Locally Constrained Homomorphisms on Graphs of Bounded Treewidth and Bounded Degree*

Steven Chaplick[1,**], Jiří Fiala[1,***], Pim van 't Hof[2],
Daniël Paulusma[3], and Marek Tesař[1]

[1] Department of Applied Mathematics, Charles University, Prague, Czech Republic
{chaplick,fiala,tesar}@kam.mff.cuni.cz
[2] Department of Informatics, University of Bergen, Norway
pim.vanthof@ii.uib.no
[3] School of Engineering and Computing Sciences, Durham University, UK
daniel.paulusma@durham.ac.uk

Abstract. A homomorphism from a graph G to a graph H is locally bijective, surjective, or injective if its restriction to the neighborhood of every vertex of G is bijective, surjective, or injective, respectively. We prove that the problems of testing whether a given graph G allows a homomorphism to a given graph H that is locally bijective, surjective, or injective, respectively, are NP-complete, even when G has pathwidth at most 5, 4 or 2, respectively, or when both G and H have maximum degree 3. We complement these hardness results by showing that the three problems are polynomial-time solvable if G has bounded treewidth and in addition G or H has bounded maximum degree.

1 Introduction

All graphs considered in this paper are finite, undirected, and have neither self-loops nor multiple edges. A *graph homomorphism* from a graph $G = (V_G, E_G)$ to a graph $H = (V_H, E_H)$ is a mapping $\varphi : V_G \to V_H$ that maps adjacent vertices of G to adjacent vertices of H, i.e., $\varphi(u)\varphi(v) \in E_H$ whenever $uv \in E_G$. The notion of a graph homomorphism is well studied in the literature due to its many practical and theoretical applications; we refer to the textbook of Hell and Nešetřil [20] for a survey.

We write $G \to H$ to indicate the existence of a homomorphism from G to H. We call G the *guest graph* and H the *host graph*. We denote the vertices of H by $1, \ldots, |H|$ and call them *colors*. The reason for doing this is that graph homomorphisms generalize graph colorings: there exists a homomorphism from

* This paper is supported by the Natural Sciences Engineering Research Council of Canada (NSERC), the Research Council of Norway (197548/F20), EPSRC (EP/G043434/1) and the Royal Society (JP100692).

** Supported by the ESF GraDR EUROGIGA grant as project GACR GIG/11/E023 and the NSERC grants of: K. Cameron and C. Hoàng (Wilfrid Laurier University), D. Corneil (University of Toronto), and P. Hell (Simon Fraser University).

*** Supported by MŠMT ČR grant LH12095 and GAČR grant P202/12/G061.

a graph G to a complete graph on k vertices if and only if G is k-colorable. The problem of testing whether $G \to H$ for two given graphs G and H is called the Hom problem. If only the guest graph is part of the input and the host graph is *fixed*, i.e., not part of the input, then this problem is denoted as H-Hom. The classical result in this area is the Hell-Nešetřil dichotomy theorem which states that H-Hom is solvable in polynomial time if H is bipartite, and NP-complete otherwise [19].

We consider so-called *locally constrained* homomorphisms. The *neighborhood* of a vertex u in a graph G is denoted $N_G(u) = \{v \in V_G \mid uv \in E_G\}$. If for every $u \in V_G$ the restriction of φ to the neighborhood of u, i.e., the mapping $\varphi_u : N_G(u) \to N_H(\varphi(u))$, is injective, bijective, or surjective, then φ is said to be *locally injective*, *locally bijective*, or *locally surjective*, respectively. Locally bijective homomorphisms are also called *graph coverings*. They originate from topological graph theory [3,26] and have applications in distributed computing [1,2,5] and in constructing highly transitive regular graphs [4]. Locally injective homomorphisms are also called *partial graph coverings*. They have applications in models of telecommunication [11] and in distance constrained labeling [12]. Moreover, they are used as indicators of the existence of homomorphisms of derivative graphs [27]. Locally surjective homomorphisms are also called *color dominations* [25]. In addition they are known as *role assignments* due to their applications in social science [9,28,29]. Just like locally bijective homomorphisms they also have applications in distributed computing [7].

If there exists a homomorphism from a graph G to a graph H that is locally bijective, locally injective, or locally surjective, respectively, then we write $G \xrightarrow{B} H$, $G \xrightarrow{I} H$, and $G \xrightarrow{S} H$, respectively. We denote the decision problems that are to test whether $G \xrightarrow{B} H$, $G \xrightarrow{I} H$, or $G \xrightarrow{S} H$ for two given graphs G and H by LBHom, LIHom and LSHom, respectively. All three problems are known to be NP-complete when both guest and host graphs are given as input (see below for details), and attempts have been made to classify their computational complexity when only the guest graph belongs to the input and the host graph is fixed. The corresponding problems are denoted by H-LBHom, H-LIHom, and H-LSHom, respectively. The H-LSHom problem is polynomial-time solvable either if H has no edge or if H is bipartite and has at least one connected component isomorphic to an edge; in all other cases H-LSHom is NP-complete, even for the class of bipartite graphs [13]. The complexity classification of H-LBHom and H-LIHom is still open, although many partial results are known; we refer to the papers [11,24] and to the survey by Fiala and Kratochvíl [10] for both NP-complete and polynomially solvable cases.

Instead of fixing the host graph, another natural restriction is to only take guest graphs from a special graph class. Heggernes et al. [21] proved that LBHom is Graph Isomorphism-complete when the guest graph is chordal, and polynomial-time solvable when the guest graph is interval. In contrast, LSHom is NP-complete when the guest graph is chordal and polynomial-time solvable when the guest graph is proper interval, whereas LIHom is NP-complete even for guest graphs that are proper interval [21]. It is also known that the problems

LBHom and LSHom are polynomial-time solvable when the guest graph is a tree [14].

In this paper we focus on the following line of research. The *core* of a graph G is a minimum subgraph F of G such that there exists a homomorphism from G to F. Dalmau, Kolaitis and Vardi [8] proved that the Hom problem is polynomial-time solvable when the guest graph belongs to any fixed class of graphs whose cores have bounded treewidth. In particular, this result implies that Hom is polynomial-time solvable when the guest graph has bounded treewidth. Grohe [17] strengthened the result of Dalmau et al. [8] by proving that under a certain complexity assumption (namely $\mathsf{FPT} \neq \mathsf{W}[1]$) the Hom problem can be solved in polynomial time if and only if this condition holds.

Our Contribution. We investigate whether the aforementioned results of Dalmau et al. [8] and Grohe [17] remain true when we consider locally constrained homomorphisms instead of general homomorphisms. In Section 2, we provide a negative answer to this question by showing that the problems LBHom, LSHom and LIHom are NP-complete already in the restricted case where the guest graph has pathwidth at most 5, 4 or 2, respectively. We also show that the three problems are NP-complete even if both the guest graph and the host graph have maximum degree 3. The latter result shows that locally constrained homomorphisms problems behave more like unconstrained homomorphisms on graphs of bounded degree than on graphs of bounded treewidth, as it is known that, for example, C_5-Hom is NP-complete on subcubic graphs [15].

On the positive side, in Section 3, we show that all three problems can be solved in polynomial time if we bound the treewidth of the guest graph and at the same time bound the maximum degree of the guest graph or the host graph. Because a graph class of bounded maximum degree has bounded treewidth if and only if it has bounded clique-width [18], all three problems are also polynomial-time solvable when we bound the clique-width and the maximum degree of the guest graph.

Preliminaries. Let G be a graph. The *degree* of a vertex v in G is denoted by $d_G(v) = |N_G(v)|$, and $\Delta(G) = \max_{v \in V_G} d_G(v)$ denotes the maximum degree of G. Let φ be a homomorphism from G to a graph H. Moreover, let G' be an induced subgraph of G, and let φ' be a homomorphism from G' to H. We say that φ *extends* (or, equivalently, is an *extension* of) φ' if $\varphi(v) = \varphi'(v)$ for every $v \in V_{G'}$.

A *tree decomposition* of G is a tree $T = (V_T, E_T)$, where the elements of V_T, called the *nodes* of T, are subsets of V_G such that the following three conditions are satisfied:

1. for each vertex $v \in V_G$, there is a node $X \in V_T$ with $v \in X$,
2. for each edge $uv \in E_G$, there is a node $X \in V_T$ with $\{u, v\} \subseteq X$,
3. for each vertex $v \in V_G$, the set of nodes $\{X \mid v \in X\}$ induces a connected subtree of T.

The *width* of a tree decomposition T is the size of a largest node X minus one. The *treewidth* of G, denoted by $\mathrm{tw}(G)$, is the minimum width over all possible tree decompositions of G. A *path decomposition* of G is a tree decomposition T of G where T is a path. The *pathwidth* of G is the minimum width over all possible path decompositions of G. By definition, the pathwidth of G is at least as high as its treewidth. A tree decomposition T is *nice* [22] if T is a binary tree, rooted in a root R such that the nodes of T belong to one of the following four types:

1. a *leaf node* X is a leaf of T,
2. an *introduce node* X has one child Y and $X = Y \cup \{v\}$ for some vertex $v \in V_G \setminus Y$,
3. a *forget node* X has one child Y and $X = Y \setminus \{v\}$ for some vertex $v \in Y$,
4. a *join node* X has two children Y, Z satisfying $X = Y = Z$.

2 NP-Completeness Results

For the NP-hardness results in Theorem 1 below we use a reduction from the 3-Partition problem. This problem takes as input a multiset A of $3m$ integers, denoted in the sequel by $\{a_1, a_2, \dots, a_{3m}\}$, and a positive integer b, such that $\frac{b}{4} < a_i < \frac{b}{2}$ for all $i \in \{1, \dots, 3m\}$ and $\sum_{1 \le i \le 3m} a_i = mb$. The task is to determine whether A can be partitioned into m disjoint sets $A_1, \dots, A_m$ such that $\sum_{a \in A_i} a = b$ for all $i \in \{1, \dots, m\}$. Note that the restrictions on the size of each element in A implies that each set A_i in the desired partition must contain exactly three elements, which is why such a partition $A_1, \dots, A_m$ is called a 3-*partition* of A. The 3-Partition problem is strongly NP-complete [16], i.e., it remains NP-complete even if the problem is encoded in unary.

Theorem 1. *The following three statements hold:*

(i) LBHom *is* NP*-complete on input pairs* (G, H) *where* G *has pathwidth at most* 5 *and* H *has pathwidth at most* 3*;*
(ii) LSHom *is* NP*-complete on input pairs* (G, H) *where* G *has pathwidth at most* 4 *and* H *has pathwidth at most* 3*;*
(iii) LIHom *is* NP*-complete on input pairs* (G, H) *where* G *has pathwidth at most* 2 *and* H *has pathwidth at most* 2*.*

Proof. We only prove statement (i) here; the similar but easier proofs of statements (ii) and (iii) have been omitted.

Note that LBHom is in NP. Given an instance (A, b) of 3-Partition, we construct two graphs G and H as follows; see Figures 1 and 2 for some helpful illustrations. The construction of G starts by taking $3m$ disjoint cycles $C_1, \dots, C_{3m}$ of length b, one for each element of A. For each $i \in \{1, \dots, 3m\}$, the vertices of C_i are labeled $u^i_1, \dots, u^i_b$ and we add, for each $j \in \{1, \dots, b\}$, two new vertices p^i_j and q^i_j as well as two new edges $u^i_j p^i_j$ and $u^i_j q^i_j$. We then add three new vertices x, y and z. Vertex x is made adjacent to vertices $p^i_1, p^i_2 \dots, p^i_{a_i}$ and $q^i_1, q^i_2 \dots, q^i_{a_i}$ for every $i \in \{1, \dots, 3m\}$. Finally, the vertex y is made adjacent to every vertex p^i_j that is not adjacent to x, and the vertex z is made adjacent to every vertex q^i_j that is not adjacent to x. This finishes the construction of G.

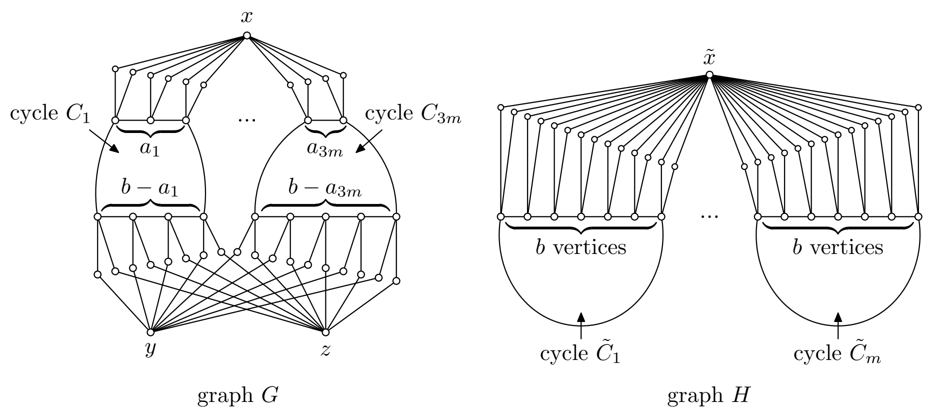

Fig. 1. A schematic illustration of the graphs G and H that are constructed from a given instance (A, b) of 3-Partition in the proof of statement (i) in Theorem 1. See also Figure 2 for a more detailed illustration of the "leftmost" part of G and the "rightmost" part of H, including more labels.

To construct H, we take m disjoint cycles $\tilde{C}_1, \ldots, \tilde{C}_m$ of length b, where the vertices of each cycle $\tilde{C}_i$ are labeled $\tilde{u}^i_1, \ldots, \tilde{u}^i_b$. For each $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, b\}$, we add two vertices $\tilde{p}^i_j$ and $\tilde{q}^i_j$ and make both of them adjacent to $\tilde{u}^i_j$. Finally, we add a vertex $\tilde{x}$ and make it adjacent to each of the vertices $\tilde{p}^i_j$ and $\tilde{q}^i_j$. This finishes the construction of H.

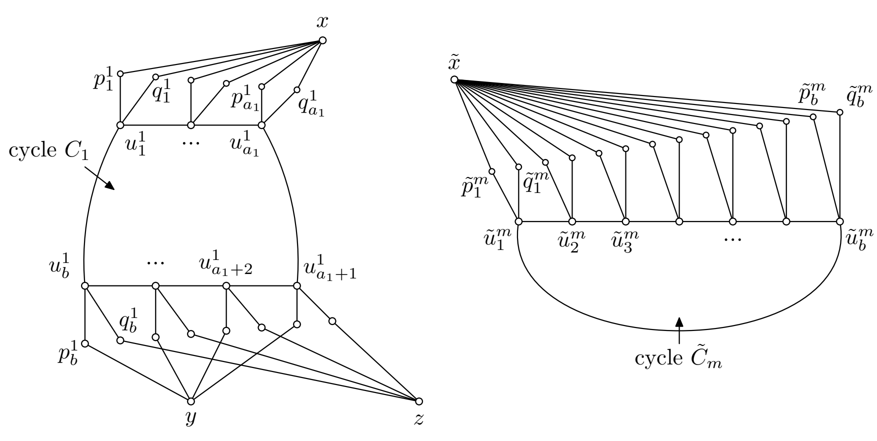

Fig. 2. More detailed illustration of parts of the graphs G and H in Figure 1.

We now show that there exists a locally bijective homomorphism from G to H if and only if (A, b) is a yes-instance of 3-Partition.

Let us first assume that there exists a locally bijective homomorphism φ from G to H. Since φ is a degree-preserving mapping, we must have $\varphi(x) = \tilde{x}$. Moreover, since φ is locally bijective, the restriction of φ to $N_G(x)$ is a bijection from $N_G(x)$ to $N_H(\tilde{x})$. Again using the definition of a locally bijective mapping, this time considering the neighborhoods of the vertices in $N_H(\tilde{x})$, we deduce that there is a bijection from the set $N_G^2(x) := \{u_j^i \mid 1 \leq i \leq 3m, 1 \leq j \leq a_i\}$, i.e., from the set of vertices in G at distance 2 from x, to the set $N_H^2(\tilde{x}) := \{\tilde{u}_j^k \mid 1 \leq k \leq m, 1 \leq j \leq b\}$ of vertices that are at distance 2 from $\tilde{x}$ in H. For every $k \in \{1, \ldots, m\}$, we define a set $A_k \subseteq A$ such that A_k contains element $a_i \in A$ if and only if $\varphi(u_1^i) \in \{\tilde{u}_1^k, \ldots, \tilde{u}_b^k\}$. Since φ is a bijection from $N_G^2(x)$ to $N_H^2(\tilde{x})$, the sets $A_1, \ldots, A_m$ are disjoint; moreover each element $a_i \in A$ is contained in exactly one of them. Observe that the subgraph of G induced by $N_G^2(x)$ is a disjoint union of $3m$ paths of lengths $a_1, a_2, \ldots, a_{3m}$, respectively, while the subgraph of H induced by $N_H^2(\tilde{x})$ is a disjoint union of m cycles of length b each. The fact that φ is a homomorphism and therefore never maps adjacent vertices of G to non-adjacent vertices in H implies that $\sum_{a \in A_i} a = b$ for all $i \in \{1, \ldots, m\}$. Hence $A_1, \ldots, A_m$ is a 3-partition of A.

For the reverse direction, suppose there exists a 3-partition $A_1, \ldots, A_m$ of A. We define a mapping φ as follows. We first set $\varphi(x) = \varphi(y) = \varphi(z) = \tilde{x}$. Let $A_i = \{a_r, a_s, a_t\}$ be any set of the 3-partition. We map the vertices of the cycles C_r, C_s, C_t that are at distance 2 from x to the vertices of the cycle $\tilde{C}_i$ in the following way: $\varphi(u_j^r) = \tilde{u}_j^i$ for each $j \in \{1, \ldots, a_r\}$, $\varphi(u_j^s) = \tilde{u}_{a_r+j}^i$ for each $j \in \{1, \ldots, a_s\}$, and $\varphi(u_j^t) = \tilde{u}_{a_r+a_s+j}^i$ for each $j \in \{1, \ldots, a_t\}$. The vertices of C_r, C_s and C_t that are at distance more than 2 from x in G are mapped to vertices of $\tilde{C}_i$ such that the vertices of C_r, C_s and C_t appear in the same order as their images on $\tilde{C}_i$. In particular, we set $\varphi(u_j^r) = \tilde{u}_j^i$ for each $j \in \{a_r + 1, \ldots, b\}$; the vertices of the cycles C_s and C_t that are at distance more than 2 from x are mapped to vertices of $\tilde{C}_i$ analogously. After the vertices of the cycles $C_1, \ldots, C_{3m}$ have been mapped in the way described above, it remains to map the vertices p_j^i and q_j^i for each $i \in \{1, \ldots, 3m\}$ and $j \in \{1, \ldots, b\}$.

Let p_j^i, q_j^i be a pair of vertices in G that are adjacent to x, and let u_j^i be the second common neighbor of p_j^i and q_j^i. Suppose $\tilde{u}_\ell^k$ is the image of u_j^i, i.e., suppose that $\varphi(u_j^i) = \tilde{u}_\ell^k$. Then we map p_j^i and q_j^i to $\tilde{p}_\ell^k$ and $\tilde{q}_\ell^k$, respectively. We now consider the neighbors of y and z in G. By construction, the neighborhood of y consists of the $2mb$ vertices in the set $\{p_j^i \mid a_{i+1} \leq j \leq b\}$, while $N_G(z) = \{q_j^i \mid a_{i+1} \leq j \leq b\}$.

Observe that $\tilde{x}$, the image of y and z, is adjacent to two sets of mb vertices: one of the form $\tilde{p}_\ell^k$, the other of the form $\tilde{q}_\ell^k$. Hence, we need to map half the neighbors of y to vertices of the form $\tilde{p}_\ell^k$ and half the neighbors of y to vertices of the form $\tilde{q}_\ell^k$ in order to make φ a locally bijective homomorphism. The same should be done with the neighbors of z. For every vertex $\tilde{u}_\ell^k$ in H, we do as follows. By construction, exactly three vertices of G are mapped to $\tilde{u}_\ell^k$, and exactly two of those vertices, say u_j^i and u_h^g, are at distance 2 from y in G. We set $\varphi(p_j^i) = \tilde{p}_\ell^k$ and $\varphi(p_h^g) = \tilde{q}_\ell^k$. We also set $\varphi(q_j^i) = \tilde{q}_\ell^k$ and $\varphi(q_h^g) = \tilde{p}_\ell^k$. This completes the definition of the mapping φ.

Since the mapping φ preserves adjacencies, it clearly is a homomorphism. In order to show that φ is locally bijective, we first observe that the degree of every vertex in G is equal to the degree of its image in H; in particular, $d_G(x) = d_G(y) = d_G(z) = d_H(\tilde{x}) = mb$. From the above description of φ we get a bijection between the vertices of $N_H(\tilde{x})$ and the vertices of $N_G(v)$ for each $v \in \{x, y, z\}$. For every vertex p_j^i that is adjacent to x and u_j^i in G, its image $\tilde{p}_\ell^k$ is adjacent to the images $\tilde{x}$ of x and $\tilde{u}_\ell^k$ of u_j^i. For every vertex p_j^i that is adjacent to y (respectively z) and u_j^i in G, its image $\tilde{p}_\ell^k$ or $\tilde{q}_\ell^k$ is adjacent to $\tilde{x}$ of y (respectively z) and $\tilde{u}_\ell^k$ of u_j^i. Hence the restriction of φ to $N_G(p_j^i)$ is bijective for every $i \in \{1, \ldots, 3m\}$ and $j \in \{1, \ldots, b\}$, and the same clearly holds for the restriction of φ to $N_G(q_j^i)$. The vertices of each cycle C_i are mapped to the vertices of some cycle $\tilde{C}_k$ in such a way that the vertices and their images appear in the same order on the cycles. This, together with the fact that the image $\tilde{u}_\ell^k$ of every vertex u_j^i is adjacent to the images $\tilde{p}_\ell^k$ and $\tilde{q}_\ell^k$ of the neighbors p_j^i and q_j^i of u_j^i, shows that the restriction of φ to $N_G(u_j^i)$ is bijective for every $i \in \{1, \ldots, 3m\}$ and $j \in \{1, \ldots, b\}$. We conclude that φ is a locally bijective homomorphism from G to H.

In order to show that the pathwidth of G is at most 5, let us first consider the subgraph of G depicted on the left-hand side of Figure 2; we denote this subgraph by L_1, and we say that the cycle C_1 *defines* the subgraph L_1. The graph L_1' that is obtained from L_1 by deleting vertices x, y, z and edge $u_1^1 u_b^1$ is a caterpillar, i.e., a tree in which there is a path containing all vertices of degree more than 1. Since caterpillars are well-known to have pathwidth 1, graph L_1' has a path decomposition P_1' of width 1. Starting with P_1', we can now obtain a path decomposition of the graph L_1 by simply adding vertices x, y, z and u_1^1 to each node of P_1'; this path decomposition has width 5. Every cycle C_i in G defines a subgraph L_i of G in the same way C_1 defines the subgraph L_1. Suppose we have constructed a path decomposition P_i of width 5 of the subgraph L_i for each $i \in \{1, \ldots, 3m\}$ in the way described above. Since any two subgraphs L_i and L_j with $i \neq j$ have only the vertices x, y, z in common, and these three vertices appear in all nodes of each of the path decompositions P_i, we can arrange the $3m$ path decompositions $P_1, \ldots, P_{3m}$ in such a way that we obtain a path decomposition P of G of width 5. Hence G has pathwidth at most 5. Similar but easier arguments can be used to show that H has pathwidth at most 3. □

We now consider the case where we bound the maximum degree of G instead of the treewidth of G. An *equitable partition* of a connected graph G is a partition of its vertex set in blocks $B_1, \ldots, B_k$ such that any vertex in B_i has the same number $m_{i,j}$ of neighbors in B_j. We call the matrix $M = (m_{i,j})$ corresponding to the coarsest equitable partition of G (in which the blocks are ordered in some canonical way; cf. [1]) the *degree refinement matrix* of G, denoted as $\mathrm{drm}(G)$. We will use the following lemma; a proof of the first statement in this lemma can be found in the paper of Fiala and Kratochvíl [11], whereas the second statement is due to Kristiansen and Telle [25].

Lemma 1. *Let G and H be two graphs. Then the following two statements hold:*

(i) if $G \xrightarrow{I} H$ *and* $\mathrm{drm}(G) = \mathrm{drm}(H)$*, then* $G \xrightarrow{B} H$*;*
(ii) if $G \xrightarrow{S} H$ *and* $\mathrm{drm}(G) = \mathrm{drm}(H)$*, then* $G \xrightarrow{B} H$*.*

Kratochvíl and Křivánek [23] showed that K_4-LBHom is NP-complete, where K_4 denotes the complete graph on four vertices. Since a graph G allows a locally bijective homomorphism to K_4 only if G is 3-regular, K_4-LBHom is NP-complete on 3-regular graphs. The degree refinement matrix of a 3-regular graph is the 1×1 matrix whose only entry is 3. Consequently, due to Lemma 1, K_4-LBHom is equivalent to K_4-LIHom and to K_4-LSHom on 3-regular graphs. This yields the following result.

Theorem 2. *The problems* LBHom*,* LIHom *and* LSHom *are* NP*-complete on input pairs* (G, K_4) *where* G *has maximum degree* 3*.*

3 Polynomial-Time Results

In Section 2, we showed that LBHom, LIHom and LSHom are NP-complete when either the treewidth or the maximum degree of the guest graph is bounded. In this section, we show that all three problems become polynomial-time solvable if we bound both the treewidth and the maximum degree of G. For the problems LBHom and LIHom, our polynomial-time result follows from reformulating these problems as constraint satisfaction problems and applying a result of Dalmau et al. [8]; we omit the proof details.

Theorem 3. *The problems* LBHom *and* LIHom *can be solved in polynomial time when* G *has bounded treewidth and* G *or* H *has bounded maximum degree.*

To our knowledge, locally surjective homomorphisms have not yet been expressed as homomorphisms between relational structures. Hence, in the proof of Theorem 4 below, we present a polynomial-time algorithm for LSHom when G has bounded treewidth and bounded maximum degree. We first introduce some additional terminology.

Let φ be a locally surjective homomorphism from G to H. Let $v \in V_G$ and $p \in V_H$. If $\varphi(v) = p$, i.e., if φ maps vertex v to color p, then we say that p is *assigned* to v. By definition, for every vertex $v \in V_G$, the set of colors that are assigned to the neighbors of v in G is exactly the neighborhood of $\varphi(v)$ in H. Now suppose we are given a homomorphism φ' from an induced subgraph G' of G to H. For any vertex $v \in V_{G'}$, we say that v *misses* a color $p \in V_H$ if $p \in N_H(\varphi'(v)) \setminus \varphi(N_{G'}(v))$, i.e., if φ' does not assign p to any neighbor of v in G', but any locally surjective homomorphism φ from G to H that extends φ' assigns p to some neighbor of v in G'.

Let T be a nice tree decomposition of G rooted in R. For every node $X \in V_T$, we define G_X to be the subgraph of G induced by the vertices of X together with the vertices of all the nodes that are descendants of X. In particular, we have $G_R = G$.

Definition 1. *Let $X \in V_T$, and let $c : X \to V_H$ and $\mu : X \to 2^{V_H}$ be two mappings. The pair (c, μ) is* feasible *for G_X if there exists a homomorphism φ from G_X to H satisfying the following three conditions:*

(i) $c(v) = \varphi(v)$ *for every* $v \in X$*;*
(ii) $\mu(v) = N_H(\varphi(v)) \setminus \varphi(N_{G_X}(v))$ *for every* $v \in X$*;*
(iii) $\varphi(N_G(v)) = N_H(\varphi(v))$ *for every* $v \in V_{G_X} \setminus X$.

In other words, a pair (c, μ) consists of a coloring c of the vertices of X, together with a collection of sets $\mu(v)$, one for each $v \in X$, consisting of exactly those colors that v misses. Informally speaking, a pair (c, μ) is feasible for G_X if there is a homomorphism $\varphi : G_X \to H$ such that φ "agrees" with the coloring c on the set X, and such that none of the vertices in $V_{G_X} \setminus X$ misses any color. The idea is that if a pair (c, μ) is feasible, then such a homomorphism φ might have an extension φ^* that is a locally surjective homomorphism from G to H. After all, for any vertex $v \in X$ that misses a color when considering φ, this color might be assigned by φ^* to a neighbor of v in the set $V_G \setminus V_{G_X}$.

We now prove a result for LSHom similar to Theorem 3.

Theorem 4. *The problem* LSHom *can be solved in polynomial time when G has bounded treewidth and G or H has bounded maximum degree.*

Proof. Let (G, H) be an instance of LSHom such that the treewidth of the guest graph G is bounded. Throughout the proof, we assume that the maximum degree of H is bounded, and show that the problem can be solved in polynomial time under these restrictions. Since $G \xrightarrow{S} H$ implies that $\Delta(G) \geq \Delta(H)$, our polynomial-time result applies also if we bound the maximum degree of G instead of H.

We may assume without loss of generality that both G and H are connected, as otherwise we just consider all pairs (G_i, H_j) separately, where G_i is a connected component of G and H_j is a connected component of H. Because G has bounded treewidth, we can compute a tree decomposition of G of width $\mathrm{tw}(G)$ in linear time using Bodlaender's algorithm [6]. We transform this tree decomposition into a nice tree decomposition T of G with width $\mathrm{tw}(G)$ with at most $4|V_G|$ nodes using the linear-time algorithm of Kloks [22]. Let R be the root of T and let $k = \mathrm{tw}(G) + 1$.

For each node $X \in V_T$, let F_X be the set of all feasible pairs (c, μ) for G_X. For every feasible pair $(c, \mu) \in F_X$ and every $v \in X$, it holds that $\mu(v)$ is a subset of $N_H(c(v))$. Since $|X| \leq k$ and $|N_H(c(v))| \leq \Delta(H)k$ for every $v \in X$ and every mapping $c : X \to V_H$, this implies that $|F_X| \leq |V_H|^k 2^{\Delta(H)k}$ for each $X \in V_T$. As we assumed that both k and $\Delta(H)$ are bounded by a constant, the set F_X is of polynomial size with respect to $|V_H|$.

The algorithm considers the nodes of T in a bottom-up manner, starting with the leaves of T and processing a node $X \in V_T$ only after its children have been processed. For every node X, the algorithm computes the set F_X in the way described below. We distinguish between four different cases. The correctness of each of the cases easily follows from the definition of a locally surjective homomorphism and Definition 1.

1. *X is a leaf node of T.* We consider all mappings $c : X \to V_H$. For each mapping c, we check whether c is a homomorphism from G_X to H. If not, then we discard c, as it can not belong to a feasible pair due to condition (i) in Definition 1. For each mapping c that is not discarded, we compute the unique mapping μ satisfying $\mu(v) = N_H(c(v)) \setminus c(N_{G_X}(v))$ for each $v \in X$, and we add the pair (c, μ) to F_X. It follows from condition (ii) that the obtained set F_X indeed contains all feasible pairs for G_X. As there is no vertex in $V_{G_X} \setminus X$, every pair (c, μ) trivially satisfies condition (iii). The computation of F_X can be done in $O(|V_H|^k k(\Delta(H) + k))$ time in this case.
2. *X is a forget node.* Let Y be the child of X in T, and let $\{u\} = Y \setminus X$. Observe that $(c, \mu) \in F_X$ if and only if there exists a feasible pair $(c', \mu') \in F_Y$ such that $c(v) = c'(v)$ and $\mu(v) = \mu'(v)$ for every $v \in X$, and $\mu'(u) = \emptyset$. Hence we examine each $(c', \mu') \in F_Y$ and check whether $\mu'(u) = \emptyset$ is satisfied. If so, we first restrict (c', μ') on X to get (c, μ) and then we insert the obtained feasible pair into F_X. This procedure needs $O(|F_Y| k\Delta(H))$ time in total.
3. *X is an introduce node.* Let Y be the child of X in T, and let $\{u\} = X \setminus Y$. Observe that $(c, \mu) \in F_X$ if and only if there exists a feasible pair $(c', \mu') \in F_Y$ such that, for every $v \in Y$, it holds that $c(v) = c'(v)$, $\mu(v) = \mu'(v) \setminus c(u)$ if $uv \in E_G$, and $\mu(v) = \mu'(v)$ if $uv \notin E_G$. Hence, for each $(c', \mu') \in F_Y$, we consider all $|V_H|$ mappings $c : X \to V_H$ that extend c'. For each such extension c, we test whether c is a homomorphism from G_X to H by checking the adjacencies of $c(u)$ in H. If not, then we may safely discard c due to condition (i) in Definition 1. Otherwise, we compute the unique mapping $\mu : X \to 2^{V_H}$ satisfying
$$\mu(v) = \begin{cases} N_H(c(u)) \setminus c(N_{G_X}(u)) & \text{if } v = u \\ \mu'(v) \setminus c(u) & \text{if } v \neq u \text{ and } uv \in E_G \\ \mu'(v) & \text{if } v \neq u \text{ and } uv \notin E_G \ , \end{cases}$$
and we add the pair (c, μ) to F_X; due to condition (ii), this pair (c, μ) is the unique feasible pair containing c. Computing the set F_X takes at most $O(|F_Y||V_H| k\Delta(H))$ time in total.
4. *X is a join node.* Let Y and Z be the two children of X in T. Observe that $(c, \mu) \in F_X$ if and only if there exist feasible pairs $(c_1, \mu_1) \in F_Y$ and $(c_2, \mu_2) \in F_Z$ such that, for every $v \in X$, $c(v) = c_1(v) = c_2(v)$ and $\mu(v) = \mu_1(v) \cap \mu_2(v)$. Hence the algorithm considers every combination of $(c_1, \mu_1) \in F_Y$ with $(c_2, \mu_2) \in F_Z$ and if they agree on the first component c, the other component μ is determined uniquely by taking the intersection of $\mu_1(v)$ and $\mu_2(v)$ for every $v \in X$. This procedure computes the set F_X in $O(|F_Y||F_Z| k\Delta(H))$ time in total.

Finally, observe that a locally surjective homomorphism from G to H exists if and only if there exists a feasible pair (c, μ) for G_R such that $\mu(v) = \emptyset$ for all $v \in R$. Since T has at most $4|V_G|$ nodes, we obtain a total running time of $O(|V_G|(|V_H|^k 2^{\Delta(H)k})^2 k\Delta(H))$. As we assumed that both $k = \mathrm{tw}(G) + 1$ and $\Delta(H)$ are bounded by a constant, our algorithm runs in polynomial time. $\square$

Note that Theorem 3 can be derived by solving LIHom using a dynamic programming approach that strongly resembles the one for LSHom described in the proof of Theorem 4, together with the fact that (G, H) is a yes-instance of LBHom if and only if it is a yes-instance for both LIHom and LSHom. In a dynamic programming algorithm for solving LIHom, instead of keeping track of sets $\mu(v)$ of colors that a vertex $v \in X$ is missing, we keep track of sets $\alpha(v)$ of colors that have already been assigned to the neighbors of a vertex $v \in X$. This is because in a locally injective homomorphism from G to H, no color may be assigned to more than one neighbor of any vertex. In this way we can adjust Definition 1 in such a way that it works for locally injective instead of locally surjective homomorphisms. We omit further details, but we expect that a dynamic programming algorithm of this kind will have smaller hidden constants in the running time estimate than the more general method of Dalmau et al. [8].

We conclude this section with one more polynomial-time result, the proof of which has been omitted. It is known that the problems LBHom and LSHom are polynomial-time solvable when G is a tree [14], and consequently when G has treewidth 1. We claim that the same holds for the LIHom problem.

Theorem 5. *The* LIHom *problem can be solved in polynomial time when* G *has treewidth* 1*.*

4 Conclusion

Theorem 5 states that LIHom can be solved in polynomial time when the guest graph has treewidth 1, while Theorem 1 implies that the problem is NP-complete when the guest graph has treewidth 2. This shows that the bound on the pathwidth in the third statement of Theorem 1 is best possible. We leave it as an open problem to determine whether the bounds on the pathwidth in the other two statements of Theorem 1 can be reduced further.

Acknowledgements. We would like to thank Isolde Adler for posing the research questions that we addressed in our paper and for helpful discussions. The fourth author also thanks Jan Arne Telle for fruitful discussions.

References

1. D. Angluin, Local and global properties in networks of processors, Proc. STOC 1980, 82–93, 1980.
2. D. Angluin and A. Gardiner, Finite common coverings of pairs of regular graphs, J. Comb. Theory Ser. B 30 (1981) 184–187.
3. N. Biggs, Algebraic Graph Theory, Cambridge University Press, 1974.
4. N. Biggs, Constructing 5-arc transitive cubic graphs, J. London Math. Society II. 26 (1982) 193–200.
5. H.L. Bodlaender, The classification of coverings of processor networks, J. Par. Distrib. Comp. 6 (1989), 166–182.
6. H.L. Bodlaender, A linear-time algorithm for finding tree-decompositions of small treewidth, SIAM J. Comp. 25(6) (1996) 1305–1317.

7. J. Chalopin, Y. Métivier and W. Zielonka, Election, naming and cellular edge local computations, Proc. ICGT 2004, LNCS 3256, pp. 242–256, 2004.
8. V. Dalmau, P.G. Kolaitis and M.Y. Vardi, Constraint satisfaction, bounded treewidth, and finite-variable logics, Proc. CP 2002, LNCS 2470, pp. 223–254, 2002.
9. M.G. Everett and S. Borgatti, Role coloring a graph, Mathematical Social Sciences 21 (1991) 183–188.
10. J. Fiala and J. Kratochvíl, Locally constrained graph homomorphisms – Structure, complexity, and applications. Comp. Sci. Review 2 (2008) 97–111.
11. J. Fiala and J. Kratochvíl, Partial covers of graphs, Disc. Math. Graph Theory 22 (2002) 89–99.
12. J. Fiala, J. Kratochvíl and T. Kloks, Fixed-parameter complexity of λ-labelings, Discr. Appl. Math. 113 (2001) 59–72.
13. J. Fiala and D. Paulusma, A complete complexity classification of the role assignment problem, Theor. Comp. Sci. 349 (2005) 67–81.
14. J. Fiala and D. Paulusma, Comparing universal covers in polynomial time, Theory Comp. Syst. 46 (2010) 620–635.
15. A. Galluccio, P. Hell and J. Nešetřil, The complexity of H-colouring of bounded degree graphs. Discr. Math. 222 (2000) 101–109
16. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-completeness. W. H. Freeman & Co., New York, 1979.
17. M. Grohe, The complexity of homomorphism and constraint satisfaction problems seen from the other side, J. ACM 54(2007).
18. F. Gurski and E. Wanke, The tree-width of clique-width bounded graphs without $K_{n,n}$, Proc. WG 2000, LNCS 1928, pp. 196–205, 2000.
19. P. Hell and J. Nešetřil, On the complexity of H-colouring, J. Comb. Theory Ser. B 48 (1990) 92–110.
20. P. Hell and J. Nešetřil, Graphs and Homomorphisms, Oxford University Press, 2004.
21. P. Heggernes, P. van 't Hof and D. Paulusma, Computing role assignments of proper interval graphs in polynomial time, J. Discr. Alg. 14 (2012) 173–188.
22. T. Kloks. *Treewidth, Computations and Approximations.* LNCS 842, Springer, 1994.
23. J. Kratochvíl and M. Křivánek, On the computational complexity of codes in graphs, Proc. MFCS 1988, LNCS 324, pp. 396–404, 1988.
24. J. Kratochvíl, A. Proskurowski and J.A. Telle, Covering regular graphs, J. Comb. Theory Ser. B 71 (1997) 1–16.
25. P. Kristiansen and J.A. Telle, Generalized H-coloring of graphs, Proc. ISAAC 2000. LNCS 1969, pp. 456–466, 2000.
26. W.S. Massey, Algebraic Topology: An Introduction, Harcourt, Brace and World, 1967.
27. J. Nešetřil, Homomorphisms of derivative graphs, Discr. Math. 1 (1971) 257–268.
28. A. Pekeč and F.S. Roberts, The role assignment model nearly fits most social networks, Mathematical Social Sciences 41 (2001) 275–293.
29. F.S. Roberts and L. Sheng, How hard is it to determine if a graph has a 2-role assignment? Networks 37 (2001) 67–73.