy the clause c), respectively. Similarly, connect the vertex c' to x_1, y_1, and z_1 by three distinct paths of lengths $4d + c_z - c_x$, $4d + c_z - c_y$, and $4d$, respectively. Analogously, this ensures that, for $r'_{c'} = 4d + c_z - 1$, the ball $B_{r'_{c'}}(c')$ contains the opposite vertices to $B_{r_c}(c)$ in P^x, P^y, and P^z (whose respective indices correspond to the respective variable values that do not satisfy the clause c), while ensuring that no unwanted shortcuts exist in G.

- For each clause $c \in \mathcal{C}$ and each variable $q \in \mathcal{X}$ such that c does not contain q, in G, connect the vertices c and c' to q_1 by distinct paths of length $3d$. These paths ensure that the balls $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$ described above contain every other variable path completely, while ensuring that no unwanted shortcuts exist in G.
- Let S be the set of all the vertices that currently exist in G. For each $x \in \mathcal{X}$, in G, make a vertex f_x and connect it to each vertex in S except those in P^x via a distinct path of length $6d$. Finally, for all $x, y \in \mathcal{X}$, make f_x adjacent to f_y. This ensures that, for each $x \in \mathcal{X}$, the ball $B_{6d}(f_x)$ contains every vertex in G except for those in P^x, while ensuring that no unwanted shortcuts exist in G. This completes the construction of G (see Figure 2).

Set $k := |\mathcal{X}|$ and $\mathcal{B}$ to contain $V(G)$, $B_{6d}(f_x)$ for all $x \in \mathcal{X}$, and $B_{r_c}(c)$ and $B_{r'_{c'}}(c')$ for all $c \in \mathcal{C}$.

Correctness of the Reduction. Suppose, given an instance ϕ of NAE-INTEGER-3-SAT, that the reduction from the subsection above returns $(G, \mathcal{B}, k)$ as an instance of NON-CLASH.

Lemma 20. *ϕ is a YES-instance of* NAE-INTEGER-3-SAT *if and only if $(G, \mathcal{B}, k)$ is a YES-instance of* NON-CLASH.

Proof. First, suppose that ϕ is a YES-instance of NAE-INTEGER-3-SAT. We construct a positive teaching map T as follows. In the NAE-satisfying variable assignment for ϕ, for each variable $x \in \mathcal{X}$, if the integer j is assigned to x, then place x_j in $T(V(G))$. For each $c \in \mathcal{C}$, set $T(B_{r_c}(c)) := \{c\}$ and $T(B_{r'_{c'}}(c')) := \{c'\}$. Lastly, for each $x \in X$, $T(B_{6d}(f_x))$ contains f_x and one arbitrary vertex from each of the $k - 1$ variable paths it contains. We prove that T is non-clashing for $\mathcal{B}$.

For any two balls $B_1, B_2 \in \mathcal{B}$ centered at vertices of the form c or c' for the same or different clauses, T satisfies the non-clashing condition since $T(B_1)$ contains its center (some c or c') while B_2 does not contain this vertex. Indeed, the radius of B_2 is less than $5d$, while the distance between any two vertices of the form c or c' is at least $6d$ since any shortest path between them contains a vertex from a variable path. For any $x, y \in \mathcal{X}$, T satisfies the non-clashing condition for $B_{6d}(f_x)$ and $B_{6d}(f_y)$ since $T(B_{6d}(f_x))$ contains a vertex in P^y while $B_{6d}(f_y)$ does not. For any $x \in \mathcal{X}$ and $c \in \mathcal{C}$, T satisfies the non-clashing condition for $B_{6d}(f_x)$ and $B_{r_c}(c)$, as well as $B_{6d}(f_x)$ and $B_{r_{c'}}(c')$, since $T(B_{6d}(f_x))$ contains f_x while $B_{r_c}(c)$ and $B_{r_{c'}}(c')$ do not as $r_c, r_{c'} < 5d$ while c and c' are at distance $6d$ from f_x. For any $x \in \mathcal{X}$, T satisfies the non-clashing condition for $V(G)$ and $B_{6d}(f_x)$ since $T(V(G))$ contains a vertex in P^x while $B_{6d}(f_x)$ does not. Finally, for any $c \in \mathcal{C}$, T satisfies the non-clashing condition for $V(G)$ and $B_{r_c}(c)$, as well as $V(G)$ and $B_{r_{c'}}(c')$, since, among the variables contained in the clause c, $T(V(G))$ contains at least one vertex from one of those variable paths whose index satisfies c, and at least one vertex from one of those variable paths whose index does not satisfy c. As can be recalled from the construction, this implies that $B_{r_c}(c)$ and $B_{r_{c'}}(c')$ do not contain the respective vertices. Thus, T satisfies the non-clashing property for all pairs of balls in $\mathcal{B}$.

Now, we prove the reverse direction, so suppose that $(G, \mathcal{B}, k)$ is a YES-instance of NON-CLASH and that T is the corresponding teaching map. For all $x \in \mathcal{X}$, in order for T to satisfy the non-clashing condition for $B_{6d}(f_x)$ and $V(G)$, we have that $T(V(G))$ contains at least one vertex from P^x as $B_{6d}(f_x) \subset V(G)$ and $V(G) \setminus B_{6d}(f_x)$ is restricted to the vertices in P^x. Since $k = |\mathcal{X}|$, we in fact have that $T(V(G))$ contains exactly one vertex from P^x for all $x \in \mathcal{X}$. Extract a variable assignment for ϕ from $T(V(G))$ as follows. For each $x \in X$, assign the variable x the value of the index of the unique vertex contained in both P^x and $T(V(G))$. We prove that this is an NAE-satisfying variable assignment for ϕ.

[2] "Connect two vertices u, v by a path of length p" means to make u and v the endpoints of a path of length p.

W.l.o.g., let $c \in \mathcal{C}$ be a clause containing the variables $x, y, z \in \mathcal{X}$. In order for T to satisfy the non-clashing condition for $V(G)$ and $B_{r_c}(c)$, $T(V(G))$ must contain at least one vertex in P^x, P^y or P^z that is not contained in $B_{r_c}(c)$. Analogously, in order for T to satisfy the non-clashing condition for $V(G)$ and $B_{r'_{c'}}(c')$, $T(V(G))$ must contain at least one vertex in P^x, P^y or P^z that is not contained in $B_{r'_{c'}}(c')$. Recall that all of the vertices in P^x, P^y, and P^z that are not contained in $B_{r_c}(c)$ have respective indices that correspond to the respective variable values that satisfy the clause c. Similarly, recall that all of the vertices in P^x, P^y, and P^z that are not contained in $B_{r_{c'}}(c')$ have respective indices that correspond to the respective variable values that do not satisfy the clause c. As these arguments hold for any clause $c \in \mathcal{C}$, the variable assignment extracted above corresponds to an NAE-satisfying variable assignment for ϕ. □

Now, we are ready to proceed with the proof of Theorem 19.

Proof of Theorem 19. Lemma 20 establishes the correctness of the polynomial-time reduction from the beginning of Section 5. To complete the proof, it remains to show that $\texttt{fvs}(G) + \texttt{pw}(G) + k$ is bounded above by a function of $|\mathcal{X}|$. This clearly holds for k which is, by definition, $|\mathcal{X}|$. Deleting from G the vertices x_1, x_d, and f_x for all $x \in \mathcal{X}$ results in an acyclic graph G'; in particular G has a feedback vertex set of size $3|\mathcal{X}|$. To establish a bound on $\texttt{pw}(G)$, it now suffices to show that G' also has bounded pathwidth. Note that G' consists of a set of connected components, each of which is either a subdivided caterpillar (this is what remains of each component containing a variable path) or a vertex (of the form c or c') with multiple pendent subdivided caterpillars and (simple) paths. Since deleting one further vertex from each connected component may only reduce the pathwidth by 1 and we need a single such deletion operation to reach a graph class of constant pathwidth (see Section 2), we also obtain that $\texttt{pw}(G)$ is bounded by a function of $|\mathcal{X}|$. □

6 Concluding Remarks

Our computational upper and lower bounds provide a near-comprehensive understanding of the complexity of computing the positive non-clashing teaching dimension. Apart from our contributions to the previously studied strict setting, we consider it notable that our work is the first to also tackle the complexity of non-clashing teaching in the non-strict setting—*i.e.*, the more general (and arguably more natural) case where not all possible concepts are present.

One open question highlighted by our work concerns the tiny remaining gap between the algorithmic lower and upper bounds obtained in Theorem 4 and Proposition 5. In particular, is there a way to improve the running time of the latter algorithm to $2^{\mathcal{O}(|V(G)| \cdot d \cdot k)}$ and make the bounds tight? More general directions for future work are to perform a similar complexity analysis in the non-positive setting and to consider approximation algorithms.

References

Baris Akgun, Maya Cakmak, Jae W. Yoo, and Andrea L. Thomaz. Trajectories and keyframes for kinesthetic teaching: A human-robot interaction perspective. In *Proc. of the 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI 2012)*, pp. 391–398, 2012.

Dana Angluin. Inductive inference of formal languages from positive data. *Information and Control*, 45:117–135, 1980a.

Dana Angluin. Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, 21:46–62, 1980b.

Shai Ben-David and Ami Litman. Combinatorial variability of Vapnik-Chervonenkis classes with applications to sample compression schemes. *Discrete Applied Mathematics*, 86:3–25, 1998.

Cornelius Brand, Robert Ganian, and Kirill Simonov. A parameterized theory of PAC learning. In *Proc. of the 37th AAAI Conference on Artificial Intelligence (AAAI 2023)*, pp. 6834–6841, 2023.

Karl Bringmann, Danny Hermelin, Matthias Mnich, and Erik Jan van Leeuwen. Parameterized complexity dichotomy for steiner multicut. *Journal of Computer and System Sciences*, 82(6): 1020–1043, 2016.

Daniel S. Brown and Scott Niekum. Machine teaching for inverse reinforcement learning: Algorithms and applications. In *Proc. of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)*, volume 33, pp. 7749–7758, 2019.

Jérémie Chalopin, Victor Chepoi, Shay Moran, and Manfred K. Warmuth. Unlabeled sample compression schemes and corner peelings for ample and maximum classes. *Journal of Computer and System Sciences*, 127:1–28, 2022.

Jérémie Chalopin, Victor Chepoi, Fionn Mc Inerney, Sébastien Ratel, and Yann Vaxès. Sample compression schemes for balls in graphs. *SIAM Journal on Discrete Mathematics*, 37(4):2585–2616, 2023.

Jérémie Chalopin, Victor Chepoi, Fionn Mc Inerney, and Sébastien Ratel. Non-clashing teaching maps for balls in graphs. In *Proc. of the 37th Annual Conference on Learning Theory (COLT 2024)*, volume 247 of *Proceedings of Machine Learning Research*, pp. 840–875, 2024.

Yuxin Chen, Adish Singla, Oisin Mac Aodha, Pietro Perona, and Yisong Yue. Understanding the role of adaptivity in machine teaching: The case of version space learners. In *Advances in Neural Information Processing Systems (NeurIPS 2018)*, volume 31, pp. 1483–1493, 2018.

Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

François Denis. Learning regular languages from simple positive examples. *Machine Learning*, 44: 37–66, 2001.

Reinhard Diestel. *Graph Theory, 6th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2024.

Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

Rodney G. Downey, Patricia A. Evans, and Michael R. Fellows. Parameterized learning complexity. In *Proc. of the 6th Annual ACM Conference on Computational Learning Theory (COLT 1993)*, pp. 51–57, 1993.

Pål Grønås Drange, Markus S. Dregi, and Pim van 't Hof. On the computational complexity of vertex integrity and component order connectivity. *Algorithmica*, 76(4):1181–1202, 2016.

Eduard Eiben, Robert Ganian, Iyad A. Kanj, Sebastian Ordyniak, and Stefan Szeider. The computational complexity of concise hypersphere classification. In *Proc. of the International Conference on Machine Learning (ICML 2023)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 9060–9070, 2023a.

Eduard Eiben, Sebastian Ordyniak, Giacomo Paesani, and Stefan Szeider. Learning small decision trees with large domain. In *Proc. of the 32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*, pp. 3184–3192, 2023b.

Shaun Fallat, David Kirkpatrick, Hans U. Simon, Abolghasem Soltani, and Sandra Zilles. On batch teaching without collusion. *Journal of Machine Learning Research*, 24:40:1–40:33, 2023.

Michael R Fellows and Sam Stueckle. The immersion order, forbidden subgraphs and the complexity of network integrity. *J. Combin. Math. Combin. Comput*, 6(1):23–32, 1989.

S. Floyd. *Space-bounded learning and the Vapnik-Chervonenkis dimension*. PhD thesis, U.C. Berkeley, 1989.

Robert Ganian and Viktoriia Korchemna. The complexity of Bayesian network learning: Revisiting the superstructure. In *Advances in Neural Information Processing Systems (NeurIPS 2021)*, volume 34, pp. 430–442, 2021.

Ziyuan Gao, Christoph Ries, Hans U. Simon, and Sandra Zilles. Preference-based teaching. *Journal of Machine Learning Research*, 18:1–32, 2017.

Tatsuya Gima and Yota Otachi. Extended MSO model checking via small vertex integrity. *Algorithmica*, 86(1):147–170, 2024.

Tatsuya Gima, Tesshu Hanaka, Masashi Kiyomi, Yasuaki Kobayashi, and Yota Otachi. Exploring the gap between treedepth and vertex cover through vertex integrity. *Theoretical Computer Science*, 918:60–76, 2022.

Sally A. Goldman and Michael J. Kearns. On the complexity of teaching. *Journal of Computer and System Sciences*, 50(1):20–31, 1995.

Sally A. Goldman and H. David Mathias. Teaching a smarter learner. *Journal of Computer and System Sciences*, 52(2):255–267, 1996.

Tesshu Hanaka, Michael Lampis, Manolis Vasilakis, and Kanae Yoshiwatari. Parameterized vertex integrity revisited. In *Proc. of the 49th International Symposium on Mathematical Foundations of Computer Science (MFCS 2024)*, volume 306 of *LIPIcs*, pp. 58:1–58:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

Mark K. Ho, Michael Littman, James MacGlashan, Fiery Cushman, and Joseph L. Austerweil. Showing versus doing: Teaching by demonstration. In *Advances in Neural Information Processing Systems (NeurIPS 2016)*, volume 33, pp. 3027–3035, 2016.

Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001.

David Kirkpatrick, Hans U. Simon, and Sandra Zilles. Optimal collusion-free teaching. In *Proc. of the 30th International Conference on Algorithmic Learning Theory (ALT 2019)*, volume 98 of *Proceedings of Machine Learning Research*, pp. 506–528, 2019.

Dima Kuzmin and Manfred K. Warmuth. Unlabelled compression schemes for maximum classes. *Journal of Machine Learning Research*, 8:2047–2081, 2007.

Yuanzhi Li and Yingyu Liang. Learning mixtures of linear regressions with nearly optimal complexity. In *Proc. of the 31st Conference on Learning Theory (COLT 2018)*, volume 75 of *Proceedings of Machine Learning Research*, pp. 1125–1144, 2018.

N. Littlestone and M. K. Warmuth. Relating data compression and learnability. *Unpublished*, 1986.

Farnam Mansouri, Yuxin Chen, Ara Vartanian, Jerry Zhu, and Adish Singla. Preference-based batch and sequential teaching: Towards a unified view of models. In *Advances in Neural Information Processing Systems (NeurIPS 2019)*, volume 32, pp. 9195–9205, 2019.

Shike Mei and Xiaojin Zhu. Using machine teaching to identify optimal training-set attacks on machine learners. In *Proc. of the 29th AAAI Conference on Artificial Intelligence (AAAI 2015)*, volume 29, pp. 2871–2877, 2015.

Sebastian Ordyniak and Stefan Szeider. Parameterized complexity of small decision tree learning. In *Proc. of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*, pp. 6454–6462, 2021.

Ingo Schwab, Wolfgang Pohl, and Ivan Koychev. Learning to recommend from positive evidence. In *Proc. of the 5th International Conference on Intelligent User Interfaces (IUI 2000)*, pp. 241–247, 2000.

Ayumi Shinohara and Satoru Miyano. Teachability in computational learning. *New Generation Computing*, 8:337–347, 1991.

Hans U. Simon. Tournaments, johnson graphs, and NC-teaching. In *Proc. of the 34th International Conference on Algorithmic Learning Theory (ALT 2023)*, volume 201 of *Proceedings of Machine Learning Research*, pp. 1411–1428, 2023.

Andreas Stolcke and Stephen Omohundro. Inducing probabilistic grammars by Bayesian model merging. In *Proc. of the 2nd International Colloquium on Grammatical Inference (ICGI 1994)*, pp. 106–118, 1994.

Jan Arne Telle, José Hernández-Orallo, and Cèsar Ferri. The teaching size: computable teachers and learners for universal languages. *Machine Learning*, 108(8-9):1653–1675, 2019.

Andrea L. Thomaz and Maya Cakmak. Learning about objects with human teachers. In *Proc. of the 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI 2009)*, pp. 15–22, 2009.

Chunlin Wang, Chris Ding, Richard F. Meraz, and Stephen R. Holbrook. PSoL: a positive sample only learning algorithm for finding non-coding RNA genes. *Bioinformatics*, 22(21):2590–2596, 2006.

Malik Yousef, Segun Jung, Louise C. Showe, and Michael K. Showe. Learning from positive examples when the negative class is undetermined - microRNA gene identification. *Algorithms for Molecular Biology*, 3(2), 2008.

Xuezhou Zhang, Xaiojin Zhu, and Stephen Wright. Training set debugging using trusted items. In *Proc. of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*, pp. 4482–4489, 2018.

Xiaojin Zhu. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *Proc. of the 29th AAAI Conference on Artificial Intelligence (AAAI 2015)*, volume 29, pp. 4083–4087, 2015.

Xiaojin Zhu, Adish Singla, Sandra Zilles, and Anna N. Rafferty. An overview of machine teaching. *Arxiv:1801.05927*, 2018. URL `https://arxiv.org/abs/1801.05927`.

Sandra Zilles, Steffen Lange, Robert Holte, and Martin Zinkevich. Models of cooperative teaching and learning. *Journal of Machine Learning Research*, 12:349–384, 2011.