Going Deep and Going Wide: Counting Logic and Homomorphism Indistinguishability over Graphs of Bounded Treedepth and Treewidth

Eva Fluck
RWTH Aachen University, Germany

Tim Seppelt
RWTH Aachen University, Germany

Gian Luca Spitzer
RWTH Aachen University, Germany

Abstract

We study the expressive power of first-order logic with counting quantifiers, especially the k-variable and quantifier-rank-q fragment C_q^k, using homomorphism indistinguishability. Recently, Dawar, Jakl, and Reggio (2021) proved that two graphs satisfy the same C_q^k-sentences if and only if they are homomorphism indistinguishable over the class $\mathcal{T}_q^k$ of graphs admitting a k-pebble forest cover of depth q. Their proof builds on the categorical framework of game comonads developed by Abramsky, Dawar, and Wang (2017). We reprove their result using elementary techniques inspired by Dvořák (2010). Using these techniques we also give a characterisation of guarded counting logic. Our main focus, however, is to provide a graph theoretic analysis of the graph class $\mathcal{T}_q^k$. This allows us to separate $\mathcal{T}_q^k$ from the intersection of the graph class $\mathcal{TW}_{k-1}$, that is graphs of treewidth less or equal $k-1$, and $\mathcal{TD}_q$, that is graphs of treedepth at most q if q is sufficiently larger than k. We are able to lift this separation to the semantic separation of the respective homomorphism indistinguishability relations. A part of this separation is to prove that the class $\mathcal{TD}_q$ is homomorphism distinguishing closed, which was already conjectured by Roberson (2022).

2012 ACM Subject Classification Mathematics of computing → Graph theory; Theory of computation → Finite Model Theory

Keywords and phrases Treewidth, treedepth, homomorphism indistinguishability, counting first-order logic

Digital Object Identifier 10.4230/LIPIcs.CSL.2024.27

Related Version *Full Version*: `https://arxiv.org/abs/2308.06044` [9]

Funding *Tim Seppelt*: German Research Foundation (DFG) via RTG 2236/2 (UnRAVeL), European Union (ERC, SymSim, 101054974)
Gian Luca Spitzer: German Research Foundation (DFG) via RTG 2236/2 (UnRAVeL)

Acknowledgements We would like to thank Martin Grohe and Daniel Neuen for fruitful discussions.

1 Introduction

Since the 1980s, first-order logic with counting quantifiers C plays a decisive role in finite model theory. In this extension of first-order logic with quantifiers $\exists^{\geq t}x$ ("there exists at least t many x"), properties which can be expressed in first-order logic only with formulae of length depending on t can be expressed succinctly. Of particular interest are the k-variable and quantifier-depth-q fragments C^k and C_q of C, which enjoy rich connections to graph algorithms [8], algebraic graph theory [7, 16], optimisation [16, 27], graph neural networks [21, 30, 15], and category theory [5, 1].

The intersection of these fragments, the fragment $\mathsf{C}_q^k \coloneqq \mathsf{C}^k \cap \mathsf{C}_q$ of all C-formulae with k-variables and quantifier-depth q, has received much less attention [25]. In this work, we study the expressivity of C_q^k using homomorphism indistinguishability.

Homomorphism indistinguishability is an emerging framework for measuring the expressivity of equivalence relations comparing graphs. Two graphs G and H are *homomorphism indistinguishable* over a graph class $\mathcal{F}$ if for all $F \in \mathcal{F}$ the number of homomorphisms from F to G is equal to the number of homomorphisms from F to H. Many natural equivalence relations between graphs including isomorphism [19], quantum isomorphism [20], cospectrality [7], and feasibility of integer programming relaxations for graph isomorphism [16, 27] can be characterised as homomorphism indistinguishability relations over certain graph classes. Establishing such characterisations is intriguing since it allows to use tools from structural graph theory to study equivalence relations between graphs [26, 28]. Furthermore, the expressivity of homomorphism counts themselves is of practical interest [23, 14].

Equivalence with respect to C^k and C_q has been characterised by Dvořák [8] and Grohe [13] as homomorphism indistinguishability over the classes $\mathcal{TW}_{k-1}$ of graphs of treewidth $\leq k-1$ and $\mathcal{TD}_q$ of graphs of treedepth $\leq q$, respectively. Recently, Dawar, Jakl, and Reggio [5] proved that two graphs satisfy the same C^k_q-sentences if and only if they are homomorphism indistinguishable over the class $\mathcal{T}^k_q$ of graphs admitting a k-pebble forest cover of depth q. Their proof builds on the categorical framework of game comonads developed in [1].

As a first step, we reprove their result using elementary techniques inspired by Dvořák [8]. The general idea is to translate between sentences in C and graphs from which homomorphism are counted in an inductive fashion. By carefully imposing structural constraints, we are able to extend the original correspondence from [8] between C^k and graphs of treewidth at most $k-1$ to C_q and graphs of treedepth at most q, reproducing a result of [13], and finally to C^k_q and $\mathcal{T}^k_q$. This simple and uniform proof strategy also yields the following result on guarded counting logic GC^k_q. Guarded counting logic plays a crucial role in the theory of properties of higher arity expressible by graph neural networks [15]. Towards this goal we introduce a new graph class called $\mathcal{GT}^k_q$, which is closely related to $\mathcal{T}^k_q$.

▶ **Theorem 1.** *Let $k, q \geq 1$. Two graphs G and H are GC^k_q-equivalent if and only if they are homomorphism indistinguishable over $\mathcal{GT}^k_q$.*

The main contribution of this work, however, concerns the relationship between the graph classes $\mathcal{T}^k_q$ and the class $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ of graphs which have treewidth at most $k-1$ *and* treedepth at most q. Given the results of [8, 13], one might think that elementary equivalence with respect to sentences in $\mathsf{C}^k_q = \mathsf{C}^k \cap \mathsf{C}_q$ is characterised by homomorphism indistinguishability with respect to $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$. The central result of this paper asserts that this intuition is wrong. As a fist step towards this, we prove that the graph class $\mathcal{T}^k_q$ and $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ are distinct if q is sufficiently larger than k. All logarithms in this work are to the base 2.

▶ **Theorem 2.** *For $q \geq 3$, $\mathcal{T}^2_q \subsetneq \mathcal{TW}_1 \cap \mathcal{TD}_q$, and for $2 \leq k-1 \leq \frac{q}{3+\log q}$, $\mathcal{T}^k_q \subsetneq \mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.*

Towards Theorem 2, we give an equivalent characterisation of $\mathcal{T}^k_q$ via a monotone cops-and-robber game, which is essentially the standard game for treewidth where one additionally counts the number of rounds the cops need to capture the robber. Here, "monotone" refers to a restriction of Cops, who is only allowed to move cops that are not adjacent to the current escape-space of the Robber. Building on [10], we then prove that $\mathcal{T}^k_q$ is a proper subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$, for q sufficiently larger than k. Additionally, we provide an analysis of various notions designed to restrict both width and depth of a decomposition and show that all of them are equivalent. Adding to the original definition of $\mathcal{T}^k_q$ via k-pebble forest covers of depth q, which can be interpreted as treedepth decompositions augmented by a width measure, we introduce a way to measure the depth of tree decompositions. Finally, we define k-construction trees of elimination depth q, another equivalent notion, which relates to the machinery used by Dvořák [8].

However, the, let us say syntactical, separation of the graph classes $\mathcal{T}_q^k$ and $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ from Theorem 2 does not suffice to separate their homomorphism indistinguishability relations semantically. In fact, it could well be that all graphs which are homomorphism indistinguishable over $\mathcal{T}_q^k$ are also homomorphism indistinguishable over $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.

That such phenomena do not arise under certain mild assumptions was recently conjectured by Roberson [26]. His conjecture asserts that every graph class which is closed under taking minors and disjoint unions is homomorphism distinguishing closed. Here, a graph class $\mathcal{F}$ is *homomorphism distinguishing closed* if it satisfies the following maximality condition: For every graph $F \notin \mathcal{F}$, there exists two graphs G and H which are homomorphism indistinguishable over $\mathcal{F}$ but have different numbers of homomorphism from F.

Since $\mathcal{T}_q^k$, $\mathcal{TW}_{k-1}$, and $\mathcal{TD}_q$ are closed under disjoint unions and minors, the confirmation of Roberson's conjecture would readily imply the semantic counterpart of Theorem 2. Unfortunately, Roberson's conjecture is wide open and has been confirmed only for the class of all planar graphs [26], $\mathcal{TW}_{k-1}$ [22], and for graph classes which are essentially finite [28]. Guided by [22], we add to this short list of examples:

▶ **Theorem 3.** *For $q \geq 1$, the class $\mathcal{TD}_q$ is homomorphism distinguishing closed.*

Combining this with the results of [22], we get that $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ is homomorphism distinguishing closed as well. We then set out to separate homomorphism indistinguishability over $\mathcal{T}_q^k$ and $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$. Despite not being able to prove that $\mathcal{T}_q^k$ is homomorphism distinguishing closed, we prove that the homomorphism distinguishing closure of $\mathcal{T}_q^k$, i.e. the smallest homomorphism distinguishing closed superclass of $\mathcal{T}_q^k$, is a proper subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$, for q sufficiently larger than k. Written out, Theorem 4 asserts that whenever q is sufficiently large in terms of k, then there exist graphs which are homomorphism indistinguishable over $\mathcal{T}_q^k$ but not over $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.

▶ **Theorem 4.** *For $q \geq 3$, $\mathrm{cl}(\mathcal{T}_q^2) \subsetneq \mathcal{TW}_1 \cap \mathcal{TD}_q$, and for $2 \leq k-1 \leq \frac{q}{3+\log q}$, $\mathrm{cl}(\mathcal{T}_q^k) \subsetneq \mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.*

Besides obtaining Theorem 4, we distil the challenge of proving that $\mathcal{T}_q^k$ is homomorphism distinguishing closed to the question whether the monotone variant of the cops-and-robber game is equivalent to the non-monotone variant. In general this equivalence between the monotone and non-monotone variant of a graph searching game is a non-trivial property. There are games where the two variants are equivalent, such as the games corresponding to treewidth [29] and treedepth [11], as well as games where they are not, such as games corresponding to directed treewidth [18] or hypertreewidth [12].

2 Preliminaries

Notation. By $[k]$ we denote the set $\{1, \dots, k\}$. For a finite set X, we write 2^X for the power set of X. For a function f, we denote the domain of f by $\mathrm{dom}(f)$. The *image* of f is the set $\mathrm{img}(f) \coloneqq \{f(x) \mid x \in \mathrm{dom}(f)\}$. The *restriction* of a function $f\colon A \to C$ to some set $B \subseteq A$ is the function $f|_B\colon B \to C$ with $f|_B(x) = f(x)$ for $x \in B$. For functions $f\colon A \to C$, $g\colon B \to C$ that agree on $A \cap B$, we write $f \sqcup g$ for the union of f and g, that is, the function mapping x to $f(x)$ if $x \in A$ and to $g(x)$ if $x \in B$.

We use bold letters to denote tuples. The tuple elements are denoted by the corresponding regular letter together with an index. For example, $\boldsymbol{a}$ stands for the tuple $(a_1, \dots, a_n)$.

Graphs and Labels. A graph G is a tuple $(V(G), E(G))$, where $V(G)$ is a finite set of vertices and $E(G) \subseteq \binom{V(G)}{2}$ is the set of edges. We usually write uv or vu to denote the edge $\{u, v\} \in E(G)$. Unless otherwise specified, all graphs are assumed to be simple: They are undirected, unweighted and contain neither loops nor parallel edges. We denote the class of all graphs by $\mathcal{G}$.

A *k-labelled graph* G is a graph together with a partial function $\nu_G \colon [k] \rightharpoonup V(G)$ that assigns labels from the finite set $[k] = \{1, \dots, k\}$ to vertices of G. A label thus occurs at most once in a graph, a single vertex can have multiple labels, and not all labels have to be assigned. By $L_G = \mathrm{img}(\nu_G)$ we denote the set of labelled vertices of G. A graph where every vertex has at least one label is called *fully labelled*. We denote the class of all k-labelled graphs by $\mathcal{G}_k$.

For $\ell \in [k]$ and $v \in V(G)$, we write $G(\ell \to v)$ to denote the graph obtained from G by setting $\nu_{G(\ell \to v)}(\ell) = v$. We can *remove* a label ℓ from a graph G, which yields a copy G' of G where $\nu_{G'}(\ell) = \bot$ and $\nu_{G'}(\ell') = \nu_G(\ell')$ for all $\ell' \neq \ell$. The *product*[1] $G_1 G_2$ of two labelled graphs is the graph obtained by taking the disjoint union of G_1 and G_2, identifying vertices with the same label, and suppressing any parallel edges that might be created.

We call H a *subgraph* of G if H can be obtained from G by removing vertices and edges. H is a *minor* of G if it can be obtained from G by removing vertices, removing edges, and *contracting* edges. We contract an edge uv by removing it and identifying u and v. For labelled graphs, the new vertex is labelled by the union of labels of u and v.

A graph is *connected* if there exists a path between any two vertices. A *tree* is a graph where any two vertices are connected by exactly one path. The disjoint union of one or more trees is called a *forest*. A *rooted tree* (T, r) is a tree T together with some designated vertex $r \in V(T)$, the *root* of T. A *rooted forest* $(F, \boldsymbol{r})$ is a disjoint union of rooted trees. The *height* of a rooted tree is equal to the number of vertices on the longest path from the root to the leaves. The height of a rooted forest is the maximum height over all its connected components.

At times, the following alternative definition is more convenient. We can view a rooted forest $(F, \boldsymbol{r})$ as a pair $(V(F), \preceq)$, where $\preceq$ is a partial order on $V(F)$ and for every $v \in V(F)$ the elements of the set $\{u \in V(F) \mid u \preceq v\}$ are pairwise comparable: The minimal elements of $\preceq$ are precisely the roots of F, and for any rooted tree (T, r) that is part of F we let $v \preceq w$ if v is on the unique path from r to w.

The height of a rooted forest $(F, \boldsymbol{r})$ is then given by the length of the longest $\preceq$-chain. A rooted tree (T', r') is a *subtree* of a tree (T, r) if $V(T') \subseteq V(T)$ and $\preceq^{T'}$ is the restriction of $\preceq^{T}$ to $V(T')$. Note that the subgraph of T induced by $V(T')$ might not be a tree, since the vertices of T' can be interleaved with vertices that do not belong to T'. We call a subtree T' of T *connected* if its induced subgraph on T is connected.

Homomorphisms. A *homomorphism* from a graph F to a graph G is a map $h \colon V(F) \to V(G)$ satisfying $uv \in E(F) \implies h(u)h(v) \in E(G)$. For k-labelled graphs, we additionally require that $h(\nu_F(\ell)) = \nu_G(\ell)$ for all $\ell \in \mathrm{dom}(\nu_F)$. We denote the set of homomorphisms from F to G by $\mathrm{Hom}(F, G)$. The number of homomorphisms from F to G we denote by $\mathrm{hom}(F, G) := |\mathrm{Hom}(F, G)|$. We write $\mathrm{Hom}(F, G; a_1 \mapsto b_1, \dots, a_n \mapsto b_n)$ to denote the set of homomorphism $h \colon F \to G$ satisfying $h(a_i) = b_i$ for $i \in [n]$. Two graphs G and H are *homomorphism indistinguishable* over a graph class $\mathcal{F}$ if $\mathrm{hom}(F, G) = \mathrm{hom}(F, H)$ for all $F \in \mathcal{F}$.

[1] Categorically speaking, this is a coproduct of labelled graphs or a pushout of graphs.

Logic of Graphs. We will mainly consider *counting first-order logic* C. C extends regular first-order logic FO by quantifiers $\exists^{\geq t}$, for $t \in \mathbb{N}$. Consequently, we can build a C-formula in the usual way from atomic formulae; variables $x_1, x_2, \dots$; logical operators $\wedge, \vee, \to, \neg$; and quantifiers $\forall, \exists, \exists^{\geq t}$. The atomic formulae in the language of graphs are $E\alpha\beta$ and $\alpha = \beta$ for arbitrary variables α, β.

An occurrence of a variable x is called *free* if it is not in the scope of any quantifier. The *free variables* $\operatorname{free}(\varphi)$ of a formula φ are precisely those that have a free occurrence in φ. A formula without free variables is called a *sentence*. We often write $\varphi(x_1, \dots, x_n)$ to denote that the free variables of φ are among $x_1, \dots, x_n$. For a graph G, it usually depends on the interpretation of the free variables whether $G \models \varphi(x_1, \dots, x_n)$. We write $G, v_1, \dots, v_n \models \varphi(x_1, \dots, x_n)$ or $G \models \varphi(v_1, \dots, v_n)$ if G satisfies φ when x_i is interpreted by v_i. We might also give an explicit *interpretation function* $\mathfrak{I}\colon \operatorname{free}(\varphi) \to V(G)$, writing $G, \mathfrak{I} \models \varphi$.

We generalise the notion of C-equivalence, writing $G, v_1, \dots, v_n \equiv_{\mathsf{C}} H, w_1, \dots, w_n$ to denote that for all formulae $\varphi(x_1, \dots, x_n) \in \mathsf{C}$ it holds that $G \models \varphi(v_1, \dots, v_n) \Leftrightarrow H \models \varphi(w_1, \dots, w_n)$. Note that for labelled graphs, such an interpretation function is implicit: If the indices of $\operatorname{free}(\varphi)$ are a subset of the labels of G, then we can interpret the variables x_i by the vertex with the label i, that is, $\mathfrak{I}(x_i) = \nu(i)$. The semantics of C can then be stated succinctly in terms of label assignments.

▶ **Definition 5** (C semantics of labelled graphs)**.** *Let $\varphi \in \mathsf{C}$ and let G be a labelled graph, such that $\nu(i) \in V(G)$ for all $x_i \in \operatorname{free}(\varphi)$. Then $G \models \varphi$ if*

- $\varphi = (x_i = x_j)$ *and* $\nu(i) = \nu(j)$,
- $\varphi = E x_i x_j$ *and* $\nu(i)\nu(j) \in E(G)$,
- $\varphi = \neg\psi$ *and* $G \not\models \psi$,
- $\varphi = \psi \vee \vartheta$ *and* $G \models \psi$ *or* $G \models \vartheta$, *or*
- $\varphi = \exists^{\geq t} x_\ell \psi(x_\ell)$ *and there exist distinct* $v_1, \dots, v_t$, *such that* $G(\ell \to v_i) \models \psi$ *for all* $i \in [t]$.

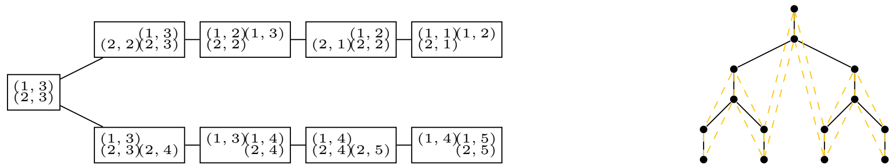
Note that for labelled graphs this is equivalent to extending the standard semantics of FO by the following rule: It is $G, v_1, \dots, v_n \models \exists^{\geq t} y \psi(x_1, \dots, x_n, y)$ if there exist distinct elements $u_1, \dots, u_t \in V(G)$ such that $G \models \psi(v_1, \dots, v_n, u_i)$ for all $i \in [t]$.

We sometimes write $\exists^{=t} x\varphi(x)$ for $\exists^{\geq t} x\varphi(x) \wedge \neg\exists^{\geq t+1} x\varphi(x)$. We also write $\top$ for $\forall x(x = x)$ and $\bot$ for $\neg\top$. As we already did above, we will often restrict ourselves to the connectives $\neg, \vee$ and the quantifier $\exists^{\geq t}$. This set of symbols is indeed equally expressive by De Morgan's laws and observing that $\exists x\varphi(x) \equiv \exists^{\geq 1} x\varphi(x)$ and $\forall x\varphi(x) \equiv \neg\exists x\neg\varphi(x)$.

The quantifier rank $\operatorname{qr}(\cdot)$ of a formula is defined inductively as follows. It is $\operatorname{qr}(\varphi) = 0$ for atomic formulae φ, $\operatorname{qr}(\neg\varphi) = \operatorname{qr}(\varphi)$, $\operatorname{qr}(\varphi \vee \psi) = \max\{\operatorname{qr}(\varphi), \operatorname{qr}(\psi)\}$ and $\operatorname{qr}(\exists^{\geq t} x\varphi) = 1 + \operatorname{qr}(\varphi)$. The *quantifier-rank-$q$ fragment* C_q of counting first order logic consists of all formulae of quantifier rank at most q.

Instead of restricting the quantifer rank, we can also restrict the number of distinct variables that are allowed to occur in a formula. By C^k we denote the *k-variable fragment* of C, consisting of all formulae using at most k different variables. Similarly, the *k-variable quantifier-rank-q fragment* is defined as $\mathsf{C}^k_q := \mathsf{C}^k \cap \mathsf{C}_q$. Note that these are purely syntactic definitions.

Treewidth and Treedepth. Treewidth is a structural graph parameter that measures how close a graph is to being a tree. It is usually defined in terms of *tree decompositions*.



(a) Tree decomposition of the grid $G_{2\times 5}$ of width 3.

(b) A forest cover of the grid $G_{2\times 7}$ of height 6. The edges of the original grid are dashed.

Figure 1 Tree decomposition and forest cover of grids.

▶ **Definition 6.** *A tree decomposition (T, β) of a graph G is a tree T together with a function $\beta\colon V(T) \to 2^{V(G)}$ satisfying*

- $\bigcup_{t\in V(T)} \beta(t) = V(G)$*,*
- *for all $uv \in E(G)$ there is a $t \in V(T)$ with $u, v \in \beta(t)$,*
- *for all $v \in V(G)$ the set $\beta^{-1}(v) = \{t \in V(T) \mid v \in \beta(t)\}$ is connected in T.*

The sets $\beta(t)$ are called *bags*. The *width* of a tree decomposition is $\max_{t\in V(T)} |\beta(t)| - 1$. The *treewidth* $\mathrm{tw}(G)$ of a graph G is the minimum width over all tree decompositions of G. We denote the class of graphs of treewidth at most k by $\mathcal{TW}_k$. For an example of a tree decomposition, see Figure 1a.

Treedepth, on the other hand, can be thought of as measuring how close a graph is to being a star. Alternatively, we may think of it as extending the notion of height beyond rooted forests. It is defined for a graph G as the minimum height of a forest F over the vertices of G, such that all edges in G have an ancestor-descendant relationship in F.

▶ **Definition 7.** *A* forest cover *of a graph G is a rooted forest $(F, \boldsymbol{r})$ with $V(F) = V(G)$, such that for every edge $uv \in E(G)$ it holds that either $u \preceq v$ or $v \preceq u$.*

The *treedepth* $\mathrm{td}(G)$ of G is the minimum height of a forest cover of G. We denote the class of all graphs of treedepth at most q by $\mathcal{TD}_q$. For an example of a forest cover, see Figure 1b.

It is possible to construct a tree decomposition from a forest cover $(F, \boldsymbol{r})$. This is achieved by considering a path of bags, each containing the vertices on a path from $\boldsymbol{r}$ to the leaves of F. It is not hard to see that there is an ordering of these bags that satisfies the conditions of Definition 6. This yields the following relation between treedepth and treewidth.

▶ **Fact 8.** *For every graph G, it holds that* $\mathrm{tw}(G) \le \mathrm{td}(G) - 1$.

Both treewidth and treedepth enjoy characterisations in terms of node searching games, the so called *cops-and-robber games*. The general cops-and-robber game is played on a graph G by Cops, controlling a number of cops; and Robber, controlling a single robber. The cops and the robber are positioned on vertices of G. The goal of Cops is to place a cop on the robber's position, while the robber tries to avoid capture by moving along paths free from cops. The players play in rounds where first Cops announces the next position(s) of the cops (with possible restriction on how many cops may be moved and where they may be positioned) and then Robber moves the robber along some path avoiding all vertices where before and after his move there is a cop. Treewidth can be characterised by the minimum number of cops needed to capture the robber where neither the movement of Cops nor Robber is restricted (see e.g. [29]). A well-known characterisation of treedepth is the minimum number of cops needed if Cops is not allowed to move a cop after it is positioned on the graph (see e.g. [11]). It is equivalent to count the number of rounds the game is played, without restricting the number of cops that can be used by Cops, as long as only one cop can be moved per round. Therefore we use the following unified definition.

▶ **Definition 9** (q-round k-cops-and-robber game). *Let G be a graph and let $k, q \geq 1$. The* monotone q-round k-cops-and-robber game mon-$\mathrm{CR}_q^k(G)$ *is defined as follows:*

We play the game on G' which is constructed from G by adding a disjoint k-vertex clique K. The cop positions are sets $X \in \binom{V(G')}{k}$, the robber position is a vertex $v \in G$. The game is initiated with all cops positioned on K and the robber on any vertex in $V(G)$ of his choice. If the cops are at positions X and robber at vertex v we write (X, v) for the position of the game. For $X \subseteq V(G')$ and $v \in G$, we call the connected component γ_v^X of the graph $G \setminus X$, with $v \in \gamma_v^X$, the robber escape space*. If the cop strategy only depends on γ_v^X and not the precise vertex that the robber occupies, we write (X, γ_v^X) for the position of the game.*

In round $i \leq q$,

- *Cops can move from the set X_i to a set X_{i+1} if $|X_i \cap X_{i+1}| = k-1$ and $\gamma_v^{X_i} \supseteq \gamma_v^{X_i \cap X_{i+1}}$,*
- *Robber moves along some (possibly empty) path $v_i P v_{i+1}$, where no (inner) vertex is in $X_i \cap X_{i+1}$.*
- *Cops wins if $v_{i+1} \in X_{i+1}$.*

Robber wins if Cops has not won after q rounds.

If we drop the condition $\gamma_v^{X_i} \supseteq \gamma_v^{X_i \cap X_{i+1}}$ in the movement of the cop, we call this the non-monotone *variant of the game and write $\mathrm{CR}_q^k(G)$.*

We write $\mathrm{CR}_q(G)$ instead of $\mathrm{CR}_q^q(G)$ and $\mathrm{CR}^k(G)$ instead of $\mathrm{CR}_{|V(G)|}^k(G)$. Treewidth and treedepth can be characterised in terms of winning strategies for these games.

▶ **Lemma 10** ([29, Theorem 1.4] and [11, Theorem 4]). *Let G be a graph. Let $k, q \geq 1$.*

1. *G has treewidth at most k iff Cops has a winning strategy for $\mathrm{CR}^{k+1}(G)$ iff Cops has a winning strategy for* mon-$\mathrm{CR}^{k+1}(G)$*.*
2. *G has treedepth at most q iff Cops player has a winning strategy for $\mathrm{CR}_q(G)$ iff Cops has a winning strategy for* mon-$\mathrm{CR}_q(G)$*.*

3 Graph Decompositions Accounting for Treewidth and Treedepth Simultaneously

In this section, we reconcile treewidth and treedepth by introducing graph decompositions which account simultaneously for depth and width. These efforts yield various equivalent characterisations of the graph class $\mathcal{T}_q^k$, a subclass both of $\mathcal{TW}_{k-1}$ and $\mathcal{TD}_q$, the classes of graphs of treewidth $\leq k-1$ and treedepth $\leq q$, respectively. By introducing a variant of the standard cops-and-robber game which captures $\mathcal{T}_q^k$ and adapting a result from [10], we show that $\mathcal{T}_q^k$ is a proper subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ if q is sufficiently larger than k.

3.1 Four Characterisations for $\mathcal{T}_q^k$

We start with the original definition of the class $\mathcal{T}_q^k$, which incorporates width into forest covers from treedepth. This definition has first been introduced as k-traversal in [1].

▶ **Definition 11.** *Let G be graph and $k \geq 1$. A* k-pebble forest cover *of G is a tuple $(F, \boldsymbol{r}, p)$, where $(F, \boldsymbol{r})$ is a rooted forest over the vertices $V(G)$ and a pebbling function $p \colon V(G) \to [k]$ such that:*

- *If $uv \in E(G)$, then $u \preceq v$ or $v \preceq u$ in $(F, \boldsymbol{r})$.*
- *If $uv \in E(G)$ and $u \prec v$ in $(F, \boldsymbol{r})$, then for every $w \in V(G)$ with $u \prec w \preceq v$ in $(F, \boldsymbol{r})$ it holds that $p(u) \neq p(w)$.*

$(F, \boldsymbol{r}, p)$ has depth $q \geq 1$ if $(F, \boldsymbol{r})$ has height q. We write $\mathcal{T}_q^k$ for the class of all graphs G admitting a k-pebble forest cover of depth q.

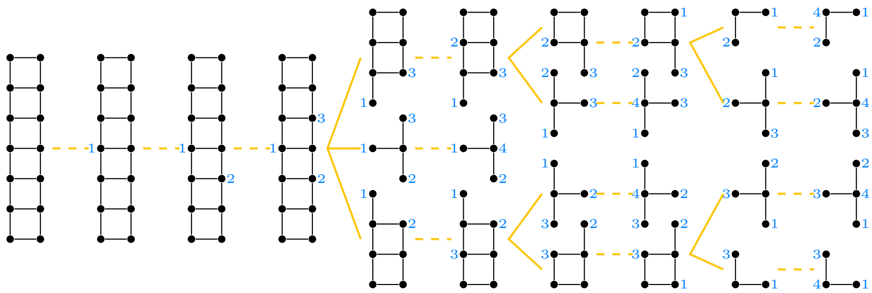


Figure 2 A 4-construction tree for the grid $G_{2\times 7}$ of elimination depth 6. Edges entering elimination nodes are dashed.

The class $\mathcal{T}_q^k$ can also be defined by measuring the depth of a tree decomposition (T, β). Crucially, it does not suffice to take the height of T into account since this notion is not robust. For example, it is well known that one can alter a tree decomposition by subdividing any edge multiple times and copying the bag of the child node. This transformation does neither change the width of the decomposition, nor does it affect the information how to decompose the graph. However, the height of the tree will change drastically under this transformation. It turns out that the following is the right definition:

▶ **Definition 12.** *Let G be a graph. A tuple (T, r, β) is a* rooted tree decomposition *of G if (T, β) is a tree decomposition of G and $r \in V(T)$. The* depth *of (T, β) is*

$$\mathrm{dp}(T, \beta) \coloneqq \min_{r \in V(T)} \mathrm{dp}(T, r, \beta) \qquad \textit{where} \qquad \mathrm{dp}(T, r, \beta) \coloneqq \max_{v \in V(T)} \left| \bigcup_{t \preceq v} \beta(t) \right|.$$

Lastly we define a construction inspired by Dvořák [8], that enables us to use their proof technique to study the expressive power of first-order logic with counting quantifiers using homomorphism indistinguishability (see Figure 2).

▶ **Definition 13.** *Let G be a (possibly labelled) graph. A k-*construction tree *for G is a tuple (T, λ, r), where T is a tree rooted at r and $\lambda\colon V(T) \to \mathcal{G}_k$ is a function assigning k-labelled graphs to the nodes of T such that:*

1. *$\lambda(r) = G$,*
2. *all leaves $\ell \in V(T)$ are assigned fully labelled graphs,*
3. *all internal nodes $t \in V(T)$ with exactly one child t' are* elimination nodes*, that is $\lambda(t)$ can be obtained from $\lambda(t')$ by removing one label, and*
4. *all internal nodes $t \in V(T)$ with more than one child are* product nodes*, that is $\lambda(t)$ is the product of its children.*

The elimination depth *of a construction tree (T, λ, r) is the maximum number of elimination nodes on any path from the root r to a leaf. If G has a k-construction tree of elimination depth $\leq q$, we say that G is (k, q)-*constructible*. We write $\mathcal{L}_q^k$ for the class of all k-labelled (k, q)-constructible graphs.*

It turns out that all three notions coincide.

▶ **Theorem 14.** *Let $k, q \geq 1$. For every graph G, the following are equivalent:*

1. *G is (k, q)-constructible,*
2. *G has a tree decomposition of width $k - 1$ and depth q,*
3. *$G \in \mathcal{T}_q^k$, that is G admits a k-pebble forest cover of depth q.*

The equivalence of Items 1 and 2 is proven by a carefully choosing a tree decomposition such tha the bags are identified with the labelled vertices of the construction tree. For the equivalence of Items 2 and 3, we follow the proof of [3, Theorem 19] and observe that their construction preserves depth. Details can be found in the full version [9].

▶ **Corollary 15.** *Let $k, q \geq 1$. The class $\mathcal{T}_q^k$ is minor-closed, closed under taking disjoint unions, and a subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.*

Given Theorem 14, Corollary 15 is immediate. Dawar, Jakl, and Reggio reduced the proofs of the results of Grohe and Dvořák [13, 8] to a “combinatorial core” [5, Remark 17], which amounts to showing that the classes $\mathcal{TW}_k$ and $\mathcal{TD}_q$ are closed under contracting edges. To that end, Corollary 15 illustrates the benefits of characterising $\mathcal{T}_q^k$ in terms of tree decompositions (Definition 12): Proving that pebble forest covers are preserved under edge contractions requires a non-trivial amount of bookkeeping while the analogous statement for tree decomposition is straightforward.

We conclude with a characterisation of $\mathcal{T}_q^k$ in terms of a cops-and-robber game.

▶ **Lemma 16.** *The Cops win the game* $\text{mon-CR}_q^k(G)$ *if and only if $G \in \mathcal{T}_q^k$.*

The proof of this lemma follows the same strategies as the proof for the monotone version of the cops-and-robber game for treewidth (see for example [24]). Details can be found in the full version [9].

3.2 Separating $\mathcal{T}_q^k$ from $\mathcal{TW}_k \cap \mathcal{TD}_q$ Syntactically

We aim to show that the graph class $\mathcal{T}_q^k$ is a proper subclass of $\mathcal{TW}_k \cap \mathcal{TD}_q$. Since $\mathcal{T}_q^q = \mathcal{TD}_q = \mathcal{TW}_{q-1} \cap \mathcal{TD}_q$, one can only hope to separate the classes if q is larger than k. Using the characterisations of $\mathcal{T}_q^k$ via a cops-and-robber game, we show that there are indeed graphs which do not admit a decomposition where the width is bounded by the treewidth and simultaneously the depth bounded by the treedepth. The graph we consider is the $(h \times \ell)$-grid $G_{h\times\ell}$ with $h < \ell$. It is well known that $\text{tw}(G_{h\times\ell}) = h$ and $\text{td}(G_{h\times\ell}) \leq h\lceil\log(\ell+1)\rceil$, cf. Figure 1. We give a lower bound to the number of rounds q that the robber can survive in a, possibly non-monotone, game $\text{CR}_q^{h+1}(G_{h\times\ell})$, which is linear in both ℓ and h.

▶ **Lemma 17.** *For $1 < h < \ell - 2$ and $q \leq \frac{h(\ell-h+2)}{4}$, Robber wins the game* $\text{CR}_q^{h+1}(G_{h\times\ell})$.

The proof of Lemma 17 builds upon [10]. The winning strategy of Robber is to always stay in the component with the most vertices. We find a lower bound on the size of this component in terms of the number of rounds played and prove that Cops can only force this bound to shrink by two vertices each round, for the majority of the rounds. We additionally show that for $h > 3$ Cops can indeed force the component to shrink by two vertices each round and thus in this case the bound given in Lemma 17 is tight up to an additive term depending only on h.

For $h = 1$, the proof idea of Lemma 17 is not applicable as on a path there are separators of size two that separate the path into three components of roughly equal size. Despite that, one may observe that such a separator does not benefit Cops as from such a position he would always have to combine two of these components into a larger one. Thus Cops can only move along the path and shrink the escape space of Robber by one vertex. This case is covered in the original proof of [10].

▶ **Lemma 18** ([10, Theorems 5 and 7])**.** *Let $\ell \geq 1$. Robber wins the game* $\text{CR}_q^2(G_{1\times\ell})$ *if and only if $q \leq \lceil\frac{\ell-1}{2}\rceil$.*

With an appropriate choice of ℓ and a small alteration to the graph $G_{k-1\times\ell}$ that ensures that the treedepth of the graph is exactly q, we can prove the following:

▶ **Theorem 2.** *For $q \geq 3$, $\mathcal{T}_q^2 \subsetneq \mathcal{TW}_1 \cap \mathcal{TD}_q$, and for $2 \leq k-1 \leq \frac{q}{3+\log q}$, $\mathcal{T}_q^k \subsetneq \mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.*

Detailed proofs can be found in the full version [9]. The reader should note that the proof of the lower bound on the number of rounds even holds for the non-monotone game, which in turn allows us to lift this result to the semantic level of homomorphism indistinguishability.

4 Homomorphism Indistinguishability

In this section, we turn to investigating $\mathcal{T}_q^k$ in terms of homomorphism indistinguishability. It turns out that the representation of $\mathcal{T}_q^k$ in terms of construction trees offers a great framework for obtaining characterisations of logical equivalence. The general idea will be to use these trees to inductively construct C-formulae that capture homomorphism counts. Not only does this approach generalise results from [8, 13], it also yields an intuitive characterisation of C_q^k-equivalence. This provides a more elementary proof of a result from [5].

Moreover, the constructive nature of our proof strategy proves fruitful in obtaining additional characterisations of fragments of C. The general idea is to impose natural restrictions on the construction trees, such that a fragment $\mathsf{L} \subsetneq \mathsf{C}$ already suffices to capture homomorphism counts. By choosing these restrictions carefully, the resulting subclass of $\mathcal{T}_q^k$ is then still large enough to capture L-equivalence. We illustrate this point by giving a characterisation of *guarded* counting logic GC.

We conclude by *semantically* separating $\mathcal{T}_q^k$ and $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$. More formally, we show that, for q sufficiently larger than k, there exist graphs G and H which are homomorphism indistinguishable over $\mathcal{T}_q^k$ but have different numbers of homomorphisms from some graph in $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.

4.1 Homomorphism Indistinguishability over $\mathcal{T}_q^k$ is C_q^k-Equivalence

In his 2010 paper [8], Dvořák showed that C^k-equivalence is equivalent to homomorphism indistinguishability over $\mathcal{TW}_k$. It turns out that his techniques generalise remarkably well to construction trees. To begin with, we make a few observations on how the operations that make up construction trees interact with homomorphism counts.

First, observe that when a graph F is fully labelled there can be at most one homomorphism $h\colon F \to G$, which is entirely determined by the label positions in G.

▶ **Observation 19.** *Let F be a fully labelled graph and let L_F denote the set of labels. Then there exists a unique homomorphism $h\colon F \to G$ if for all labels $i, j \in L_F$*

- $\nu_F(i) = \nu_F(j) \implies \nu_G(i) = \nu_G(j)$,
- $\nu_F(i)\nu_F(j) \in E(F) \implies \nu_G(i)\nu_G(j) \in E(G)$.

Further, for $h \in \mathrm{Hom}(F_1F_2, G)$ the restrictions $h|_{V(F_1)}$ and $h|_{V(F_2)}$ are homomorphisms, and since two homomorphisms $g\colon F_1 \to G$ and $h\colon F_2 \to G$ must agree on vertices with the same label, $g \sqcup h$ is well-defined and a homomorphism from F_1F_2 to G. This implies the following for products.

▶ **Observation 20.** *For labelled graphs F_1, F_2, G, it holds that* $\hom(F_1F_2, G) = \hom(F_1, G) \cdot \hom(F_2, G)$.

Finally, we can also relate the homomorphism counts from graphs F and F', whenever F' is obtained from F by removing some label ℓ. Then in any homomorphism $h\colon F' \to G$ the image of $\nu_F(\ell)$ is no longer necessarily $\nu_G(\ell)$. Hence, we can obtain $\hom(F,G)$ by moving the label ℓ to different vertices in G and tallying up the homomorphisms from F to those graphs. We may write this succinctly as

$$\hom(F', G) = \sum_{v \in V(G)} \hom(F, G(\ell \to v)),$$

or slightly more verbose as follows.

▶ **Observation 21.** *Let F' be the graph obtained from F by removing a single label ℓ. Then $\hom(F', G) = m$ if and only if there exists a decomposition $m = \sum_i c_i m_i$ with $c_i, m_i \in \mathbb{N}$, such that:*

- *There exist exactly c_i vertices v with $\hom(F, G(\ell \to v)) = m_i$.*
- *There exist exactly $c := \sum_i c_i$ vertices v with $\hom(F, G(\ell \to v)) \neq 0$.*

The crucial insight is that the conditions above are all definable in C. In particular, the condition for fully labelled graphs can be expressed as a conjunction of atomic formulae using at most $|L_F|$ different variables. This allows us to prove the following lemma by induction over a construction tree. The proofs of the lemmas in this section can be found in the full version [9].

▶ **Lemma 22.** *Let $F \in \mathcal{L}_q^k$ be a k-labelled graph, and let $m \geq 0$. Then there exists a formula $\varphi_m \in \mathsf{C}_q^k$ such that for each k-labelled graph G with $L_F \subseteq L_G$, $G \models \varphi_m$ if and only if $\hom(F, G) = m$.*

Ideally, we would like to prove the converse in a similar manner. Given some C_q^k-formula ψ that distinguishes two graphs G and H, construct a graph $F \in \mathcal{L}_q^k$ with $\hom(F, G) \neq \hom(F, H)$ by induction over the structure of ψ. While graphs are too rigid in this regard, such a construction will be possible using *linear combinations* of graphs.[2]

For a class of (labelled) graphs $\mathcal{F}$, we let $\mathbb{R}\mathcal{F}$ be the class of finite formal linear combinations with real coefficients of graphs $F \in \mathcal{F}$. We linearly extend the function hom to $\mathbb{R}\mathcal{G}$ by defining

$$\hom(\mathfrak{F}, G) = \hom(\sum_i c_i F_i, G) := \sum_i c_i \cdot \hom(F_i, G),$$

for $\mathfrak{F} = \sum_i c_i F_i$.

The following observation shows that homomorphism indistinguishability over $\mathcal{F}$ and over $\mathbb{R}\mathcal{F}$ is essentially the same. This allows us to reason about linear combinations instead of graphs.

▶ **Observation 23.** *Let G, H be graphs and let $\mathfrak{F} \in \mathbb{R}\mathcal{F}$. If $\hom(\mathfrak{F}, G) \neq \hom(\mathfrak{F}, H)$, then there is already an $F \in \mathcal{F}$ with $\hom(F, G) \neq \hom(F, H)$.*

The product of two linear combinations is defined in the natural way, where the graph products distribute over the sum. We also remove any graphs with loops that might have been created from the resulting linear combination. This definition preserves the property that $\hom(\mathfrak{F}_1\mathfrak{F}_2, H) = \hom(\mathfrak{F}_1, H)\hom(\mathfrak{F}_2, H)$ and admits the following interpolation lemma.

[2] These linear combinations are called "quantum graphs" in [8].

▶ **Lemma 24** ([8, Lemma 5]). *Let $\mathcal{F}$ be a class of graphs and let $\mathfrak{F} \in \mathbb{R}\mathcal{F}$. If S^-, S^+ are disjoint finite sets of real numbers, then there exists a linear combination $\mathfrak{F}[S^-; S^+] \in \mathbb{R}\mathcal{G}$, such that for any graph G*

- $\hom(\mathfrak{F}[S^-; S^+], G) = 1$ *if* $\hom(\mathfrak{F}, G) \in S^+$, *and*
- $\hom(\mathfrak{F}[S^-; S^+], G) = 0$ *if* $\hom(\mathfrak{F}, G) \in S^-$.

Moreover, if $\mathcal{F}$ is closed under taking products then $\mathfrak{F}[S^-; S^+] \in \mathbb{R}\mathcal{F}$.

With this result, we may construct for a formula $\psi \in \mathsf{C}_q^k$ and $n \in \mathbb{N}$ a linear combination $\mathfrak{F}_{\psi,n}$ such that for all graphs G of size n it holds that $\hom(\mathfrak{F}_{\psi,n}, G) = 1$ if $G \models \psi$ and $\hom(\mathfrak{F}_{\psi,n}, G) = 0$ otherwise. We say that $\mathfrak{F}_{\psi,n}$ *models ψ for graphs of size n.*

▶ **Lemma 25.** *Let $k, q \geq 1$ and let φ be a C_q^k-formula. Then for every $n \geq 1$ there exists an $\mathfrak{F} \in \mathbb{R}\mathcal{L}_q^k$ modelling φ for graphs of size n.*

The proof is by structural induction on φ and exploits how homomorphism counts change under label deletions and taking products. Interpolation is used to define negation and to renormalise homomorphism counts to 0 or 1. The construction has the property that labels in the components of $\mathfrak{F}$ correspond to free variables of φ. This correspondence yields the following corollary.

▶ **Corollary 26.** *Let $k, q \geq 1$ and let φ be a C_q^k-sentence. Then for every $n \geq 1$ there exists an $\mathfrak{F} \in \mathbb{R}\mathcal{T}_q^k$ modelling φ for graphs of size n.*

We can now prove the main result of this section.

▶ **Theorem 27.** *Let $k, q \geq 1$. Two graphs G and H are C_q^k-equivalent if and only if they are homomorphism indistinguishable over $\mathcal{T}_q^k$.*

Proof. Suppose there exists a graph $F \in \mathcal{T}_q^k \subseteq \mathcal{L}_q^k$ with $\hom(F, G) \neq \hom(F, H)$. Then by Lemma 22 there exist C_q^k-sentences φ_m^F for $m \geq 0$ such that $G \models \varphi_m^F$ iff $\hom(F, G) = m$. Consequently, there exists an m with $G \models \varphi_m^F$ and $H \not\models \varphi_m^F$, so G and H cannot satisfy the same C_q^k-sentences.

Suppose now that there exists a sentence $\varphi \in \mathsf{C}_q^k$ with $G \models \varphi$ and $H \not\models \varphi$. Without loss of generality, we may assume $|G| = |H| =: n$. Then, by Corollary 26, there is an $\mathfrak{F} \in \mathbb{R}\mathcal{T}_q^k$ that models φ for graphs of size n, that is, $\hom(\mathfrak{F}, G) \neq \hom(\mathfrak{F}, H)$. By Observation 23, this already implies the existence of an $F \in \mathcal{T}_q^k$ with $\hom(F, G) \neq \hom(F, H)$. ◀

By dropping the restriction on one of the two parameters in Theorem 27, we recover the original results of Dvořák [8] and Grohe [13]:

▶ **Corollary 28.** *Let $k, q \geq 1$. Let G and H be graphs.*

1. *G and H are C^k-equivalent iff they are homomorphism indistinguishable over $\mathcal{TW}_{k-1}$.*
2. *G and H are C_q-equivalent iff they are homomorphism indistinguishable over $\mathcal{TD}_q$.*

4.2 Guarded Fragments

Given the constructive nature of the arguments in Section 4.1, it is interesting to investigate whether the same strategy can be used to obtain results for different fragments of C by restricting construction trees in some way. An example where this works well is guarded counting logic GC.

In the guarded fragment GC, quantifiers are restricted to range over the neighbours of a vertex. Formally, we require that quantifiers only occur in the form $\exists^{\geq t} y(Exy \land \psi(z_1, \dots, z_n, y))$, where x and y are distinct variables.

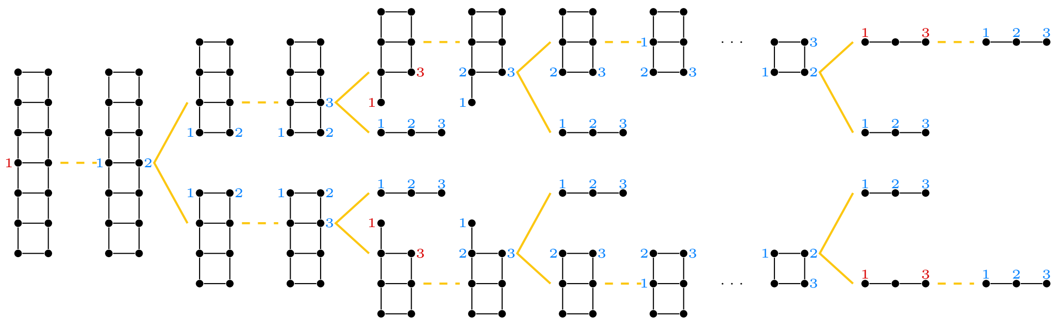


Figure 3 A guarded 3-construction tree of elimination depth 7 for the grid $G_{2\times 7}$ with one labelled vertex. Edges entering elimination nodes are dashed. At every labelled graph, those labels that may be removed are marked blue, those that may not be removed are marked red. The dotted omitted part of the construction tree follows the same pattern.

Since GC-formulae necessarily have a free variable, it is not immediately obvious how to define GC-equivalence on graphs. One option is to consider GC-equivalence of graphs together with a distinguished vertex. This works, and we will, in fact, obtain a characterisation for precisely this relation. However, we would prefer to study the landscape of homomorphism indistinguishability relations on graphs without additional structure. The following natural definition of GC-equivalence allows us to lift our result to graphs without a dinstinguished vertex.

▶ **Definition 29** (GC-equivalence). *Let G and H be unlabelled graphs. We say that G and H are GC-equivalent, in symbols $G \equiv_{\mathsf{GC}} H$, if there exists a bijection $f \colon V(G) \to V(H)$ such that $G, v \models \varphi(x) \iff H, f(v) \models \varphi(x)$ for all $v \in V(G)$ and $\varphi \in \mathsf{GC}$.*

To apply our proof strategy to GC, we need to restrict the construction trees such that guarded quantifiers suffice to express homomorphism counts. Observe that the quantifiers are only needed to describe how the number of homomorphisms $F \to G$ changes by removing a label from F. More precisely, we make use of the fact that removing a label ℓ from F is the same as moving it around in G and tallying up the resulting homomorphisms. Now if ℓ is adjacent to some other label ℓ', then the only positions of ℓ in G that contribute to the final homomorphism count are adjacent to ℓ'. Consequently, it will suffice to quantify over the neighbours of ℓ'.

▶ **Definition 30.** *Let $k, q \geq 1$. By $\mathcal{GL}^k_q$ we denote the class of k-labelled graphs that admit a k-construction tree of elimination-depth q with the additional restriction that labels can only be removed if they have a labelled neighbour.*

We observe that in Figure 2 there are nodes where labels without labelled neighbors are removed. In Figure 3, we depict a construction tree without such nodes of the same graph. We remark that all graphs in $\mathcal{GL}^k_q$ are labelled, as a single label can never be removed. Under these restrictions, the argument from Lemma 22 goes through using only guarded quantifiers.

▶ **Lemma 31.** *Let $F \in \mathcal{GL}^k_q$. Then for each $m \geq 0$ there is a formula $\varphi_m \in \mathsf{GC}^k_q$ such that for appropriately labelled graphs G it holds that $\hom(F, G) = m$ iff $G \models \varphi_m$.*

The proof of the converse – showing that there exists for each $\psi \in \mathsf{GC}^k_q$ an $\mathfrak{F} \in \mathbb{R}\mathcal{GL}^k_q$ modelling ψ – also goes through nearly unchanged.

▶ **Lemma 32.** *Let $\varphi \in \mathsf{GC}^k_q$. Then there is an $\mathfrak{F} \in \mathbb{R}\mathcal{GL}^k_q$ modelling φ for graphs of size n.*

The analogues of these two lemmas already sufficed to prove Theorem 27. Here, however, we still need to be mindful of any remaining labels. Concretely, Lemma 31 and Lemma 32 imply the following for GC sentences.

▶ **Corollary 33.** *Let G, v and H, w be graphs together with a single labelled vertex. Then the following are equivalent.*

1. *For all $\psi(x) \in \mathsf{GC}^k_q$, it holds $G, v \models \psi(x) \iff H, w \models \psi(x)$.*
2. $\hom(F, G) = \hom(F, H)$ *for all $F \in \mathcal{GL}^k_q$.*

While this is already a nice result, ideally we would like to make a statement about general, unlabelled, graphs. Fortunately, simply removing all labels from $F \in \mathcal{GL}^k_q$ turns out to induce the equivalence relation described in Definition 29.

Let us denote by $\mathcal{GT}^k_q$ the class of graphs in $\mathcal{GL}^k_q$ with all labels removed. Then we can state the following theorem, characterising GC^k_q-equivalence in terms of homomorphism indistinguishability. The details can be found in the full version [9].

▶ **Theorem 1.** *Let $k, q \geq 1$. Two graphs G and H are GC^k_q-equivalent if and only if they are homomorphism indistinguishable over $\mathcal{GT}^k_q$.*

We remark that in [2] the logic GC was studied with comonadic means. In this work, winning strategies for Duplicator in guarded bisimulation games were characterised as coKleisli morphisms with respect to a suitably defined comonad. This is in contrast to the comonadic Lovász-type theorem of [5] which applies to logical equivalences which can be characterised as coKleisli isomorphisms. Thus, Theorem 1 does not seem to be immediate from [2, 5].

4.3 Separating $\mathcal{T}^k_q$ from $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ Semantically

By Theorem 2, the graph class $\mathcal{T}^k_q$ is a proper subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$. Despite that, it could well be that the homomorphism indistinguishability relations of the two graph classes (and via Theorem 27 also C^k_q-equivalence) coincide, i.e. $G \equiv_{\mathcal{T}^k_q} H$ if and only if $G \equiv_{\mathcal{TW}_{k-1} \cap \mathcal{TD}_q} H$ for all graphs G and H. It turns out that this is not the case.

In general, establishing that the homomorphism indistinguishability relations $\equiv_{\mathcal{F}_1}$ and $\equiv_{\mathcal{F}_2}$ of two graph classes $\mathcal{F}_1 \neq \mathcal{F}_2$ are distinct is a notoriously hard task. Pivotal tools for accomplishing this were introduced by Roberson in [26]. He defines the *homomorphism distinguishing closure* $\mathrm{cl}(\mathcal{F})$ of a graph class $\mathcal{F}$ as the graph class

$$\mathrm{cl}(\mathcal{F}) := \{F \text{ graph} \mid \forall G, H.\ G \equiv_{\mathcal{F}} H \implies \hom(F, G) = \hom(F, H)\}.$$

A graph class $\mathcal{F}$ is *homomorphism distinguishing closed* if $\mathcal{F} = \mathrm{cl}(\mathcal{F})$. This is the case if and only if for every $F \notin \mathcal{F}$ there exist two graphs G and H homomorphism indistinguishable over $\mathcal{F}$ and satisfying that $\hom(F, G) \neq \hom(F, H)$. Therefore, homomorphism distinguishing closed graph classes may be thought of as maximal in terms of homomorphism indistinguishability.

Roberson conjectures that *every graph class which is closed under taking minors and disjoint unions is homomorphism distinguishing closed.* A confirmation of this conjecture would aid separating homomorphism indistinguishability relations and in turn all equivalence relations between graphs which have such characterisations, cf. [27]. In particular, it would readily imply that $\equiv_{\mathcal{T}^k_q}$ and $\equiv_{\mathcal{TW}_{k-1} \cap \mathcal{TD}_q}$ are distinct, cf. Corollary 15. Unfortunately, the conjecture's assertion is only known to be true for the class of planar graphs [26], $\mathcal{TW}_k$ [22] and graph classes arising from finite graph classes [28]. Towards separating $\equiv_{\mathcal{T}^k_q}$ and $\equiv_{\mathcal{TW}_{k-1} \cap \mathcal{TD}_q}$, we first add to this list by proving the following:

▶ **Theorem 3.** *For $q \geq 1$, the class $\mathcal{TD}_q$ is homomorphism distinguishing closed.*

The proof of Theorem 3 follows the proof in [22] of the assertion that the class $\mathcal{TW}_k$ is homomorphism distinguishing closed for all $k \geq 0$. Central to it is a construction of highly similar graphs from [26] which is reminiscent of the CFI-construction [4]. With these ingredients, it suffices to prove that Duplicator wins the model comparison game characterising C_q-equivalence on these CFI-like graphs constructed over a graph of high treedepth. To that end, we build a Duplicator strategy from a Robber strategy for the game from Definition 9. The connection between model comparison and node searching games via CFI-constructions is well-known [17, 6].

Crucial for the aforementioned argument is that Robber wins the *non-monotone* node searching game characterising bounded treedepth. Indeed, it cannot be assumed that Cops plays monotonously since he must shadow Spoiler's moves. Since we are unable to establish a non-monotone node searching game characterising $\mathcal{T}_q^k$, we cannot conclude along the same lines that $\mathcal{T}_q^k$ is homomorphism distinguishing closed. Nevertheless, we separate $\equiv_{\mathcal{T}_q^k}$ and $\equiv_{\mathcal{TW}_{k-1} \cap \mathcal{TD}_q}$. The details are deferred to the full version [9].

▶ **Theorem 4.** *For $q \geq 3$, $\mathrm{cl}(\mathcal{T}_q^2) \subsetneq \mathcal{TW}_1 \cap \mathcal{TD}_q$, and for $2 \leq k-1 \leq \frac{q}{3+\log q}$, $\mathrm{cl}(\mathcal{T}_q^k) \subsetneq \mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.*

Proving that $\mathcal{T}_q^k$ is characterised by Robber winning the (non-monotone) game CR_q^k, c.f. Lemma 16, would immediately imply that $\mathcal{T}_q^k$ is homomorphism distinguishing closed.

In the introduction, we mentioned that it is tempting to assume that C_q^k-equivalence coincides with homomorphism indistinguishability over $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ because $\mathsf{C}_q^k = \mathsf{C}^k \cap \mathsf{C}_q$. However, our results imply that there are properties definable in both C^k and C_q that are not definable in C_q^k.

▶ **Corollary 34.** *For $2 \leq k-1 \leq \frac{q}{3+\log q}$, there exist sentences $\varphi \in \mathsf{C}^k$ and $\psi \in \mathsf{C}_q$ such that $\varphi \equiv \psi$, but for all sentences $\vartheta \in \mathsf{C}_q^k$ it holds that $\vartheta \not\equiv \varphi$.*

Proof. By Theorem 4, for suitable k, q, there exist graphs G and H such that $G \equiv_{\mathcal{T}_q^k} H$ and there exists $F \in \mathcal{TW}_{k-1} \cap \mathcal{TD}_q$ such that $m := \hom(F, G) \neq \hom(F, H)$. By Lemma 22 and Theorem 14, there exist sentences $\varphi \in \mathsf{C}^k$ and $\psi \in \mathsf{C}_q$ such that $\hom(F, K) = m \iff K \models \varphi \iff K \models \psi$ for every graph K. However, this property cannot be defined in C_q^k since G and H satisfy the same C_q^k-sentences by Theorem 27. ◀

5 Outlook

We studied the expressive power of the counting logic fragment C_q^k with tools from homomorphism indistinguishability. After giving an elementary and uniform proof of theorems from [8, 13, 5], we showed that the graph class $\mathcal{T}_q^k$, whose homomorphism indistinguishability relation characterises C_q^k-equivalence, is a proper subclass of $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$. Finally, we showed that homomorphism indistinguishability over $\mathcal{T}_q^k$ is not the same as homomorphism indistinguishability over $\mathcal{TW}_{k-1} \cap \mathcal{TD}_q$.

The main problem remaining open is to tighten Theorem 4 by proving that the graph class $\mathcal{T}_q^k$ is homomorphism distinguishing closed, as predicted by Roberson's conjecture. Our contribution in this direction is a reduction to a purely graph theoretic problem: Proving that the class $\mathcal{T}_q^k$ is characterised by a *non-monotone* cops-and-robber game, cf. Lemma 16, would be sufficient to yield this claim. Exploring whether intertwining node searching and model comparison games can help to verify Roberson's conjecture in other cases seems a tempting direction for future research.

With slight reformulations, our results might yield insights into the ability of the Weisfeiler–Leman algorithm to determine subgraph counts after a fixed number of rounds [25, 22].

References

1 Samson Abramsky, Anuj Dawar, and Pengming Wang. The pebbling comonad in Finite Model Theory. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavík, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005129.

2 Samson Abramsky and Dan Marsden. Comonadic Semantics for Guarded Fragments. In *Proceedings of the 36th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '21. IEEE Press, 2021. doi:10.1109/LICS52264.2021.9470594.

3 Samson Abramsky and Nihil Shah. Relating structure and power: Comonadic semantics for computational resources. *Journal of Logic and Computation*, 31(6):1390–1428, September 2021. doi:10.1093/logcom/exab048.

4 Jin-Yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, December 1992. doi:10.1007/BF01305232.

5 Anuj Dawar, Tomáš Jakl, and Luca Reggio. Lovász-Type Theorems and Game Comonads. In *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13, June 2021. doi:10.1109/LICS52264.2021.9470609.

6 Anuj Dawar and David Richerby. The power of counting logics on restricted classes of finite structures. In Jacques Duparc and Thomas A. Henzinger, editors, *Computer Science Logic*, pages 84–98. Springer Berlin Heidelberg, 2007. doi:10.1007/978-3-540-74915-8_10.

7 Holger Dell, Martin Grohe, and Gaurav Rattan. Lovász Meets Weisfeiler and Leman. *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, pages 40:1–40:14, 2018. doi:10.4230/LIPICS.ICALP.2018.40.

8 Zdeněk Dvořák. On recognizing graphs by numbers of homomorphisms. *Journal of Graph Theory*, 64(4):330–342, August 2010. doi:10.1002/jgt.20461.

9 Eva Fluck, Tim Seppelt, and Gian Luca Spitzer. Going Deep and Going Wide: Counting Logic and Homomorphism Indistinguishability over Graphs of Bounded Treedepth and Treewidth, 2023. doi:10.48550/arXiv.2308.06044.

10 Martin Fürer. Weisfeiler-Lehman Refinement Requires at Least a Linear Number of Iterations. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*, volume 2076 of *Lecture Notes in Computer Science*, pages 322–333. Springer, 2001. doi:10.1007/3-540-48224-5_27.

11 Archontia C. Giannopoulou, Paul Hunter, and Dimitrios M. Thilikos. LIFO-search: A min–max theorem and a searching game for cycle-rank and tree-depth. *Discrete Applied Mathematics*, 160(15):2089–2097, October 2012. doi:10.1016/j.dam.2012.03.015.

12 Georg Gottlob, Nicola Leone, and Francesco Scarcello. Robbers, marshals, and guards: game theoretic and logical characterizations of hypertree width. *Journal of Computer and System Sciences*, 66(4):775–808, 2003. Special Issue on PODS 2001. doi:10.1016/S0022-0000(03)00030-8.

13 Martin Grohe. Counting Bounded Tree Depth Homomorphisms. In *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '20, pages 507–520, New York, NY, USA, 2020. Association for Computing Machinery. event-place: Saarbrücken, Germany. doi:10.1145/3373718.3394739.

14 Martin Grohe. word2vec, node2vec, graph2vec, X2vec: Towards a Theory of Vector Embeddings of Structured Data. In Dan Suciu, Yufei Tao, and Zhewei Wei, editors, *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2020, Portland, OR, USA, June 14-19, 2020*, pages 1–16. ACM, 2020. doi:10.1145/3375395.3387641.

15 Martin Grohe. The Logic of Graph Neural Networks. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, pages 1–17. IEEE, 2021. doi:10.1109/LICS52264.2021.9470677.

16 Martin Grohe, Gaurav Rattan, and Tim Seppelt. Homomorphism Tensors and Linear Equations. In Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff, editors, *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*, volume 229 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 70:1–70:20, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.ICALP.2022.70`.

17 Lauri Hella. Logical Hierarchies in PTIME. *Information and Computation*, 129(1):1–19, August 1996. `doi:10.1006/inco.1996.0070`.

18 Stephan Kreutzer and Sebastian Ordyniak. Digraph decompositions and monotonicity in digraph searching. *Theor. Comput. Sci.*, 412(35):4688–4703, 2011. `doi:10.1016/j.tcs.2011.05.003`.

19 László Lovász. Operations with structures. *Acta Mathematica Academiae Scientiarum Hungarica*, 18(3):321–328, September 1967. `doi:10.1007/BF02280291`.

20 Laura Mančinska and David E. Roberson. Quantum isomorphism is equivalent to equality of homomorphism counts from planar graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 661–672, 2020. `doi:10.1109/FOCS46700.2020.00067`.

21 Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks. *AAAI*, 33:4602–4609, July 2019. `doi:10.1609/aaai.v33i01.33014602`.

22 Daniel Neuen. Homomorphism-Distinguishing Closedness for Graphs of Bounded Tree-Width, April 2023. `doi:10.48550/arXiv.2304.07011`.

23 Hoang Nguyen and Takanori Maehara. Graph homomorphism convolution. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7306–7316. PMLR, 13–18 July 2020. URL: `https://proceedings.mlr.press/v119/nguyen20c.html`.

24 Roman Rabinovich. *Graph complexity measures and monotonicity*. PhD thesis, RWTH Aachen University, 2013. URL: `https://publications.rwth-aachen.de/record/230227`.

25 Gaurav Rattan and Tim Seppelt. Weisfeiler–Leman and Graph Spectra. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2268–2285. Society for Industrial and Applied Mathematics, 2023. `doi:10.1137/1.9781611977554.ch87`.

26 David E. Roberson. Oddomorphisms and homomorphism indistinguishability over graphs of bounded degree, June 2022. `doi:10.48550/arXiv.2206.10321`.

27 David E. Roberson and Tim Seppelt. Lasserre Hierarchy for Graph Isomorphism and Homomorphism Indistinguishability. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 101:1–101:18, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.ICALP.2023.101`.

28 Tim Seppelt. Logical Equivalences, Homomorphism Indistinguishability, and Forbidden Minors. In Jérôme Leroux, Sylvain Lombardy, and David Peleg, editors, *48th International Symposium on Mathematical Foundations of Computer Science (MFCS 2023)*, volume 272 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 82:1–82:15, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.MFCS.2023.82`.

29 Paul D. Seymour and Robin Thomas. Graph Searching and a Min-Max Theorem for Tree-Width. *J. Comb. Theory, Ser. B*, 58(1):22–33, 1993. `doi:10.1006/jctb.1993.1027`.

30 Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*, 2019. URL: `https://openreview.net/forum?id=ryGs6iA5Km`.