The Exponential-Time Complexity of Counting Quantum Graph Homomorphisms

Hubie Chen
Department of Computer Science and Information Systems, Birkbeck, University of London, London, United Kingdom
hubiechen@gmail.com

Radu Curticapean
Basic Algorithms Research Copenhagen (BARC)
IT University of Copenhagen, Copenhagen, Denmark
radu.curticapean@gmail.com

Holger Dell
IT University of Copenhagen, Copenhagen, Denmark
Basic Algorithms Research Copenhagen (BARC)
hold@itu.dk

Abstract

Many graph parameters can be expressed as homomorphism counts to fixed target graphs; this includes the number of independent sets and the number of k-colorings for any fixed k. Dyer and Greenhill (RSA 2000) gave a sweeping complexity dichotomy for such problems, classifying which target graphs render the problem polynomial-time solvable or #P-hard. In this paper, we give a new and shorter proof of this theorem, with previously unknown tight lower bounds under the exponential-time hypothesis. We similarly strengthen complexity dichotomies by Focke, Goldberg, and Živný (SODA 2018) for counting *surjective* homomorphisms to fixed graphs. Both results crucially rely on our main contribution, a complexity dichotomy for evaluating linear combinations of homomorphism numbers to fixed graphs. In the terminology of Lovász (Colloquium Publications 2012), this amounts to counting homomorphisms to *quantum graphs*.

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis; Mathematics of computing → Graph theory

Keywords and phrases graph homomorphisms, exponential-time hypothesis, counting complexity, complexity dichotomy, surjective homomorphisms

Related Version Proceedings version appears at the 45th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2019).

Funding Hubie Chen acknowledges the support of Spanish Project TIN2017-86727-C2-2-R. Radu Curticapean was partly supported by ERC grant SYSTEMATICGRAPH (No. 725978) and VILLUM Foundation grant 16582 while working on this project.

1 Introduction

The classification program in counting complexity strives to identify comprehensive classes of counting problems that are well-behaved enough to allow for exhaustive complexity classifications [16, 9, 7, 4, 6, 5]. Particularly good candidates for such classes are counting variants of the Constraint Satisfaction Problem (#CSP) [4, 3]. In the general #CSP, a problem instance is defined by a set of *variables* $V = \{v_1, \ldots, v_n\}$, each taking values from a *domain* D. The computational task is to determine the number of assignments $a : V \to D$ from variables to domain elements, subject to the requirement that a satisfies a set of *constraints* that are part of the input. Each constraint is applied to a tuple of variables and restricts the admissible assignments to that tuple.

In this full generality, the #CSP framework can easily express #P-hard problems such as counting satisfying assignments to Boolean formulas in CNF, or counting the proper k-colorings of graphs G, for fixed $k \in \mathbf{N}$. For instance, to count k-colorings, interpret the vertices of G as variables over the domain $\{1, \dots, k\}$ and constrain variable pairs corresponding to adjacent vertices to have distinct assignments.

Among other properties, the complexity of #CSP depends on the types of constraints present in the instance. This motivates the study of $\#\mathrm{CSP}(\mathcal{F})$ for fixed constraint sets $\mathcal{F}$, where only instances with constraints from $\mathcal{F}$ are allowed as input. After a wealth of research, a full dichotomy for these problems is known by now: For every finite set $\mathcal{F}$, the problem $\#\mathrm{CSP}(\mathcal{F})$ has been shown to be either polynomial-time solvable or #P-hard, with an explicit decidable dichotomy criterion [5, 17]. Dichotomies are known even in weighted settings [7] that arise in statistical physics in the context of *partition functions*.

1.1 Graph homomorphisms

The full dichotomy for $\#\mathrm{CSP}(\mathcal{F})$ was predated by numerous results for special cases, with a particular focus on *graph homomorphisms* [16, 6, 24]. Given graphs G and H, a homomorphism from G to H is a function $h : V(G) \to V(H)$ such that any edge $uv \in E(G)$ is mapped to an edge $h(u)h(v) \in E(H)$. Homomorphisms from G to H are sometimes also called *H-colorings* of G, since they generalize q-colorings for fixed $q \in \mathbf{N}$ by taking $H = K_q$.

We write $\mathrm{Hom}(G, H)$ for the number of homomorphisms from G to H. For a fixed graph H, the computational problem $\mathrm{Hom}(\star, H)$ asks to compute $\mathrm{Hom}(G, H)$ on input a graph G. This is indeed a particular $\#\mathrm{CSP}(\mathcal{F})$ problem: Viewing $V(G)$ as variables and $V(H)$ as domain, a homomorphism h corresponds to an assignment from variables to domain elements that respects certain constraints on variable pairs: If u and v are connected by an edge in G, then its assignments $h(u)$ and $h(v)$ must be such that $h(u)h(v)$ is an edge of H. Following this interpretation, it can be seen that the class of problems $\mathrm{Hom}(\star, H)$ for fixed H correspond exactly to $\#\mathrm{CSP}(\mathcal{F})$ problems where $\mathcal{F}$ contains only a single constraint, and this constraint depends only (symmetrically) on two variables.

Despite these restrictions, many interesting counting problems on graphs can be expressed as $\mathrm{Hom}(\star, H)$ for suitable choices of H. This includes the number of independent sets in a graph, the number of k-colorings for fixed k, and certain partition functions from statistical physics. In a seminal result, Dyer and Greenhill proved a full classification for the complexity of $\mathrm{Hom}(\star, H)$ when H is an undirected graph that may contain self-loops. In the following, we say that a graph is *reflexive* if every vertex features a self-loop, and we say that it is *irreflexive* if no vertex does. Note that bipartite graphs are irreflexive.

▶ **Theorem 1** (Dyer and Greenhill [16]). *Let H be a fixed undirected graph. If each connected component of H is a complete bipartite graph or a reflexive complete graph, then* $\mathrm{Hom}(\star, H)$ *can be computed in polynomial time. Otherwise the problem is* #P*-hard, even on irreflexive input graphs.*

This exhaustive dichotomy was extended in numerous ways, including a setting where H has edge-weights and the weight of a homomorphism is the product of edge-weights in the image, counted with multiplicities [6, 24, 8]: Given an input graph G, the task is to determine the sum of weights of all homomorphisms from G to H, a quantity that occurs naturally in statistical physics. The case of directed graphs was also fully classified [15, 7]. Furthermore, a variant was investigated that asks to determine the number of homomorphisms modulo a fixed prime [18, 22, 23], but a full dichotomy was not yet obtained for such problems.

Our Contribution: New proof of Theorem 1 with tight lower bound under ETH.

Using techniques originally introduced by Lovász [27], we significantly shorten the proof of Theorem 1. Our new proof also gives tight conditional lower bounds on the running times needed to solve the #P-hard cases: For a k-vertex graph H and an n-vertex graph G, the quantity $\mathrm{Hom}(G, H)$ can be computed in time roughly $O(k^n)$ using exhaustive search. It was shown by Cygan et al. [13] that $\mathrm{Hom}(G, H)$ cannot be computed in time $\exp(o(n \log k))$ when both G and H are input, unless the widely-believed *exponential-time hypothesis* (ETH) by Impagliazzo and Paturi [25] fails. However, this result leaves open the possibility of $\exp(o(n))$-time algorithms for particular fixed graphs H for which $\mathrm{Hom}(\star, H)$ is #P-hard. We rule out such algorithms under ETH. In fact, we only require the counting exponential-time hypothesis #ETH, introduced in [14]. This makes the result slightly stronger, since ETH implies #ETH.

▶ **Theorem 2.** *For every hard graph H in Theorem 1, the problem $\mathrm{Hom}(\star, H)$ cannot be computed in $\exp(o(n))$ time on n-vertex input graphs unless #ETH fails. This holds even for bipartite and irreflexive inputs with $O(n)$ edges.*

1.2 Surjective homomorphisms

Focke, Goldberg, and Živný [20] used Theorem 1 as a starting point to classify the complexity of counting homomorphisms with surjectivity constraints. We call a homomorphism h from G to H *surjective* if its image contains every vertex and every edge of H. That is, for every vertex $v \in V(H)$, the preimage $h^{-1}(v)$ is non-empty, and for every edge $st \in E(H)$, there is at least one edge between the sets $h^{-1}(s)$ and $h^{-1}(t)$ in G. This notion can be relaxed by requiring surjectivity only on a subset of the vertices and edges of H. For instance, *vertex-surjective homomorphisms* only require every vertex to be hit. Likewise, a *compaction* is a vertex-surjective homomorphism from G to H that hits all non-loop edges of H.

The above authors proved a dichotomy theorem for counting vertex-surjective homomorphisms to fixed graphs H [20], discovering that the dichotomy criterion for these problems coincides with that for standard homomorphisms. They proved a similar dichotomy for counting compactions and showed that there are significantly fewer polynomial-time solvable cases.

▶ **Theorem 3** (Focke, Goldberg, and Živný [20]). *Let H be a fixed graph. The problem $\mathrm{VertSurj}(\star, H)$ is polynomial-time solvable if every connected component of H is a complete bipartite graph or a reflexive complete graph. The problem $\mathrm{Comp}(\star, H)$ is polynomial-time solvable if every component of H is an irreflexive star or a reflexive complete graph of size at most two. In all other cases, the problems are #P-hard, even on irreflexive inputs.*

Our Contribution: Simplified and strengthened version of Theorem 3.

We define a problem that jointly generalizes the problems $\mathrm{VertSurj}(\star, H)$ and $\mathrm{Comp}(\star, H)$ in a natural way. To this end, we consider target graphs H in which some edges and vertices of H are *marked*. A *partially surjective homomorphism* then is a homomorphism h whose image includes all marked objects of H; we write $\mathrm{PartSurj}(G, H)$ for their number. With appropriate choices of markings, this can be seen to generalize various quantities, such as homomorphisms, surjective and vertex-surjective homomorphisms, and compactions. We obtain the following complexity dichotomy, from which Theorem 3 easily follows.

▶ **Theorem 4.** *Let H be a graph in which some edges and/or vertices are marked, and let $\mathcal{D}(H)$ be the set of graphs obtainable from H by deleting marked objects.*

- *If every graph in* $\mathcal{D}(H)$ *is a disjoint union of complete bipartite graphs and reflexive complete graphs, then* $\mathrm{PartSurj}(\star, H)$ *is polynomial-time solvable.*
- *Otherwise,* $\mathrm{PartSurj}(\star, H)$ *is* $\#\mathrm{P}$*-hard and cannot be computed in* $\exp(o(n))$ *time on* n*-vertex input graphs unless #ETH fails. This holds even for bipartite and irreflexive inputs with* $O(n)$ *edges.*

1.3 Our techniques: Homomorphisms to quantum graphs

While the class of homomorphism problems $\mathrm{Hom}(\star, H)$ to fixed H subsumes many interesting counting problems for graphs, there are also natural problems that cannot be expressed in this framework. This includes the number of perfect matchings in a graph [21, 28]. To give another example that is more similar to homomorphism counts, recall that counting 3-colorings in a graph is expressible as $\mathrm{Hom}(\star, K_3)$. However, counting surjective 3-colorings (colorings that use all three colors) cannot be expressed as $\mathrm{Hom}(\star, H)$ for a fixed graph H. This is because, for any graph G, the number of surjective 3-colorings is

$$\mathrm{VertSurj}(G, K_3) = \mathrm{Hom}(G, K_3) - 3 \cdot \mathrm{Hom}(G, K_2) + 3 \cdot \mathrm{Hom}(G, K_1)\,. \tag{1}$$

However, the expression of a graph parameter as a linear combination of homomorphism counts $\mathrm{Hom}(\star, H)$ is known to be unique, see [27, Exercise 5.51], ruling out the existence of a graph H with $\mathrm{VertSurj}(\star, K_3) = \mathrm{Hom}(\star, H)$.

More generally, the uniqueness of such expressions implies that closing the class of homomorphism counts under point-wise linear combinations gives a strictly richer class of graph parameters. Following Lovász's terminology [27, Chapter 6], we call these graph parameters homomorphism counts to *quantum graphs*. Here, a quantum graph $\overline{H}$ is a formal linear combination

$$\overline{H} = \sum_{H \in \mathcal{C}} \alpha_H H$$

for a finite set of constituent graphs $\mathcal{C}$ where each $H \in \mathcal{C}$ has an associated coefficient $\alpha_H \in \mathbf{Q}$. The canonical linear extension of homomorphism counts to quantum graphs $\overline{H}$ then reads

$$\mathrm{Hom}(G, \overline{H}) = \sum_{H \in \mathcal{C}} \alpha_H \cdot \mathrm{Hom}(G, H).$$

In other words, every finite (point-wise) linear combination of homomorphism counts to fixed graphs can be expressed as a homomorphism count to a fixed quantum graph. The computational problem $\mathrm{Hom}(\star, \overline{H})$ for fixed $\overline{H}$ is to compute $\mathrm{Hom}(G, \overline{H})$ for a given input G.

As exemplified in (1), problems that do not immediately appear to be linear combinations of homomorphism counts may in fact be expressible in this format. For instance, all partially surjective homomorphism counts can be expressed as linear combinations of ordinary homomorphism counts.

Our Contribution: Dichotomy for homomorphisms to quantum graphs.

We prove that the complexity of counting homomorphisms to fixed graphs enjoys a very favorable monotonicity property. (A similar phenomenon was already observed for linear combinations of homomorphism counts *from* fixed graphs [10, 12].)

Let $\overline{H}$ be a fixed quantum graph that is properly normalized, that is, its constituents are pairwise non-isomorphic and all coefficients are non-zero. Then, for any constituent H of $\overline{H}$, the problem $\mathrm{Hom}(\star, H)$ reduces to $\mathrm{Hom}(\star, \overline{H})$ under polynomial-time Turing reductions.

That is, given access to an oracle that delivers the quantity $\mathrm{Hom}(G', \overline{H})$ on any query G', we can compute $\mathrm{Hom}(G, H)$ for any input graph G and any constituent H of $\overline{H}$. In particular, if $\mathrm{Hom}(\star, H)$ is #P-hard, then any linear combination of homomorphism counts containing the summand $\mathrm{Hom}(\star, H)$ is #P-hard.

Moreover, to determine $\mathrm{Hom}(G, H)$ for an n-vertex graph G, our reduction only needs to query graphs G' with $n + c$ vertices, with c depending only on $\overline{H}$. This makes the reduction very suitable in the exponential-time setting: An algorithm with running time $O(b^n)$ for $\mathrm{Hom}(\star, \overline{H})$ would imply $O(b^n)$ time algorithms for any constituent problem $\mathrm{Hom}(\star, H)$. We use the complexity monotonicity of quantum graphs to obtain our final dichotomy theorem, stated as follows.

▶ **Theorem 5.** *Let $\overline{H} = \sum_{i=1}^{k} \alpha_i H_i$ be a fixed quantum graph, where $H_1, \ldots, H_k$ are fixed pairwise non-isomorphic graphs and $\alpha_1, \ldots, \alpha_k \in \mathbf{Q} \setminus \{0\}$ are fixed.*

- *If the problem $\mathrm{Hom}(\star, H_i)$ can be solved in polynomial time for every $i \in [k]$, then so can $\mathrm{Hom}(\star, \overline{H})$.*
- *If there is some $i \in [k]$ such that $\mathrm{Hom}(\star, H_i)$ is #P-hard, then so is $\mathrm{Hom}(\star, \overline{H})$. In this case, unless #ETH fails, $\mathrm{Hom}(\star, \overline{H})$ cannot be solved in time $\exp(o(n))$, even for bipartite and irreflexive input graphs with $O(n)$ edges.*

The quantum graph $\overline{H}$ in this theorem may have negative coefficients; if $\overline{H}$ has only positive coefficients, the #P-hardness of $\mathrm{Hom}(\star, \overline{H})$ can already be derived from Theorem 1.

Organization of the paper

After introducing notions related to homomorphisms and exponential-time complexity in Section 2, we prove the dichotomy theorem for homomorphisms to quantum graphs (Theorem 5) in Section 3. Using the complexity monotonicity of homomorphism numbers to quantum graphs, we sketch the proof of the exponential-time Dyer–Greenhill theorem (Theorem 2) in Section 4. Finally, we derive the dichotomy for partially surjective homomorphisms (Theorem 4) in Section 5.

Due to lack of space, some proofs are deferred to the full version.

2 Preliminaries

Let $\mathcal{G}$ be the set of all unlabeled and undirected finite graphs. These graphs may have self-loops but no parallel edges. In the remainder of this section, let $G, H \in \mathcal{G}$. We denote the vertex set of G with $V(G)$ and the edge set with $E(G)$.

Homomorphisms and graph algebra:

Let $\mathrm{Hom}(G, H)$ be the number of homomorphisms from G to H, that is, functions $h : V(G) \to V(H)$ such that any edge $uv \in E(G)$ is mapped to an edge $h(u)h(v) \in E(H)$. For fixed H, we write $\mathrm{Hom}(\star, H)$ for the graph parameter that maps input graphs G to $\mathrm{Hom}(G, H)$.

Our proofs rely on a result of Borgs et al. [2, Lemma 4.2], who show that the graph function Hom, when viewed as a matrix, has certain non-singular finite submatrices. We use the following extension, which we derive from the original result in the full version.

▶ **Lemma 6.** *For any set of pairwise non-isomorphic graphs $H_1, \ldots, H_k$, there exist irreflexive graphs $F_1, \ldots, F_k$ such that the $k \times k$ matrix M with $M[i, j] = \mathrm{Hom}(F_i, H_j)$ is invertible.*

Even though $H_1, \dots, H_k$ may feature self-loops, the lemma guarantees the existence of irreflexive graphs $F_1, \dots, F_k$. In fact, these graphs can even be guaranteed to be 3-colorable.

Our proofs also rely upon two binary operations on graphs (which can be viewed as graph products) and their effects on homomorphism counts: The *disjoint union* of graphs, and its "dual", the *tensor product*.

▶ **Definition 7.** *Let A, B be graphs on disjoint vertex sets. The disjoint union $A \cup B$ has vertex set $V(A) \cup V(B)$ and consists of a copy of A and one of B.*

The tensor product $A \otimes B$ is the graph on vertex set $V(A) \times V(B)$ where (u, v) and (u', v') are adjacent if and only if $(u, u') \in E(A)$ and $(v, v') \in E(B)$.

From a matrix perspective, the adjacency matrix of $A \cup B$ is a block matrix with blocks corresponding to A and B, and the adjacency matrix of $A \otimes B$ is the Kronecker product of the respective adjacency matrices. The following identities hold for all vertex-disjoint graphs G, F, A, B:

$$\mathrm{Hom}(G \cup F, A) = \mathrm{Hom}(G, A) \cdot \mathrm{Hom}(F, A)\,, \text{ and} \tag{2}$$

$$\mathrm{Hom}(G, A \otimes B) = \mathrm{Hom}(G, A) \cdot \mathrm{Hom}(G, B)\,. \tag{3}$$

If additionally G is connected, then we also have

$$\mathrm{Hom}(G, A \cup B) = \mathrm{Hom}(G, A) + \mathrm{Hom}(G, B)\,. \tag{4}$$

The proofs are elementary and can be found in [27, (5.28)–(5.30)].

Exponential-time complexity:

The *counting exponential time hypothesis* (#ETH) of Dell et al. [14], adapted from the decision setting of Impagliazzo, Paturi, and Zane [25, 26], asserts that there is no $\exp(o(m))$ time algorithm to count the satisfying assignments of a given 3-CNF formula with m clauses. We use the following stringent type of polynomial-time reduction:

▶ **Definition 8** (Linear Reduction)**.** *Let $f, g : \mathcal{G} \to \mathbf{Q}$ be two graph parameters. We write $f \preceq g$ if there is a polynomial-time Turing reduction from f to g that, on input a graph with m edges, queries only graphs with at most $O(m)$ edges.*

Note that $\preceq$ is a reflexive and transitive relation; it is called *size-preserving* reducibility in [19, p. 422]. If $f \preceq g$, then an algorithm with running time $\exp(o(m))$ for g on m-edge graphs would imply one for f.

3 Counting homomorphisms to quantum graphs

We are ready to prove Theorem 5, the dichotomy for counting homomorphisms to quantum graphs. We establish the theorem via the following proposition on the complexity monotonicity for counting homomorphisms to quantum graphs.

▶ **Proposition 9** (Complexity Monotonicity)**.** *Fix any quantum graph*

$$\overline{H} = \sum_{j=1}^{k} \alpha_j H_j$$

with pairwise non-isomorphic graphs $H_1, \dots, H_k$ and coefficients $\alpha_1, \dots, \alpha_k \in \mathbf{Q} \setminus \{0\}$. For every fixed $j \in [k]$, we then have

$$\mathrm{Hom}(\star, H_j) \preceq \mathrm{Hom}(\star, \overline{H}).$$

Furthermore, if the input graph G for $\mathrm{Hom}(\star, H_j)$ is irreflexive, then all queries for $\mathrm{Hom}(\star, \overline{H})$ are irreflexive as well.

Proof. Without loss of generality, let $j = 1$. By Lemma 6, there exist irreflexive graphs $F_1, \dots, F_k$ such that the matrix M with $M[i, j] = \mathrm{Hom}(F_i, H_j)$ is invertible. On input a graph G, we first construct the graphs $G \cup F_i$ for all $i \in [k]$. By (2), we obtain the following linear equation for every $i \in [k]$:

$$\mathrm{Hom}(G \cup F_i, \overline{H}) = \sum_{j=1}^{k} \alpha_j \,\mathrm{Hom}(G, H_j) \cdot M[i, j]\,. \tag{5}$$

The set of these equations for all $i \in [k]$ forms a linear equation system $b = Mx$, with $b_i = \mathrm{Hom}(G \cup F_i, \overline{H})$ for all $i \in [k]$ and $x_j = \alpha_j \,\mathrm{Hom}(G, H_j)$ for all $j \in [k]$. Thus if G is the input and we wish to compute $\mathrm{Hom}(G, H_1)$ using the oracle for $\mathrm{Hom}(\star, \overline{H})$, we use the following procedure:

1. Compute the vector $b \in \mathbf{Q}^k$ using k queries to $\mathrm{Hom}(\star, \overline{H})$.
2. Output the number $(M^{-1}b)_1/\alpha_1$.

This indeed yields $\mathrm{Hom}(G, H_1)$, because $\alpha_1 \,\mathrm{Hom}(G, H_1) = (M^{-1}b)_1$ and $\alpha_1 \neq 0$ hold. Since $H_1, \dots, H_k$ is fixed, we can hard-code the constants α_j and graphs F_j, for $j \in [k]$, as well as the matrix M^{-1} into the reduction. The reduction itself runs in linear time to prepare the queries $G \cup F_i$. Given as input an m-edge graph G, it only issues queries on graphs with $m + C$ edges, where C is a fixed constant depending only on $\overline{H}$. If G is irreflexive, then so are all query graphs $G \cup F_i$ for $i \in [k]$, since all F_i are irreflexive. ◀

Theorem 5 follows easily from Proposition 9 and Theorem 2.

4 Revisiting the Dyer-Greenhill dichotomy

We outline our new proof of Theorem 1 and classify the complexity of $\mathrm{Hom}(\star, H)$. Our proof also gives a tight lower bound under #ETH, resulting in Theorem 2.

Throughout this section, let us say that a graph H is *hom-easy* if every connected component of H is either a complete bipartite graph $K_{a,b}$ for $a, b \in \mathbf{N}$ or a reflexive complete graph K_q° for $q \in \mathbf{N}$. It is straightforward to check that $\mathrm{Hom}(\star, H)$ can be solved in linear time if H is a hom-easy graph. If H is not hom-easy, we call H *hom-hard.* In the remainder of the section, we show how to establish the #P-hardness of $\mathrm{Hom}(\star, H)$ for hom-hard graphs H in three steps.

Step 1: Ensuring bipartiteness

Rather than working directly with H, we proceed to its *bipartite double cover* $H \otimes K_2$. Recall from (3) that

$$\mathrm{Hom}(G, H \otimes K_2) = \mathrm{Hom}(G, H) \cdot \mathrm{Hom}(G, K_2)$$

holds for all graphs G. Since K_2 is hom-easy, we can compute $\mathrm{Hom}(G, K_2)$ in linear time, and this readily implies

$$\mathrm{Hom}(\star, H \otimes K_2) \preceq \mathrm{Hom}(\star, H)\,. \tag{6}$$

Hence, in order to establish hardness of $\mathrm{Hom}(\star, H)$, it suffices to establish hardness of $\mathrm{Hom}(\star, H \otimes K_2)$ for the bipartite graph $H \otimes K_2$. Note that $H \otimes K_2$ is hom-hard if H is hom-hard. We remark that (6) is a significant shortcut compared to the original proof of Dyer and Greenhill [16], a large part of which dealt with non-bipartite graphs H.

Step 2: Isolating 2-neighborhoods

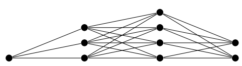
Similar to the original proof [16], we successively isolate induced subgraphs from $H \otimes K_2$ until reaching a hard base case. We provide the details of this step in Section 4.1.

Given a bipartite graph B and $v \in V(B)$, let B_v denote the subgraph induced by vertices of distance at most 2 from v. We show $\mathrm{Hom}(\star, B_v) \preceq \mathrm{Hom}(\star, B)$ for all $v \in V(B)$ by using the monotonicity for quantum graph homomorphisms. This reduction may happen to be useless for some vertices $v \in V(B)$, as B_v may be a complete bipartite graph $K_{a,b}$ or B itself. If this holds for all $v \in V(B)$, we call B an *impasse*.

Starting at $B = H \otimes K_2$, we repeatedly pick a vertex $v \in V(B)$ and set $B := B_v$ until reaching an impasse P. We show that the vertices in the above process can be chosen to ensure that P is not a $K_{a,b}$. Since $\mathrm{Hom}(\star, P) \preceq \mathrm{Hom}(\star, H \otimes K_2)$ follows, it remains to prove hardness for this impasse P.

Step 3: Exploded four-vertex paths

A structural argument shows that any impasse P that is not a $K_{a,b}$ is in fact a 4-vertex path $P(a_1, a_2, a_3, a_4)$ in which the i-th vertex is replaced by a positive number a_i of clones. For example, $P(1, 3, 4, 2)$ is the following graph:



In Section **??**, we prove the hardness of $\mathrm{Hom}(\star, P)$ with $P = P(a_1, a_2, a_3, a_4)$ for arbitrary fixed integers a_1, a_2, a_3, a_4 The reduction starts from the #P-hard problem of counting independent sets, for which #ETH rules out $2^{o(m)}$ time algorithms [11]. In the special case $P = P(1, 1, 1, 1)$, a simple reduction is possible: Since $P = \otimes$ holds, (3) implies

$$\mathrm{Hom}(G, P) = \mathrm{Hom}(G, \;) \cdot \mathrm{Hom}(G, \;)\,. \tag{7}$$

Since $\mathrm{Hom}(G, \;)$ counts precisely the independent sets of G, and $\mathrm{Hom}(G, \;)$ is non-zero and can be computed in linear time, the reduction $\mathrm{Hom}(\star, \;) \preceq \mathrm{Hom}(\star, P)$ is immediate.

Putting the steps together

Given a hom-hard graph H, the three steps outlined above identify a graph $P = P(a_1, a_2, a_3, a_4)$ for $a_1, a_2, a_3, a_4 \in \mathbf{N}$ such that

$$\mathrm{Hom}(\star, P) \preceq \ldots \preceq \mathrm{Hom}(\star, H \otimes K_2) \preceq \mathrm{Hom}(\star, H). \tag{8}$$

By establishing hardness of $\mathrm{Hom}(\star, P)$, we thus prove hardness of $\mathrm{Hom}(\star, H)$.

4.1 Details of Step 2: Successively isolating 2-neighborhoods

Let B be a bipartite, hom-hard graph, initially $B = H \otimes K_2$. Since B is hom-hard, it does not contain a connected component that is a complete bipartite graph. We find a hom-hard impasse P with $\mathrm{Hom}(\star, P) \preceq \mathrm{Hom}(\star, H \otimes K_2)$ by transitioning successively to proper induced subgraphs of B, in a manner similar to the bipartite case of [16, Theorem 1.1]. In the following, we describe one step of this process.

For any vertex $v \in V(B)$, recall that B_v is the subgraph of B induced by vertices at distance at most 2 from v. We prove $\mathrm{Hom}(\star, B_v) \preceq \mathrm{Hom}(\star, B)$. To this end, we first show in Lemma 10 how to compute the sum $\sum_{v\in V(B)} \mathrm{Hom}(G, B_v)$ on input G with an oracle for $\mathrm{Hom}(\star, B)$. Combining this with Proposition 9, we will then extract $\mathrm{Hom}(G, B_v)$ for any fixed vertex $v \in V(B)$ from the sum in Proposition 11.

▶ **Lemma 10.** *Let B be a bipartite graph and let G be a connected bipartite graph with bipartition $V(G) = L \cup R$. Let G^a_L be derived from G by adding an "apex" vertex a that is adjacent to all of R, and let G^a_R be derived by adding an apex vertex a adjacent to all of L. Then*

$$\mathrm{Hom}(G^a_L, B) + \mathrm{Hom}(G^a_R, B) = \sum_{v\in V(B)} \mathrm{Hom}(G, B_v)\,. \tag{9}$$

Proof. For any $v \in V(B)$, we write $\mathrm{Hom}(G^a_L, B \mid a \to v)$ for the number of homomorphisms from G^a_L to B that map a to v, with an analogous definition for G^a_R. We observe that

$$\mathrm{Hom}(G^a_L, B) + \mathrm{Hom}(G^a_R, B) = \sum_{v\in V(B)} \mathrm{Hom}(G^a_L, B \mid a \to v) + \mathrm{Hom}(G^a_R, B \mid a \to v)\,, \tag{10}$$

because the set of homomorphisms h from G^a_L to B can be partitioned according to the image $h(a) = v$ and the same applies to homomorphisms from G^a_R. In the remainder of the proof, we establish that, for all $v \in V(B)$,

$$\mathrm{Hom}(G^a_L, B \mid a \to v) + \mathrm{Hom}(G^a_R, B \mid a \to v) = \mathrm{Hom}(G, B_v). \tag{11}$$

Together with (10), this implies (9). To prove (11), fix any vertex $v \in V(B)$. We say that a homomorphism h from G^a_L or G^a_R to B is an *extension* of a homomorphism g from G to B_v if h agrees with g on all of $V(G)$, and h also maps the additional vertex a in G^a_L or G^a_R to v.

We first claim that any homomorphism h from G^a_L or G^a_R to B with $h(a) = v$ is an extension of some homomorphism g from G to B_v. Secondly, we claim that for any homomorphism g from G to B_v, there is precisely one homomorphism h from either G^a_L or G^a_R to B that is an extension of g. Then (11) follows.

For the first claim, let h be a homomorphism from G^a_L to B with $h(a) = v$. (The argument for homomorphisms from G^a_R is analogous.) Then h maps R to the neighborhood of v in B: Since a has edges to all of R in G^a_L, there must be edges from $h(a) = v$ to all of $h(R)$ in B. Furthermore, since G is connected, $h(L)$ is contained in the neighborhood of $h(R)$. It follows that the entire image of h is contained in B_v, so the restriction g of h to $V(G)$ is a homomorphism from G to B_v. Hence h is an extension of g, proving the first claim.

For the second claim, let X be the bipartition side of B_v not containing v. Consider a homomorphism g from G to B_v. Since G is connected, either g maps R to X, or g maps L to X.

1. In the first case, we can extend g to a map h from G^a_L to B via $h(a) = v$, and we show that h is indeed a homomorphism: By definition, g preserves edges on G and the image of G is the subgraph B_v of B. Since g maps R to X, and X is the neighborhood of v in B_v by definition of B_v, we see that h maps the edges aw for $w \in R$ in G^a_L to edges of B_v. Thus h is an extension of g. Furthermore, the map h' from G^a_R to B obtained from g by setting $h'(a) = v$ is not a homomorphism, since v and R are all mapped to X, which is an independent set in B_v.

2. In the second case, we can extend g to a homomorphism h from G^a_R to B as above. By a symmetric argument, h maps the edges aw for $w \in L$ in G^a_R to edges of B_v. Thus h is an extension of g.

Hence, the homomorphisms h from G^a_L and G^a_R to B are extensions of homomorphisms g from G to B_v, and each g has precisely one extension from either G^a_L or G^a_R. This establishes (11), thus concluding the proof. ◀

With Lemma 10 at hand, we can readily reduce $\mathrm{Hom}(\star, B_v)$ to $\mathrm{Hom}(\star, B)$.

▶ **Proposition 11.** *For every bipartite graph B and every vertex $v \in V(B)$, we have $\mathrm{Hom}(\star, B_v) \preceq \mathrm{Hom}(\star, B)$.*

Proof. Let G be the input for $\mathrm{Hom}(\star, B_v)$. Without loss of generality, we can assume that G is connected and bipartite with $V(G) = L \cup R$.

Let G^a_L and G^a_R be the graphs derived from G in Lemma 10. Both have $O(n)$ vertices and $O(n+m)$ edges, with $n = |V(G)|$ and $m = |E(G)|$. By (9),

$$\mathrm{Hom}(G^a_L, B) + \mathrm{Hom}(G^a_R, B) = \sum_{v \in V(B)} \mathrm{Hom}(G, B_v)\,.$$

We can compute the left-hand side with an oracle for $\mathrm{Hom}(\star, B)$. On the right-hand side, no graphs cancel when collecting terms for isomorphic graphs B_v, as all coefficients in the sum are 1. Since B is fixed, all graphs and coefficients are fixed, and Proposition 9 gives $\mathrm{Hom}(\star, B_v) \preceq \mathrm{Hom}(\star, B)$. ◀

We establish the hardness of $\mathrm{Hom}(\star, B)$ by reduction from $\mathrm{Hom}(\star, B_v)$ for some $v \in V(B)$ such that B_v is hom-hard and has fewer vertices than B. This is possible unless B is an impasse or hom-easy. We prove that every connected hom-hard impasse is an exploded 4-vertex path. Since B_v is connected, this implies that disconnected hom-hard graphs cannot be an impasse, because we can transition to a hom-hard connected component in this case.

▶ **Lemma 12.** *Let B be a connected bipartite graph, but not a complete bipartite graph. If for every $v \in V(B)$ the graph B_v is either complete bipartite or equal to B, then B is isomorphic to $P(a_1, a_2, a_3, a_4)$ for positive integers a_1, a_2, a_3, a_4.*

Proof. Let $V(B) = L \cup R$ be a bipartition of B. We define:
- S as the set of all $v \in V(B)$ such that B_v is equal to B.
- T as the set of all $v \in V(B)$ such that B_v is a complete bipartite graph.

The sets S and T partition $V(B)$ by the assumptions on B. Let $A_1 = L \cap T$; $A_2 = R \cap S$; $A_3 = L \cap S$; $A_4 = R \cap T$. We claim that these sets witness that B is an exploded 4-vertex path $P(|A_1|, |A_2|, |A_3|, |A_4|)$.

By definition of L and R, each of these four sets are independent in B and there are no edges between A_1 and A_3 or between A_2 and A_4. There are also no edges between A_1 and A_4: To see this, suppose for contradiction that $uv \in A_1 \times A_4$ is an edge of B. Thus we have $u \in N(v)$. Since B_v is the complete bipartite graph induced by $N(v) \subseteq L$ and $N(N(v)) \subseteq R$, the vertex u is adjacent to all vertices of $N(N(v))$. By the symmetric argument for B_u, the vertex v is adjacent to all vertices of $N(N(u))$. Together, this implies $N(v) = N(N(N(v)))$ and $N(u) = N(N(N(u)))$, and thus by connectivity of B that $N(v) = L$ and $N(u) = R$. This contradicts with the assumption that B is not a complete bipartite graph, thus B cannot contain an edge between A_1 and A_4.

By definition of S, all vertices of A_2 are adjacent to all vertices of L and all vertices of A_3 are adjacent to all vertices of R. Thus $B[A_1, A_2]$, $B[A_2, A_3]$, and $B[A_3, A_4]$ are complete bipartite graphs. This establishes that B is isomorphic to an exploded 4-vertex path $P(|A_1|, |A_2|, |A_3|, |A_4|)$. If at least one of the sets $A_1, \ldots, A_4$ were empty, then B would be isomorphic to a complete bipartite graph. Since this is not the case, all sets $A_1, \ldots, A_4$ are non-empty. This concludes the proof of the claim. ◀

4.2 Putting the steps together

Proof of Theorem 2. If H is a reflexive complete graph or a complete bipartite graph, then $\mathrm{Hom}(G, H)$ can be easily computed in polynomial time: If H is a reflexive complete graph, then $\mathrm{Hom}(G, H) = |V(H)|^{|V(G)|}$ holds. If H is a complete bipartite graph $K_{a,b}$, then $\mathrm{Hom}(G, H) = 0$ if G is not bipartite. Moreover, if $G = (U, V, E)$ is bipartite and connected, then $\mathrm{Hom}(G, H) = a^{|U|}b^{|V|} + a^{|V|}b^{|U|}$ holds and is easy to compute as well. Disjoint unions of such graphs can be handled by (2) and (4).

Otherwise, assume H is a hom-hard graph. Then $B := H \otimes K_2$ is a bipartite hom-hard graph and we have $\mathrm{Hom}(\star, H \otimes K_2) \preceq \mathrm{Hom}(\star, H)$ by (6). Now continue replacing B with a hom-hard graph B_v for $v \in V(B)$ until an impasse P is reached. By successive applications of Proposition 11, we have $\mathrm{Hom}(\star, P) \preceq \mathrm{Hom}(\star, H \otimes K_2)$. By Lemma 12, the graph P is a non-trivial exploded 4-vertex path. The problem $\mathrm{Hom}(\star, P)$ is #P-hard and hard under #ETH, as we show in the full version of this paper. These hardness results transfer under $\preceq$-reductions to $\mathrm{Hom}(\star, H)$, which proves the claim. ◀

5 Counting partially surjective homomorphisms

Finally, we prove a dichotomy for $\mathrm{PartSurj}(\star, H)$, thus establishing Theorem 4. For a fixed graph H with marked vertices and edges, let $\mathcal{D}(H)$ denote the set of graphs obtainable from H by deleting marked objects. We first show in Lemma 13 that $\mathrm{PartSurj}(\star, H)$ can be expressed as a linear combination of functions $\mathrm{Hom}(\star, F)$ for $F \in \mathcal{D}(H)$. Then we apply Theorem 5 to classify the complexity of these linear combinations.

▶ **Lemma 13.** *For every graph H with markings, there is a quantum graph $\overline{F} = \sum_{F \in \mathcal{D}(H)} \alpha_F F$ such that $\mathrm{PartSurj}(G, H) = \mathrm{Hom}(G, \overline{F})$ holds for all graphs G. After collecting for isomorphic graphs, we have $\alpha_H = 1$ and $\alpha_F < 0$ for every graph $F \in \mathcal{D}(H)$ obtained by deleting at most one marked edge from H.*

Proof of Lemma 13. Let G be a graph and H be a graph with possible markings. We first express $\mathrm{Hom}(\star, H)$ as a linear combination of functions $\mathrm{PartSurj}(\star, F)$:

$$\mathrm{Hom}(G, H) = \sum_F \mathrm{PartSurj}(G, F)\,, \tag{12}$$

where the sum is over all distinct labeled graphs F that can be obtained by deleting marked vertices or edges of H, and these graphs F inherit the markings from H.

To prove (12), note that the left side of (12) counts all homomorphisms h from G to H. We group them according to which marked objects of H they hit. Let $M \subseteq V(H) \cup E(H)$ be the set of marked objects. For a set $S \subseteq M$, we say that h *has type* S if the image of h contains S and is disjoint from $M \setminus S$. Every homomorphism has exactly one type. Note that $uv \in S$ implies $\{u, v\} \subseteq S$, because h is a homomorphism. Thus removing $M \setminus S$ from both M and H yields a possibly-marked graph $H - (M \setminus S)$. Moreover, $\mathrm{PartSurj}(G, H - (M \setminus S))$ is exactly the number of homomorphisms of type S by definition of PartSurj. This proves (12).

Next, we "invert" (12). We have

$$\mathrm{PartSurj}(G,H) = \mathrm{Hom}(G,H) - \sum_{F<H} \mathrm{PartSurj}(G,F)\,, \tag{13}$$

where the sum is over all labeled subgraphs F of H that are distinct from H and obtained by deleting some marked vertices and edges. Since the graphs F are strictly smaller than H, this inductively proves that $\mathrm{PartSurj}(G,H)$ can be written as a linear combination:

$$\mathrm{PartSurj}(G,H) = \sum_{F} \alpha_F \,\mathrm{Hom}(G,F)\,. \tag{14}$$

It is immediate that $\alpha_H = 1$ and $\alpha_F < 0$ holds for all F obtained by deleting a single marked edge, since only the first step in the induction contributes to these coefficients. ◀

Using Lemma 13 in combination with Theorem 5, we obtain the classification for partially surjective homomorphisms.

Proof of Theorem 4. By Lemma 13, there exists a quantum graph $\overline{F}$ with constituents from $\mathcal{D}(H)$ such that $\mathrm{PartSurj}(G,H) = \mathrm{Hom}(G,\overline{F})$. It follows that $\mathrm{PartSurj}(\star,H)$ and $\mathrm{Hom}(\star,\overline{F})$ are the same problem.

Recall the notions of hom-easy and hom-hard graphs from Section 4. If every graph $F \in \mathcal{D}(H)$ is hom-easy, then $\mathrm{Hom}(\star,\overline{F})$ is polynomial-time solvable. Otherwise, there are hom-hard graphs $F \in \mathcal{D}(H)$, and it only remains to find one with $\alpha_F \neq 0$ in order for Theorem 5 to yield the hardness of $\mathrm{Hom}(\star,\overline{F})$.

If H itself is hom-hard, then we pick $F = H$ and obtain $\alpha_F \neq 0$ by Lemma 13. Otherwise, H is hom-easy, so every connected component of H is a K_q° or a $K_{a,b}$. We check that only one marked edge e^* needs to be deleted from H to obtain a hom-hard graph $F \in \mathcal{D}(H)$:

- If H contains a component C with marked edges and $C = K_q^\circ$ for $q \geq 3$ or $C = K_{a,b}$ for $a,b > 1$, we can choose e^* to be any marked edge in C.
- If H contains a component $C = K_2^\circ$ with at least one marked self-loop, we can choose e^* to be any marked self-loop in C.

If neither of these conditions applies to H, then it can be checked that $\mathcal{D}(H)$ contains only hom-easy graphs. Thus, if $\mathcal{D}(H)$ contains any hom-hard graphs at all, then there is an edge e^* such that $F = H - e^*$ is hom-hard. Lemma 13 then implies $\alpha_F \neq 0$, so Theorem 5 gives hardness of $\mathrm{Hom}(\star,\overline{F})$. ◀

Now Theorem 3 can be easily rederived from Theorem 4 as follows.

Proof of Theorem 3. We have $\mathrm{VertSurj}(\star,H) = \mathrm{PartSurj}(\star,H)$ where all vertices of H are marked but none of its edges. If H is a disjoint union of complete bipartite graphs and reflexive complete graphs, then all of its induced subgraphs are also of this type. Thus $\mathrm{VertSurj}(\star,H)$ is polynomial-time computable by Theorem 4. On the other hand, if H is not of this type, then $\mathrm{VertSurj}(\star,H)$ is #P-hard by Theorem 4.

We have $\mathrm{Comp}(\star,H) = \mathrm{PartSurj}(\star,H)$ where all vertices and non-loop edges of H are marked. If H is a disjoint union of irreflexive stars and reflexive complete graphs of size at most two, then deleting vertices or non-loop edges of H again yields a graph of this type. Thus $\mathrm{Comp}(\star,H)$ is polynomial-time computable by Theorem 4. On the other hand, if H is not of this type, then a non-loop edge can be deleted to obtain a graph H' that contains a connected component that is neither a reflexive clique nor an irreflexive biclique. But then $\mathrm{Comp}(\star,H)$ is #P-hard by Theorem 4. ◀

6 Conclusion

We consider Theorem 2 as an initial step towards a fine-grained understanding of general #CSP problems, and we believe that our shortened proof can be used to simplify and strengthen other dichotomy results for #CSP following in the wake of Dyer and Greenhill's seminal result [16]. Techniques based on quantum graphs might also advance the state of the art for open problems regarding approximate and modular homomorphism counting.

An interesting open problem is to improve Theorem 2 to more precise running time bounds under the strong exponential-time hypothesis. Doing so however is challenging, as non-trivial improvements upon the running time $O(k^n)$ are possible for some #P-hard patterns H. For example, Björklund et al. [1] prove that the number of proper k-colorings, which is equal to $\mathrm{Hom}(G, K_k)$, can be computed in time $2^n \cdot n^{O(1)}$ for any $k \in \mathbf{N}$.

References

1. Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Computing the Tutte polynomial in vertex-exponential time. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 677–686. IEEE Computer Society, 2008. doi:10.1109/FOCS.2008.40.
2. Christian Borgs, Jennifer T. Chayes, Jeff Kahn, and László Lovász. Left and right convergence of graphs with bounded degree. *Random Struct. Algorithms*, 42(1):1–28, 2013. doi:10.1002/rsa.20414.
3. Andrei A. Bulatov. The complexity of the counting constraint satisfaction problem. In *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Track A: Algorithms, Automata, Complexity, and Games*, pages 646–661, 2008. doi:10.1007/978-3-540-70575-8_53.
4. Andrei A. Bulatov and Víctor Dalmau. Towards a dichotomy theorem for the counting constraint satisfaction problem. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 562–571, 2003. doi:10.1109/SFCS.2003.1238229.
5. Andrei A. Bulatov, Martin E. Dyer, Leslie Ann Goldberg, Markus Jalsenius, Mark Jerrum, and David Richerby. The complexity of weighted and unweighted #CSP. *J. Comput. Syst. Sci.*, 78(2):681–688, 2012. doi:10.1016/j.jcss.2011.12.002.
6. Andrei A. Bulatov and Martin Grohe. The complexity of partition functions. *Theor. Comput. Sci.*, 348(2-3):148–186, 2005. doi:10.1016/j.tcs.2005.09.011.
7. Jin-Yi Cai and Xi Chen. Complexity of counting CSP with complex weights. *J. ACM*, 64(3):19:1–19:39, 2017. doi:10.1145/2822891.
8. Jin-Yi Cai, Xi Chen, and Pinyan Lu. Graph homomorphisms with complex values: A dichotomy theorem. *SIAM J. Comput.*, 42(3):924–1029, 2013. doi:10.1137/110840194.
9. Jin-Yi Cai, Pinyan Lu, and Mingji Xia. The complexity of complex weighted Boolean #CSP. *J. Comput. Syst. Sci.*, 80(1):217–236, 2014. doi:10.1016/j.jcss.2013.07.003.
10. Hubie Chen and Stefan Mengel. Counting answers to existential positive queries: A complexity classification. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, pages 315–326, 2016. doi:10.1145/2902251.2902279.
11. Radu Curticapean. Block interpolation: A framework for tight exponential-time counting complexity. *Inf. Comput.*, 261(Part):265–280, 2018. doi:10.1016/j.ic.2018.02.008.
12. Radu Curticapean, Holger Dell, and Dániel Marx. Homomorphisms are a good basis for counting small subgraphs. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 210–223. ACM, 2017. doi:10.1145/3055399.3055502.

13 Marek Cygan, Fedor V. Fomin, Alexander Golovnev, Alexander S. Kulikov, Ivan Mihajlin, Jakub Pachocki, and Arkadiusz Socala. Tight bounds for graph homomorphism and subgraph isomorphism. In *SODA*, pages 1643–1649. SIAM, 2016. doi:10.1137/1.9781611974331.ch112.
14 Holger Dell, Thore Husfeldt, Dániel Marx, Nina Taslaman, and Martin Wahlen. Exponential time complexity of the permanent and the Tutte polynomial. *ACM Trans. Algorithms*, 10(4):21:1–21:32, 2014. doi:10.1145/2635812.
15 Martin E. Dyer, Leslie Ann Goldberg, and Mike Paterson. On counting homomorphisms to directed acyclic graphs. *J. ACM*, 54(6):27, 2007. doi:10.1145/1314690.1314691.
16 Martin E. Dyer and Catherine S. Greenhill. The complexity of counting graph homomorphisms. *Random Struct. Algorithms*, 17(3-4):260–289, 2000. doi:10.1002/1098-2418(200010/12)17:3/4<260::AID-RSA5>3.0.CO;2-W.
17 Martin E. Dyer and David Richerby. An effective dichotomy for the counting constraint satisfaction problem. *SIAM J. Comput.*, 42(3):1245–1274, 2013. doi:10.1137/100811258.
18 John Faben and Mark Jerrum. The complexity of parity graph homomorphism: An initial investigation. *Theory of Computing*, 11:35–57, 2015. doi:10.4086/toc.2015.v011a002.
19 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory (Texts in Theoretical Computer Science. An EATCS Series).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. doi:10.1007/3-540-29953-X.
20 Jacob Focke, Leslie Ann Goldberg, and Stanislav Živný. The complexity of counting surjective homomorphisms and compactions. In Artur Czumaj, editor, *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 1772–1781. SIAM, 2018. doi:10.1137/1.9781611975031.116.
21 Michael Freedman, László Lovász, and Alexander Schrijver. Reflection positivity, rank connectivity, and homomorphism of graphs. *Journal of the American Mathematical Society*, 20(1):37–51, 2007. doi:10.1090/S0894-0347-06-00529-7.
22 Andreas Göbel, Leslie Ann Goldberg, and David Richerby. The complexity of counting homomorphisms to cactus graphs modulo 2. *TOCT*, 6(4):17:1–17:29, 2014. doi:10.1145/2635825.
23 Andreas Göbel, Leslie Ann Goldberg, and David Richerby. Counting homomorphisms to square-free graphs, modulo 2. *TOCT*, 8(3):12:1–12:29, 2016. doi:10.1145/2898441.
24 Leslie Ann Goldberg, Martin Grohe, Mark Jerrum, and Marc Thurley. A complexity dichotomy for partition functions with mixed signs. *SIAM J. Comput.*, 39(7):3336–3402, 2010. doi:10.1137/090757496.
25 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.
26 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.
27 László Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications*. American Mathematical Society, 2012. URL: https://www.ams.org/bookstore-getitem/item=COLL-60.
28 Alexander Schrijver. Graph invariants in the edge model. In Martin Grötschel, Gyula O. H. Katona, and Gábor Sági, editors, *Building Bridges: Between Mathematics and Computer Science*, pages 487–498. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. doi:10.1007/978-3-540-85221-6_16.

A Homomorphisms from loop-free to looped graphs

In this section of the appendix, we prove Lemma 6, building upon a result of Borgs et al. [2, Lemma 4.2]. We use the following formulation for the proof, which is analogous to [27, Proposition 5.44.b] with rows and columns exchanged, see also [27, Proposition 5.43].

▶ **Lemma 14.** *If $H_1, \ldots, H_k$ are pairwise non-isomorphic irreflexive graphs, then there exist irreflexive graphs $F_1, \ldots, F_k$ such that the matrix M with $M[i,j] = \mathrm{Hom}(F_i, H_j)$ is invertible.*

Note that in this setting, the graphs on both left-hand side *and* right-hand side are required to be irreflexive. We will show later how to drop the requirement on the right-hand side. Lemma 14 can be obtained as a corollary to Lemma 4.2 in [2] as follows.

Proof of Lemma 14. Let $\mathcal{H} = \{H_1, \ldots, H_k\}$ and let $\mathcal{S} \supseteq \mathcal{H}$ be some finite set of pairwise non-isomorphic irreflexive graphs that is closed under surjective homomorphisms. Lemma 4.2 in [2] now states that the $(|\mathcal{S}| \times |\mathcal{S}|)$-matrix $\mathrm{Hom}(\mathcal{S}, \mathcal{S})$ is invertible. Thus the $(|\mathcal{S}| \times k)$-submatrix $\mathrm{Hom}(\mathcal{S}, \mathcal{H})$ has full rank k, and there exists a size-k subset $\mathcal{F} = \{F_1, \ldots, F_k\} \subseteq \mathcal{S}$ such that the $(k \times k)$-matrix $\mathrm{Hom}(\mathcal{F}, \mathcal{H})$ is invertible. This is exactly the matrix M, and the claim follows. ◀

▶ Remark 15. In general, it is not possible to choose $\{F_1, \ldots, F_k\} = \{H_1, \ldots, H_k\}$ in Lemma 14, see [27, Exercise 5.46].

We now extend Lemma 14 to the asymmetric situation where the graphs H on the right may have loops, but the graphs F on the left must not. That is, the homomorphism vector $(\mathrm{Hom}(F, H))_F$ determines the isomorphism type of H even when F ranges only over graphs that do not have loops. As in Lemma 14, we in fact show these vectors to be linearly independent. We remark that the reverse statement is false, since $\mathrm{Hom}(F, H) = 0$ holds whenever F contains a loop while H does not.

For our proof, we use a property of the tensor product of graphs, as defined in Definition 7. Namely, while two isomorphic graphs H, H' lead to isomorphic graphs $H \otimes A$ and $H' \otimes A$ for any graph A, the converse is not generally true. However, if A is not bipartite (and in particular, if A is a triangle), then the converse does hold, see [27, Theorem 5.37].

▶ **Lemma 16.** *If H, H' are non-isomorphic unweighted graphs that may contain self-loops, then $H \otimes K_3$ and $H' \otimes K_3$ are non-isomorphic as well.*

With these preliminaries at hand, we can prove Lemma 6.

Proof of Lemma 6. Let $H_1, \ldots, H_k$ be pairwise non-isomorphic graphs that may contain loops. We must show that there exist irreflexive graphs $F_1, \ldots, F_k$ such that the $(k \times k)$-matrix A with $A[i, j] = \mathrm{Hom}(F_i, H_j)$ is invertible.

By Lemma 16, the graphs $H_1 \otimes K_3, \ldots, H_k \otimes K_3$ are pairwise non-isomorphic. Moreover, they are irreflexive, since K_3 is irreflexive. Since the graphs $H_j \otimes K_3$ are pairwise non-isomorphic irreflexive graphs, their Hom-columns over irreflexive graphs are linearly independent by Lemma 14. More precisely, there are irreflexive graphs $F_1, \ldots, F_k$ such that the $(k \times k)$-matrix M with $M[i, j] = \mathrm{Hom}(F_i, H_j \otimes K_3)$ is invertible. Note that $M[i, j] = A[i, j] \cdot \mathrm{Hom}(F_i, K_3)$ holds by (3). Since M is invertible, it does not have any all-0 rows, and in particular $\mathrm{Hom}(F_i, K_3) \neq 0$ holds for all $i \in \{1, \ldots, k\}$. Since A is obtained from the invertible matrix M by multiplying each row with a non-zero scalar $1/\mathrm{Hom}(F_i, K_3)$, the matrix A is invertible as well. ◀