



An Algorithmic Framework for Locally Constrained Homomorphisms

Laurent Bulteau, Konrad Dabrowski, Noleen Köhler, Sebastian Ordyniak,
Daniël Paulusma

► To cite this version:

Laurent Bulteau, Konrad Dabrowski, Noleen Köhler, Sebastian Ordyniak, Daniël Paulusma. An Algorithmic Framework for Locally Constrained Homomorphisms. International Workshop on Graph-Theoretic Concepts in Computer Science, 2022, Tübingen, Germany. pp.114-128, 10.1007/978-3-031-15914-5_9 . hal-04495171

HAL Id: hal-04495171

<https://hal.science/hal-04495171>

Submitted on 8 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Algorithmic Framework for Locally Constrained Homomorphisms

Laurent Bulteau ✉ 

LIGM, CNRS, Université Gustave Eiffel, France

Konrad K. Dabrowski ✉ 

Newcastle University, UK

Noleen Köhler ✉

Université Paris-Dauphine, PSL University, CNRS UMR7243, LAMSADE, Paris, France

Sebastian Ordyniak ✉ 

University of Leeds, UK

Daniël Paulusma ✉ 

University of Durham, UK

Abstract

A homomorphism f from a guest graph G to a host graph H is locally bijective, injective or surjective if for every $u \in V(G)$, the restriction of f to the neighbourhood of u is bijective, injective or surjective, respectively. The corresponding decision problems, LBHOM, LIHOM and LSHOM, are well studied both on general graphs and on special graph classes. Apart from complexity results when the problems are parameterized by the treewidth and maximum degree of the guest graph, the three problems still lack a thorough study of their parameterized complexity. This paper fills this gap: we prove a number of new FPT, W[1]-hard and para-NP-complete results by considering a hierarchy of parameters of the guest graph G . For our FPT results, we do this through the development of a new algorithmic framework that involves a general ILP model. To illustrate the applicability of the new framework, we also use it to prove FPT results for the ROLE ASSIGNMENT problem, which originates from social network theory and is closely related to locally surjective homomorphisms.

2012 ACM Subject Classification Mathematics of computing → Graph theory

Keywords and phrases locally constrained homomorphism, parameterized complexity, deletion set/modulator to small components, role assignment, integer linear programming

Funding *Konrad K. Dabrowski*: Supported by EPSRC grant EP/V00252X/1.

Sebastian Ordyniak: Supported by EPSRC grant EP/V00252X/1.

Daniël Paulusma: Supported by Leverhulme Trust grant RPG-2016-258.

1 Introduction

A *homomorphism* from a graph G to a graph H is a mapping $\phi : V(G) \rightarrow V(H)$ such that $\phi(u)\phi(v) \in E(H)$ for every $uv \in E(G)$. Graph homomorphisms generalise graph colourings (let H be a complete graph) and have been intensively studied over a long period of time, both from a structural and an algorithmic perspective. We refer to the textbook of Hell and Nešetřil [44] for a further introduction.

We write $G \rightarrow H$ if there exists a homomorphism from G to H ; here, G is called the *guest graph* and H is the *host graph*. We denote the corresponding decision problem by HOM, and if H is fixed, that is, not part of the input, we write H -HOM. The renowned Hell-Nešetřil dichotomy [42] states that H -HOM is polynomial-time solvable if H is bipartite, and NP-complete otherwise. We denote the vertices of H by $1, \dots, |V(H)|$ and call them *colours*. The reason for doing this is that graph homomorphisms generalise graph colourings: there exists a homomorphism from a graph G to the complete graph on k vertices if and only if G is k -colourable.

Instead of fixing the host graph H , one can also restrict the structure of the guest graph G by bounding some graph parameter. A classical result states that HOM is polynomial-time solvable when the guest graph G has bounded treewidth [16, 37]. The *core* of a graph G is the subgraph F of G such that $G \rightarrow F$ and there is no proper subgraph F' of F with $G \rightarrow F'$ (the core is unique up to isomorphism [43]). Dalmau, Kolaitis and Vardi [18] proved that the HOM problem is polynomial-time solvable even if the core of the guest graph G has bounded treewidth. This result was strengthened by Grohe [40], who proved that if $\text{FPT} \neq \text{W}[1]$, then HOM can be solved in polynomial time if and only if this condition holds.

Locally constrained homomorphisms. We are interested in three well-studied variants of graph homomorphisms that occur after placing constraints on the neighbourhoods of the vertices of the guest graph G . Consider a homomorphism ϕ from a graph G to a graph H . We say that ϕ is locally injective, locally bijective or locally surjective for $u \in V(G)$ if the restriction ϕ_u to the neighbourhood $N_G(u) = \{v \mid uv \in E(G)\}$ of u is injective, bijective or surjective. We say that ϕ is *locally injective*, *locally bijective* or *locally surjective* if it is locally injective, locally bijective, or locally surjective for every $u \in V(G)$. We denote these *locally constrained* homomorphisms by $G \xrightarrow{B} H$, $G \xrightarrow{L} H$ and $G \xrightarrow{S} H$, respectively.

The three variants have been well studied in several settings over a long period of time. For example, locally injective homomorphisms are also known as *partial graph coverings* and are used in telecommunications [29], in distance constrained labelling [28] and as indicators of the existence of homomorphisms of derivative graphs [58]. Locally bijective homomorphisms originate from topological graph theory [4, 57] and are more commonly known as *graph coverings*. They are used in distributed computing [2, 3, 7] and in constructing highly transitive regular graphs [5]. Locally surjective homomorphisms are sometimes called *colour dominations* [52]. They have applications in distributed computing [12, 13] and in social science [26, 63, 66, 69]. In the latter context they are known as *role assignments*, as we will explain in more detail below.

Let LBHOM, LIHOM and LSHOM be the three problems of deciding, for two graphs G and H , whether $G \xrightarrow{B} H$, $G \xrightarrow{L} H$ or $G \xrightarrow{S} H$ holds, respectively. As before, we write H -LBHOM, H -LIHOM and H -LSHOM in the case where the host graph H is fixed. Out of the three problems, only the complexity of H -LSHOM has been completely classified, both for general graphs and bipartite graphs [32]. We refer to a series of papers [1, 6, 29, 31, 49, 50, 55] for polynomial-time solvable and NP-complete cases of H -LBHOM and H -LIHOM; see also the survey by Fiala and Kratochvíl [30]. Some more recent results include sub-exponential algorithms for H -LBHOM, H -LIHOM and H -LSHOM on string graphs [60] and complexity results for H -LBHOM for host graphs H that are multigraphs [51] or that have semi-edges [9].

In our paper we assume that both G and H are part of the input. We note a fundamental difference between locally injective homomorphisms on one hand and locally bijective and surjective homomorphisms on the other hand. Namely, for connected graphs G and H , we must have $|V(G)| \geq |V(H)|$ if $G \xrightarrow{B} H$ or $G \xrightarrow{S} H$, whereas H might be arbitrarily larger than G if $G \xrightarrow{L} H$ holds. For example, if we let G be a complete graph, then $G \xrightarrow{L} H$ holds if and only if H contains a clique on at least $|V(G)|$ vertices.

The above difference is also reflected in the complexity results for the three problems under input restrictions. In fact, LIHOM is closely related to the SUBGRAPH ISOMORPHISM problem and is usually the hardest problem. For example, LBHOM is GRAPH ISOMORPHISM-complete on chordal guest graphs, but polynomial-time solvable on interval guest graphs and LSHOM is NP-complete on chordal guest graphs, but polynomial-time solvable on proper interval guest graphs [41]. In contrast, LIHOM is NP-complete even on complete guest graphs G , which follows from a reduction from the CLIQUE problem via the aforementioned equivalence:

$G \xrightarrow{L} H$ holds if and only if H contains a clique on at least $|V(G)|$ vertices.

To give another example, LBHOM, LSHOM and LIHOM are NP-complete for guest graphs G of path-width at most 5, 4 and 2, respectively [15] (all three problems are polynomial-time solvable if G is a tree [15, 33]). Note that these hardness results imply that the aforementioned polynomial-time result on HOM for guest graphs G of bounded treewidth [16, 37] does not carry over to any of the three locally constrained homomorphism problems. It is also known that LBHOM [48], LSHOM [52] and LIHOM [29] are NP-complete even if G is cubic and H is the complete graph K_4 on four vertices, but polynomial-time solvable if G has bounded treewidth and one of the two graphs G or H has bounded maximum degree [15].

An Application. Locally surjective homomorphisms from a graph G to a graph H are known as H -role assignments in social network theory. We will include this topic in our investigation and provide some brief context. Suppose we are given a social network of individuals whose properties we aim to characterise. Can we assign each individual a role such that individuals with the same role relate in the same way to other individuals with some role, using exactly h different roles in total? To formalise this question, we model the network as a graph G , where vertices represent individuals and edges represent the existence of a relationship between two individuals. We now ask whether G has an h -role assignment, that is, a function f that assigns each vertex $u \in V(G)$ a role $f(u) \in \{1, \dots, h\}$, such that $f(V(G)) = \{1, \dots, h\}$ and for every two vertices u and v , if $f(u) = f(v)$ then $f(N_G(u)) = f(N_G(v))$.

Role assignments were introduced by White and Reitz [69] as *regular equivalences* and were called *role colourings* by Everett and Borgatti [26]. We observe that two adjacent vertices u and v may have the same role, that is, $f(u) = f(v)$ is allowed (so role assignments are not proper colourings). Hence, a connected graph G has an h -role assignment if and only if $G \xrightarrow{S} H$ for some connected graph H with $|V(H)| = h$, as long as we allow H to have self-loops (while we assume that G is a graph with no self-loops).

The ROLE ASSIGNMENT problem is to decide, for a graph G and an integer h , whether G has an h -role assignment. If h is fixed, we denote the problem h -ROLE ASSIGNMENT. Whereas 1-ROLE ASSIGNMENT is trivial, 2-ROLE ASSIGNMENT is NP-complete [66]. In fact, h -ROLE ASSIGNMENT is NP-complete for planar graphs ($h \geq 2$) [64], cubic graphs ($h \geq 2$) [65], bipartite graphs ($h \geq 3$) [62], chordal graphs ($h \geq 3$) [68] and split graphs ($h \geq 4$) [20]. Very recently, Pandey, Raman and Sahlo [61] gave an $n^{\mathcal{O}(h)}$ -time algorithm for ROLE ASSIGNMENT on general graphs and an $f(h)n^{\mathcal{O}(1)}$ -time algorithm on forests.

Our Focus. We continue the line of study in [15] and focus on the following research question:

For which parameters of the guest graph do LBHOM, LSHOM and LIHOM become fixed-parameter tractable?

We will also apply our new techniques towards answering this question for the ROLE ASSIGNMENT problem. In order to address our research question, we need some additional terminology. A graph parameter p *dominates* a parameter q if there is a function f such that $p(G) \leq f(q(G))$ for every graph G . If p dominates q but q does not dominate p , then p is *more powerful (less restrictive)* than q . We denote this by $p \triangleright q$. If neither p dominates q nor q dominates p , then p and q are *incomparable (orthogonal)*. Given the para-NP-hardness results on LBHOM, LSHOM and LIHOM for graph classes of bounded path-width [15], we will consider a range of graph parameters that are less powerful than path-width. In this way we aim to increase our understanding of the (parameterized) complexity of LBHOM, LSHOM and LIHOM.

guest graph parameter	LIHOM	LBHOM	LSHOM
$ V(G) $	XP, W[1]-hard [21]	FPT	FPT
vertex cover number	XP (Theorem 21), W[1]-hard	FPT	FPT
c -deletion set number (fixed c)	para-NP-c ($c \geq 2$) (Theorem 24)	FPT	FPT
fracture number	para-NP-c	FPT (Theorem 18)	FPT (Theorem 18)
tree-depth	para-NP-c	para-NP-c (Theorem 28)	para-NP-c (Theorem 28)
path-width	para-NP-c [15]	para-NP-c [15]	para-NP-c [15]
treewidth	para-NP-c	para-NP-c	para-NP-c
maximum degree	para-NP-c [29]	para-NP-c [48]	para-NP-c [52]
treewidth plus maximum degree	XP, W[1]-hard	XP [15]	XP [15]
feedback vertex set number	para-NP-c	para-NP-c (Theorem 29)	para-NP-c (Theorem 29)

■ **Table 1** Table of results. The results in blue are the new results proven in this paper. The results in black are either known results, some of which are now also implied by our new results, or follow immediately from other results in the table; in particular, for a graph G , $ds_c(G) \geq fr(G)$ if $c \leq fr(G) - 1$, and $ds_c(G) \leq fr(G)$ if $c \geq fr(G)$. Also note that LIHOM is W[1]-hard when parameterized by $|V(G)|$, as CLIQUE is W[1]-hard when parameterized by the clique number [21], so as before, we can let G be the complete graph in this case.

For an integer $c \geq 1$, a c -deletion set of a graph G is a subset $S \subseteq V(G)$ such that every connected component of $G \setminus S$ has at most c vertices. The c -deletion set number $ds_c(G)$ of a graph G is the minimum size of a c -deletion set in G . If $c = 1$, then we obtain the *vertex cover number* $vc(G)$ of G . The c -deletion set number is also known as *vertex integrity* [23]. It is closely related to the *fracture number* $fr(G)$, introduced in [24], which is the minimum k such that G has a k -deletion set on at most k vertices. Note that $fr(G) \leq \max\{c, ds_c(G)\}$ holds for every integer c . The *feedback vertex set number* $fv(G)$ of a graph G is the size of a smallest set S such that $G \setminus S$ is a forest. We write $tw(G)$, $pw(G)$ and $td(G)$ for the treewidth, path-width and tree-depth of a graph G , respectively; see [59] for more information, in particular on tree-depth. It is known that

$$tw(G) \triangleright pw(G) \triangleright td(G) \triangleright fr(G) \triangleright ds_c(G) (\text{fixed } c) \triangleright vc(G) \triangleright |V(G)|,$$

where the second relationship is proven in [8] and the others follow immediately from their definitions (see also Section 2.2). It is readily seen that

$$tw(G) \triangleright fv(G) \triangleright ds_2(G)$$

and that $fv(G)$ is incomparable with the parameters $pw(G)$, $td(G)$, $fr(G)$ and $ds_c(G)$ for every fixed $c \geq 3$ (consider e.g. a tree of arbitrarily large path-width and the disjoint union of arbitrarily many triangles).

Our Results. We prove a number of new parameterized complexity results for LBHOM, LSHOM and LIHOM by taking some property of the guest graph G as the parameter. In particular, we consider the graph parameters above. Our two main results, which are proven in Section 4, show that LBHOM and LSHOM are fixed-parameter tractable parameterized by the fracture number of G . These two results cannot be strengthened to the tree-depth of the guest graph, for which we prove para-NP-completeness in Section 6. Note that the latter results imply the known para-NP-completeness results for path-width of the guest graph [15]. In Section 6 we also prove that LBHOM and LSHOM are para-NP-complete when parameterized by the feedback vertex set number of the guest graph. This result and the para-NP-hardness for tree-depth motivated us to consider the fracture number as a natural remaining graph parameter for obtaining an fpt algorithm.

The above para-NP-completeness results for LBHOM in fact even hold for 3-FOLDCOVER, the restriction of LBHOM to input pairs (G, H) where $|V(G)| = 3|V(H)|$. The k -FOLDCOVER

problem was introduced in [7] (where it was called the k -GRAPH COVERING problem). In fact, the aforementioned result of [15] on LBHOM for path-width is the first proof that 3-FOLDCOVER is NP-complete, and the proof can easily be adapted for $k \geq 4$, as observed by Klavík [47].

In Section 5 we prove that LIHOM is in XP and W[1]-hard when parameterized by the vertex cover number, or equivalently, the c -deletion set number for $c = 1$. We then show that the XP-result for LIHOM cannot be generalised to hold for $c \geq 2$. In fact, in the same section, we will determine the complexity of LIHOM on graphs with c -deletion set number at most k for every fixed pair of integers c and k . Our results for LBHOM, LSHOM and LIHOM are summarised, together with the known results, in Table 1.

Algorithmic Framework. The FPT algorithms for LBHOM and LSHOM are proven via a new algorithmic framework (described in detail in Section 3) that involves a reduction to an integer linear program (ILP) that has a wider applicability. To illustrate this, in Section 4 we also use our general framework to prove that ROLE ASSIGNMENT is FPT when parameterized by $c + ds_c$. We emphasize that in our framework the host graph H is not fixed, but part of the input, in contrast to other frameworks that include the locally constrained homomorphism problems (and that consequently work for more powerful graph parameters), such as the framework of locally checkable vertex partitioning problems [10, 67] or the framework of Gerber and Kobler [39] based on (feasible) interval degree constraint matrices.

Techniques. The main ideas behind our algorithmic ILP framework are as follows. Let G and H be the guest and host graphs, respectively. First, we observe that if G has a c -deletion of size at most k and there is a locally surjective homomorphism from G to H , then H must also have a c -deletion set of size at most k . However it does not suffice to compute c -deletion sets D_G and D_H for G and H , guess a partial homomorphism h from D_G to D_H , and use the structural properties of c -deletion sets to decide whether h can be extended to a desired homomorphism from G to H . This is because a homomorphism from G to H does not necessarily map D_G to D_H . Moreover, even if it did, vertices in $G \setminus D_G$ can still be mapped to vertices in D_H . Consequently, components of $G \setminus D_G$ can still be mapped to more than one component of $H \setminus D_H$. This makes it difficult to decompose the homomorphism from G to H into small independent parts. To overcome this challenge, we prove that there are small sets D_G and D_H of vertices in G and H , respectively, such that every locally surjective homomorphism from G to H satisfies:

1. the pre-image of D_H is a subset of D_G ,
2. D_H is a c' -deletion set for H for some c' bounded in terms of only $c + k$, and
3. all but at most k components of $G \setminus D_G$ have at most c vertices and, while the remaining components can be arbitrary large, their treewidth is bounded in terms of $c + k$.

As D_G and D_H are small, we can enumerate all possible homomorphisms from some subset of D_G to D_H . Condition 2 allows us to show that any locally surjective homomorphism from G to H can be decomposed into locally surjective homomorphisms from a small set of components of $G \setminus D_G$ (plus D_G) to one component of $H \setminus D_H$ (plus D_H). This enables us to formulate the question of whether a homomorphism from a subset of D_G to a subset of D_H can be extended to a desired homomorphism from G to H in terms of an ILP. Finally, Condition 3 allows us to efficiently compute the possible parts of the decomposition, that is, which (small) sets of components of $G \setminus D_G$ can be mapped to which components of D_H .

2 Preliminaries

We use standard notation from graph theory, as can be found in e.g. [19]. Let G be a graph. We denote the vertex set and edge set of G by $V(G)$ and $E(G)$, respectively. Let $X \subseteq V(G)$ be a set of vertices of G . The *subgraph of G induced by X* , denoted $G[X]$, is the graph with vertex set X and edge set $E(G) \cap [X]^2$. Whenever the underlying graph is clear from the context, we will sometimes refer to an induced subgraph simply by its set of vertices. We use $G \setminus X$ to denote the subgraph of G induced by $V(G) \setminus X$. Similarly, for $Y \subseteq E(G)$ we let $G \setminus Y$ be the subgraph of G obtained by deleting all edges in Y from G .

For a graph G and a vertex $u \in V(G)$, we let $N_G(u) = \{v \mid uv \in E(G)\}$ and $N_G[v] = N_G(u) \cup \{u\}$ denote the open and closed neighbourhood of u in G , respectively. We let $\Delta(G)$ be the maximum degree of G . Recall that we assume that the guest graph G does not contain self-loops, while the host graph H is permitted to have self-loops. In this case, by definition, $u \in N_H(u)$ if $uu \in E(H)$.

2.1 Parameterized Complexity

In parameterized complexity [17, 22, 35], the complexity of a problem is studied not only with respect to the input size, but also with respect to some problem parameter(s). The core idea behind parameterized complexity is that the combinatorial explosion resulting from the NP-hardness of a problem can sometimes be confined to certain structural parameters that are small in practical settings. We now proceed to the formal definitions.

A *parameterized problem* Q is a subset of $\Omega^* \times \mathbb{N}$, where Ω is a fixed alphabet. Each instance of Q is a pair (I, κ) , where $\kappa \in \mathbb{N}$ is called the *parameter*. A parameterized problem Q is *fixed-parameter tractable* (FPT) [17, 22, 35], if there is an algorithm, called an *FPT-algorithm*, that decides whether an input (I, κ) is a member of Q in time $f(\kappa) \cdot |I|^{\mathcal{O}(1)}$, where f is a computable function and $|I|$ is the size of the input instance. The class FPT denotes the class of all fixed-parameter tractable parameterized problems.

A parameterized problem Q is *FPT-reducible* to a parameterized problem Q' if there is an algorithm, called an *FPT-reduction*, that transforms each instance (I, κ) of Q into an instance (I', κ') of Q' in time $f(\kappa) \cdot |I|^{\mathcal{O}(1)}$, such that $\kappa' \leq g(\kappa)$ and $(I, \kappa) \in Q$ if and only if $(I', \kappa') \in Q'$, where f and g are computable functions. By *fpt-time*, we denote time of the form $f(\kappa) \cdot |I|^{\mathcal{O}(1)}$, where f is a computable function. Based on the notion of FPT-reducibility, a hierarchy of parameterized complexity, *the W-hierarchy* $= \bigcup_{t \geq 0} W[t]$, where $W[t] \subseteq W[t+1]$ for all $t \geq 0$, has been introduced, in which the 0-th level $W[0]$ is the class FPT. The notions of hardness and completeness have been defined for each level $W[i]$ of the W-hierarchy for $i \geq 1$ [17, 22]. It is commonly believed that $W[1] \neq \text{FPT}$ (see [17, 22]). This assumption has served as the main working hypothesis of fixed-parameter intractability. The class XP contains parameterized problems that can be solved in $\mathcal{O}(|I|^{f(\kappa)})$ time, where f is a computable function. It contains the class $W[t]$, for all $t \geq 0$, and every problem in XP is polynomial-time solvable when the parameter is bounded by a constant. The class **paraNP** is the class of parameterized problems that can be solved by non-deterministic algorithms in $f(\kappa) \cdot |I|^{\mathcal{O}(1)}$ time, where f is a computable function. A problem is *paraNP-hard* if it is NP-hard for a constant value of the parameter [35].

2.2 Graph Parameters

A (k, c) -*extended deletion set* for G is a set $D \subseteq V(G)$ such that:

- every component of $G \setminus D$ either has at most c vertices or has a c -deletion set of size at most k and
- at most k components of $G \setminus D$ have more than c vertices.

We need the following well-known fact:

► **Proposition 1** ([53]). *Let G be a graph and let k and c be natural numbers. Then, deciding whether G has a c -deletion set of size at most k is fixed-parameter tractable parameterized by $k + c$.*

Tree-depth. Tree-depth is closely related to treewidth, and the structure of graphs of bounded tree-depth is well understood [59]. A useful way of thinking about graphs of bounded tree-depth is that they are (sparse) graphs with no long paths.

The *tree-depth* of an undirected graph G , denoted by $\text{td}(G)$, is the smallest natural number k such that there is an undirected rooted forest F with vertex set $V(G)$ of height at most k for which G is a subgraph of $C(F)$, where $C(F)$ is called the *closure* of F and is the undirected graph with vertex set $V(F)$ having an edge between u and v if and only if u is an ancestor of v in F . A forest F for which G is a subgraph of $C(F)$ is also called a *tree-depth decomposition*, whose *depth* is equal to the height of the forest plus one. Informally a graph has tree-depth at most k if it can be embedded in the closure of a forest of height k . Note that if G is connected, then it can be embedded in the closure of a tree instead of a forest.

Treewidth. A *tree-decomposition* \mathcal{T} of a graph G is a pair (T, χ) , where T is a tree and χ is a function that assigns each tree node t a set $\chi(t) \subseteq V(G)$ of vertices such that the following conditions hold:

- (P1) For every edge $uv \in E(G)$, there is a tree node t such that $u, v \in \chi(t)$.
- (P2) For every vertex $v \in V(G)$, the set of tree nodes t with $v \in \chi(t)$ induces a non-empty subtree of T .

The sets $\chi(t)$ are called *bags* of the decomposition \mathcal{T} and $\chi(t)$ is the bag associated with the tree node t . The *width* of a tree-decomposition (T, χ) is the size of a largest bag minus 1. The *treewidth* of a graph G , denoted by $\text{tw}(G)$, is the minimum width over all tree-decompositions of G .

Relationships and Properties. The following proposition summarizes the known relationships between the parameters we consider.

► **Proposition 2** ([59]). *Let G be a graph and let k and c be natural numbers. Then:*

- if G has a c -deletion set of size at most k , then $\text{td}(G) \leq k + c$.
- if G has a (k', c) -extended deletion set of size at most k , then $\text{td}(G) \leq k' + k + c$.
- $\text{tw}(G) \leq \text{td}(G)$.

2.3 Locally Constrained Homomorphisms

We always allow self-loops for the host graph, but not for the guest graph (see also Section 1). Here we show some basic properties of locally constrained homomorphisms.

► **Observation 3.** *Let G and H be non-empty connected graphs and let ϕ be a locally surjective homomorphism from G to H . Then ϕ is surjective.*

Proof. Suppose not, and let C be the set of vertices in $V(H) \setminus \phi(V(G))$. Note that $C \neq \emptyset$ (because otherwise ϕ is surjective) and $\phi(V(G)) \neq \emptyset$ (because G is non-empty). Because H is connected, there is an edge $uv \in E(H)$ such that $u \in V(C)$ and $v \in \phi(V(G))$. But then, the mapping $\phi_x : N_G(x) \rightarrow N_H(v)$ is not surjective for any vertex $x \in \phi^{-1}(v)$. \blacktriangleleft

► **Observation 4.** Let G and H be non-empty connected graphs with a homomorphism ϕ from G to H and let $I \subseteq \phi(V(G))$. Let $P = \phi^{-1}(I)$ and $\phi_R = \phi|_P$. If ϕ is a locally injective, surjective or bijective homomorphism, then ϕ_R is a locally injective, surjective or bijective homomorphism, respectively, from $G[P]$ to $H[I]$.

Proof. Clearly, ϕ_R is a homomorphism from $G[P]$ to $H[I]$ and since ϕ_R is a restriction of ϕ , it follows that if ϕ is locally injective, then so is ϕ_R . It remains to show that if ϕ is locally surjective, then so is ϕ_R . Suppose, for contradiction, that ϕ is locally surjective, but ϕ_R is not. Then there is a vertex $v \in P$ such that $\phi_R(N_G(v) \cap P) \subsetneq N_H(\phi_R(v)) \cap I$. However, since ϕ does not map any vertex in $V(G) \setminus P$ to a vertex of I , it follows that $\phi(N_G(v)) \cap I \subsetneq N_H(\phi(v)) \cap I$, so $\phi(N_G(v)) \neq N_H(\phi(v))$. Thus ϕ is not surjective, a contradiction. \blacktriangleleft

► **Observation 5.** Let G and H be graphs, let $D \subseteq V(G)$, and let ϕ be a homomorphism from G to H . Then, for every component C_G of $G \setminus D$ such that $\phi(C_G) \cap \phi(D) = \emptyset$, there is a component C_H of $H \setminus \phi(D)$ such that $\phi(C_G) \subseteq C_H$. Moreover, if ϕ is locally injective/surjective/bijective, then $\phi_R = \phi|_{D \cup C_G}$ is a homomorphism from $G' = G[D \cup C_G]$ to $H' = H[\phi(D) \cup C_H]$ that is locally injective/surjective/bijective for every $v \in V(C_G)$.

Proof. Suppose for a contradiction that this is not the case. Then, there is a component C_G of $G \setminus D$ and an edge $uv \in E(C_G)$ such that $\phi(u)$ and $\phi(v)$ are in different components of $H \setminus \phi(D)$. Therefore, $\phi(u)\phi(v) \notin E(H)$, contradicting our assumption that ϕ is a homomorphism.

Towards showing the second statement, first note that ϕ_R is a homomorphism from G' to H' . Moreover, $N_G[v] = N_{G'}[v]$ for every vertex $v \in V(C_G)$, so if ϕ is locally injective/surjective/bijective for a vertex $v \in V(C_G)$, then so is ϕ_R . \blacktriangleleft

The following lemma is a basic but crucial observation showing that if $G \xrightarrow{s} H$ and G has a small c -deletion set, then so does H .

► **Lemma 6.** Let G and H be non-empty connected graphs, let $D \subseteq V(G)$ be a c -deletion set for G , and let ϕ be a locally surjective homomorphism from G to H . Then $\phi(D)$ is a c -deletion set for H .

Proof. Suppose not, then there is a component C_H of $H \setminus \phi(D)$ such that $|C_H| > c$. By Observation 3, it follows that ϕ is surjective and therefore $\phi^{-1}(C_H)$ is defined. Let $v \in \phi^{-1}(C_H)$. Then $v \notin D$ and therefore v is in some component C_G of $G \setminus D$. Observation 5 implies that $\phi_R = \phi|_{D \cup C_G}$ is a homomorphism from $G[D \cup C_G]$ to $H[\phi(D) \cup C_H]$ that is locally surjective for every $v \in V(C_G)$.

Now $|V(C_G)| < |V(C_H)|$, so there must be a vertex in $V(C_H) \setminus \phi_R(C_G)$. Because C_H is connected, there is an edge $xy \in E(C_H)$ such that $x \in V(C_H) \setminus \phi_R(C_G)$ and $y \in \phi_R(V(C_G))$. But then, the mapping $\phi_z : N_G(z) \rightarrow N_H(y)$ is not surjective for any vertex $z \in \phi_R^{-1}(y)$. \blacktriangleleft

2.4 Integer Linear Programming

Given a set X of variables and a set C of linear constraints (i.e. inequalities) over the variables in X with integer coefficients, the task in the feasibility variant of *integer linear*

programming (ILP) is to decide whether there is an assignment $\alpha : X \rightarrow \mathbb{Z}$ of the variables satisfying all constraints in C . We will use the following well-known result by Lenstra [54].

► **Proposition 7** ([27, 36, 45, 54]). *ILP is fpt parameterized by the number of variables.*

3 Our Algorithmic Framework

In this section we present our main algorithmic framework that will allow us to show that LSHOM, LBHOM and ROLE ASSIGNMENT are fpt parameterized by $k + c$, whenever the guest graph has c -deletion set number at most k . To illustrate the main ideas behind our framework, let us first explain these ideas for the examples of LSHOM and LBHOM. In this case we are given G and H and we know that G has a c -deletion set of size at most k . Because of Lemma 6, it then follows that if (G, H) is a yes-instance of LSHOM or LBHOM, then H also has a c -deletion set of size at most k . Informally, our next step, which is given in Section 3.1, is to compute a small (i.e. with size bounded by a function of $k + c$) set Φ of partial locally surjective homomorphisms such that (1) every locally surjective homomorphism from G to H augments some $\phi_P \in \Phi$ and (2) for every $\phi_P \in \Phi$, the domain of ϕ_P is a (k, c) -extended deletion set of G and the co-domain of ϕ_P is a c' -deletion set of H , where c' is bounded by a function of $k + c$. Here and in what follows, we say that a function $\phi : V(G) \rightarrow V(H)$ *augments* (or is an *augmentation* of) a partial function $\phi_P : V_G \rightarrow V_H$, where $V_G \subseteq V(G)$ and $V_H \subseteq V(H)$ if $v \in V_G \Leftrightarrow \phi(v) \in V_H$ and $\phi|_{V_G} = \phi_P$. This allows us to reduce our problems to (boundedly many) subproblems of the following form: Given a (k, c) -extended deletion set D_G for G , a c' -deletion set D_H for H , and a locally surjective (respectively bijective) homomorphism ϕ_P from D_G to D_H , find a locally surjective homomorphism ϕ from G to H that augments ϕ_P . In Section 3.2 we will then show how to formulate this subproblem as an integer linear program and in Section 3.3 we will show that we can efficiently construct and solve the ILP for this subproblem. Importantly, our ILP formulation will allow us to solve a much more general problem, where the host graph H is not explicitly given, but defined in terms of a set of linear constraints. We will then exploit this in Section 4 to solve not only LSHOM and LBHOM, but also the ROLE ASSIGNMENT problem.

3.1 Partial Homomorphisms for the Deletion Set

For a graph G and $m \in \mathbb{N}$ we let $D_G^m := \{v \in V(G) \mid \deg_G(v) \geq m\}$. The aim of this subsection is to show that there is a small set Φ of partial homomorphisms such that every locally surjective (respectively bijective) homomorphism from G to H augments some $\phi_P \in \Phi$ and, for every $\phi_P \in \Phi$, the domain of ϕ_P is a (k, c) -extended deletion set for G of size at most k and its co-domain is a c' -deletion set of size at most k for H . The main idea behind finding this set Φ is to consider the set of high degree vertices in G and H , i.e. the sets D_G^{k+c} and D_H^{k+c} . As it turns out (see Lemma 8), for every subset $D \subseteq D_G^{k+c}$, D is a $(k - |D|, c)$ -extended deletion set for G of size at most k and D_H^{k+c} is a c' -deletion set for H of size at most k , where $c' = kc(k + c)$. Moreover, as we will show in Lemma 9, every locally surjective (respectively bijective) homomorphism from G to H has to augment a locally surjective (respectively bijective) homomorphism from some induced subgraph of $G[D_G^{k+c}]$ to $D_H = D_H^{k+c}$. Intuitively, this holds because for every locally surjective homomorphism, only vertices of high degree in G can be mapped to a vertex of high degree in H and for every vertex in H , there must be a vertex in G that is mapped to it.

► **Lemma 8.** *Let G be a graph. If G has a c -deletion set of size at most k , then the set D_G^{k+c} is a $kc(k + c)$ -deletion set of size at most k . Furthermore, every subset $D \subseteq D_G^{k+c}$ is a*

$(k - |D|, c)$ -extended deletion set of G .

Proof. Let D' be a c -deletion set of G of size at most k . Then every vertex $v \in V(G) \setminus D'$ has degree at most $k + c - 1$, as each of its neighbours lies either in its own component of $G \setminus D'$ or in D' . Hence $D_G^{k+c} \subseteq D'$ and therefore $|D_G^{k+c}| \leq k$. Let C_1, \dots, C_m be the components of $G \setminus D'$ that contain a vertex adjacent to a vertex in $D' \setminus D_G^{k+c}$. Since $|D' \setminus D_G^{k+c}| \leq k$ and every vertex in $D' \setminus D_G^{k+c}$ has degree at most $k + c - 1$, we find that $m \leq k(k + c - 1)$ and $|C_1 \cup \dots \cup C_m \cup (D' \setminus D_G^{k+c})| \leq k + kc(k + c - 1) \leq kc(k + c)$. Since every component in $G \setminus D_G^{k+c}$ is either contained in a component of $G \setminus D'$ or contained in $C_1 \cup \dots \cup C_m \cup (D' \setminus D_G^{k+c})$, we find that D_G^{k+c} is a $kc(k + c)$ -deletion set.

Let $D \subseteq D_G^{k+c} \subseteq D'$. We will show that D is a $(k - |D|, c)$ -extended deletion set of G . The components of $G \setminus D$ that contain no vertices from $D' \setminus D$ are components of $G \setminus D'$ and thus have size at most c . Consider a component C of $G \setminus D$ that contains at least one vertex from $D' \setminus D$. Let $D_C = V(C) \cap (D' \setminus D)$. Every component of $C \setminus D_C$ is a component of $G \setminus D'$ and thus has size at most c . Moreover, D_C has size at most $|D' \setminus D| \leq k - |D|$. We conclude that every component of $G \setminus D$ either has size at most c or has a c -deletion set of size at most $k - |D|$. Furthermore, since there are at most $k - |D|$ vertices in $D_G^{k+c} \setminus D$, and every component of $G \setminus D$ that has size larger than c must contain a vertex of D_G^{k+c} , it follows that there are at most $k - |D|$ components of $G \setminus D$ that have size larger than c . This completes the proof. \blacktriangleleft

Using the above lemma, we now get to the most important result of this subsection, which informally speaking shows that there are only boundedly many possible pre-images of the vertices in D_H^{k+c} and, moreover, these pre-images are subsets of D_G^{k+c} .

► **Lemma 9.** *Let G and H be non-empty connected graphs such that G has a c -deletion set of size at most k . If there is a locally surjective homomorphism ϕ from G to H , then there is a set $D \subseteq D_G^{k+c}$ and a locally surjective homomorphism ϕ_P from $G[D]$ to $H[D_H^{k+c}]$ such that ϕ augments ϕ_P . If ϕ is locally bijective, then $D = D_G^{k+c}$ and ϕ_P is a locally bijective homomorphism.*

Proof. By Lemma 8, D_G^{k+c} is a $kc(k + c)$ -deletion set of size at most k . Furthermore, observe that for a locally surjective homomorphism ϕ from G to H , the inequality $\deg_G(v) \geq \deg_H(\phi(v))$ holds for every $v \in V(G)$ ($\deg_G(v) = \deg_H(\phi(v))$ holds in the locally bijective case). Since ϕ is surjective by Observation 3, this implies that $\phi(D_G^{k+c}) \supseteq D_H^{k+c}$ (and if ϕ is locally bijective, then $\phi(D_G^{k+c}) = D_H^{k+c}$). By Lemma 6, $\phi(D_G^{k+c})$ is a $kc(k + c)$ -deletion set for H . Let $D = \phi^{-1}(D_H^{k+c})$, so $D \subseteq D_G^{k+c}$ (note that $D = D_G^{k+c}$ if ϕ is locally bijective). Now $\phi|_D$ is a surjective map from D to D_H^{k+c} . Furthermore, $\phi(D_G^{k+c} \setminus D) \cap \phi(D) = \phi(D_G^{k+c} \setminus D) \cap D_H^{k+c} = \emptyset$. Moreover, for every $v \in V(G) \setminus D_G^{k+c}$, $\phi(v) \notin D_H^{k+c} = \phi|_D(D)$, since $\deg_G(v) \geq \deg_H(\phi(v))$. Furthermore, $\phi|_D$ is a homomorphism from $G[D]$ to $H[D_H^{k+c}]$ because ϕ is a homomorphism. We argue that $\phi|_D$ is locally surjective (bijective resp.) by contradiction. Suppose $\phi|_D$ is not locally surjective. Then there is a vertex $u \in D$ and a neighbour $v \in D_H^{k+c}$ of $\phi|_D(u)$ such that $v \notin \phi|_D(N_G(u) \cap D)$. Since ϕ is locally surjective, there must be $w \in N_G(u) \setminus D$ such that $\phi(w) = v$. This contradicts the fact that $\phi(V(G) \setminus D) \cap D_H^{k+c} = \emptyset$. Hence $\phi|_D$ is a locally surjective homomorphism. In the bijective case we just need to additionally observe that $\phi|_D$ restricted to the neighbourhood of any vertex $v \in D$ must be injective. This completes the proof. \blacktriangleleft

Finally, we show that we can easily compute all possible pre-images of D_H^{k+c} in any locally surjective (respectively bijective) homomorphism from G to H .

► **Lemma 10.** *Let G and H be non-empty connected graphs and let k, c be non-negative integers. For any $D \subseteq D_G^{k+c}$, we can compute the set Φ_D of all locally surjective (respectively bijective) homomorphisms ϕ_P from $G[D]$ to $H[D_H^{k+c}]$ in $\mathcal{O}(|D|^{|D|+2})$ time. Furthermore, $|\Phi_D| \leq |D|^{|D|}$.*

Proof. Let $D \subseteq D_G^{k+c}$ and suppose there is a surjective map $\phi_P : D \rightarrow D_H^{k+c}$. Then for every vertex $v \in D_H^{k+c}$, there must be a vertex $x \in D$ such that $\phi_P(x) = v$. Therefore $|D_H^{k+c}| \leq |D|$, so if this condition fails, then we can immediately return that $\Phi_D = \emptyset$.

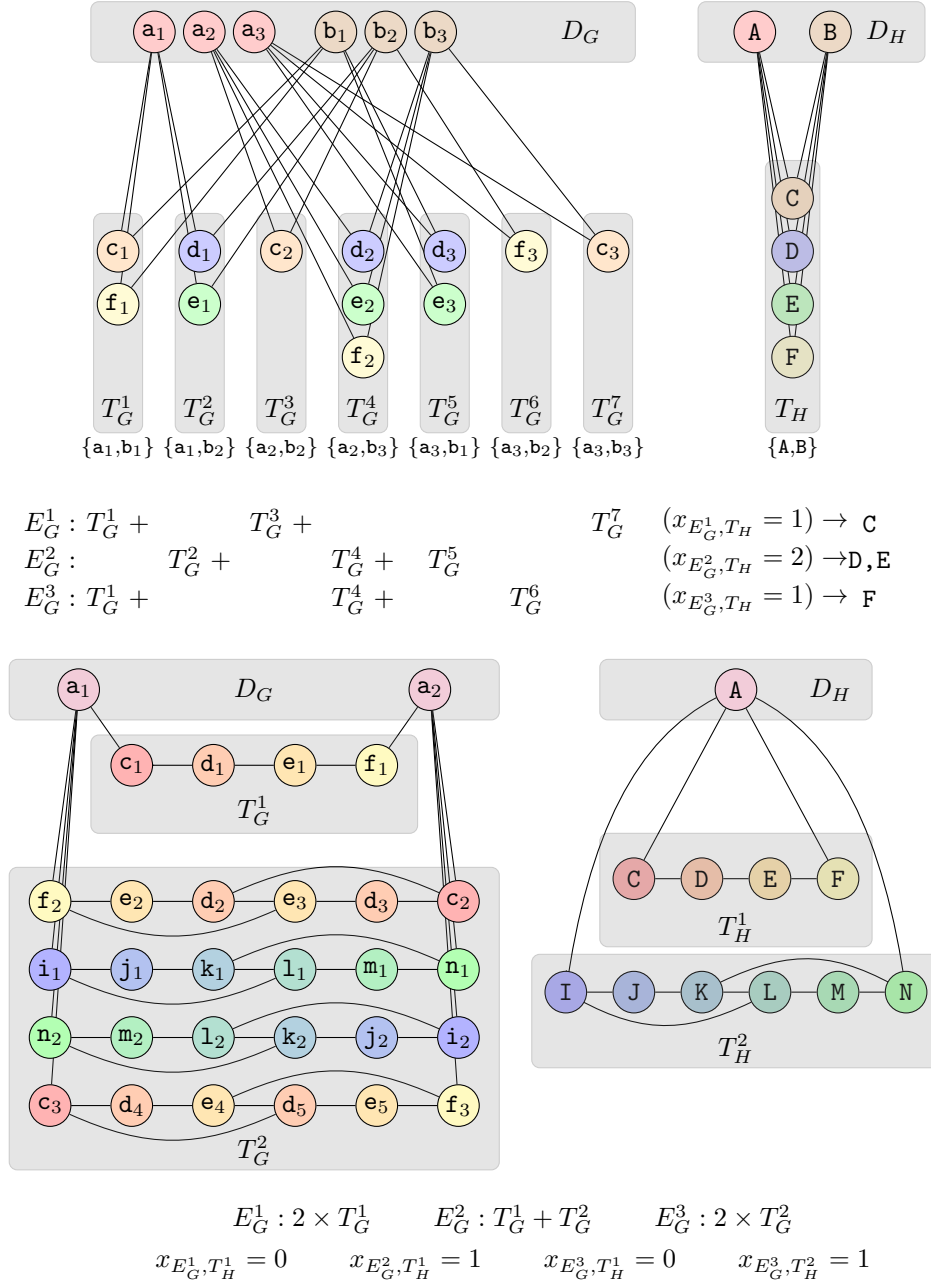
Otherwise, for each vertex of $|D|$, there are $|D_H^{k+c}| \leq |D|$ possible choices for where a map $\phi_P : D \rightarrow D_H^{k+c}$ could map this vertex. We can list all of the at most $|D|^{|D|}$ resulting maps in $\mathcal{O}(|D|^{|D|})$ time, and for each such map, we can check whether it is a locally surjective (respectively bijective) homomorphism in $\mathcal{O}(|D|^2)$ time. ◀

3.2 ILP Formulation

In this section, we will show how to formulate the subproblem obtained in the previous subsection in terms of an ILP instance. More specifically, we will show that the following problem can be formulated in terms of an ILP: given a partial locally surjective (respectively bijective) homomorphism ϕ_P from some induced subgraph D_G of G to some induced subgraph D_H of H , can this be augmented to a locally surjective (respectively bijective) homomorphism from G to H ? See the top of Figure 1 for an illustration of the subproblem for the simpler case when D_G is a vertex cover of G and we are looking for a locally bijective homomorphism. Moreover, we will actually show that for this to work, the host graph H does not need to be given explicitly, but can instead be defined by a certain system of linear constraints.

The main ideas behind our translation to ILP are as follows. Suppose that there is a locally surjective (respectively bijective) homomorphism ϕ from G to H that augments ϕ_P . Because ϕ augments ϕ_P , Observation 5 implies that ϕ maps every component C_G of $G \setminus V(D_G)$ entirely to some component C_H of $H \setminus V(D_H)$, moreover, $\phi|_{V(D_G) \cup V(C_G)}$ is already locally surjective (respectively bijective) for every vertex $v \in V(C_G)$. Our aim now is to describe ϕ in terms of its parts consisting of locally surjective (respectively bijective) homomorphisms from *extensions* of D_G in G , i.e. sets of components of $G \setminus D_G$ plus D_G , to *simple extensions* of D_H in H , i.e. single components of $H \setminus D_H$ plus D_H . Note that the main difficulty comes from the fact that we need to ensure that ϕ is locally surjective (respectively bijective) for every $d \in D_G$ and not only for the vertices within the components of $G \setminus D_G$. This is why we need to describe the parts of ϕ using sets of components of $G \setminus D_G$ and not just single components. However, as we will show, it will suffice to consider only minimal extensions of D_G in G , where an extension is minimal if no subset of it allows for a locally surjective (respectively bijective) homomorphism from it to some simple extension of D_H in H . The fact that we only need to consider minimal extensions is important for showing that we can compute the set of all possible parts of ϕ efficiently (see Section 3.3). Having shown this, we can create an ILP that has one variable $x_{\text{Ext}_G \text{Ext}_H}$ for every minimal extension Ext_G and every simple extension Ext_H such that there is a locally surjective (respectively bijective) homomorphism from Ext_G to Ext_H that augments ϕ_P . The value of the variable $x_{\text{Ext}_G \text{Ext}_H}$ now corresponds to the number of parts used by ϕ that map minimal extensions isomorphic to Ext_G to simple extensions isomorphic to Ext_H that augment ϕ_P . We can then use linear constraints on these variables to ensure that:

(SB2') H contains exactly the right number of extensions isomorphic to Ext_H required by the assignment for $x_{\text{Ext}_G \text{Ext}_H}$,



■ **Figure 1** Top: A locally bijective homomorphism from a graph G (left) to a graph H (right), augmenting a partial homomorphism mapping the vertices of the vertex cover D_G into $D_H = \{A, B\}$. The i th vertex of G mapped to some vertex x of H is denoted x_i . Vertices in $G \setminus D_G$ are grouped by type (e.g. $\{c_1\}$ and $\{f_1\}$ have type T_G^1), each T_G^i is characterised by the neighbours of its vertices, recalled below each column. Vertices in $H \setminus D_H$ all have the same type T_H . Rows Ext_G^i are extensions that can be minimally ϕ_P -B-mapped to T_H (in particular, each a_i and b_i must have a neighbour in some type in Ext_G^i , which can be used as a pre-image of any vertex in $\{C, D, E, F\}$). Using Ext_G^1 once (for colour C), Ext_G^2 twice (for colours D and E) and Ext_G^3 once (for colour F) yields the given locally bijective homomorphism, and it can be verified that each a_i and b_i indeed has all four colours in its neighbourhood.

Bottom: a locally surjective homomorphism from a graph G (left) to a graph H (right), where D_G is a 6-deletion set. The extensions $\text{Ext}_G^1, \text{Ext}_G^2, \text{Ext}_G^3$ can be minimally ϕ_P -S-mapped to T_H^1 ; only Ext_G^3 can also be minimally ϕ_P -S-mapped to T_H^2 . Furthermore, T_G^1 and T_G^2 can each be weakly ϕ_P -S-mapped to some type in H (respectively T_H^1 and T_H^2). Using pair (Ext_G^2, T_H^1) and (Ext_G^3, T_H^2) once is sufficient to ensure that the mapping is locally surjective for each a_i .

- (B1') G contains exactly the right number of minimal extensions isomorphic to Ext_G required by the assignment for $x_{\text{Ext}_G \text{Ext}_H}$ (if ϕ is locally bijective),
- (S1') G contains at least the number of minimal extensions isomorphic to Ext_G required by the assignment for $x_{\text{Ext}_G \text{Ext}_H}$ (if ϕ is locally surjective),
- (S3') for every simple extension Ext_G of G that is not yet used in any part of ϕ , there is a homomorphism from Ext_G to some simple extension of D_H in H that augments ϕ_P and is locally surjective for every vertex in $\text{Ext}_G \setminus D_G$ (if ϕ is locally surjective).

Together, these constraints ensure that there is a locally surjective (respectively bijective) homomorphism ϕ from G to H that augments ϕ_P . See also the bottom of Figure 1 for an illustration of the main ideas. We are now ready to formalise these ideas. To do so, we need the following additional notation.

Given a graph D , an *extension* for D is a graph E containing D as an induced subgraph. It is *simple* if $E \setminus D$ is connected, and *complex* in general. Given two extensions $\text{Ext}_1, \text{Ext}_2$ of D , we write $\text{Ext}_1 \sim_D \text{Ext}_2$ if there is an isomorphism τ from Ext_1 to Ext_2 with $\tau(d) = d$ for every $d \in D$. Then \sim_D is an equivalence relation. Let the *types* of D , denoted \mathcal{T}_D , be the set of equivalence classes of \sim_D of simple extensions of D . We write \mathcal{T}_D^c to denote the set of types of D of size at most $|D| + c$, so $|\mathcal{T}_D^c| \leq (|D| + c)2^{\binom{|D|+c}{2}}$.

Given a complex extension E of D , let C be a connected component of $E \setminus D$. Then C has type $T \in \mathcal{T}_D$ if $E[D \cup C] \sim_D T$ (depending on the context, we also say that the extension $E[D \cup C]$ has type T). The *type-count* of E is the function $\text{tc}_E : \mathcal{T}_D \rightarrow \mathbb{N}$ such that $\text{tc}_E(T)$ for $T \in \mathcal{T}_D$ is the number of connected components of $E \setminus D$ with type T (in particular if E is simple, the type-count is 1 for E and 0 for other types). Note that two extensions are equivalent if and only if they have the same type-counts; this then also implies that there is an isomorphism τ between the two extensions satisfying $\tau(d) = d$ for every $d \in D$. We write $E \preceq E'$ if $\text{tc}_E(T) \leq \text{tc}_{E'}(T)$ for all types $T \in \mathcal{T}_D$. If E is an extension of D , we write $\mathcal{T}_D(E) = \{T \in \mathcal{T}_D \mid \text{tc}_E(T) \geq 1\}$ for the *set of types of E* and $\mathcal{E}_D(E)$ for the set of simple extensions of E . Moreover, for $T \in \mathcal{T}_D$, we write $\mathcal{E}_D(E, T)$ for the set of simple extensions in E having type T .

A *target description* is a tuple (D_H, c, CH) where D_H is a graph, c is an integer and CH is a set of linear constraints over variables x_T , $T \in \mathcal{T}_{D_H}^c$. Note that a type-count for D_H is an integer assignment of the variables x_T . A graph H satisfies the target description (D_H, c, CH) if it is an extension of D_H , $\text{tc}_H(T) = 0$ for $T \notin \mathcal{T}_{D_H}^c$, and setting $x_T = \text{tc}_H(T)$ for all $T \in \mathcal{T}_{D_H}^c$ satisfies all constraints in CH .

In what follows, we assume that the following are given: the graphs D_G, D_H , an extension G of D_G , a target description $\mathcal{D} = (D_H, c, \text{CH})$, and a locally surjective (respectively bijective) homomorphism $\phi_P : D_G \rightarrow D_H$. Let Ext_G be an extension of D_G with $\text{Ext}_G \preceq G$ and let $T_H \in \mathcal{T}_{D_H}^c$; note that we only consider $T_H \in \mathcal{T}_{D_H}^c$, because we assume that T_H is a type of a simple extension of a graph H that satisfies the target description \mathcal{D} . We say Ext_G can be *weakly ϕ_P -S-mapped* to a type T_H if there exists an augmentation $\phi : \text{Ext}_G \rightarrow T_H$ of ϕ_P such that ϕ is locally surjective for every $v \in \text{Ext}_G \setminus D_G$. We say that Ext_G can be *ϕ_P -S-mapped* (respectively *ϕ_P -B-mapped*) to a type T_H if there exists an augmentation $\phi : \text{Ext}_G \rightarrow T_H$ of ϕ_P such that ϕ is locally surjective (respectively locally bijective). Furthermore, Ext_G can be *minimally ϕ_P -S-mapped* (respectively *minimally ϕ_P -B-mapped*) to T_H if Ext_G can be ϕ_P -S-mapped (respectively ϕ_P -B-mapped) to T_H and no other extension Ext'_G with $\text{Ext}'_G \preceq \text{Ext}_G$ can be ϕ_P -S-mapped (respectively ϕ_P -B-mapped) to T_H . Let $\text{wSM}(G, D_G, \mathcal{D}, \phi_P)$ be the set of all pairs (T_G, T_H) such that $T_G \in \mathcal{T}_{D_G}(G)$ can be weakly ϕ_P -S-mapped to T_H . Let $\text{SM}(G, D_G, \mathcal{D}, \phi_P)$ be the set of all pairs (Ext_G, T_H) with $\text{Ext}_G \preceq G$, $T_H \in \mathcal{T}_{D_H}^c$ such that Ext_G can be minimally ϕ_P -S-mapped to T_H and let $\text{BM}(G, D_G, \mathcal{D}, \phi_P)$ be the set of all pairs

(Ext_G, T_H) with $\text{Ext}_G \preceq G$, $T_H \in \mathcal{T}_{D_H}^c$ such that Ext_G can be minimally ϕ_P -B-mapped to T_H . See also the bottom of Figure 1 for an illustration of these notions.

We now build a set of linear constraints. To this end, besides variables x_T for $T \in \mathcal{T}_H$, we introduce variables $x_{\text{Ext}_G T_H}$ for each $(\text{Ext}_G, T_H) \in \text{SM}$ (respectively BM), where here and in what follows $\text{wSM} = \text{wSM}(G, D_G, \mathcal{D}, \phi_P)$, $\text{SM} = \text{SM}(G, D_G, \mathcal{D}, \phi_P)$ and $\text{BM} = \text{BM}(G, D_G, \mathcal{D}, \phi_P)$.

$$(S1) \sum_{(\text{Ext}_G, T_H) \in \text{SM}} \text{tc}_{\text{Ext}_G}(T_G) * x_{\text{Ext}_G T_H} \leq \text{tc}_G(T_G) \text{ for every } T_G \in \mathcal{T}_{D_G}(G),$$

$$(B1) \sum_{(\text{Ext}_G, T_H) \in \text{BM}} \text{tc}_{\text{Ext}_G}(T_G) * x_{\text{Ext}_G T_H} = \text{tc}_G(T_G) \text{ for every } T_G \in \mathcal{T}_{D_G}(G),$$

$$(S2) \sum_{\text{Ext}_G: (\text{Ext}_G, T_H) \in \text{SM}} x_{\text{Ext}_G, T_H} = x_{T_H} \text{ for every } T_H \in \mathcal{T}_{D_H},$$

$$(B2) \sum_{\text{Ext}_G: (\text{Ext}_G, T_H) \in \text{BM}} x_{\text{Ext}_G, T_H} = x_{T_H} \text{ for every } T_H \in \mathcal{T}_{D_H},$$

$$(S3) \sum_{(T_G, T_H) \in \text{wSM}} x_{T_H} \geq 1 \text{ for every } T_G \in \mathcal{T}_{D_G}(G).$$

We refer to the bottom of Figure 1 for an illustration and note that (S1) corresponds to (S1'), (B1) corresponds to (B1'), (S2) and (B2) correspond to (SB2'), and (S3) corresponds to (S3').

► **Lemma 11.** *Let D_G and D_H be graphs, let G be an extension of D_G and let $\mathcal{D} = (D_H, c, \text{CH})$ be a target description. Moreover, let $\phi_P : V(D_G) \rightarrow V(D_H)$ be a locally surjective (respectively bijective) homomorphism from D_G to D_H . There exists a graph H satisfying \mathcal{D} and a locally surjective (respectively bijective) homomorphism ϕ augmenting ϕ_P if and only if the equation system $(\text{CH}, S1, S2, S3)$ (respectively $(\text{CH}, B1, B2)$) admits a solution.*

Proof. Towards showing the forward direction of the claim, let H be a graph satisfying $\mathcal{D} = (D_H, c, \text{CH})$ and let ϕ be a locally surjective (respectively bijective) homomorphism that augments ϕ_P .

Consider $T_H \in \mathcal{T}_{D_H}(H)$ and let $\text{Ext}_H \in \mathcal{E}_{D_H}(H, T_H)$. Let $P = G[\phi^{-1}(\text{Ext}_H)]$; note that $D_G \subseteq V(P)$ and therefore P is a (possibly) complex extension of D_G . Then because of Observation 4, we obtain that $\phi_R = \phi|_P$ is a locally surjective (respectively bijective) homomorphism that augments ϕ_P from P to Ext_H . Moreover, because of Observation 5, it follows that $P \setminus D_G$ is the union of a set $W_\phi(\text{Ext}_H)$ of components of $G - D_G$. Therefore, $W_\phi(\text{Ext}_H)$ can be ϕ_P -S-mapped (respectively ϕ_P -B-mapped) to T_H . Moreover, if $W_\phi(\text{Ext}_H)$ can be ϕ_P -S-mapped to T_H , then $W_\phi(\text{Ext}_H)$ also contains a subset $W_\phi^{\min}(\text{Ext}_H)$ of components that can be minimally ϕ_P -S-mapped to T_H .

Let $\text{wSM} = \text{wSM}(G, D_G, \mathcal{D}, \phi_P)$, $\text{SM} = \text{SM}(G, D_G, \mathcal{D}, \phi_P)$ and $\text{BM} = \text{BM}(G, D_G, \mathcal{D}, \phi_P)$. Let $X_T = \{x_{T_H} : T_H \in \mathcal{T}_{D_H}(H)\}$, $X_M = \{x_{\text{Ext}_G T_H} : (\text{Ext}_G, T_H) \in \text{SM}\}$ (respectively $X_M = \{x_{\text{Ext}_G T_H} : (\text{Ext}_G, T_H) \in \text{BM}\}$), and $X = X_T \cup X_M$. Let $\alpha : X \rightarrow \mathbb{N}$ be defined by setting:

- $\alpha(x_{T_H}) = \text{tc}_H(T_H)$ and
- $\alpha(x_{\text{Ext}_G T_H}) = |\{\text{Ext}_H \in \mathcal{E}_{D_H}(H, T_H) : \text{tc}_{\text{Ext}_G} = \text{tc}_{W_\phi^{\min}(\text{Ext}_H)}\}|$ (in the locally surjective case)
- $\alpha(x_{\text{Ext}_G T_H}) = |\{\text{Ext}_H \in \mathcal{E}_{D_H}(H, T_H) : \text{tc}_{\text{Ext}_G} = \text{tc}_{W_\phi(\text{Ext}_H)}\}|$ (in the locally bijective case)

We claim that the assignment α satisfies the equation system $(\text{CH}, S1, S2, S3)$ (respectively the equation system $(\text{CH}, B1, B2)$). Because H satisfies \mathcal{D} , it follows that α satisfies CH.

We start by showing the claim for the locally surjective case. Towards showing that (S3) is satisfied, consider a type $T_G \in \mathcal{T}_{D_G}(G)$ and let $\text{Ext}_G \in \mathcal{E}_{D_G}(G, T_G)$. Then, because of Observation 5, the mapping $\phi|_{\text{Ext}_G}$ maps Ext_G to some type $T_H \in \mathcal{T}_{D_H}(H)$ and therefore shows that Ext_G can be ϕ_P -S-mapped to T_H .

Towards showing (S1), let $T_G \in \mathcal{T}_{D_G}(G)$. Because of Observation 5, every extension $\text{Ext}_G \in \mathcal{E}_{D_G}(G, T_G)$ satisfies $\phi(\text{Ext}_G) \subseteq \text{Ext}_H$ for some simple extension Ext_H of D_H . In other words Ext_G is contained in the pre-image of exactly one simple extension Ext_H , showing that every extension $\text{Ext}_G \in \mathcal{E}_{D_G}(G, T_G)$ is counted at most once on the left side of the inequality in (S1) and therefore the left side is at most $\text{tc}_G(T_G)$.

Towards showing (S2), let $T_H \in \mathcal{T}_{D_H}$. Then, because $W_\phi^{\min}(\text{Ext}_H) \neq W_\phi^{\min}(\text{Ext}'_H)$ for every two distinct $\text{Ext}_H, \text{Ext}'_H \in \mathcal{E}_{D_H}(H, T_H)$, we obtain:

$$\begin{aligned} \sum_{\text{Ext}_G: (\text{Ext}_G, T_H) \in \text{SM}} \alpha(x_{\text{Ext}_G T_H}) &= \sum_{\text{Ext}_G: (\text{Ext}_G, T_H) \in \text{SM}} |\{ \text{Ext}_H \in \mathcal{E}_{D_H}(H, T_H) : \text{tc}_{\text{Ext}_G} = \text{tc}_{W_\phi^{\min}(\text{Ext}_H)} \}| \\ &= \text{tc}_H(T_H) \\ &= \alpha(T_H), \end{aligned}$$

as required.

Finally, if ϕ is locally bijective, we only have to show (B1) and (B2), which can be shown very similarly to (S1) and (S2). That is, (B1) can be shown very similarly to (S1) by using the additional observation that due to the definition of α in terms of $W_\phi(\text{Ext}_H)$ instead of $W_\phi^{\min}(\text{Ext}_H)$, every simple extension Ext_G also occurs in the pre-image of at least one simple extension Ext_H . Moreover, (B2) can be shown in the same manner as (S2), since $W_\phi(\text{Ext}_H) \neq W_\phi(\text{Ext}'_H)$ also holds for every two distinct $\text{Ext}_H, \text{Ext}'_H \in \mathcal{E}_{D_H}(H, T_H)$.

Towards showing the reverse direction, let $\alpha : X \rightarrow \mathbb{N}$ be an assignment satisfying the equation system (CH, S1, S2, S3) (respectively the equation system (CH, B1, B2)). Let H be the unique graph consisting of D_H and $\alpha(T_H)$ extensions of D_H of type T_H for every $T_H \in \mathcal{T}_{D_H}$. Then H satisfies (D_H, c, CH) and $\text{tc}_H(T_H) = \alpha(x_{T_H})$.

We now define a function $\phi : V(G) \rightarrow V(H)$, which will be a locally surjective (respectively, locally bijective) homomorphism that augments ϕ_P as follows.

Let \mathcal{A} be the multiset containing each pair $(\text{Ext}_G, T_H) \in \text{SM}$ (respectively BM) exactly $\alpha(x_{\text{Ext}_G T_H})$ times. Because of (S2) (respectively (B2)), there is a bijection γ_{T_H} between $\mathcal{A}_{T_H} = \{ (\text{Ext}_G, T_H) : (\text{Ext}_G, T_H) \in \mathcal{A} \}$ and $\mathcal{E}_{D_H}(H, T_H)$ for every $T_H \in \mathcal{T}_{D_H}(H)$. Let γ be the bijection between \mathcal{A} and the extensions $\mathcal{E}_{D_H}(H)$ given by $\gamma((\text{Ext}_G, T_H)) = \gamma_{T_H}((\text{Ext}_G, T_H))$.

Because the proof now diverges quite significantly for the locally surjective and locally bijective cases, we start by showing the remainder of the proof for the former case and then show how to adapt the proof in latter (easier) case.

Because of (S1), there is a function β from \mathcal{A} to the complex extensions of G such that:

- $\text{tc}_{\beta((\text{Ext}_G, T_H))} = \text{tc}_{\text{Ext}_G}$ for every $(\text{Ext}_G, T_H) \in \mathcal{A}$,
- $\beta(A) \cap \beta(A') = \emptyset$ for every two distinct A and A' in \mathcal{A} .

Let $A = (\text{Ext}_G, T_H) \in \mathcal{A}$. We set $P_A = \beta(A)$ and $I_A = \gamma(A)$. Because $A \in \text{SM}$, there is a locally surjective homomorphism ϕ_A from P_A to I_A that augments ϕ_P . Let $\mathcal{E}_\mathcal{A}$ be the set of simple extensions Ext_G in $\mathcal{E}_{D_G}(G)$ for which there is an $A \in \mathcal{A}$ such that Ext_G is an induced subgraph of $\beta(A)$. Moreover, let $\bar{\mathcal{E}}_\mathcal{A}$ be the set of all remaining simple extensions in $\mathcal{E}_{D_G}(G)$, i.e. the set of all simple extensions Ext_G in $\mathcal{E}_{D_G}(G) \setminus \mathcal{E}_\mathcal{A}$. Consider a simple extension Ext_G in $\bar{\mathcal{E}}_\mathcal{A}$. Then, because of (S3), there is a $T_H \in \mathcal{T}_{D_H}(H)$ and a corresponding extension $\text{Ext}_H \in \mathcal{E}_{D_H}(H, T_H)$ such that there is a homomorphism ϕ_{Ext_G} from Ext_G to Ext_H that augments ϕ_P , which is locally surjective for every $v \in V(\text{Ext}_G - D_G)$. We are now ready to define $\phi : V(G) \rightarrow V(H)$. That is we set $\phi(v)$ to be equal to:

- $\phi_P(v)$ if $v \in D_G$,
- $\phi_{\text{Ext}_G}(v)$ if $v \in V(\text{Ext}_G)$ for some simple extension $\text{Ext}_G \in \bar{\mathcal{E}}_\mathcal{A}$, and
- $\phi_A(v)$ if $v \in V(\text{Ext}_G - D_G)$ for some $\text{Ext}_G \in \beta(A)$ and $A \in \mathcal{A}$.

It remains to show that ϕ is a locally surjective homomorphism from G to H that augments ϕ_P . Clearly, ϕ augments ϕ_P by definition and because ϕ_{Ext_G} does so too for every simple extension Ext_G in $\bar{\mathcal{E}}_{\mathcal{A}}$, as does ϕ_A for every $A \in \mathcal{A}$. Moreover, ϕ is also a homomorphism, because every edge $\{u, v\} \in E(G)$ is contained in $G[\text{Ext}_G]$ for some simple extension Ext_G in $\mathcal{E}_{D_G}(G)$ and ϕ maps Ext_G according to some homomorphism ϕ_{Ext_G} (if $\text{Ext}_G \in \bar{\mathcal{E}}_{\mathcal{A}}$ or some homomorphism ϕ_A (otherwise). For basically the same reason, i.e. because every ϕ_{Ext_G} and every ϕ_A is locally surjective for every vertex in $V(G) \setminus D_G$, ϕ is locally surjective for every vertex $v \in V(G) \setminus D_G$. Towards showing that ϕ is also locally surjective for every $d \in D_G$, let n_H be any neighbour of $\phi(d)$ in H . If $n_H \in \phi_P(D_G)$, then there is a neighbour n_G of d in D_G with $\phi(n_G) = n_H$, because ϕ_P is locally surjective. If on the other hand $n_H \in V(\text{Ext}_H - D_H)$ for some $\text{Ext}_H \in T_{D_H}(H, T_H)$ with $T_H \in \mathcal{T}_{D_H}(H)$, then there is a neighbour n_G of d in $\beta(\gamma^{-1}(T_H))$ with $\phi(n_G) = n_H$, because ϕ (restricted to $\beta(\gamma^{-1}(T_H))$) is a locally surjective homomorphism from $\beta(\gamma^{-1}(T_H))$ to Ext_H .

This completes the proof for the locally surjective case. We now complete the proof for the locally bijective case. First note that because of (B1), the function β from \mathcal{A} to the complex extensions of G is bijective. Moreover, if $A = (\text{Ext}_G, T_H) \in \mathcal{A}$, then because $A \in \text{BM}$, there is a locally bijective homomorphism ϕ_A from $P_A = \beta(A)$ to $I_A = \gamma(A)$ that augments ϕ_P . This now allows us to directly define $\phi : V(G) \rightarrow V(H)$. That is, we set $\phi(v)$ to be equal to:

- $\phi_P(v)$ if $v \in D_G$ and
- $\phi_A(v)$ if $v \in V(\text{Ext}_G - D_G)$ for some $\text{Ext}_G = \beta(A)$ and $A \in \mathcal{A}$.

It remains to show that ϕ is a locally bijective homomorphism from G to H that augments ϕ_P . Note that we can assume that ϕ is already a locally surjective homomorphism that augments ϕ_P , using the same arguments as for the locally surjective case. Thus it only remains to show that ϕ is also locally injective for every $d \in D_G$. Suppose not, then there are two distinct neighbours n_G and n'_G that are mapped to the same neighbour n_H of $\phi(d)$ in H . This is clearly not possible if both n_G and n'_G are in D_G because ϕ_P is locally bijective on D_G . Moreover, this can also not be the case if exactly one of n_G and n'_G is in D_G , because then $n_H \in V(D_H)$, but because ϕ augments ϕ_P , the other cannot be mapped to D_H . Therefore, we can assume that n_G and n'_G are outside of D_G . Let $\text{Ext}_H \in \mathcal{E}_{D_H}(H)$ be the simple extension containing n_H . Then, n_G and n'_G must be mapped by $\phi_{\gamma^{-1}(\text{Ext}_H)}$, but this is not possible because $\phi_{\gamma^{-1}(\text{Ext}_H)}$ is locally bijective. \blacktriangleleft

3.3 Constructing and Solving the ILP

The main aim of this section is to show the following theorem.

► **Theorem 12.** *Let G be a graph, let D_G be a (k, c) -extended deletion set (respectively a c -deletion set) of size at most k for G , let $\mathcal{D} = (D_H, c', \text{CH})$ be a target description and let $\phi_P : D_G \rightarrow D_H$ be a locally surjective (respectively bijective) homomorphism from D_G to D_H . Then, deciding whether there is a locally surjective (respectively bijective) homomorphism that augments ϕ_P from G to any graph satisfying CH is fpt parameterized by $k + c + c'$.*

To prove Theorem 12, we need to show that we can construct and solve the ILP instance given in the previous section. The main ingredient for the proof of Theorem 12 is Lemma 17, which shows that we can efficiently compute the sets wSM , SM , and BM . We start by showing that the set $\mathcal{T}_{D_G}(G)$ can be computed efficiently and has small size.

► **Lemma 13.** *Let G be a graph and let D_G be a (k, c) -extended deletion set of size at most k for G . Then, $\mathcal{T}_{D_G}(G)$ has size at most $k + (|D_G| + c)2^{\binom{|D_G|+c}{2}}$ and computing $\mathcal{T}_{D_G}(G)$ and tc_G is fpt parameterized by $|D_G| + k + c$.*

Proof. Because $|\mathcal{T}_G(G) \setminus \mathcal{T}_G^c| \leq k$ and $|\mathcal{T}_G^c| \leq (|D_G| + c)2^{\binom{|D_G|+c}{2}}$, we obtain that $|\mathcal{T}_{D_G}(G)| \leq k + (|D_G| + c)2^{\binom{|D_G|+c}{2}}$. Moreover, we can compute $\mathcal{T}_{D_G}(G)$ starting from the empty set and adding a simple extension $G[D_G \cup C]$ for some component C of $G \setminus D_G$ if $G[D_G \cup C]$ is not equivalent with respect to \sim_{D_G} to any element already added to $\mathcal{T}_{D_G}(G)$. Note that checking whether $G[D_G \cup C] \sim_D G[D_G \cup C']$ for two components C and C' of $G \setminus D_G$ is fpt parameterized by $|D_G| + k + c$, because $G[D_G \cup C]$ has treewidth at most $|D_G| + k + c$ for every component C of $G \setminus D_G$ (because of Proposition 2) and graph isomorphism is fpt parameterized by treewidth [56]. The same procedure can now also be used to compute all the non-zero entries of the function tc_G (i.e. the entries where $\text{tc}_G(T) \neq 0$), which provides us with a compact representation of tc_G . \blacktriangleleft

The following lemma is crucial for computing the sets SM and BM that are required to construct the ILP instance. Informally, we will show that if $(\text{Ext}_G, \text{Ext}_H) \in \text{SM}$ (or $(\text{Ext}_G, \text{Ext}_H) \in \text{BM}$), then Ext_G consists of only boundedly many (in terms of some function of the parameters) components, which will allow us to enumerate all possibilities for Ext_G in fpt-time.

► **Lemma 14.** *Let D_G and D_H be graphs and let ϕ_P be a locally surjective (respectively locally bijective) homomorphism from D_G to D_H . Moreover, let Ext_G be an extension of D_G that can be minimally ϕ_P -S-mapped (respectively minimally ϕ_P -B-mapped) to an extension Ext_H of D_H . Then, $\text{Ext}_G \setminus D_G$ consists of at most $|D_G||\text{Ext}_H \setminus D_H|$ components.*

Proof. We first show the statement of the lemma for the case when ϕ_P is locally surjective and therefore Ext_G can be minimally ϕ_P -S-mapped to Ext_H . Let $\phi : V(\text{Ext}_G) \rightarrow V(\text{Ext}_H)$ be a locally surjective homomorphism that augments ϕ_P and exists because Ext_G can be ϕ_P -S-mapped to Ext_H . Let Ext'_G be an extension of Ext_G with $\text{Ext}'_G \preceq \text{Ext}_G$. Then, because of Observation 5, it follows that $\phi|_{\text{Ext}'_G}$ is a homomorphism from Ext'_G to Ext_H that is locally surjective for every $v \in \text{Ext}'_G \setminus D_G$. Therefore, $\phi|_{\text{Ext}'_G}$ is a locally surjective homomorphism from Ext'_G to Ext_H if and only if Ext'_G is such that $\phi|_{\text{Ext}'_G}$ is locally surjective for every $d \in D_G$. That is, for every $d \in D_G$ and every neighbour n_H of $\phi(d)$ in Ext'_H , there has to exist a neighbour n_G of d in Ext'_G such that $\phi(n_G) = n_H$. Since this clearly holds if $n_H \in D_H$, because ϕ_P is a locally surjective homomorphism from D_G to D_H , we can assume that the above only has to hold for every $d \in D_G$ and $n_H \in \text{Ext}_H \setminus D_H$. Because ϕ is a locally surjective homomorphism from Ext_G to Ext_H , it follows that for every $d \in D_G$ and every neighbour n_H of $\phi(d)$ in Ext_H , there is a component, say C_{d,n_H} , containing a neighbour n_G of d in Ext_G such that $\phi(n_G) = \phi(n_H)$; note that because ϕ augments ϕ_P , it follows that $n_G \notin D_G$ because $n_H \notin D_H$. Let Ext'_G be the extension of D_G consisting of D_G and all components C_{d,n_H} for every $d \in D_G$ and $n_H \in \text{Ext}_H \setminus D_H$ as above. Then, $\phi|_{\text{Ext}'_G}$ is a locally surjective homomorphism from Ext'_G to Ext_H and since Ext_G is minimally ϕ_P -S-mapped to Ext_H and $\text{Ext}'_G \preceq \text{Ext}_G$, it follows that $\text{Ext}'_G = \text{Ext}_G$. However, $\text{Ext}'_G \setminus D_G$ consists of at most one component for every $d \in D_G$ and every $n_H \in \text{Ext}_H \setminus D_H$ and therefore it consists of at most $|D_G||\text{Ext}_H \setminus D_H|$ components, which concludes the proof for the case when ϕ_P is locally surjective.

It remains to show the statement of the lemma for the case when ϕ_P is locally bijective and Ext_G is minimally ϕ_P -B-mapped to Ext_H . Let $\phi : V(\text{Ext}_G) \rightarrow V(\text{Ext}_H)$ be a locally bijective homomorphism that augments ϕ_P and exists because Ext_G can be ϕ_P -B-mapped to Ext_H . Because ϕ is locally bijective, it is also locally surjective and therefore we can obtain the components C_{d,n_H} of $\text{Ext}_G \setminus D_G$ for $d \in D_G$ and $n_H \in \text{Ext}_H \setminus D_H$ using the same arguments as in the case when ϕ was locally surjective. As before, let Ext'_G be the extension of D_G containing all components C_{d,n_H} . Then, as we showed above, $\phi|_{\text{Ext}'_G}$ is a locally

surjective homomorphism from Ext'_G to Ext_H . Moreover, $\phi|_{\text{Ext}'_G}$ is also locally injective, because so is ϕ . Therefore, $\phi|_{\text{Ext}'_G}$ is a locally bijective homomorphism from Ext'_G to Ext_H , which because Ext_G can be minimally ϕ_P -B-mapped to Ext_H implies that $\text{Ext}_G = \text{Ext}'_G$, which concludes the proof of the lemma, because Ext'_G consists of at most $|D_G||\text{Ext}_H \setminus D_H|$ components. \blacktriangleleft

The following proposition is a slight generalisation of [15, Theorem 4] and will allow us to efficiently decide whether an extension Ext_G can be (weakly) S-mapped (respectively B-mapped) to some extension Ext_H .

► **Lemma 15** ([15, Theorem 4]). *Let G and H be graphs and let $\phi_P : D_G \rightarrow D_H$ be a locally surjective (respectively bijective) homomorphism from D_G to D_H for some subgraphs D_G of G and D_H of H . Then deciding whether there is a locally surjective (respectively bijective) homomorphism from G to H that augments ϕ_P can be achieved in $\mathcal{O}(|V(G)|((|V(H)2^{\Delta(H)})^\omega)^2\omega\Delta(H))$ time and is therefore fpt parameterized by $\omega + |V(H)|$, where ω is the treewidth of G .*

Proof. In [15, Theorem 4], the authors provide an algorithm that, given a graph G and a graph H , decides in $\mathcal{O}(|V(G)|((|V(H)2^{\Delta(H)})^\omega)^2\omega\Delta(H))$ time whether there is a locally surjective homomorphism from G to H , where ω is the treewidth of G . The algorithm uses a standard dynamic programming approach on a tree decomposition of G (of width ω), and it is straightforward to verify that the algorithm can be adapted with only minor modifications to an algorithm using the same run-time that decides whether there is a locally bijective homomorphism from G to H . Similarly, it is straightforward to adapt their algorithm to the case that one is additionally given a locally surjective (respectively bijective) homomorphism ϕ_P from some induced subgraph D_G of G to some induced subgraph D_H of H and one only looks for a locally surjective (respectively bijective) homomorphism from G to H that augments ϕ_P . \blacktriangleleft

The following corollary now follows directly from Lemma 15 and the definition of (weakly) S-mapped (respectively B-mapped).

► **Corollary 16.** *Let D_G and D_H be graphs and let ϕ_P be a locally surjective (respectively bijective) homomorphism from D_G to D_H . Let Ext_G be an extension of D_G having treewidth at most ω and let Ext_H be an extension of D_H . Then, testing whether Ext_G can be weakly ϕ_P -S-mapped, ϕ_P -S-mapped, or ϕ_P -B-mapped to Ext_H is fpt parameterized by $\omega + |\text{Ext}_H|$.*

We are now ready to show that we can efficiently compute the sets wSM, SM, and BM, which is the last crucial step towards constructing the ILP instance.

► **Lemma 17.** *Let G be a graph, let D_G be a (k, c) -extended deletion set (respectively a c -deletion set) of size at most k for G , let $\mathcal{D} = (D_H, c', \text{CH})$ be a target description and let ϕ_P be a locally surjective (respectively bijective) homomorphism from D_G to D_H . Then, the sets $\text{wSM} = \text{wSM}(G, D_G, \mathcal{D}, \phi_P)$ and $\text{SM} = \text{SM}(G, D_G, \mathcal{D}, \phi_P)$ (respectively the set $\text{BM} = \text{BM}(G, D_G, \mathcal{D}, \phi_P)$) can be computed in fpt-time parameterized by $k + c + c'$ and $|\text{SM}|$ (respectively $|\text{BM}|$) is bounded by a function depending only on $k + c + c'$. Moreover, the number of variables in the equation system $(\text{CH}, S1, S2, S3)$ (respectively $(\text{CH}, B1, B2)$) is bounded by a function depending only on $k + c + c'$.*

Proof. We only show the lemma for the set SM, since the proof for the set wSM can be seen as a special case and the proof for the set BM is identical. Let $(\text{Ext}_G, T_H) \in \text{SM}$. Then, Ext_G is an extension of D_G with $\text{Ext}_G \preceq G$, $T_H \in \mathcal{T}_{D_H}^c$, and Ext_G can be minimally ϕ_P -S-mapped to

T_H . Because Ext_G can be minimally ϕ_P -S-mapped to Ext_H , Lemma 14 implies that $\text{Ext}_G \setminus D_G$ consists of at most $\ell = |D_G||\text{Ext}_H \setminus D_H|$ components and, because $\text{Ext}_G \preceq G$, these are also components of $G \setminus D_G$. Therefore, there are at most $(|\mathcal{T}_{D_G}(G)|)^\ell$ non-isomorphic possibilities for Ext_G , which together with Lemma 13 and the facts that $\ell \leq kc'$ and $|\mathcal{T}_{D_H}^{c'}| \leq (k+c')2^{\binom{k+c'}{2}}$ shows that $|\text{SM}| \leq (|\mathcal{T}_{D_G}(G)|)^\ell |\mathcal{T}_{D_H}^{c'}| \leq (k + (k+c)2^{\binom{k+c}{2}})^\ell ((k+c')2^{\binom{k+c'}{2}})$. Therefore, $|\text{SM}|$ is bounded by a function depending only on $k+c+c'$. Towards showing that we can compute SM is fpt-time parameterized by $k+c+c'$, first note that the set $\mathcal{T}_{D_G}(G)$ can be computed in fpt-time parameterized by $k+c$ using Lemma 13. Similarly, the set $\mathcal{T}_{D_H}^{c'}$ can be computed in fpt-time parameterized by $k+c'$ using the same idea as in Lemma 13. This now allows us to compute the set \mathcal{A} containing all non-isomorphic possibilities for Ext_G , i.e. the set of all extensions Ext_G of D_G with $\text{Ext}_G \preceq G$ and $\sum_{T_G \in \mathcal{T}_{D_G}(G)} \text{tc}_{\text{Ext}_G}(T_G) \leq \ell$ in fpt-time parameterized by $k+c+c'$, i.e. in time at most $(|\mathcal{T}_{D_G}(G)|)^\ell$. But then, SM is equal to the set of all pairs $(\text{Ext}_G, \text{Ext}_H) \in \mathcal{A} \times \mathcal{T}_{D_H}^{c'}$ such that Ext_G can be minimally ϕ_P -S-mapped to Ext_H . Moreover, for every such pair $(\text{Ext}_G, \text{Ext}_H)$ we can test in fpt-time parameterized by $k+c+c'$ whether Ext_G can be ϕ_P -S-mapped to Ext_H using Corollary 16, because the treewidth of Ext_G is at most $k+c$ (Proposition 2). Therefore, we can compute SM by enumerating all pairs $(\text{Ext}_G, \text{Ext}_H) \in \mathcal{A} \times \mathcal{T}_{D_H}^{c'}$, testing for each of them whether Ext_G can be ϕ_P -S-mapped to Ext_H using Corollary 16, and keeping only those pairs $(\text{Ext}_G, \text{Ext}_H)$ such that Ext_G can be ϕ_P -S-mapped to Ext_H and Ext_G is inclusion-wise minimal among all pairs $(\text{Ext}'_G, \text{Ext}_H)$. \blacktriangleleft

We are now ready to prove the main result of this subsection.

Proof of Theorem 12. We first compute the sets wSM and SM (respectively the set BM), which because of Lemma 17 can be achieved in fpt-time parameterized by $k+c+c'$. This now allows us to construct the ILP instance \mathcal{I} given by the equation system (CH,S1,S2,S3) (respectively the equation system (CH,B1,B2)) in fpt-time parameterized by $k+c+c'$. Moreover, because the number of variables in \mathcal{I} is bounded by a function of $k+c+c'$ and we can employ Proposition 7 to solve \mathcal{I} in fpt-time parameterized by $k+c+c'$. Finally, because of Lemma 11, it follows that \mathcal{I} has a solution if and only if there is a locally surjective (respectively bijective) homomorphism that augments ϕ_P from G to any graph satisfying CH, which completes the proof of the theorem. \blacktriangleleft

4 Applications of Our Algorithmic Framework

In this section we show the main results of our paper, which can be obtained as an application of our framework given in the previous section. Our first result implies that LSHOM and LBHOM are fpt parameterized by the fracture number of the guest graph.

► **Theorem 18.** *LSHOM and LBHOM are fpt parameterized by $k+c$, where k and c are such that the guest graph G has a c -deletion set of size at most k .*

Proof. Let G and H be non-empty connected graphs such that G has a c -deletion set of size at most k . Let $D_H = H[D_H^{k+c}]$. We first verify whether H has a c -deletion set of size at most k using Proposition 1. Because of Lemma 6, we can return that there is no locally surjective (and therefore also no bijective) homomorphism from G to H if this is not the case. Therefore, we can assume in what follows that H also has a c -deletion set of size at most k , which together with Lemma 8 implies that $V(D_H)$ is a $kc(k+c)$ -deletion set of size at most k for H . Therefore, using Lemma 13, we can compute tc_H in fpt-time parameterized by

$k + c$. This now allows us to obtain a target description $\mathcal{D} = (D_H, c', \text{CH})$ with $c' = kc(k + c)$ for H , i.e. \mathcal{D} is satisfied only by the graph H , by adding the constraint $x_T = \text{tc}_H(T_H)$ to CH for every simple extension type $T_H \in \mathcal{T}_{D_H}^{c'}$; note that $\mathcal{T}_{D_H}^{c'}$ can be computed in fpt-time parameterized by $k + c$ because of Lemma 13.

Because of Lemma 9, we obtain that there is a locally surjective (respectively bijective) homomorphism ϕ from G to H if and only if there is a set $D \subseteq D_G^{k+c}$ and a locally surjective (respectively bijective) homomorphism ϕ_P from $D_G = G[D]$ to D_H such that ϕ augments ϕ_P . Therefore, we can solve LSHOM by checking, for every $D \subseteq D_G^{k+c}$ and every locally surjective homomorphism ϕ_P from $D_G = G[D]$ to D_H , whether there is a locally surjective homomorphism from G to H that augments ϕ_P . Note that there are at most 2^k subsets D and because of Lemma 10, we can compute the set Φ_D for every such subset in $\mathcal{O}(k^{k+2})$ time. Furthermore, because of Lemma 8, D is a $(k - |D|, c)$ -extended deletion set of size at most k for G . Therefore, for every $D \subseteq D_G^{k+c}$ and $\phi_P \in \Phi_D$, we can employ Theorem 12 to decide in fpt-time parameterized by $k + c$ (because $c' = kc(k + c)$), whether there is a locally surjective (respectively bijective) homomorphism from G to a graph satisfying \mathcal{D} that augments ϕ_P . Since H is the only graph satisfying \mathcal{D} , this completes the proof of the theorem. \blacktriangleleft

The proof of the following theorem is similar to the proof of Theorem 18. The major difference is that H is not given. Instead, we use Theorem 12 for a selected set of target descriptions. Each of these target descriptions enforces that graphs satisfying it have to be connected and have precisely h vertices, where h is part of the input for the ROLE ASSIGNMENT problem. Furthermore, we ensure that every graph H satisfying the requirements of the ROLE ASSIGNMENT problem must satisfy at least one of the selected target descriptions. The size of the set of considered target descriptions depends only on c and k , as it is sufficient to consider any small graph D_H and types of small simple extensions of D_H .

► **Theorem 19.** *ROLE ASSIGNMENT is fpt parameterized by $k + c$, where k and c are such that G has a c -deletion set of size at most k .*

Proof. Let G be a non-empty connected graph such that G has a c -deletion set of size at most k and let $h \geq 1$ be an integer.

In order to use Theorem 12 in this case, we need to ensure that the target descriptions used enforce that H is connected and has h vertices. Therefore for a fixed graph D on at most k vertices, we let CON_D be the set of all minimal sets $S \subseteq \mathcal{T}_D^{k+c}$ such that any extension H of D , which contains exactly the types in S is connected. Since $|\mathcal{T}_D^{k+c}|$ is bounded by $(2k+c)2^{\binom{2k+c}{2}}$, we can compute CON_D by considering every $S \subseteq \mathcal{T}_D^{k+c}$ and checking whether an extension $T \in \mathcal{T}_D$ of D containing precisely the types in S is connected. Since $|V(T)| \leq k + (k+c) \cdot |S|$ and checking connectivity takes linear time (using BFS or DFS) we can compute CON_D in time depending only on k and c . For $S \in \text{CON}_D$, we set CH_S to be the set of equations containing $x_T \geq 1$ for every $T \in S$ and $|V(D_H)| + \sum_{T \in \mathcal{T}_{D_H}^c} (|V(T)| - |V(D_H)|) * x_T = h$. Note that for D and $S \in \text{CON}_D$, any graph H satisfying the target description $(D, c+k, \text{CH}_S)$ is connected and has h vertices.

If there is a connected graph H on h vertices and a locally surjective homomorphism ϕ from G to H , then by Lemma 9 there is a set $D \subseteq D_G^{k+c}$ and a locally surjective homomorphism ϕ_P from $D_G = G[D]$ to $D_H = H[D_H^{k+c}]$ such that ϕ augments ϕ_P . Note that by Lemmas 6 and 8, D_H is a $(k+c)$ -deletion set of size at most k . This implies firstly that D_H is a graph on at most k vertices. Secondly, H is an extension of D_H such that $\text{tc}_H(T) = 0$ for $T \notin \mathcal{T}_{D_H}^{c+k}$ and, since H is also connected and has h vertices, H satisfies the target description $(D_H, c+k, \text{CH}_S)$ for at least one $S \in \text{CON}_{D_H}$.

Therefore, we can solve the ROLE ASSIGNMENT problem by checking for every $D \subseteq D_G^{k+c}$, every graph D_H on no more than k vertices, every $S \in \text{CON}_{D_H}$ and every locally surjective homomorphism ϕ_P from $D_G = G[D]$ to D_H , whether there is a graph H satisfying the target description $(D_H, k+c, \text{CH}_S)$ and a locally surjective homomorphism from G to H that augments ϕ_P . Note that there are at most 2^k subsets D . Furthermore, there are at most $k2^{\binom{k}{2}}$ graphs on at most k vertices and for each we can compute CON_D in time depending only on k and c . For each such graph D_H , there are at most $|\text{CON}_D| \leq 2^{(2k+c)2^{\binom{2k+c}{2}}}$ subsets S to consider. Lastly, because of Lemma 10, for every $D \subseteq D_G^{k+c}$, and any graph D_H on no more than k vertices, we can compute the set of locally surjective homomorphisms ϕ_P from $G[D]$ to D_H in time $\mathcal{O}(k^{k+2})$ time and there are at most $|D|^{|D|}$ ϕ_P to consider.

By Lemma 8, D is a $(k - |D|, c)$ -extended deletion set of size at most k for G . Therefore, for every $D \subseteq D_G^{k+c}$, every graph D_H on no more than k vertices, every $S \in \text{CON}_{D_H}$ and every locally surjective homomorphism ϕ_P from $D_G = G[D]$ to D_H , we can employ Theorem 12 to decide in fpt-time parameterized by $k+c$, whether there is a graph H satisfying $(D_H, c+k, \text{CH}_S)$ and a locally surjective homomorphism from G to H that augments ϕ_P . This completes the proof. \blacktriangleleft

5 Locally Injective Homomorphisms

The following result is well known. We include a proof for completeness.

► **Theorem 20 (Folklore).** *LIHOM is $\text{W}[1]$ -hard parameterized by $|V(G)|$. In particular, it is $\text{W}[1]$ -hard for all structural parameters of G .*

Proof. Let G be a complete graph on k vertices, and let H be an arbitrary graph. There exists a locally injective homomorphism ϕ from G to H if and only if H contains a clique K on k vertices. Indeed, for the forward direction, pick K to be the image of $V(G)$ under ϕ . Then $|K| = |V(G)| = k$ by the local injectivity of ϕ , and K is a clique. For the reverse direction, let ϕ be any bijection between $V(G)$ and K . The result follows from the fact that CLIQUE is $\text{W}[1]$ -hard. \blacktriangleleft

The locally injective case is more difficult in our setting since, in general, surjectivity helps to transfer structural parameters on G to similar structures on H (for example, in LSHOM and LBHOM the image of a deletion set is also a deletion set by Lemma 6). In LIHOM however, and even in the restricted case of graphs with bounded vertex cover number, no such property can be used to help find the image of a vertex cover, and exponential-time enumerations appear to be necessary. On the positive side, once such a partial mapping from a vertex cover of G to H has been found, our ILP framework can still be applied to map the remaining vertices in FPT-time. This leads to an XP algorithm for vertex cover number (Theorem 21). Interestingly, this result does not extend to c -deletion set number for $c > 1$: even if the mapping of the deletion set can be guessed, the fact that the non-trivial remaining components must be mapped to distinct subgraphs of H makes the problem difficult (see Theorem 24).

► **Theorem 21.** *LIHOM is in XP parameterized by the vertex cover number of G .*

Proof. As for the surjective and bijective cases, we employ a two-step algorithm that first guesses the image of the vertex cover through a partial homomorphism, then runs an ILP to map the remaining vertices. The ILP only requires FPT-time, however the first step needs an exhaustive enumeration of subsets of H (in the injective case, the image of a vertex cover does not have to be a vertex cover), hence the XP running time.

We use the definitions of *types* and *extensions* from Section 3.2. Let G be a connected graph with a vertex cover D_G of size k . Note that connected components of $G \setminus D_G$ are single vertices. We can thus define the type of a vertex $v \in G \setminus D_G$ to be the type of the component $\{v\}$. Note that there are at most $n_t = |\mathcal{T}_{D_G}(G)| \leq 2^{|D_G|}$ types in G .

The first step of the algorithm consists of guessing a partial homomorphism ϕ_P between D_G and H (there are $|V(G)|^{|D_G|}$ such homomorphisms). From now on, we look for locally injective homomorphisms ϕ from G to H such that $\phi(v) \in D_H \Rightarrow v \in D_G$ (such a function ϕ is more simply called a *solution*). We write $D_H = \phi_P(D_G)$. ϕ is *stable* if it is an augmentation of ϕ_P (i.e. if $\phi(v) \in D_H \Leftrightarrow v \in D_G$).

Let ϕ be a solution. For a vertex $h \in V(H)$, let $P_h = \phi^{-1}(h)$ be the pre-image of h . Sets P_h form a partition of $V(G)$, two distinct vertices of P_h may not share a neighbour in G (otherwise, ϕ would not be locally injective for this common neighbour). Thus, $P_h \setminus D_G$ has size at most $|D_G|$ (since by connectivity of G , each vertex in $G \setminus D_G$ has at least one neighbour in D_G), and may not contain two vertices with the same type. In particular there are at most $n_t^{|D_G|}$ distinct pre-images (up to equivalence).

Guess the pre-image of each $h \in D_H$ (for a total of $n_t^{|D_G|^2}$ branches), and let $D'_G = \bigcup_{h \in D_H} P_h$. Then D'_G is a vertex cover of G with size at most $|D_G|^2$. Define $\phi'_P : D'_G \rightarrow D_H$ such that $\phi'_P(v) = h$ whenever $v \in P_h$. Thus ϕ is a locally injective homomorphism extending ϕ'_P and $\phi(v) \in D_H \Leftrightarrow v \in D'_G$. Without loss of generality, we thus assume that we look for a stable solution ϕ (equivalently, we can set $D_G := D'_G$ and $\phi_P := \phi'_P$). Also note that all edges in $H \setminus D_H$ can be safely ignored (since no solution ϕ would map an edge $\{u, v\}$ of G to such an edge of H , and there is no surjectivity constraint), so that D_H is a vertex cover of H . Finally, we can assume that ϕ_P is locally injective (on its domain D_G), and that two vertices u, u' with $\phi_P(u) = \phi_P(u')$ do not share a neighbour in G , since otherwise no stable solution exist.

A (possibly empty) subset P of $V(G \setminus D_G)$ is a *candidate pre-image* of $h \in V(H \setminus D_H)$ if the following conditions hold:

1. $\phi_P(N_G(P)) \subseteq N_H(h) \cap D_H$,
2. any two vertices in P do not share a neighbour,
3. P contains at most one vertex from every type in G and
4. P has size at most $|D_G|$.

By the remarks above, given a stable solution ϕ , $P_h = \phi^{-1}(h)$ is a candidate pre-image of h . Conversely, building ϕ using candidate pre-images only leads to a stable solution, as formalised below.

▷ **Claim 22.** If $\phi : G \rightarrow H$ satisfies $\phi(v) = \phi_P(v)$ for $v \in D_G$ and $\phi^{-1}(h)$ is a candidate pre-image of h for each $h \in V(H \setminus D_H)$, then ϕ is a stable solution.

Proof. First note that ϕ is a homomorphism, i.e. $(\phi(u), \phi(v))$ is an edge in H for each edge (u, v) in G (since ϕ_P is a homomorphism for $u, v \in D_G$, by Condition 1 for $u \in D_G$ and $v \notin D_G$, and the case $u, v \notin D_G$ is impossible since D_G is a vertex cover). Pick $v \in V(G)$ and $u, u' \in N_G(v)$. We prove that $\phi(u) \neq \phi(u')$. If $u, u' \in D_G$, then u, u' share a neighbour and $\phi_P(u) \neq \phi_P(u')$. If $u \in D_G$ and $u' \notin D_G$, then $\phi(u) \in D_H$ and $\phi(u') \notin D_H$. If $u, u' \notin D_G$, then u, u' share a neighbour and they cannot be in the same candidate pre-image (by Condition 2). ◁

Also note that if $D_G \cup P \sim_{D_G} D_G \cup P'$ and $D_H \cup \{h\} \sim_{D_H} D_H \cup \{h'\}$, then P is a candidate pre-image of h if and only if P' is a candidate pre-image of h' . Let ICM be the set of pairs (Ext_G, T_H) , $\text{Ext}_G \in \mathcal{E}_{D_G}(G)$, $T_H \in \mathcal{T}_H$ such that Ext_G contains an extension $D_G \cup P$,

T_H contains an extension $D_H \cup \{h\}$, and P is a candidate pre-image of h (note that there is no minimality constraint for pairs in ICM).

We now build the ILP computing the pre-images of vertices in $H \setminus D_H$. Introduce a variable x_{Ext_G, T_H} for each pair $(\text{Ext}_G, T_H) \in \text{ICM}$. This variable represents the number of vertices h with type T_h whose pre-image P_h has $P_h \cup D_G \in \text{Ext}_G$. We introduce two types of constraints (see below). Constraint (I1) enforces that the pre-images P_h form a partition of $V(G \setminus D_G)$ by counting vertices of each type in each Ext_G and checking that the sum corresponds to the count in G . Constraint (I2) corresponds to the fact that each vertex in H needs to be assigned a (possibly empty) pre-image (the number of pairs involving a type T_H must correspond to the type-count of T_H in H).

$$(I1) \sum_{(\text{Ext}_G, T_H) \in \text{ICM}} \text{tc}_{\text{Ext}_G}(T_G) * x_{\text{Ext}_G, T_H} = \text{tc}_G(T_G) \text{ for every } T_G \in \mathcal{T}_{D_G}(G),$$

$$(I2) \sum_{\text{Ext}_G: (\text{Ext}_G, T_H) \in \text{ICM}} x_{\text{Ext}_G, T_H} = \text{tc}_H(T_H) \text{ for every } T_H \in \mathcal{T}_{D_H}.$$

From the above remarks, a stable solution ϕ yields a feasible solution for constraints (I1, I2). Conversely, a solution to the ILP gives integers x_{Ext_G, T_H} : for each pair (Ext_G, T_H) , pick $x = x_{\text{Ext}_G, T_H}$ new vertices h_1, \dots, h_x with type T_H , pick x sets P_1, \dots, P_x in $G \setminus D_G$ such that $P_i \cup D_G \in \text{Ext}_G$ (each P_i being disjoint from previously selected sets). Assign $\phi(v) = h_i$ for each $v \in P_i$, and $\phi(v) = \phi_P(v)$ for $v \in D_G$. Then, over all pairs (Ext_G, T_H) , the sets P_i form a partition of $V(G \setminus D_G)$ so ϕ is well defined, and each $\phi^{-1}(h)$ is a candidate pre-image of H . By the claim above, ϕ is indeed a locally injective homomorphism from G to H . \blacktriangleleft

Corollary 23. *For any constant k , LIHOM is polynomial-time solvable for graphs G with 1-deletion set number at most k .*

We actually obtain the following dichotomy for the complexity of LIHOM, where the $c = 1, k \geq 1$ case is already given by Corollary 23.

Theorem 24. *Let $c, k \geq 1$. Then LIHOM is polynomial-time solvable on guest graphs with a c -deletion set of size at most k if either $c = 1$ and $k \geq 1$ or $c = 2$ and $k = 1$; otherwise, it is NP-complete.*

Theorem 24 follows from Corollary 23 and the following three lemmas.

Lemma 25. *LIHOM is polynomial-time solvable for graphs G with a 2-deletion set number at most 1.*

Proof. Let G and H be connected graphs such that G has 2-deletion set number at most 1. If G has a 2-deletion set containing no vertices, then G contains at most two vertices, in which case we can solve LIHOM in polynomial time. Otherwise, we can find a 2-deletion set $\{v\}$ in polynomial time by trying all possibilities for v . Let p be the number of edges in $G[N_G(v)]$ and let w be a vertex of H . We claim that there is a locally injective homomorphism ϕ from G to H such that $\phi(v) = w$ if and only if $H[N_H(w)]$ has a matching on at least p edges and $d_G(v) \leq d_H(w)$.

Indeed, if such a locally injective homomorphism ϕ exists, then $d_G(v) \leq d_H(w)$ because ϕ is locally injective. Furthermore, for every edge xy in $G[N_G(v)]$, the homomorphism ϕ maps the vertices x and y to adjacent vertices of $H[N_H(w)]$, and since ϕ is locally injective, it cannot map two vertices of $N_G(v)$ to the same vertex in $N_H(w)$. Therefore $H[N_H(w)]$ must have a matching on at least p edges.

Now suppose that $H[N_H(w)]$ has a matching M on at least p edges and $d_G(v) \leq d_H(w)$. For each edge xy in $G[N_G(v)]$, let $\phi(x)$ and $\phi(y)$ be the endpoints of an edge in M (choosing a different edge of M for each edge xy). For the remaining vertices $x \in N_G(v)$, assign the

remaining vertices of $N_H(w)$ arbitrarily, such that no two vertices of $N_G(v)$ are assigned the same value (this can be done since $d_G(v) \leq d_H(w)$). Let $\phi(x) = w$ for all remaining vertices of G (i.e. the vertex v and all vertices non-adjacent to v that have a common neighbour with v). By construction, ϕ is a locally injective homomorphism from G to H .

The size of a maximum matching in a graph can be found in polynomial time [25]. Thus, by branching over the possible vertices $w \in V(H)$, we obtain a polynomial-time algorithm for LIHOM. \blacktriangleleft

To prove NP-hardness for results in Lemmas 26 and 27 below, we use a reduction from the H' -PARTITION problem when $H' = P_3$ (the 3-vertex path) or K_3 (the 3-vertex complete graph), respectively. Let H' be a fixed graph on h vertices. The H' -PARTITION problem takes as input a graph G' on hn vertices and the task is to decide whether the vertex set of G' can be partitioned into sets V_1, \dots, V_n , each of size h , such that $G'[V_i]$ contains H' as a subgraph for all $i \in \{1, \dots, n\}$. This problem is known to be NP-complete if $H' \in \{K_3, P_3\}$ [38, 46].

► **Lemma 26.** *For $c \geq 2$ and $k \geq 2$, LIHOM is NP-hard on graphs G with c -deletion set number k .*

Proof. We first consider the case when $k = 2$. Consider an instance G' of the P_3 -PARTITION problem on $3n$ vertices, where $n \geq c$. We construct a graph G as follows. For $i \in \{1, \dots, n\}$, add vertices a_i, b_i, c_i and d_i and edges $a_i b_i$ and $c_i d_i$. Then add vertices u and v and make u adjacent to a_i, b_i and d_i and v adjacent to a_i, c_i and d_i for all $i \in \{1, \dots, n\}$. Finally, add the edge uv . Note that $\{u, v\}$ is a minimum-size c -deletion set for G since $\deg_G(u) = \deg_G(v) > c$. Now let H be the graph obtained from G' by adding two vertices u' and v' that are adjacent to all the vertices in $V(G')$ and to each other. We claim that there is a locally injective homomorphism ϕ from G to H if and only if G' is a yes-instance of the P_3 -PARTITION problem.

Suppose that G' is a yes-instance of the P_3 -PARTITION problem and, for $i \in \{1, \dots, n\}$, let v_i^1, v_i^2, v_i^3 be the three vertices in V_i , such that v_i^2 is adjacent to v_i^1 and v_i^3 (v_i^1 may or may not be adjacent to v_i^3). Let $\phi : V(G) \rightarrow V(H)$ be the function such that $\phi(u) = u'$, $\phi(v) = v'$, and for $i \in \{1, \dots, n\}$, $\phi(a_i) = v_i^1$, $\phi(b_i) = \phi(c_i) = v_i^2$ and $\phi(d_i) = v_i^3$. Then ϕ is a locally injective homomorphism from G to H .

Now suppose that ϕ is a locally injective homomorphism from G to H . Now $\deg_G(u) = \deg_G(v) = 3n + 1$. Since H has $3n + 2$ vertices and ϕ is a locally injective homomorphism, it follows that $\phi(u)$ and $\phi(v)$ must be universal vertices in H . By symmetry, we may therefore assume that $\phi(u) = u'$ and $\phi(v) = v'$. Now u is adjacent to v and the vertices a_i, b_i and d_i for all $i \in \{1, \dots, n\}$. Similarly, v is adjacent to u and the vertices a_i, c_i and d_i for all $i \in \{1, \dots, n\}$. Since $\deg_G(u) = 3n + 1$, and ϕ is locally injective, it follows that $\phi(\{a_i, b_i, d_i \mid i \in \{1, \dots, n\}\}) = V(G')$. Similarly, since $\deg_G(v) = 3n + 1$, it follows that $\phi(\{a_i, c_i, d_i \mid i \in \{1, \dots, n\}\}) = V(G')$. Therefore $\phi(\{b_i \mid i \in \{1, \dots, n\}\}) = \phi(\{c_i \mid i \in \{1, \dots, n\}\})$. Renumbering the indices of the c_i and d_i vertices if necessary, we may therefore assume by symmetry that $\phi(b_i) = \phi(c_i)$ for all $i \in \{1, \dots, n\}$. Now, for all $i \in \{1, \dots, n\}$, the vertices a_i and b_i are adjacent in G , so $\phi(a_i)$ and $\phi(b_i)$ are adjacent in H . Furthermore the vertices c_i and d_i are adjacent in G , so $\phi(c_i) = \phi(b_i)$ and $\phi(d_i)$ are adjacent in H . We now set $V_i = \{\phi(a_i), \phi(b_i), \phi(d_i)\}$ and note that the V_i sets partition $V(G')$, and that $G'[V_i]$ contains a P_3 subgraph for all $i \in \{1, \dots, n\}$. This completes the proof of the case when $k = 2$.

To extend the proof to graphs with c -deletion number $k > 2$, we add $(k - 1)$ universal vertices to H and replace u with a k -clique K each of whose vertices is adjacent to v and a_i, b_i and d_i for all $i \in \{1, \dots, n\}$. \blacktriangleleft

► **Lemma 27.** *For $c \geq 3$ and $k \geq 1$, LIHOM is NP-hard on graphs G with c -deletion set number k .*

Proof. We first consider the case when $k = 1$. Consider an instance G' of the K_3 -PARTITION problem on $3n$ vertices, where $n \geq c$. Let H be the graph obtained from G' by adding a universal vertex w . Let G be the graph obtained by taking the disjoint union of n copies of K_3 and adding a universal vertex v . Note that $\{v\}$ forms a minimum-size c -deletion set for G since $\deg_G(v) > c$. We claim that there is a locally injective homomorphism ϕ from G to H if and only if G' is a yes-instance of the K_3 -PARTITION problem.

Indeed, suppose there is such a ϕ . Since ϕ is locally injective and the graphs G and H each have $3n$ vertices, the universal vertex v must be mapped to a universal vertex of H ; without loss of generality, we may therefore assume that $\phi(v) = w$. Since v and w are universal vertices of the same degree, it follows that ϕ is a bijection from $V(G)$ to $V(H)$. Every K_3 in the disjoint union part of G must therefore be mapped to a K_3 in $H \setminus \{w\} = G'$. Therefore G' is a yes-instance of the K_3 -PARTITION problem.

Now suppose that G' is a yes-instance of the K_3 -PARTITION problem. We let $\phi(v) = w$, and map the vertices of each K_3 in the disjoint union part of G to some V_i from the K_3 -partition of H , mapping each K_3 to a different set V_i . Clearly this is a locally injective homomorphism. This completes the proof of the case when $k = 1$. To extend the proof to graphs with c -deletion number $k > 1$, we add $(k - 1)$ universal vertices to G and H . ◀

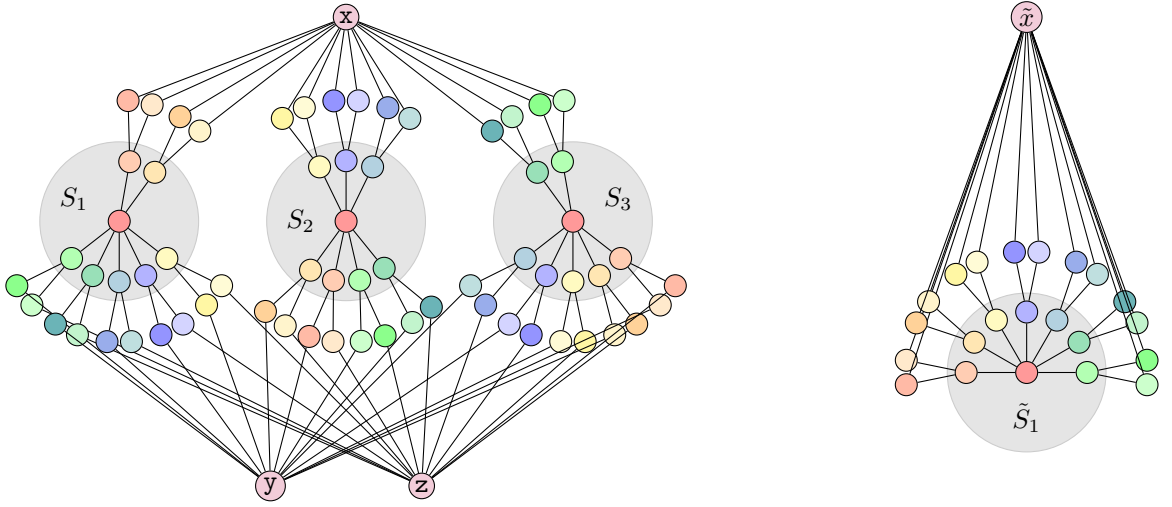
6 Bounded Tree-depth and Feedback Vertex Set Number

By Theorem 24, we already obtained paraNP-hardness for LIHOM parameterized by tree-depth or feedback vertex set number. In this section we show that our tractability results for LSHOM and LBHOM cannot be significantly extended, since both problems become paraNP-hard parameterized by tree-depth. Furthermore, the reduction we give here also provides paraNP-hardness for both LSHOM and LBHOM parameterized by the feedback vertex set number. We show this by replacing cycles with stars in the reduction provided in [15] for path-width. This strengthens their result from path-width to tree-depth and feedback vertex set number.

► **Theorem 28.** *LBHOM, or more specifically, 3-FOLDCOVER, and LSHOM are NP-complete on input pairs (G, H) where G has tree-depth at most 6 and H has tree-depth at most 4.*

Proof. First note that LBHOM, 3-FOLDCOVER and LSHOM are in NP. To prove NP-hardness for 3-FOLDCOVER and LSHOM we use a reduction from the 3-PARTITION problem. This problem takes as input a multiset A of $3m$ integers, denoted in what follows 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 $\sum_{1 \leq i \leq 3m} a_i = mb$. 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 [38], i.e. it remains NP-complete even if the problem is encoded in unary.

We first prove NP-hardness for 3-FOLDCOVER, which implies NP-hardness for LBHOM. Given an instance (A, b) of 3-PARTITION, we construct an instance of 3-FOLDCOVER consisting of connected graphs G and H with $|V(G)| = 3|V(H)|$ as follows. To construct G we take $3m$ disjoint copies S_1, \dots, S_{3m} of $K_{1,b}$ (stars), one for each element of A . For each $i \in \{1, \dots, 3m\}$, the vertices of S_i are labelled c^i, u_1^i, \dots, u_b^i , where c^i is the vertex of degree b in S_i (the centre



■ **Figure 2** An instance of LBHOM consisting of the graph G (left) and the graph H (right) corresponding to the instance (A, b) of 3-PARTITION, where $A = \{2, 3, 2\}$ and $b = 7$. As (A, b) is a yes-instance of the 3-PARTITION problem, there is a locally bijective homomorphism from G to H which is indicated by colours.

of the star). We add two new vertices p_j^i and q_j^i for each $i \in \{1, \dots, 3m\}$, $j \in \{1, \dots, b\}$, as well as two new edges $u_j^i p_j^i$ and $u_j^i q_j^i$. We then add three new vertices x , y and z . The vertex x is made adjacent to the vertices $p_1^i, p_2^i, \dots, p_{a_i}^i$ and $q_1^i, q_2^i, \dots, q_{a_i}^i$ for every $i \in \{1, \dots, 3m\}$. Finally, the vertex y is made adjacent to every vertex p_j^i that is not adjacent to x , and the vertex z is made adjacent to every vertex q_j^i that is not adjacent to x . This completes the construction of G . Note that $|V(G)| = 3|V(H)|$. For an example see Figure 2.

To construct H , we take m disjoint copies $\tilde{S}_1, \dots, \tilde{S}_m$ of $K_{1,b}$, where the vertices of each star \tilde{S}_i are labelled $\tilde{c}^i, \tilde{u}_1^i, \dots, \tilde{u}_b^i$. For each $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, b\}$, we add two vertices \tilde{p}_j^i and \tilde{q}_j^i and make both of them adjacent to \tilde{u}_j^i . Finally, we add a vertex \tilde{x} and make it adjacent to each of the vertices \tilde{p}_j^i and \tilde{q}_j^i . This finishes the construction of H . For an illustration see Figure 2.

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 $\phi(x) = \tilde{x}$. 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 neighbourhoods 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 $k \in \{1, \dots, m\}$, we define a set $A_k \subseteq A$ such that A_k contains element $a_i \in A$ if and only if $\phi(u_1^i) \in \{\tilde{u}_1^k, \dots, \tilde{u}_b^k\}$. Since ϕ is a bijection from $N_G^2(x)$ to $N_H^2(\tilde{x})$, the sets A_1, \dots, A_m are disjoint; moreover each element $a_i \in A$ is contained in exactly one of them. Since ϕ is degree preserving, each c^i has to be mapped onto a \tilde{c}^j (in the special case when $b = 3$, we can argue this using the distance to x as before). Additionally, since ϕ is locally bijective for every $i \in \{1, \dots, 3m\}$, there is a bijection from $N_G(c^i) = \{u_1^i, \dots, u_b^i\}$ to $N_H(\tilde{c}^j) = \{\tilde{u}_1^j, \dots, \tilde{u}_b^j\}$ for the $j \in \{1, \dots, m\}$ for which $\phi(c^i) = \tilde{c}^j$. Combining this and the previous argument implies that $\sum_{a \in A_i} a = b$ for all $i \in \{1, \dots, m\}$. Hence A_1, \dots, A_m is a 3-partition of A .

For the reverse direction, suppose there exists a 3-partition A_1, \dots, A_m of A . We define a mapping ϕ as follows. We first set $\phi(x) = \phi(y) = \phi(z) = \tilde{x}$. Let $A_i = \{a_r, a_s, a_t\}$ be any set of the 3-partition. We map the vertices of S_r, S_s, S_t to the vertices of \tilde{S}_i in the following way: $\phi(c_r) = \phi(c_s) = \phi(c_t) = \tilde{c}_i$; $\phi(u_j^r) = \tilde{u}_j^i$ for each $j \in \{1, \dots, b\}$; $\phi(u_j^s) = \tilde{u}_{a_r+j}^i$ for each $j \in \{1, \dots, a_s + a_t\}$ and $\phi(u_j^s) = \tilde{u}_{a_r+j-b}^i$ for $j \in \{a_s + a_t + 1, \dots, b\}$; and $\phi(u_j^t) = \tilde{u}_{a_r+a_s+j}^i$ for each $j \in \{1, \dots, a_t\}$ and $\phi(u_j^t) = \tilde{u}_{a_r+j-b}^i$ for $j \in \{a_t + 1, \dots, b\}$. It remains to map the vertices p_j^i and q_j^i for each $i \in \{1, \dots, 3m\}$ and $j \in \{1, \dots, b\}$. 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 neighbour of p_j^i and q_j^i . Suppose \tilde{u}_ℓ^k is the image of u_j^i , i.e. suppose that $\phi(u_j^i) = \tilde{u}_\ell^k$. Then we map p_j^i and q_j^i to \tilde{p}_ℓ^k and \tilde{q}_ℓ^k , respectively. We now consider the neighbours of y and z in G . By construction, the neighbourhood 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}_ℓ^k , the other of the form \tilde{q}_ℓ^k . Hence, we need to map half the neighbours of y to vertices of the form \tilde{p}_ℓ^k and half the neighbours of y to vertices of the form \tilde{q}_ℓ^k in order to make ϕ a locally bijective homomorphism. The same should be done with the neighbours of z . For every vertex \tilde{u}_ℓ^k in H , we do as follows. By construction, exactly three vertices of G are mapped to \tilde{u}_ℓ^k , and exactly two of these vertices, say u_j^i and u_h^g , are at distance 2 from y in G . We set $\phi(p_j^i) = \tilde{p}_\ell^k$ and $\phi(p_h^g) = \tilde{q}_\ell^k$. We also set $\phi(q_j^i) = \tilde{q}_\ell^k$ and $\phi(q_h^g) = \tilde{p}_\ell^k$. This completes the definition of the mapping ϕ . For an illustration of the map ϕ , see Figure 2.

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, $d_G(x) = d_G(y) = d_G(z) = d_H(\tilde{x}) = 2mb$. 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}_ℓ^k is adjacent to the images \tilde{x} of x and \tilde{u}_ℓ^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}_ℓ^k or \tilde{q}_ℓ^k is adjacent to \tilde{x} of y (respectively z) and \tilde{u}_ℓ^k of u_j^i . Hence the restriction of ϕ to $N_G(p_j^i)$ is bijective for every $i \in \{1, \dots, 3m\}$ and $j \in \{1, \dots, b\}$, and the same clearly holds for the restriction of ϕ to $N_G(q_j^i)$. The vertices of each star S_i are mapped to the vertices of some star \tilde{S}_k in such a way that the centres are mapped to centres. This, together with the fact that the image \tilde{u}_ℓ^k of every vertex u_j^i is adjacent to the images \tilde{p}_ℓ^k and \tilde{q}_ℓ^k of the neighbours 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 $i \in \{1, \dots, 3m\}$ and $j \in \{1, \dots, b\}$. Finally, the neighbourhood of c^i is clearly mapped to the neighbourhood of $\phi(c^i)$ for every $i \in \{1, \dots, 3m\}$. We conclude that ϕ is a locally bijective homomorphism from G to H .

In order to show that the tree-depth of G is at most 6, we construct a rooted tree T as follows. We let z be the root of T and add one child y . For y we add one child x . We construct $3m$ children c^1, \dots, c^{3m} of x . Furthermore, to each c^i we add b children u_1^i, \dots, u_b^i . Finally, each u_j^i gets two children p_j^i and q_j^i . It is easy to observe that G is a subgraph of $C(T)$ and since T has depth 5, this implies $\text{td}(G) \leq 6$. Furthermore, for H we can use a very similar approach. We let \tilde{T} be the tree obtained from T by removing z, y from T and letting x be the root. After renaming the vertices appropriately, H is a subgraph of $C(\tilde{T})$ and hence $\text{td}(H) \leq 4$. This completes the proof for 3-FOLD COVER and therefore LBHOM.

In order to prove NP-hardness for LSHOM we can use the same reduction as for LBHOM. For this we can argue that there is a locally bijective homomorphism from G to H , for the graphs G and H constructed above, if and only if there is a locally surjective homomorphism from G to H . While the one direction is clear, if $G \xrightarrow{B} H$ then $G \xrightarrow{S} H$, for the converse direction we can make use of the following statement due to Kristiansen and Telle [52]:

(*) If $G \xrightarrow{S} H$ and $\text{drm}(G) = \text{drm}(H)$, then $G \xrightarrow{B} H$.

Here $\text{drm}(G)$, $\text{drm}(H)$ refers to the degree refinement matrix of G or H respectively, which is defined as follows. An *equitable partition* of a connected graph G is a partition of its vertex set into blocks B_1, \dots, B_k such that every vertex in B_i has the same number $m_{i,j}$ of neighbours in B_j . Then $\text{drm}(G) = (m_{i,j})$ for $m_{i,j}$ corresponding to the coarsest equitable partition of G . We can easily observe that

$$\text{drm}(G) = \text{drm}(H) = \begin{pmatrix} 0 & 0 & 2mb & 0 \\ 0 & 0 & 2 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & b & 0 & 0 \end{pmatrix},$$

corresponding to the equitable partitions $B_1 = \{x, y, z\}$, $B_2 = \{u_j^i \mid i \in \{1, \dots, 3m\}, j \in \{1, \dots, b\}\}$, $B_3 = \{p_j^i, q_j^i \mid i \in \{1, \dots, 3m\}, j \in \{1, \dots, b\}\}$ and $B_4 = \{c^i \mid i \in \{1, \dots, 3m\}\}$ in G and a similar equitable partition in H . Hence by (*) we find that $G \xrightarrow{B} H$ if and only if $G \xrightarrow{S} H$, completing the proof for LSHOM. ◀

► **Theorem 29.** LBHOM, or more specifically, 3-FOLDCOVER, and LSHOM are NP-complete on input pairs (G, H) where G and H have feedback vertex set number at most 3 and 1, respectively.

Proof. To prove the statement we use the same reductions as in the proof of Theorem 28. This is sufficient, as the set $\{x, y, z\}$ is a feedback vertex set of G and the set $\{\bar{x}\}$ is a feedback vertex set of H for graphs G and H defined in the proof of Theorem 28. ◀

7 Conclusions

We introduced a general algorithmic framework that can be employed for a wide variety of problems related to homomorphisms on graphs that have a small fracture number. We already illustrated the applicability of the framework for three well-known variants of the locally constrained homomorphism problem, i.e. LSHOM, LBHOM, and LIHOM, as well as the ROLE ASSIGNMENT problem, by giving three FPT results and one XP result. Our complementary hardness results provide a fairly comprehensive picture concerning the parameterized complexity of the three locally constrained homomorphism problems (see also Table 1).

For future work we aim to extend our ILP-based framework. If successful, this will then also enable us to address the parameterized complexity of other graph homomorphism variants such as quasi-covers [34] and pseudo-covers [11, 13, 14]. We also recall an interesting open problem from [15]. Namely, are LBHOM and LSHOM in FPT when parameterized by the treewidth of the guest graph plus the maximum degree of the guest graph?

References

- 1 James Abello, Michael R. Fellows, and John Stillwell. On the complexity and combinatorics of covering finite complexes. *Australasian Journal of Combinatorics*, 4:103–112, 1991.
- 2 Dana Angluin. Local and global properties in networks of processors (extended abstract). *Proc. STOC 1980*, pages 82–93, 1980.
- 3 Dana Angluin and A. Gardiner. Finite common coverings of pairs of regular graphs. *Journal of Combinatorial Theory, Series B*, 30:184–187, 1981.
- 4 Norman J. Biggs. *Algebraic Graph Theory*. Cambridge University Press, 1974.

- 5 Norman J. Biggs. Constructing 5-arc transitive cubic graphs. *Journal of the London Mathematical Society II*, 26:193–200, 1982.
- 6 Ondrej Bílka, Bernard Lidický, and Marek Tesar. Locally injective homomorphism to the simple weight graphs. *Proc. TAMC 2011, LNCS*, 6648:471–482, 2011.
- 7 Hans L. Bodlaender. The classification of coverings of processor networks. *Journal of Parallel and Distributed Computing*, 6:166–182, 1989.
- 8 Hans L. Bodlaender, John R. Gilbert, Hjálmtýr Hafsteinsson, and Ton Kloks. Approximating treewidth, pathwidth, frontsize, and shortest elimination tree. *Journal of Algorithms*, 18:238–255, 1995.
- 9 Jan Bok, Jirí Fiala, Petr Hlinený, Nikola Jedlicková, and Jan Kratochvíl. Computational complexity of covering two-vertex multigraphs with semi-edges. *CoRR*, abs/2103.15214, 2021.
- 10 Binh-Minh Bui-Xuan, Jan Arne Telle, and Martin Vatshelle. Fast dynamic programming for locally checkable vertex subset and vertex partitioning problems. *Theoretical Computer Science*, 511:66–76, 2013.
- 11 Jérémie Chalopin. Local computations on closed unlabelled edges: The election problem and the naming problem. *Proc. SOFSEM 2005, LNCS*, 3381:82–91, 2005.
- 12 Jérémie Chalopin, Yves Métivier, and Wiesław Zielonka. Local computations in graphs: The case of cellular edge local computations. *Fundamenta Informaticae*, 74:85–114, 2006.
- 13 Jérémie Chalopin and Daniël Paulusma. Graph labelings derived from models in distributed computing: A complete complexity classification. *Networks*, 58:207–231, 2011.
- 14 Jérémie Chalopin and Daniël Paulusma. Packing bipartite graphs with covers of complete bipartite graphs. *Discrete Applied Mathematics*, 168:40–50, 2014.
- 15 Steven Chaplick, Jirí Fiala, Pim van ’t Hof, Daniël Paulusma, and Marek Tesar. Locally constrained homomorphisms on graphs of bounded treewidth and bounded degree. *Theoretical Computer Science*, 590:86–95, 2015.
- 16 Chandra Chekuri and Anand Rajaraman. Conjunctive query containment revisited. *Theoretical Computer Science*, 239:211–229, 2000.
- 17 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.
- 18 Víctor Dalmau, Phokion G. Kolaitis, and Moshe Y. Vardi. Constraint satisfaction, bounded treewidth, and finite-variable logics. *Proc. CP 2002, LNCS*, 2470:310–326, 2002.
- 19 Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer Verlag, New York, 2nd edition, 2000.
- 20 Mitre Costa Dourado. Computing role assignments of split graphs. *Theoretical Computer Science*, 635:74–84, 2016.
- 21 Rodney G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness II: on completeness for W[1]. *Theoretical Computer Science*, 141:109–131, 1995.
- 22 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.
- 23 Pål Grønås Drange, Markus S. Dregi, and Pim van ’t Hof. On the computational complexity of vertex integrity and component order connectivity. *Algorithmica*, 76:1181–1202, 2016.
- 24 Pavel Dvořák, Eduard Eiben, Robert Ganian, Dusan Knop, and Sebastian Ordyniak. Solving integer linear programs with a small number of global variables and constraints. *Proc. IJCAI 2017*, pages 607–613, 2017.
- 25 Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.
- 26 Martin G. Everett and Stephen P. Borgatti. Role colouring a graph. *Mathematical Social Sciences*, 21:183–188, 1991.
- 27 Michael R. Fellows, Daniel Lokshtanov, Neeldhara Misra, Frances A. Rosamond, and Saket Saurabh. Graph layout problems parameterized by vertex cover. In *ISAAC*, Lecture Notes in Computer Science, pages 294–305. Springer, 2008.
- 28 Jirí Fiala, Ton Kloks, and Jan Kratochvíl. Fixed-parameter complexity of lambda-labelings. *Discrete Applied Mathematics*, 113:59–72, 2001.

- 29 Jirí Fiala and Jan Kratochvíl. Partial covers of graphs. *Discussiones Mathematicae Graph Theory*, 22:89–99, 2002.
- 30 Jirí Fiala and Jan Kratochvíl. Locally constrained graph homomorphisms - structure, complexity, and applications. *Computer Science Review*, 2:97–111, 2008.
- 31 Jirí Fiala, Jan Kratochvíl, and Attila Pór. On the computational complexity of partial covers of theta graphs. *Discrete Applied Mathematics*, 156:1143–1149, 2008.
- 32 Jirí Fiala and Daniël Paulusma. A complete complexity classification of the role assignment problem. *Theoretical Computer Science*, 349:67–81, 2005.
- 33 Jirí Fiala and Daniël Paulusma. Comparing universal covers in polynomial time. *Theory of Computing Systems*, 46:620–635, 2010.
- 34 Jirí Fiala and Marek Tesar. Dichotomy of the H -Quasi-Cover problem. *Proc. CSR 2013, LNCS*, 7913:310–321, 2013.
- 35 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*, volume XIV of *Texts in Theoretical Computer Science. An EATCS Series*. Springer Verlag, Berlin, 2006.
- 36 András Frank and Éva Tardos. An application of simultaneous diophantine approximation in combinatorial optimization. *Combinatorica*, 7(1):49–65, 1987.
- 37 Eugene C. Freuder. Complexity of k -tree structured constraint satisfaction problems. *Proc. AAAI 1990*, pages 4–9, 1990.
- 38 Michael Randolph Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979.
- 39 Michael U. Gerber and Daniel Kobler. Algorithms for vertex-partitioning problems on graphs with fixed clique-width. *Theoretical Computer Science*, 299(1-3):719–734, 2003.
- 40 Martin Grohe. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *Journal of the ACM*, 54:1:1–1:24, 2007.
- 41 Pinar Heggenes, Pim van 't Hof, and Daniël Paulusma. Computing role assignments of proper interval graphs in polynomial time. *Journal of Discrete Algorithms*, 14:173–188, 2012.
- 42 Pavol Hell and Jaroslav Nešetřil. On the complexity of h -coloring. *Journal of Combinatorial Theory, Series B*, 48:92–110, 1990.
- 43 Pavol Hell and Jaroslav Nešetřil. The core of a graph. *Discrete Mathematics*, 109:117–126, 1992.
- 44 Pavol Hell and Jaroslav Nešetřil. *Graphs and Homomorphisms*. Oxford University Press, 2004.
- 45 Ravi Kannan. Minkowski's convex body theorem and integer programming. *Math. Oper. Res.*, 12(3):415–440, 1987.
- 46 David G. Kirkpatrick and Pavol Hell. On the completeness of a generalized matching problem. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, page 240–245, New York, NY, USA, 1978. Association for Computing Machinery.
- 47 Pavel Klavík. *Extension Properties of Graphs and Structures*. PhD thesis, Charles University, Prague, 2017.
- 48 Jan Kratochvíl. Regular codes in regular graphs are difficult. *Discrete Mathematics*, 133:191–205, 1994.
- 49 Jan Kratochvíl, Andrzej Proskurowski, and Jan Arne Telle. Covering regular graphs. *Journal of Combinatorial Theory, Series B*, 71:1–16, 1997.
- 50 Jan Kratochvíl, Andrzej Proskurowski, and Jan Arne Telle. On the complexity of graph covering problems. *Nordic Journal of Computing*, 5:173–195, 1998.
- 51 Jan Kratochvíl, Jan Arne Telle, and Marek Tesar. Computational complexity of covering three-vertex multigraphs. *Theoretical Computer Science*, 609:104–117, 2016.
- 52 Petter Kristiansen and Jan Arne Telle. Generalized H -coloring of graphs. *Proc. ISAAC 2000, LNCS*, 1969:456–466, 2000.
- 53 Martin Kronegger, Sebastian Ordyniak, and Andreas Pfandler. Backdoors to planning. *Artif. Intell.*, 269:49–75, 2019.
- 54 Hendrik W. Lenstra Jr. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983.

- 55 Bernard Lidický and Marek Tesar. Complexity of locally injective homomorphism to the theta graphs. *Proc. IWOCA 2010, LNCS*, 6460:326–336, 2010.
- 56 Daniel Lokshtanov, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. Fixed-parameter tractable canonization and isomorphism test for graphs of bounded treewidth. *SIAM J. Comput.*, 46(1):161–189, 2017.
- 57 William S. Massey. *Algebraic Topology: An Introduction*. Harcourt, Brace and World, 1967.
- 58 Jaroslav Nešetřil. Homomorphisms of derivative graphs. *Discrete Mathematics*, 1:257–268, 1971.
- 59 Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity: Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and Combinatorics*. Springer, 2012.
- 60 Karolina Okrasa and Paweł Rzażewski. Subexponential algorithms for variants of the homomorphism problem in string graphs. *Journal of Computer and System Sciences*, 109:126–144, 2020.
- 61 Sukanya Pandey, Venkatesh Raman, and Vibha Sahlot. Parameterizing role coloring on forests. *Proc. SOFSEM 2021, LNCS*, 12607:308–321, 2021.
- 62 Sukanya Pandey and Vibha Sahlot. Role coloring bipartite graphs. *CoRR*, abs/2102.01124, 2021.
- 63 Aleksandar Pekeč and Fred S. Roberts. The role assignment model nearly fits most social networks. *Mathematical Social Sciences*, 41:275–293, 2001.
- 64 Christopher Purcell and M. Puck Rombach. On the complexity of role colouring planar graphs, trees and cographs. *Journal of Discrete Algorithms*, 35:1–8, 2015.
- 65 Christopher Purcell and M. Puck Rombach. Role colouring graphs in hereditary classes. *CoRR*, abs/1802.10180, 2018.
- 66 Fred S. Roberts and Li Sheng. How hard is it to determine if a graph has a 2-role assignment? *Networks*, 37:67–73, 2001.
- 67 Jan Arne Telle and Andrzej Proskurowski. Algorithms for vertex partitioning problems on partial k -trees. *SIAM Journal on Discrete Mathematics*, 10(4):529–550, 1997.
- 68 Pim van ’t Hof, Daniël Paulusma, and Johan M. M. van Rooij. Computing role assignments of chordal graphs. *Theoretical Computer Science*, 411:3601–3613, 2010.
- 69 Douglas R. White and Karl P. Reitz. Graph and semigroup homomorphisms on networks of relations. *Social Networks*, 5:193–235, 1983.