Number of Variables for Graph Differentiation and the Resolution of GI Formulas

Jacobo Torán
Universität Ulm, Germany

Florian Wörz
Universität Ulm, Germany

Abstract

We show that the number of variables and the quantifier depth needed to distinguish a pair of graphs by first-order logic sentences exactly match the complexity measures of clause width and positive depth needed to refute the corresponding graph isomorphism formula in propositional narrow resolution.

Using this connection, we obtain upper and lower bounds for refuting graph isomorphism formulas in (normal) resolution. In particular, we show that if k is the number of variables needed to distinguish two graphs with n vertices each, then there is an $n^{\mathrm{O}(k)}$ resolution refutation size upper bound for the corresponding isomorphism formula, as well as lower bounds of 2^{k-1} and k for the tree-like resolution size and resolution clause space for this formula. We also show a (normal) resolution size lower bound of $\exp\big(\Omega(k^2/n)\big)$ for the case of colored graphs with constant color class sizes.

Applying these results, we prove the first exponential lower bound for graph isomorphism formulas in the proof system SRC-1, a system that extends resolution with a global symmetry rule, thereby answering an open question posed by Schweitzer and Seebach.

2012 ACM Subject Classification Theory of computation → Proof complexity; Theory of computation → Complexity theory and logic; Mathematics of computing → Graph theory

Keywords and phrases Proof Complexity, Resolution, Narrow Width, Graph Isomorphism, k-variable fragment first-order logic $\mathcal{L}_k$, Immerman's Pebble Game, Symmetry Rule, SRC-1

Digital Object Identifier 10.4230/LIPIcs.CSL.2022.36

Related Version *Full Version*: `https://eccc.weizmann.ac.il/report/2021/097/` [37]

Funding This research was supported by the Deutsche Forschungsgemeinschaft (DFG) under project number 430150230, "Complexity measures for solving propositional formulas".

Acknowledgements We thank Pascal Schweitzer for helpful discussions. We would also like to thank the anonymous reviewers for many insightful comments and suggestions.

1 Introduction

In an attempt to give a logical characterization of polynomial-time decidable graph properties, as well as a description of general classes of graph canonization algorithms, Immerman identified certain fragments of first-order logic suitable for expressing graph properties [21, 22]. In this setting, for such a language $\mathcal{L}$ of first-order logic sentences, two graphs G and H are *$\mathcal{L}$-equivalent*, denoted by $G \equiv_{\mathcal{L}} H$, if for all sentences $\psi \in \mathcal{L}$ it holds that $G \vDash \psi \iff H \vDash \psi$. Immerman noticed that the number of variables needed for expressing a property is a good complexity measure and defined the *k-variable fragment of first-order logic* $\mathcal{L}_k$ as the set of first-order logic formulas with the edge and equality relations that use at most k different variables (possibly re-quantifying them). He also defined the stronger class $\mathcal{C}_k$ by adding counting quantifiers to the class $\mathcal{L}_k$ and defined two pebble games for proving (non)equivalence of structures in these classes.

30th EACSL Annual Conference on Computer Science Logic (CSL 2022).
Editors: Florin Manea and Alex Simpson; Article No. 36; pp. 36:1–36:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

It was shown in [8] that two graphs are $\mathcal{C}_k$-equivalent if and only if they cannot be distinguished with the $(k-1)$-dimensional Weisfeiler–Leman algorithm, a well-known method for testing graph isomorphism. Roughly speaking, the 1-dimensional Weisfeiler–Leman (WL) algorithm [41, 40], or color refinement algorithm, identifies non-isomorphic colored graphs by updating in a series of steps the original vertex colors according to the multiset of colors of their neighbors. This basic step is applied repeatedly until the coloring stabilizes. This procedure can be generalized to the k-dimensional Weisfeiler–Leman algorithm (k-WL) by partitioning the set of k-tuples of vertices into automorphism-invariant equivalence classes (see e. g., [8, 23, 24] for excellent overviews of the powers and limits of this procedure).

The graph isomorphism problem (GraphIso), deciding whether two given graphs are isomorphic, has been intensively studied, as it is one of the few problems in NP that is not known to be complete for this class nor to be in P. Also unknown is whether the problem is in co-NP. It had been conjectured that GraphIso is solvable using the k-dimensional Weisfeiler–Leman algorithm, with k being sublinear in the number of vertices of the graphs. However, this was shown to be false in the seminal work of Cai, Fürer, and Immerman [8], using the $\mathcal{C}_k$ pebble game as a central tool. The Weisfeiler–Leman method still plays a central role in the algorithmic research on GraphIso; for example, Babai's celebrated algorithm for GraphIso [4] uses the k-WL method as a subroutine, with k being polylogarithmic in the number of vertices.

The field of proof complexity provides a different approach for studying the complexity of the GraphIso problem. Roughly speaking, in this setting, one tries to find out the smallest size of a proof in a concrete system of the fact that two graphs are non-isomorphic. It holds that GraphIso is in co-NP if and only if there is a concrete proof system with polynomial-size proofs of non-isomorphism. Similar to the Cook–Reckhow program [10] for the unsatisfiability problem UNSAT, this defines a clear line of research trying to provide superpolynomial size lower bounds for refuting graph (non)isomorphism formulas in stronger and stronger proof systems. The situation is even more interesting here than in the SAT case, since it would not be too surprising if GraphIso $\in$ co-NP, and this would imply the existence of polynomial-size proofs for the problem in some system. In fact, GraphIso is in co-AM [5], a randomized version of co-NP.

A first example of such a lower bound was given in [36], where it was shown that a family of unsatisfiable formulas encoding pairs of non-isomorphic graphs in a natural way requires exponential-size resolution refutations. These graphs are based on the CFI construction from [8]. The lower bound can be explained as an "encoding" of the Tseitin tautologies [38] into graph isomorphism instances. This result has been extended to stronger proof systems: In [7], the authors proved linear degree lower bounds for the algebraic systems Polynomial Calculus and Positivstellensatz by studying graphs arising from Tseitin tautologies. They furthermore characterized the power of the Weisfeiler–Leman algorithm in terms of an algebraic proof system lying between degree-k Nullstellensatz and degree-k Polynomial Calculus. Moreover, it has been shown in [3, 28, 18] that the expressive power of k-WL lies between the k-th and $(k+1)$-st level of the canonical Sherali–Adams LP hierarchy [34]. By the construction in [8], no sublinear level of Sherali–Adams suffices to decide GraphIso. Again, building on the work of [8], it was shown in [30] that there exist pairs of non-isomorphic n-vertex graphs such that any Sum-of-Squares proof of non-isomorphism must have degree $\Omega(n)$. In related work [9], it was shown that no sublinear level of the Lasserre hierarchy suffices to decide GraphIso.

Very recently, a different view was considered by Schweitzer and Seebach in [33] by introducing symmetry rules into the picture. The authors proved that resolution extended with the well-known symmetry rule SRC-2 from Krishnamurthy [26] has polynomial-size refutations for all the instances of the graph isomorphism problem for which exponential size lower bounds for (normal) resolution are known. They pointed to the search for hard instances of graph isomorphism for resolution extended with the existing symmetry rules that define the proof systems SRC-1, SRC-2, and SRC-3, a hierarchy of systems with more and more powerful symmetry rules [1, 35]. They pose the question of whether graph non-isomorphism formulas have superpolynomial resolution complexity in any of these proof systems. These are very interesting questions since finding symmetries in a formula in order to be able to apply Krishnamurthy's rules is closely related to graph isomorphism. Finding lower bounds for non-isomorphism in a system with symmetry rules can be seen as finding lower bounds for proving non-isomorphism with the help of an "isomorphism subroutine".

1.1 Our Results

We show a strong connection between the $\mathcal{L}_k$ fragment of first-order logic and the propositional resolution proof system. This is done by proving that the number of variables and the quantifier depth simultaneously needed to distinguish two graphs G and H in first-order logic exactly corresponds to the width and positive depth of a narrow resolution refutation of the unsatisfiable formula $\mathrm{ISO}(G, H)$ stating that the graphs are isomorphic (Theorem 17). Narrow resolution [17] is a slight variation of (normal) resolution that allows a distinction by cases rule, allowing to deal with the inconveniences of having long clauses in the formula. As in the case of the clause width measure [6], narrow width allows, in our case, to derive upper and lower bounds for the size of the resolution refutations of non-isomorphism. Furthermore, we show that narrow width also provides a lower bound for the clause space needed in resolution, as it is the case for the standard width measure. In particular, we prove that for any pair of non-isomorphic graphs (G, H) with n vertices each and $k \in \mathbb{N}$:

- If $G \not\equiv_{\mathcal{L}_k} H$, then there is a (normal) resolution refutation of $\mathrm{ISO}(G, H)$ of size $n^{\mathrm{O}(k)}$;
- if $G \equiv_{\mathcal{L}_k} H$, then every tree-like resolution refutation of $\mathrm{ISO}(G, H)$ has size $\geq 2^k$;
- if $G \equiv_{\mathcal{L}_k} H$, then every (normal) resolution refutation of $\mathrm{ISO}(G, H)$ has clause space $\geq k + 1$; and
- for a pair of graph colorings (λ, μ) with $(G, \lambda) \equiv_{\mathcal{L}_k} (H, \mu)$, every (normal) resolution refutation of $\mathrm{ISO}(G, H)$ has size $\exp\big(\Omega(k^2/m^2)\big)$, where $m := \sum_{v \in G} |\text{color-class}(v)|$.

The last result allows to directly derive resolution size lower bounds from Immerman's pebble game for $\mathcal{L}_k$. We use this result to prove that a version of the multipede graphs defined in [11] has exponential resolution size lower bounds. We also observe that Krishnamurthy's SRC-1 symmetry rule cannot be applied to the isomorphism formulas for asymmetric graphs and conclude that the resolution size lower bound for the multipede graphs also holds for the SRC-1 system. This provides the first example of a class of graphs whose isomorphism formulas have exponential size lower bounds for the size of resolution refutations with one of the symmetry rules, thus solving a question from [33].

1.2 Organization of This Paper

The rest of this paper is organized as follows. In Section 2, we introduce resolution complexity measures, narrow resolution, and Krishnamurthy's symmetry rules, as well as the graph isomorphism formulas and Immerman's pebble game. Then, in Section 3, we prove the connection between narrow resolution width and $\mathcal{L}_k$. This yields the upper bounds on resolution size and the lower bounds on tree-like resolution size for refuting $\mathrm{ISO}(G, H)$. The exponential lower bound for the size of SRC-1 graph isomorphism formula refutations is shown in Section 4. Finally, in Section 5, clause space lower bounds for proving graph non-isomorphism in resolution are shown.

Due to space reasons some proofs have been omitted from this version. They can be found in the full-length version of the paper [37].

2 Preliminaries

We let $\mathbb{N}$ denote the set of positive integers. For $n \in \mathbb{N}$, we let $[n] := \{k \in \mathbb{N} \mid 1 \le k \le n\}$.

A *literal* ℓ over a Boolean variable x is either x itself or its negation $\overline{x} := \neg x$. For a literal ℓ, we put $\overline{\ell} := \neg x$ if $\ell = x$, and $\overline{\ell} := x$ if $\ell = \neg x$; and call ℓ and $\overline{\ell}$ *complementary* literals. A *clause* $C = (\ell_1 \vee \cdots \vee \ell_k)$ is a (possibly empty) disjunction of literals ℓ_i. We let the symbol $\square$ denote the contradictory *empty clause* (the clause without any literals). A *CNF formula* $F = C_1 \wedge \cdots \wedge C_m$ is a conjunction of clauses. It is often advantageous to think of clauses as sets of literals and CNF formulas as sets of clauses (i. e., sets of sets). The *set of variables occurring in a clause* C will be denoted by $\mathrm{Vars}(C)$. The notion of the set of variables in a clause is extended to CNF formulas by taking unions. An *assignment/restriction* α for a CNF formula F is a function that maps some subset of $\mathrm{Vars}(F)$, denoted by $\mathrm{Dom}(\alpha)$, to $\{0,1\}$. We will consider the graph of this function and call this set also an assignment. We let $|\alpha| := |\mathrm{Dom}(\alpha)|$ be the *size* of α. We denote the empty assignment with ε. By naturally extending α by the definition $\alpha(\overline{x}) := \overline{\alpha(x)}$, we can define the result of applying α to C, which we denote by $C|_\alpha$: one deletes all occurrences of literals ℓ from C, where $\alpha(\ell) = 0$; if there is a literal $\ell \in C$ with $\alpha(\ell) = 1$, then $C|_\alpha = 1$. The notation $F|_\alpha$ denotes the formula, where all clauses containing a literal ℓ with $\alpha(\ell) = 1$ are deleted and each remaining clause C is replaced by $C|_\alpha$. If ℓ is a literal that is not assigned by α, and $a \in \{0,1\}$, then $\alpha\{\ell = a\}$ denotes the *extension of* α with $\big(\alpha\{\ell = a\}\big)(x) := \alpha(x)$ for all $x \notin \{\ell, \overline{\ell}\}$ and $\big(\alpha\{\ell = a\}\big)(\ell) = a$ as well as $\big(\alpha\{\ell = a\}\big)(\overline{\ell}) = 1 - a$.

2.1 Resolution and Complexity Measures

If $B \vee x$ and $C \vee \overline{x}$ are clauses, then the *resolution rule* allows the derivation of the clause $R := (B \vee C)$. In the resolution rule, we call $B \vee x$ and $C \vee \overline{x}$ the *parents* and R the *resolvent*.

▶ **Definition 1.** *A* resolution derivation *of a clause D from a CNF formula F (denoted by $\pi : F \vdash D$) is an ordered sequence of clauses $\pi = (C_1, \dots, C_t)$ such that $C_t = D$, and each clause C_i, for $i \in [t]$, is*

(1) *either an* axiom clause $C_i \in F$,

(2) *or a* weakening *of a clause C_j with $j < i$, i. e., $C_i \supseteq C_j$,*

(3) *or is derived from clauses C_j and C_k with $j < k < i$ by the* resolution rule.

A derivation of the empty clause from an unsatisfiable CNF formula F is called refutation.

To every refutation π, we can associate a *refutation DAG* G_π: The clauses of the refutations label the vertices of the DAG; for every application of the resolution rule we include edges from the parents to the resolvent; and for each application of the weakening rule we include edges from the original to the weakened clauses. We say that a resolution refutation π is *tree-like* if G_π is a tree.

▶ **Definition 2.** *The* size *of a resolution refutation π, denoted* $\mathrm{Size}(\pi)$*, is defined to be the number of vertices in the underlying refutation DAG G_π.*

The width of a clause C *is defined by* $\mathrm{Width}(C) := |C|$, *whereas the* width of a formula F *is given by* $\mathrm{Width}(F) := \max_{C \in F} \mathrm{Width}(C)$. *Similarly, we put* $\mathrm{Width}(\pi) := \max_{i \in [t]} \mathrm{Width}(C_i)$ *for a refutation $\pi = (C_1, \dots, C_t)$.*

The depth $\mathrm{Depth}(\pi)$ *of a refutation π is the length of a longest path in the underlying refutation DAG G_π.*

In the following, we will consider the one-sided version of depth, called *positive depth*, that was recently introduced in [31].

▶ **Definition 3.** *If C and R are clauses with $C \setminus R = \{\ell\}$, we say that the literal ℓ is* introduced *from R to C. The* positive depth of a clause *C in a resolution refutation π, denoted* $\mathrm{PosDepth}(C)$*, is the minimal number of negative literals introduced (while also counting re-introductions) along any (inverse) path in G_π from the empty clause to C. The* positive depth of a refutation *π is defined by $\mathrm{PosDepth}(\pi) := \max_{C \in \pi} \mathrm{PosDepth}(C)$.*

We will also refer to the clause space measure for resolution. Intuitively, the clause space of a refutation π, $\mathrm{CS}(\pi)$, is the maximum number of clauses that need to be kept in memory simultaneously when verifying the proof π. A more formal definition can be found in [14].

2.1.1 Narrow Resolution and Narrow Width

The standard definition of width is not well suited for proving size lower bounds of formulas having large width themselves (cf. [6]), like the isomorphism formulas (cf. Section 2.2). A more natural way to deal with the width concept in such formulas was introduced by Galesi and Thapen in [17] together with the concept of narrow resolution that does not take into account the width of the axioms.

▶ **Definition 4.** *A* narrow resolution derivation *of a clause D from a CNF formula F is an ordered sequence of clauses $\pi = (C_1, \ldots, C_t)$ such that $C_t = D$, and for each $i \in [t]$, the clause C_i is obtained by rule (1), (2), or (3) of a (normal) resolution derivation (Definition 1) or by the following* distinction by cases *step:*

(4) *If $(B \vee x_1 \vee \cdots \vee x_m) \in F$, and if there are clauses $C_{j_1} = (A_1 \vee \overline{x_1}), \ldots, C_{j_m} = (A_m \vee \overline{x_m})$ with $j_1 < \cdots < j_m < i$, then we can derive $C_i := (B \vee A_1 \vee \cdots \vee A_m)$ in one step.*

We write $\mathrm{N\text{-}Width}(\pi) \leq k$ if π is a narrow resolution derivation and $\mathrm{Width}(C_i) \leq k$ for all $i \in [t]$ with $C_i \notin F$.

The definition here is a slight generalization of the original one in [17] since, in rule (4), we do not require all the A_j clauses to coincide, and we allow for a subclause B to be present in the axiom clause (note, however, that the width of each A_j and B will be counted). This modification also allows an exact characterization of the number of pebbles needed in Immerman's game in terms of the width measure in narrow resolution, as shown in Theorem 17.

▶ **Definition 5.** *For a measure $\mathcal{C} \in \{\mathrm{Size}, \mathrm{Width}, \mathrm{Depth}, \mathrm{PosDepth}, \mathrm{CS}, \mathrm{N\text{-}Width}\}$, by taking the minimum over all refutations π of an unsatisfiable formula F, we define $\mathcal{C}(F \vdash \Box) := \min_{\pi : F \vdash \Box} \mathcal{C}(\pi)$ as the* size, width, depth, positive depth, clause space, and narrow width of refuting F in resolution*, respectively.*

2.1.2 Krishnamurthy's Symmetry Rules

Krishnamurthy [26] observed that symmetries arise naturally in proofs of combinatorial principles and suggested some rules to simplify such proofs.

▶ **Definition 6.** *Let L be a finite set of complementary literals. Then, a bijective mapping $f \colon L \to L$ is called a* renaming *if for every $\ell \in L$ we have $\overline{f(\ell)} = f(\overline{\ell})$. For a clause $C \subseteq L$ and a renaming f, we set $f(C) := \{f(\ell) \mid \ell \in C\}$. For a formula F with $\bigcup_{C \in F} C \subseteq L$ we put $f(F) := \{f(C) \mid C \in F\}$.*

▶ **Definition 7** (The symmetry rules, [26, 39]). *Let F be a CNF formula and C a clause that can be derived by a proof $\pi : F' \vdash C$ from a subformula $F' \subseteq F$. If there exists a renaming $f\colon \mathrm{Lits}(F) \to \mathrm{Lits}(F)$ with $f(F') \subseteq F$, then the* local symmetry rule with complementation *allows the derivation of $f(C)$ from C in one step in the extended proof system. If we have the additional restriction $F' = F$, we speak of the* global symmetry rule with complementation*. Adding the global or local rule, respectively, to the proof system resolution (i. e., we consider proofs in which each clause is inferred by resolution from two clauses listed earlier in the proof, or by the respective symmetry rule from one clause earlier in the proof) yields the proof systems* SRC-1 *and* SRC-2*.*

Allowing also to use so-called *dynamic symmetries*, i. e., symmetries in the clauses already resolved, and not restricting ourselves to symmetries in the original axioms, one can define the proof system *SRC-3*. We refer to [35].

2.2 Graph Isomorphism and GI Formulas

An *(undirected) graph* is a tuple $G = \big(V_G, E_G\big)$, where V_G is a finite set of *vertices* and $E_G \subseteq \binom{V_G}{2}$ is the set of *edges*. A *colored graph* (G, λ) is a graph G together with a function $\lambda\colon V \to \mathcal{C}$, called *coloring*, where $\mathcal{C}$ is some set of *colors*. We treat every uncolored graph as a monochromatic graph.

▶ **Definition 8.** *Two colored graphs (G, λ) and (H, μ) are* isomorphic*, denoted by $(G, \lambda) \cong (H, \mu)$, if there is a color- and edge-respecting bijection $\varphi\colon V(G) \to V(H)$, called* (color-preserving) isomorphism *from G to H, i. e., $\{u, v\} \in E_G \iff \big\{\varphi(u), \varphi(v)\big\} \in E_H$ and $\lambda(v) = \mu\big(\varphi(v)\big)$ holds for all $u, v \in V_G$. An* automorphism *of a colored graph (G, λ) is an isomorphism from (G, λ) to (G, λ). We denote by $\mathrm{Iso}(G, H)$ the set of isomorphisms between G and H and by $\mathrm{Aut}(G)$ the set of automorphisms of G.*

Every coloring $\lambda\colon V_G \to \mathcal{C}$ of a graph G induces a partition of V_G: for a color $c \in \mathrm{Im}(\lambda)$, we call $\lambda^{-1}(c) \subseteq V(G)$ a *color class* of G. The *color class size* of G is the cardinality of its largest color class. It is known that the GraphIso problem can be solved in polynomial time when the color classes have constant size [16].

We encode instances of the GraphIso problem as Boolean formulas. As explained below, the formulas used here are a slight modification of those in [36]. Throughout the paper, we will consider only isomorphism formulas corresponding to pairs of graphs having the same number of vertices.

▶ **Definition 9.** *Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two graphs with $V_G = \{v_1, \dots, v_n\}$ and $V_H = \{w_1, \dots, w_n\}$. The formula $\mathrm{ISO}(G, H)$ is defined by the following clauses:*

- ***Type 1 clauses:*** *for every $i \in [n]$ the clause $(x_{i,1} \vee x_{i,2} \vee \cdots \vee x_{i,n})$ indicating that vertex $v_i \in V_G$ is mapped to some vertex in V_H; and for every $j \in [n]$ the clause $(x_{1,j} \vee x_{2,j} \vee \cdots \vee x_{n,j})$ indicating that vertex $w_j \in V_H$ is the image of some vertex in V_G.*
- ***Type 2 clauses:*** *for every $i, j, k \in [n]$ with $i \neq j$ the clause $(\overline{x_{i,k}} \vee \overline{x_{j,k}})$ indicating that not two different vertices are mapped to the same one; and for every $i, j, k \in [n]$ with $j \neq k$ the clause $(\overline{x_{i,j}} \vee \overline{x_{i,k}})$ indicating that the variables encode a function.*
- ***Type 3 clauses:*** *for every $i, j, k, \ell \in [n]$ with $i < j$ and $k \neq \ell$ with $\{v_i, v_j\} \in E_G \Leftrightarrow \{v_k, v_\ell\} \notin E_H$, the clause $(\overline{x_{i,k}} \vee \overline{x_{j,\ell}})$ expressing the adjacency relation (an edge cannot be mapped to a non-edge and vice-versa).*

The formula $\mathrm{ISO}(G, H)$ has n^2 variables and $\mathrm{O}(n^4)$ clauses. The clauses of Type 2 and Type 3 have width 2, while the clauses of Type 1 have width n.

Clearly, these formulas are satisfiable if the corresponding graphs are isomorphic. In the original definition of the $\mathrm{ISO}(G,H)$ formulas [36], the second possibility of Type 1 and Type 2 clauses was not considered. The formulas with and without these clauses are equivalent under satisfiability. We include these clauses here in order to obtain an exact characterization of Immerman's pebble game. Including these clauses can only make the lower bounds for the resolution of these formulas for non-isomorphic graphs stronger. The situation is similar to that for other principles, like the Pigeon-Hole-Principle, where the formulas with the additional Type 1 and Type 2 clauses are called onto-functional-PHP formulas (see, e. g., [32]). We remark that PHP_n^{n+1} has exponential-size resolution proofs [20], but as noticed in [26, 39], polynomial-size proofs in SRC-1.

An advantage of the isomorphism formulas is that one can express colorings of the involved graphs G and H as partial assignments of the variables:

▶ **Definition 10.** *Let G, H be as in Definition 9 and let $\lambda\colon V_G \to \mathcal{C}$ and $\mu\colon V_H \to \mathcal{C}$ be two graph colorings. Set $\rho := \{x_{i,j} = 0 \mid i,j \in [n] \text{ with } \lambda(i) \neq \mu(j)\}$. Define the ISO-formula for the colored graphs as $\mathrm{ISO}_{\lambda,\mu}(G,H) := \mathrm{ISO}(G,H)|_\rho$.*

Observe that while every coloring can be represented by a restriction, a restriction is just a partial assignment and it does not always encode a coloring. A coloring can drastically reduce the number of variables in the isomorphism formula. We will later make use of this fact. It is not hard to see that we have $\mathrm{ISO}_{\lambda,\mu}(G,H) \in \mathrm{UNSAT} \iff (G,\lambda) \not\cong (H,\mu)$.

▶ Remark 11. Since every pair of colorings (λ,μ) of a pair of graphs (G,H) can be encoded as a restriction ρ of the formula $\mathrm{ISO}(G,H)$ as explained, a lower bound on the size of a resolution refutation of the $\mathrm{ISO}_{\lambda,\mu}$-formula for colored graphs also holds for the ISO-formula of the corresponding monochromatic graphs.

It is illustrative to contrast the $\mathrm{ISO}_{\lambda,\mu}$-formulas with the ListIso problem which asks, given two graphs G and H, where each vertex $v \in V_G$ is equipped with a list $\mathfrak{L}(v) \subseteq V_H$, if there exists an isomorphism $\varphi\colon V_G \to V_H$ such that $\varphi(v) \in \mathfrak{L}(v)$ for all $v \in V_G$. This problem can also be easily expressed as a satisfiability problem by restricting the first kind of Type 1 clauses to contain only the possibilities for each vertex (and doing analogously with the second kind of Type 1 clauses). However, this restriction would not encode a graph coloring in general. Moreover, ListIso might be harder than GraphIso as it was shown in [27] (see also [25]) that this problem is NP-complete.

2.3 Immerman's Pebble Game

▶ **Definition 12** ([21, 22])**.** *For a given language $\mathcal{L}$ (of first-order logic sentences), we say that two graphs G and H are* $\mathcal{L}$-equivalent*, denoted by $G \equiv_{\mathcal{L}} H$ if for all sentences $\psi \in \mathcal{L}$ it holds that $G \vDash \psi \iff H \vDash \psi$.*

▶ **Definition 13** (k-variable fragment of first-order logic)**.** *The* k-variable fragment of first-order logic *$\mathcal{L}_k$ is the set of first-order logic formulas that use at most k different variables (possibly re-quantifying them). Furthermore, $\mathcal{L}_{k,m}$ is the subclass of $\mathcal{L}_k$ where the quantifier depth in the formulas is restricted to m.*

By allowing counting quantifiers, we can extend $\mathcal{L}_k$ to the more expressive fragment $\mathcal{C}_k$. For a graph G, we say that it has *Weisfeiler–Leman dimension* at most k if and only if $G \not\equiv_{\mathcal{C}_{k+1}} H$ for all graphs H non-isomorphic to G.

We next describe a pebble game that is equivalent to testing $\mathcal{L}_{k,m}$-equivalence (or $\mathcal{L}_k$-equivalence for the unrestricted game) and is a variant of an Ehrenfeucht-Fraïssé game [15, 12]. We borrow the notation from [23].

▶ **Definition 14** (Immerman's pebble game, [21]). *Let $m, k \in \mathbb{N}$. For graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$ with an equal number of vertices, we define the m-move k-pebble game of Immerman as follows: The game is played by two players called Player I and Player II on the graphs G and H with k pairs of pebbles. The game proceeds in rounds, each of which is associated with a position consisting of pebble placements. The position after move $r \in [m]$ of the game is denotes by $(\vec{v}_r, \vec{w}_r) \in V_G^\ell \times V_H^\ell$ with $0 \le \ell \le k$. The initial position is the pair $\big((), ()\big)$ of empty tuples. We now describe a round of the game. Suppose the current position of the game is $(\vec{v}_r, \vec{w}_r) = \big((v_1, \ldots, v_\ell), (w_1, \ldots, w_\ell)\big)$.*

- *First, Player I chooses whether he wants to remove a pebble pair (only possible if $\ell > 0$) or to place a new pair of pebbles (only possible if $\ell < k$).*
  - *If he wants to remove a pair of pebbles, he chooses some $i \in [\ell]$ and the position of the game changes to $\big((v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_\ell), (w_1, \ldots, w_{i-1}, w_{i+1}, \ldots, w_\ell)\big)$ and the next round begins.*
  - *Otherwise, he picks a graph $K \in \{G, H\}$ and a vertex $v \in V_K$.*
- *Player II then picks a vertex $w \in V_{\hat{K}}$, where $\hat{K} := \{G, H\} \setminus \{K\}$ is the graph not chosen by Player I. The position of the game changes to*

$$(\vec{v}_{r+1}, \vec{w}_{r+1}) := \begin{cases} \big((v_1, \ldots, v_\ell, v), (w_1, \ldots, w_\ell, w)\big) & \text{if } K = G, \\ \big((v_1, \ldots, v_\ell, w), (w_1, \ldots, w_\ell, v)\big) & \text{otherwise,} \end{cases}$$

  and the next round begins.

We say that Player II survives round r of the game if and only if $G[\vec{v}_r] \cong H[\vec{w}_r]$, i. e., the map $v_i \mapsto w_i$ (for $i \in [\ell]$) is an isomorphism of the subgraphs induced by the pebbled vertices. If any difference between the induced ordered subgraphs is exposed within at most m rounds, then we say that Player I wins the m-move game. This is precisely the case when there are $i, j \in [\ell]$ such that $v_i = v_j \not\Leftrightarrow w_i = w_j$ or $\{v_i, v_j\} \in E_G \not\Leftrightarrow \{w_i, w_j\} \in E_H$ or there is an $i \in [\ell]$ such that the colors of v_i and w_i are different.

If there is no restriction on the number of rounds m being played, Player I wins the game *if he wins some round, while* Player II survives the game *if she can survive forever.*

The interpretation of a configuration $\big((v_1, \ldots, v_\ell), (w_1, \ldots, w_\ell)\big)$ is that the i-th pebble pair is placed on the vertices v_i and w_i (for $i \in [\ell]$).

3 Connection Between Narrow Resolution Width and $\mathcal{L}_k$

Immerman's pebble game can be directly translated as a Spoiler–Duplicator type game played on the ISO(G, H) formulas. This kind of game has often been used in proof complexity arguments. The game defined here is a version of the game for the characterization of resolution width from [2] except that now Spoiler cannot choose variables but clauses, and Duplicator has to satisfy some literal in the chosen clause. Very similar games have already been defined in [13] and [17]. The only difference is that in our game, Spoiler can only choose Type 1 clauses (instead of any clause as in [13] or even variables as in [17]). For some of our proofs, we need to define the witnessing games also on restricted isomorphism formulas ISO$(G, H)|_\gamma$ for some restriction γ. In this case, we say that the *Type of an axiom $C|_\gamma$* in ISO$(G, H)|_\gamma$ (1, 2, or 3) is the same as that of the original axiom C.

▶ **Definition 15** (k-witnessing game). *For $k \in \mathbb{N}$ and a restriction γ, Spoiler and Duplicator construct in rounds a partial assignment for the formula* ISO$(G, H)|_\gamma$*. Initially, $\alpha_0 = \varepsilon$. At the beginning of round i, Spoiler chooses a subset of α_{i-1} of size at most $k-1$ and a*

Type 1 clause $C|_\gamma$ in $\mathrm{ISO}(G,H)|_\gamma$. Then, Duplicator extends the chosen subset to one literal in $C|_\gamma$ (we call the obtained assignment α_i), satisfying this clause and not falsifying any clause in $\mathrm{ISO}(G,H)|_\gamma$. If this is not possible, Duplicator loses the game.

▶ **Observation 16.** *$G \not\equiv_{\mathcal{L}_k} H$ if and only if Spoiler wins the k-witnessing game on $\mathrm{ISO}(G,H)$.*

Proof. The moves of Player I in Immerman's game, placing a pebble on a vertex $v_i \in V_G$ (or a vertex $w_j \in V_H$), correspond to Spoiler choosing a Type 1 clause of the kind $(x_{i,1} \vee \cdots \vee x_{i,n})$ (respectively one of the kind $(x_{1,j} \vee \cdots \vee x_{n,j})$). Player II's answer corresponds to the literal in these clauses satisfied by Duplicator. Since Duplicator only assigns variables with 1, only Type 2 or Type 3 clauses can be falsified. Player I wins Immerman's game when two pebbles on different vertices in one graph are answered with two pebbles on the same vertex in the other graph, corresponding to a Type 2 clause being falsified, or when the pebbles contradict the local isomorphism condition, and this corresponds to a Type 3 clause being falsified in the witnessing game. ◀

Using this game, we can show an equivalence between the number of variables needed to distinguish two graphs and the width measure in narrow resolution. We also notice that the number of rounds in both games matches. Since our witnessing game is a restriction of the game in [17], the proof of the result in one direction follows similar arguments as in the result for general formulas from the mentioned paper, but the bound we obtain is slightly better.

▶ **Theorem 17.** *For $k \in \mathbb{N}$, $G \not\equiv_{\mathcal{L}_{k,m}} H$ if and only if there is a narrow width resolution refutation π of $\mathrm{ISO}(G,H)$ with $\text{N-Width}(\pi) \leq k-1$ and $\mathrm{PosDepth}(\pi) \leq m$ simultaneously.*

Proof. For the direction from left to right, suppose $G \not\equiv_{\mathcal{L}_{k,m}} H$. By Observation 16, there is a winning strategy for Spoiler in the k-witnessing game on $\mathrm{ISO}(G,H)$ in m moves. This strategy has to be able to decide for each reachable partial assignment α in the game what variables can be deleted from the assignment, and what Type 1 clause C to query next. Such a strategy can be represented as a graph whose vertices store the information (α, C) with $|\alpha| \leq k-1$. From such a vertex and for every literal $\ell \in C$, there is a directed edge pointing to the vertex (α'_ℓ, C_ℓ). Here, α'_ℓ is the assignment obtained from α by setting $\ell = 1$ and maybe deleting some values (according to the strategy of Spoiler after knowing the answer of Duplicator for C). Furthermore, C_ℓ is the Type 1 clause queried next or a clause falsified by α'_ℓ. In this last case, (α'_ℓ, C_ℓ) is a winning position for Spoiler and a sink in the strategy graph. The only source of the graph is the initial vertex (α_0, C_0), where $\alpha_0 = \varepsilon$ and C_0 is the first Type 1 clause queried by Spoiler. Observe that since we have supposed that Spoiler has a winning strategy, this graph is acyclic. It is not necessarily a tree.

We can construct a resolution refutation DAG of $\mathrm{ISO}(G,H)$ by following the strategy backwards, i. e., by inverting the strategy graph. For this, we associate with each vertex (α, C) the clause C_α, defined as the set of literals falsified by α. With an inductive argument, starting at the sinks, we show that C_α can be resolved by narrow resolution from the clauses associated with the successor vertices of (α, C). For the sink vertices (α, C), by the way the strategy graph and the witness game are defined, C is an axiom of width 2 falsified by α. Since C is an axiom, it does not count for the narrow width. Using weakening, we can identify C_α with this vertex. For an interior vertex (α, C) with $C = (\ell_1 \vee \cdots \vee \ell_n)$ and with successor vertices $(\beta_1, C_1), \ldots, (\beta_n, C_n)$, we can suppose by induction that there are clauses $C_{\beta_1}, \ldots C_{\beta_n}$ associated with the successor vertices. Each assignment β_i has the form $\beta_i = \alpha_i \cup \{\ell_i = 1\}$ with $\alpha_i \subseteq \alpha$ and $|\beta_i| \leq k-1$. Because of this, C and each C_{β_i} have exactly the pair of complementary literals $(\ell_i, \overline{\ell_i})$ and can be resolved. Using a narrow resolution step, we can resolve all these clauses with C in one step, obtaining a clause $C_{\alpha'}$ with $\alpha' \subseteq \alpha$, and with weakening, we obtain C_α.

Since the clause mapped to the source vertex has to be falsified by the empty assignment, this is the empty clause, and the process defines a correct narrow resolution of $\mathrm{ISO}(G, H)$. Notice that all the clauses in the refutation have width at most $k - 1$.

The depth of the strategy graph for Spoiler in the k-witnessing game is the maximum number of rounds m needed for Spoiler to defeat Duplicator in Immerman's $\mathcal{L}_k$-game. Following a path from the empty clause towards a clause C_α being derived by a narrow resolution step from $(\ell_1 \vee \cdots \vee \ell_n)$ and $C_{\beta_1}, \ldots, C_{\beta_n}$, one can notice that this step increases the positive depth measure by one when continuing the path towards the clauses $C_{\beta_1}, \ldots, C_{\beta_n}$ (the measure stays the same when continuing towards the axiom $(\ell_1 \vee \cdots \vee \ell_n)$). The positive depth measure also increases by at most one in any ordinary resolution step. Any weakening step does not increase the positive depth. By the correspondence between the game positions (β_i, C_i) and the clauses C_{β_i} of the proof π constructed above, this shows that we have $\text{N-Width}(\pi) \leq k - 1$ and $\text{PosDepth}(\pi) \leq m$ simultaneously.

For the other direction, consider a narrow resolution refutation π for $\mathrm{ISO}(G, H)$ of width $k - 1$. We describe a strategy for Spoiler to win the k-witnessing game. Starting at the empty clause, Spoiler queries Type 1 clauses, and with the literals satisfied by Duplicator, he keeps a set S of at most k variables $x_{i,j}$ assigned with value 1 by Duplicator. For a clause $C \in \pi$ and such a set S, we say that S *contradicts* C if the following conditions happen:

1. For every negated variable $\overline{x_{i,j}}$ in C, $x_{i,j} \in S$, and
2. for every positive variable $x_{i,j}$ in C, $x_{i,j} \notin S$ and $\exists k \in [n]$ such that $(x_{i,k} \in S$ or $x_{k,j} \in S)$.

Starting at the empty clause and with the set $S = \emptyset$, S determines the predecessor clause in the refutation π where Spoiler moves to. At each step, Spoiler makes a query, updates S, and always moves to the predecessor clause contradicted by the current S. Let C be Spoiler's clause at a certain stage and S the corresponding set of variables.

If C is the (normal) resolvent of two clauses on variable $x_{i,j}$, in case one of these clauses is a Type 1 axiom, Spoiler queries it. Otherwise, Spoiler queries any of the two Type 1 clauses in $\mathrm{ISO}(G, H)$ containing $x_{i,j}$. If Duplicator assigns value 1 to this variable, Spoiler moves to the parent clause in which this variable is negated and adds $x_{i,j}$ to S. If some other variable is given value 1 by Duplicator, Spoiler adds it to S and moves to the contradicted parent clause. In both cases, Spoiler deletes from S all the variables that are not needed for contradicting the new clause.

If C is the result of a narrow resolution step involving a Type 1 axiom D, Spoiler queries this clause. Duplicator's answer must satisfy some variable $x_{i,j} \in D$. The set S together with this variable contradicts a predecessor clause C', and this clause cannot be D unless some Type 2 axiom is falsified (see the claim below). Spoiler moves to C', and he then deletes from S all the variables that are not necessary in S for contradicting the new clause. This means keeping one variable for each negated literal in C' and at most one variable for each positive literal in C'. Because the clauses in π have narrow width at most $k - 1$, Spoiler needs to keep at most k variables in S at any moment.

If C comes from a weakening step, Spoiler just needs to forget some of the variables in S.

After each new variable set by Duplicator, if some Type 2 or Type 3 axiom of $\mathrm{ISO}(G, H)$ is falsified, Spoiler wins the game. We claim that if, at some point, S contradicts some Type 1 axiom, then S falsifies some Type 2 axiom. Suppose that S contradicts the Type 1 clause $(x_{i,1} \vee \cdots \vee x_{i,n})$. By definition, this means that $x_{i,1}, \ldots, x_{i,n} \notin S$, and thus, again, by definition, there is a set of n indices $\{k_1, \ldots, k_n\} \subseteq [n]$ such that $x_{k_1,1}, \ldots, x_{k_n,n} \in S$. In case that $\{k_1, \ldots, k_n\} = [n]$, there exists a $j \in [n]$ with $k_j = i$. Thus, $x_{k_j,j} = x_{i,j} \in S$. But then S does not contradict the clause $(x_{i,1} \vee \cdots \vee x_{i,n})$, a contradiction. In case not all k_j's are different, there are $j, j' \in [n]$ such that $j \neq j'$ but still $k_j = k_{j'}$. Since $x_{k_j,j} \in S$

as well as $x_{k_{j'},j'} = x_{k_j,j'} \in S$, the functionality axiom $(\overline{x_{k_j,j}} \vee \overline{x_{k_j,j'}})$ is falsified by S. The case in which S contradicts a Type 1 clause of the form $(x_{1,i} \vee \cdots \vee x_{n,i})$ can be treated symmetrically.

Eventually, some axiom is reached. This axiom is contradicted by the current S. If it is a Type 2 or 3 axiom, S falsifies it (these axioms have only negated literals), and Spoiler wins. As observed, if this is a Type 1 axiom, then some Type 2 axiom is falsified, and Spoiler wins.

In the described construction of a winning strategy, Spoiler always moves to the contradicted predecessor of the clause he is currently standing on. Such a move increases the positive depth of his position. Thus he needs at most m moves to win the Immerman game, where m is the positive depth of the refutation. ◀

Not surprisingly, the result above holds also for colored graphs, that is, the number of pebbles and rounds in Immerman's game on colored graphs correspond exactly to narrow width and positive depth in resolution of the isomorphism formula under the restriction encoding the coloring. We need, in fact, a version of the result for general restrictions, not only for colorings, and therefore we have to make use of the witnessing game, which is also well defined for restrictions. The proof follows the same steps as that for the result above. We state the part of the result that we will need for our results.

▶ **Observation 18.** *For $k \in \mathbb{N}$, and for every restriction γ, Spoiler has a winning strategy for the k-witnessing game on $\mathrm{ISO}(G,H)|_\gamma$ if and only if $\text{N-Width}\big(\mathrm{ISO}(G,H)|_\gamma \vdash \Box\big) \leq k-1$.*

The equivalence between the number of variables for graph differentiation and narrow width allows us to give upper and lower bounds for the size of resolution proofs for isomorphism formulas.

▶ **Theorem 19.** *Let $k \in \mathbb{N}$, and G and H be two graphs with n vertices each. If $G \not\equiv_{\mathcal{L}_k} H$, then there is a (normal) resolution refutation of $\mathrm{ISO}(G,H)$ of size $n^{\mathrm{O}(k)}$.*

Proof. By the above result, if $G \not\equiv_{\mathcal{L}_k} H$, then the narrow resolution width of $\mathrm{ISO}(G,H)$ is at most $k-1$. Since there are n^2 variables in this formula, there are at most $\sum_{i=0}^{k-1}\binom{n^2}{i}2^i \leq 2^{k-1}\left(\frac{en^2}{k-1}\right)^{k-1}$ clauses that can appear in a $(k-1)$-narrow resolution refutation of the formula. But a narrow resolution refutation is just like a normal one in which the distinction by cases is made in just one step. This can be simulated by at most n steps (with at most $n-1$ intermediate clauses that might be wider than k) in normal resolution. Using an upper bound for the partial sum of binomial coefficients, the total number of different clauses in the refutation is thus bounded by $n^{\mathrm{O}(k)}$, and it is polynomial for constant k. ◀

Observe that this result suggests a way to automatically generate short proofs for (non)-isomorphism formulas, following the same ideas as those in the algorithm proposed in [6] and [17] for general formulas. The algorithm would generate in stages all clauses that can be derived by narrow resolution of width $1, 2, 3, \ldots$, until the empty clause is derived. By the above result, the running time of this algorithm is $n^{\mathrm{O}(k)}$.

Lower bounds for narrow width also imply lower bounds on the size of a resolution refutation for $\mathrm{ISO}(G,H)$, in the same way that width lower bounds imply size lower bounds in normal resolution, as shown by Ben-Sasson and Wigderson [6]. For this, we follow the same steps as in the mentioned paper, adapted to narrow width. The general fact that narrow width provides lower bounds for resolution size has also been proved in [17]. By concentrating on the isomorphism formulas, we obtain tighter results. The next lemma is the basis for our lower bounds. It is a version in our context of [6, Lemma 3.2] or [17, Lemma 6].

▶ **Lemma 20.** *Let γ be a restriction and let ℓ be any literal in* $\mathrm{ISO}(G,H)|_\gamma$*. If Spoiler has a winning strategy for the k-witnessing game on* $\mathrm{ISO}(G,H)|_{\gamma\{\ell=1\}}$ *as well as for the $(k-1)$-witnessing game on* $\mathrm{ISO}(G,H)|_{\gamma\{\ell=0\}}$*, then he wins the k-witnessing game on* $\mathrm{ISO}(G,H)|_\gamma$*.*

Proof. We distinguish two cases depending on whether literal ℓ is positive or negative:

Case 1: $\ell = x_{i,j}$. The formula $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=1\}}$ is like $\mathrm{ISO}(G,H)|_\gamma$ without the two Type 1 clauses containing literal $x_{i,j}$ and without all occurrences of the literal $\overline{x_{i,j}}$. If Spoiler selects in the game on $\mathrm{ISO}(G,H)|_\gamma$ the same sequence of Type 1 clauses as in the game on $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=1\}}$, Duplicator either loses the game or sets a literal $x_{a,b}$ to 1 for a clause $C = (\overline{x_{a,b}} \vee \overline{x_{i,j}}) \in \mathrm{ISO}(G,H)|_\gamma$. When this happens, Spoiler restricts the assignment to $\gamma\{x_{a,b}=0\}$, and then simulates the strategy for $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=0\}}$ on $\mathrm{ISO}(G,H)|_\gamma$. If Duplicator does not assign $x_{i,j}=1$, she loses the game eventually by the assumption. If she does, then the clause C is falsified, and she also loses. Spoiler needs to keep an assignment of size at most k at any moment.

Case 2: $\ell = \overline{x_{i,j}}$. In this case, Spoiler simulates the strategy for $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=0\}}$ on the formula $\mathrm{ISO}(G,H)|_\gamma$, either winning the game or forcing Duplicator to assign $x_{i,j}=1$ (by a Type 1 clause that contains $x_{i,j}$ and which was falsified in the $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=0\}}$-game). Restricting then the assignment to this literal, Spoiler now plays the strategy for $\mathrm{ISO}(G,H)|_{\gamma\{x_{i,j}=1\}}$ and Duplicator loses. ◀

From this result, lower bounds as in [6] follow directly. The advantage here is that the width of the axioms of $\mathrm{ISO}(G,H)$ is not subtracted from the exponent of the lower bound results, as it is done in [6, Corollary 3.4].

▶ **Theorem 21.** *Let $k \in \mathbb{N}$, and G, H be two non-isomorphic graphs with n vertices each. If $G \equiv_{\mathcal{L}_k} H$, then the size of a tree-like resolution refutation of* $\mathrm{ISO}(G,H)$ *is at least 2^k.*

Lower bounds on narrow width also imply, as noted in [17], lower bounds on general resolution size. Using (a version for narrow width) from [6, Theorem 3.5], one can show that if G and H are two non-isomorphic graphs with n vertices each with $G \equiv_{\mathcal{L}_k} H$, then the size of a resolution refutation of $\mathrm{ISO}(G,H)$ is at least $\exp\big(\Omega(k^2/n^2)\big)$. However, since the maximum number k of variables needed for distinguishing G and H is at most the number of vertices n, this only provides trivial lower bounds. A way to avoid this problem is to consider graph colorings under which the number k is still large, but the number of variables in $\mathrm{ISO}(G,H)$ is smaller. Since such a coloring can be expressed as a restriction ρ applied to $\mathrm{Vars}\big(\mathrm{ISO}(G,H)\big)$, and using the fact that for every restriction ρ, the size of a resolution refutation of $\mathrm{ISO}(G,H)$ is at least the size of the refutation of the formula under the restriction, $\mathrm{ISO}(G,H)|_\rho$, we obtain Theorem 23 below.

▶ **Definition 22.** *Let (G,λ) and (H,μ) be two colored graphs. For a vertex $v \in V_G$, we set* $\text{color-class}(v) := \mu^{-1}\big(\lambda(v)\big)$*, i. e., the set of vertices in V_H that have the same color as v.*

If (G,λ) and (H,μ) are two colored graphs in n vertices each, $m := \sum_{v\in V_G} |\text{color-class}(v)|$ is between n and n^2.

▶ **Theorem 23.** *Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two non-isomorphic graphs with n vertices each, for which there is a $k \in \mathbb{N}$ and two colorings λ, μ such that $(G,\lambda) \equiv_{\mathcal{L}_k} (H,\mu)$. Then, the size of every resolution refutation of* $\mathrm{ISO}(G,H)$ *is at least* $\exp\big(\Omega(k^2/m)\big)$*, where* $m := \sum_{v\in V_G} |\text{color-class}(v)|$ *is the sum of the sizes of the color classes.*

Proof Sketch. Let $\rho := \{x_{i,j} = 0 \mid i,j \in [n] \text{ with } \lambda(i) \neq \mu(j)\}$, and consider the unsatisfiable formula $\mathrm{ISO}(G,H)|_\rho$. The set of variables of this formula is $\{x_{i,j} \mid i,j \in [n] \text{ with } \lambda(i) = \mu(j)\}$ and contains exactly $m = \sum_{v\in V_G} |\text{color-class}(v)|$ variables. Since $(G,\lambda) \equiv_{\mathcal{L}_k} (H,\mu)$, by Observation 18, $\text{N-Width}\big(\mathrm{ISO}(G,H)|_\rho \vdash \square\big) \geq k$. We following the same steps of that of [6, Theorem 3.5], with the modifications needed to deal with restrictions as done in Theorem 21.

For simplicity, let us denote $\mathrm{ISO}(G,H)|_\rho$ by F and let π be a (normal) resolution refutation of minimal size s of F. We define d and a to be $d := \lceil\sqrt{2m\ln s}\rceil$ and $a := (1-\frac{d}{2m})^{-1}$. A clause in π is called *fat* if it contains more than d literals. Let π^* be the set of fat clauses in π. We prove by induction on m that $\text{N-Width}(F \vdash \square) \leq d + \log_a(|\pi^*|)$. The result follows from this implication since $|\pi^*| \leq s$ and therefore by the way a and d are defined, $\log_a(|\pi^*|)$ is bounded by $c\sqrt{2m\ln s}$ for some constant c. The base case $m = 0$ holds trivially. For the induction case, observe that F contains at most $2m$ literals and therefore one literal ℓ appears in at least $\frac{d}{2m}|\pi^*|$ fat clauses. We consider the two refutations of the formulas $F|_{\ell=1}$ and $F|_{\ell=0}$ obtained from π by setting ℓ to 1 and to 0, respectively. Setting $\ell = 1$ removes all the clauses including literal ℓ and leaves a refutation of $F|_{\ell=1}$ with at most $(1-\frac{d}{2m})|\pi^*| = a^{-1}|\pi^*|$ fat clauses. By induction hypothesis we have $\text{N-Width}(F|_{\ell=1} \vdash \square) \leq d + \log_a(a^{-1}|\pi^*|) = d + \log_a(|\pi^*|) - 1$. Setting $\ell = 0$ produces a refutation of the formula $F|_{\ell=0}$ with less than m variables, and again by induction on m it holds $\text{N-Width}(F|_{\ell=0} \vdash \square) \leq d + \log_a(|\pi^*|)$. By applying Lemma 20 we obtain:

$$\text{N-Width}\big(F \vdash \square\big) \leq d + \log_a(|\pi^*|) \in \mathrm{O}\left(\sqrt{m \cdot \ln\Big(\mathrm{Size}\big(F \vdash \square\big)\Big)}\right).$$

Observe that since we are dealing with narrow resolution, we do not need the width of the axioms in $\mathrm{ISO}(G,H)|_\rho$ as an additional term, as in the result from [6]. It follows that $\mathrm{Size}\big(\mathrm{ISO}(G,H)|_\rho \vdash \square\big) = \exp\big(\Omega(k^2/m)\big)$. The last fact needed is that for every restriction ρ, $\mathrm{Size}\big(\mathrm{ISO}(G,H) \vdash \square\big) \geq \mathrm{Size}\big(\mathrm{ISO}(G,H)|_\rho \vdash \square\big)$. ◀

This result can then be automatically applied to graphs in which the maximum size of a color class is small.

▶ **Corollary 24.** *Let G and H be two graphs with n vertices each, and let $k \in \mathbb{N}$ and λ, μ be colorings with constant size color classes such that $(G,\lambda) \equiv_{\mathcal{L}_k} (H,\mu)$. Then, any resolution refutation of* $\mathrm{ISO}(G,H)$ *has size at least* $\exp\big(\Omega(k^2/n)\big)$.

Such constant size color classes are the case for the CFI graphs [8, 36] and the variant of the multipede graphs from [11]. In both examples, the maximum size of a color class is 4, while the number of variables needed to distinguish the graphs is linear in n. Thus, for both examples, the above result gives a resolution size lower bound of $\exp\big(\Omega(n)\big)$. One can also imagine this result being useful for proving resolution size lower bounds in cases in which not all color classes of the graphs have constant size, but the sum of the class sizes is still smaller than the number of variables needed to distinguish the graphs.

4 An Exponential Lower Bound for the Size of SRC-1 proofs for Graph (Non)Isomorphism

In this section, we show that there is a family of non-isomorphic graph pairs (G_n, H_n) that has only exponentially-long proofs of $\mathrm{ISO}(G_n, H_n)$ in the SRC-1 system. Exponential size lower bounds in SRC-1 are known [39], but not for graph isomorphism formulas. Our result is proven by observing that the global symmetry rule cannot be applied to formulas corresponding to graphs having only trivial automorphisms and restricting ourselves to such graphs.

▶ **Definition 25.** *A colored graph* (G, λ) *is called* asymmetric *if* $\mathrm{Aut}(G) = \{\mathrm{id}\}$.

To characterize the possible symmetries in an isomorphism formula, we need the notions of graph anti-automorphism and anti-isomorphism.

▶ **Definition 26.** *Let* $G = (V_G, E_G)$ *and* $H = (V_H, E_H)$ *be two graphs. An* anti-isomorphism σ *from* G *to* H *is a bijection between the vertices of* G *and* H *exchanging edges and non-edges, i. e., for all* $u, v \in V_G$*:* $\{u, v\} \in E_G \iff \{\sigma(u), \sigma(v)\} \notin E_H$*. An* anti-automorphism *of a graph* G *is an anti-isomorphism from* G *to* G*. We denote by* $\text{A-Iso}(G, H)$ *the set of anti-isomorphisms between* G *and* H *and by* $\text{A-Aut}(G)$ *the set of anti-automorphisms of* G*.*

We will also need the following simple observation.

▶ **Observation 27.** *Asymmetric graphs do not have any anti-automorphisms.*

Szeider observed in [35, Lemma 10] that if a formula is asymmetric, then the size of a resolution refutation and of an SRC-1 refutation of the formula are equal. The next lemma shows that if two graphs are asymmetric, then the corresponding ISO formula is asymmetric.

▶ **Lemma 28.** *Let* G *and* H *be two graphs with* $|V_G| = |V_H| =: n \geq 3$*, and let* $F := \mathrm{ISO}(G, H)$*. Further, let* $f \colon \mathrm{Lits}(F) \to \mathrm{Lits}(F)$ *be a renaming of the literals in* F*. Then* $f(F) \subseteq F$ *if and only if one of the following two cases hold:*

1. *There are two permutations* $\sigma, \gamma \in S_n$ *such that for every* $(i, j) \in [n] \times [n], f(x_{i,j}) = x_{\sigma(i),\gamma(j)}$ *and* $(\sigma, \gamma) \in \mathrm{Aut}(G) \times \mathrm{Aut}(H)$ *or* $(\sigma, \gamma) \in \text{A-Aut}(G) \times \text{A-Aut}(H)$*; or*
2. *there are two permutations* $\sigma, \gamma \in S_n$ *such that for every* $(i, j) \in [n] \times [n], f(x_{i,j}) = x_{\gamma(j),\sigma(i)}$ *and* $(\sigma, \gamma^{-1}) \in \mathrm{Iso}(G, H) \times \mathrm{Iso}(G, H)$ *or* $(\sigma, \gamma^{-1}) \in \text{A-Iso}(G, H) \times \text{A-Iso}(G, H)$*.*

Notice that if the graphs G and H are non-isomorphic and $f(F) \subseteq F$, then we can only be dealing with Case 1 in the lemma. Moreover, by Observation 27, if the graphs G and H do not have any non-trivial automorphisms, they cannot have anti-automorphisms either. In this case, a renaming f with $f(F) \subseteq F$ cannot exist, and therefore the global symmetry rule cannot be applied. This implies that size lower bounds for the resolution of (non)isomorphism formulas for asymmetric graphs coincide with their size lower bounds for the system SRC-1.

The Cai–Fürer–Immerman construction [8] gave graphs with a large Weisfeiler–Leman dimension, more precisely with a linear lower bound on the WL-dimension. A related construction of graphs satisfying this property, known as *multipedes*, was given in [19]. However, the resulting graphs are very large in terms of the WL-dimension. Neuen and Schweitzer improved in [29] the multipede construction combining it with size reduction techniques. Using a different construction, Dawar and Khan [11] showed how to obtain graphs whose Weisfeiler–Leman dimension is linear in the number of their vertices (as with the CFI graphs) and without any non-trivial automorphisms.

▶ **Theorem 29** ([11])**.** *For* $k \in \mathbb{N}$*, there is (a random process that produces with high probability) a family of asymmetric pairs of non-isomorphic graphs* (G_k, H_k) *with* $\mathrm{O}(k)$ *vertices, color classes of size 4, and Weisfeiler–Leman dimension* k*.*

In [11], it was furthermore demonstrated by conducting experiments that the resulting graphs provide hard examples for graph isomorphism solvers, matching the hardest-known benchmarks for graph isomorphism. The following result can be seen as a theoretical insight into this phenomenon.

Corollary 24 implies that the isomorphism formulas for the pairs (G_k, H_k) of non-isomorphic graphs from the above-mentioned construction have resolution refutations of size $\exp\bigl(\Omega(n)\bigr)$, where n is the number of vertices in the graphs (linear in the WL-dimension k). Since these graphs are asymmetric, from Lemma 28, we conclude:

▶ **Theorem 30.** *There is a (non-constructive) family of non-isomorphic graph pairs* (G_n, H_n) *with* $\mathrm{O}(n)$ *vertices each, such that any refutation of* $\mathrm{ISO}(G_n, H_n)$ *requires size* $\exp\big(\Omega(n)\big)$ *in the SRC-1 proof system.*

5 Lower Bounds on Clause Space for Proving Non-Isomorphism

Atserias and Dalmau [2] gave a combinatorial characterization of resolution width and used it to show the relation $\mathrm{CS}(F \vdash \Box) \geq \mathrm{Width}(F \vdash \Box) - \mathrm{Width}(F) + 1$ for any $F \in \mathrm{UNSAT}$. We will show in this section that this also holds for narrow width, with the advantage that, again, in this case, we do not have to worry about the width of the axioms. From this result, we obtain clause space lower bounds for the (normal) resolution of isomorphism formulas.

▶ **Definition 31** (w-NW Family)**.** *Given an unsatisfiable CNF formula* F *and a natural number* $w \in \mathbb{N}$*, we say that a family of assignments* $\mathcal{F}$ *for* F *is a* w-NW family *if all of the following properties hold:*

(1) $\mathcal{F} \neq \varnothing$,
(2) $\forall \alpha \in \mathcal{F}$ *and* $\forall C \in F$: $C|_\alpha \neq \Box$,
(3) $\forall \alpha \in \mathcal{F}$: $|\mathrm{Dom}(\alpha)| \leq w$,
(4) $\forall \alpha \in \mathcal{F}$ *and* $\forall \beta \subseteq \alpha$: $\beta \in \mathcal{F}$,
(5) $\forall \alpha \in \mathcal{F}$ *with* $\mathrm{Dom}(\alpha) \leq w - 1$ *and* $\forall C \in F|_\alpha$: $\exists \ell \in C$ *such that* $\alpha\{\ell = 1\} \in \mathcal{F}$.

▶ **Theorem 32.** *If* F *is an unsatisfiable CNF formula with* $\mathrm{N\text{-}Width}(F \vdash \Box) > w$*, then there exists a* $(w+1)$*-NW family for* F*.*

▶ **Theorem 33.** *If there is a* $(w+1)$*-NW family for an unsatisfiable CNF formula* F*, then* $\mathrm{CS}(F \vdash \Box) \geq w + 2$*.*

Proof Sketch. This follows from an adaptation of [2, Lemma 5], by noticing that the original constant for $\mathrm{Width}(F)$ vanishes by modifying point (5) of the definition of an Atserias–Dalmau family as we did. Playing the so-called Spoiler–Duplicator game on F, as in the proof of [2, Lemma 5], Duplicator has an answer to satisfy the queried clause in one round, making it not necessary for Spoiler to query the variables in a clause until he gets a satisfying assignment. ◀

▶ **Corollary 34.** *For every* $F \in \mathrm{UNSAT}$ *we have* $\mathrm{CS}(F \vdash \Box) \geq \mathrm{N\text{-}Width}(F \vdash \Box) + 1$*.*

Using Theorem 17, we obtain:

▶ **Theorem 35.** *Let* $k \in \mathbb{N}$ *and let* G *and* H *be two non-isomorphic graphs with* $G \equiv_{\mathcal{L}_k} H$*. Then* $\mathrm{CS}\big(\mathrm{ISO}(G, H) \vdash \Box\big) \geq k + 1$*.*

By the CFI construction [8], for every $n \in \mathbb{N}$, there is a pair of non-isomorphic graphs (G_n, H_n) such that G_n and H_n have $\mathrm{O}(n)$ vertices but $G_n \equiv_{\mathcal{C}_n} H_n$ (and therefore also $G_n \equiv_{\mathcal{L}_n} H_n$). Hence, for these graphs, $\mathrm{CS}\big(\mathrm{ISO}(G_n, H_n) \vdash \Box\big) \geq \big|\mathrm{Vars}\big(\mathrm{ISO}(G_n, H_n)\big)\big|^{1/2} + 1$.

6 Conclusions

We have given an exact characterization for the number of variables needed to distinguish two graphs in first-order logic in terms of the narrow resolution width needed for refuting the corresponding isomorphism formulas. This fact allowed us to obtain upper and lower bounds for the size and space of (normal) resolution refutation of such formulas. The size upper bound justifies a clause length increasing algorithm for the resolution (and solving) of isomorphism formulas of the kind proposed in [6] for general formulas.

The lower bounds techniques provide a simplified method to obtain resolution size lower bounds directly from the structure of the graphs, using the $\mathcal{L}_k$-logic, and without having to deal with the isomorphism formulas directly. All the known resolution size lower bounds for isomorphism formulas can be easily derived from this result. Moreover, we have been able to use the method to obtain exponential lower bounds for isomorphism formulas in the stronger system of SRC-1, which includes a global symmetry rule, answering a question posed in [33].

The obvious open question is to prove superpolynomial size lower bounds for isomorphism formulas in the stronger systems SRC-2 and SRC-3. However, one would need different ideas for this, since, as shown recently in [33], the families of graphs based on the CFI construction, like the ones used in all known lower bounds, have polynomial-size SRC-2 refutations.

References

1 Noriko H. Arai and Alasdair Urquhart. Local symmetries in propositional logic. In *Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, TABLEAUX 2000, St Andrews, Scotland, UK, July 3-7, 2000, Proceedings*, pages 40–51, 2000. doi:10.1007/10722086_3.

2 Albert Atserias and Víctor Dalmau. A combinatorial characterization of resolution width. *Journal of Computer and System Sciences*, 74(3):323–334, 2008. Earlier conference version in *CCC '03*. doi:10.1016/j.jcss.2007.06.025.

3 Albert Atserias and Elitza N. Maneva. Sherali–Adams relaxations and indistinguishability in counting logics. *SIAM Journal on Computing*, 42(1):112–137, 2013. Earlier conference version in *ITCS '12*. doi:10.1137/120867834.

4 László Babai. Graph isomorphism in quasipolynomial time [extended abstract]. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing (STOC '16)*, pages 684–697, 2016. Full-length version in *arXiv:1512.03547*. doi:10.1145/2897518.2897542.

5 László Babai and Shlomo Moran. Arthur–Merlin games: A randomized proof system, and a hierarchy of complexity classes. *Journal of Computer and System Sciences*, 36(2):254–276, 1988. doi:10.1016/0022-0000(88)90028-1.

6 Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001. Earlier conference versions in *STOC '99* and *CCC '99*. doi:10.1145/375827.375835.

7 Christoph Berkholz and Martin Grohe. Limitations of algebraic approaches to graph isomorphism testing. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming (ICALP '15)*, pages 155–166, 2015. doi:10.1007/978-3-662-47672-7_13.

8 Jin-yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identifications. *Combinatorica*, 12(4):389–410, 1992. Earlier conference version in *FOCS '89*. doi:10.1007/BF01305232.

9 Paolo Codenotti, Grant Schoenebeck, and Aaron Snook. Graph isomorphism and the Lasserre hierarchy. *arXiv*, 1401.0758, 2014. arXiv:1401.0758.

10 Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *The Journal of Symbolic Logic*, 44(1):36–50, 1979. doi:10.2307/2273702.

11 Anuj Dawar and Kashif Khan. Constructing hard examples for graph isomorphism. *Journal of Graph Algorithms and Applications*, 23(2):293–316, 2019. doi:10.7155/jgaa.00492.

12 Andrzej Ehrenfeucht. An application of games to the completeness problem for formalized theories. *Fundamenta Mathematicae*, 49:129–141, 1961. doi:10.4064/fm-49-2-129-141.

13 Juan Luis Esteban, Nicola Galesi, and Jochen Messner. On the complexity of resolution with bounded conjunctions. *Theoretical Computer Science*, 321(2-3):347–370, 2004. Earlier conference version in *ICALP '02*. doi:10.1016/j.tcs.2004.04.004.

14 Juan Luis Esteban and Jacobo Torán. Space bounds for resolution. *Information and Computation*, 171(1):84–97, 2001. Preliminary versions in *STACS '99* and *CSL '99*. doi:10.1006/inco.2001.2921.

15 Roland Fraïssé. Sur une nouvelle classification des systèmes de relations. *Comptes Rendus*, 230:1022–1024, 1950.

16 Merrick L. Furst, John E. Hopcroft, and Eugene M. Luks. Polynomial-time algorithms for permutation groups. In *Proceedings of the 21st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 36–41, 1980. doi:10.1109/SFCS.1980.34.

17 Nicola Galesi and Neil Thapen. Resolution and pebbling games. In *Proceedings of the 8th International Conference on Theory and Applications of Satisfiability Testing (SAT '05)*, pages 76–90, 2005. doi:10.1007/11499107_6.

18 Martin Grohe and Martin Otto. Pebble games and linear equations. *The Journal of Symbolic Logic*, 80(3):797–844, 2015. Earlier conference version in *CSL '12*. doi:10.1017/jsl.2015.28.

19 Yuri Gurevich and Saharon Shelah. On finite rigid structures. *The Journal of Symbolic Logic*, 61(2):549–562, 1996. doi:10.2307/2275675.

20 Armin Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985. doi:10.1016/0304-3975(85)90144-6.

21 Neil Immerman. Upper and lower bounds for first order expressibility. *Journal of Computer and System Sciences*, 25(1):76–98, 1982. Earlier conference version in *FOCS '80*. doi:10.1016/0022-0000(82)90011-3.

22 Neil Immerman. *Descriptive Complexity*. Graduate texts in computer science. Springer, 1999. doi:10.1007/978-1-4612-0539-5.

23 Sandra Kiefer. *Power and limits of the Weisfeiler–Leman algorithm*. Dissertation, RWTH Aachen University, 2020. doi:10.18154/RWTH-2020-03508.

24 Sandra Kiefer. The Weisfeiler–Leman algorithm: an exploration of its power. *ACM SIGLOG News*, 7(3):5–27, 2020. doi:10.1145/3436980.3436982.

25 Pavel Klavík, Dusan Knop, and Peter Zeman. Graph isomorphism restricted by lists. *Theoretical Computer Science*, 860:51–71, 2021. Earlier conference version in *WG '20*. doi:10.1016/j.tcs.2021.01.027.

26 Balakrishnan Krishnamurthy. Short proofs for tricky formulas. *Acta Informatica*, 22(3):253–275, 1985. doi:10.1007/BF00265682.

27 Anna Lubiw. Some NP-complete problems similar to graph isomorphism. *SIAM Journal on Computing*, 10(1):11–21, 1981. doi:10.1137/0210002.

28 Peter N. Malkin. Sherali–Adams relaxations of graph isomorphism polytopes. *Discrete Optimization*, 12:73–97, 2014. doi:10.1016/j.disopt.2014.01.004.

29 Daniel Neuen and Pascal Schweitzer. An exponential lower bound for individualization-refinement algorithms for graph isomorphism. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC '18)*, pages 138–150. ACM, 2018. doi:10.1145/3188745.3188900.

30 Ryan O'Donnell, John Wright, Chenggang Wu, and Yuan Zhou. Hardness of robust graph isomorphism, lasserre gaps, and asymmetry of random graphs. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '14)*, pages 1659–1677, 2014. doi:10.1137/1.9781611973402.120.

31 Theodoros Papamakarios and Alexander Razborov. Space characterizations of complexity measures and size-space trade-offs in propositional proof systems. Technical Report TR21-074, ECCC, 2021. URL: https://eccc.weizmann.ac.il/report/2021/074/.

32 Alexander A. Razborov. Proof complexity of pigeonhole principles. In *Proceedings of the 5th International Conference on Developments in Language Theory (DLT '01), Revised Papers*, pages 100–116, 2001. doi:10.1007/3-540-46011-X_8.

33 Pascal Schweitzer and Constantin Seebach. Resolution with symmetry rule applied to linear equations. In *Proceedings of the 38th International Symposium on Theoretical Aspects of Computer Science (STACS '21)*, pages 58:1–58:16, 2021. doi:10.4230/LIPIcs.STACS.2021.58.

34 Hanif D. Sherali and Warren P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. Discret. Math.*, 3(3):411–430, 1990. doi:10.1137/0403036.

35 Stefan Szeider. The complexity of resolution with generalized symmetry rules. *Theory of Computing Systems*, 38(2):171–188, 2005. Earlier conference version in *STACS '03*. doi:10.1007/s00224-004-1192-0.

36 Jacobo Torán. On the resolution complexity of graph non-isomorphism. In *Proceedings of the 16th International Conference on Theory and Applications of Satisfiability Testing (SAT '13)*, pages 52–66, 2013. doi:10.1007/978-3-642-39071-5_6.

37 Jacobo Torán and Florian Wörz. Number of variables for graph identification and the resolution of GI formulas. Technical Report TR21-097, ECCC, 2021. URL: https://eccc.weizmann.ac.il/report/2021/097.

38 Grigori S. Tseitin. On the complexity of derivation in propositional calculus. In *Studies in Constructive Mathematics and Mathematical Logic, Part 2*, volume 8 of *Seminars in Mathematics*, pages 115–125. Consultants Bureau, 1968.

39 Alasdair Urquhart. The symmetry rule in propositional logic. *Discret. Appl. Math.*, 96-97:177–193, 1999. doi:10.1016/S0166-218X(99)00039-6.

40 Boris Weisfeiler. *On Construction and Identification of Graphs*, volume 558 of *Lecture Notes in Mathematics*. Springer, 1976. doi:10.1007/BFb0089374.

41 Boris Weisfeiler and Andrei Leman. The reduction of a graph to canonical form and the algebra which appears therein. *Nauchno-Technicheskaya Informatsia, Seriya 2*, 9, 1968. Translation from Russian into English by Grigory Ryabov available under https://www.iti.zcu.cz/wl2018/pdf/wl_paper_translation.pdf.