List H-Coloring a Graph by Removing Few Vertices*

Rajesh Chitnis[1]**, László Egri[2], and Dániel Marx[2]

[1] Department of Computer Science, University of Maryland at College Park, USA, rchitnis@cs.umd.edu

[2] Computer and Automation Research Institute, Hungarian Academy of Sciences (MTA SZTAKI), Budapest, Hungary. {dmarx@cs.bme.hu, laszlo.egri@mail.mcgill.ca}

Abstract. In the deletion version of the list homomorphism problem, we are given graphs G and H, a list $L(v) \subseteq V(H)$ for each vertex $v \in V(G)$, and an integer k. The task is to decide whether there exists a set $W \subseteq V(G)$ of size at most k such that there is a homomorphism from $G \setminus W$ to H respecting the lists. We show that DL-Hom(H), parameterized by k and $|H|$, is fixed-parameter tractable for any (P_6, C_6)-free bipartite graph H; already for this restricted class of graphs, the problem generalizes Vertex Cover, Odd Cycle Transversal, and Vertex Multiway Cut parameterized by the size of the cutset and the number of terminals. We conjecture that DL-Hom(H) is fixed-parameter tractable for the class of graphs H for which the list homomorphism problem (without deletions) is polynomial-time solvable; by a result of Feder et al. [9], a graph H belongs to this class precisely if it is a bipartite graph whose complement is a circular arc graph. We show that this conjecture is equivalent to the fixed-parameter tractability of a single fairly natural satisfiability problem, *Clause Deletion Chain-SAT*.

1 Introduction

Given two graphs G and H (without loops and parallel edges; unless otherwise stated, we consider only such graphs throughout this paper), a *homomorphism* $\phi : G \to H$ is a mapping $\phi : V(G) \to V(H)$ such that $\{u, v\} \in E(G)$ implies $\{\phi(u), \phi(v)\} \in E(H)$; the corresponding algorithmic problem *Graph Homomorphism* asks if G has a homomorphism to H. It is easy to see that G has a homomorphism into the clique K_c if and only if G is c-colorable; therefore, the algorithmic study of (variants of) Graph Homomorphism generalizes the study of graph coloring problems (cf. Hell and Nešetřil [15]). Instead of graphs, one can

* Supported by ERC Starting Grant PARAMTIGHT (No. 280152)

** Supported in part by NSF CAREER award 1053605, NSF grant CCF-1161626, ONR YIP award N000141110662, DARPA/AFOSR grant FA9550-12-1-0423, a University of Maryland Research and Scholarship Award (RASA) and a Summer International Research Fellowship from University of Maryland.

consider homomorphism problems in the more general context of relational structures. Feder and Vardi [12] observed that the standard framework for Constraint Satisfaction Problems (CSP) can be formulated as homomorphism problems for relational structures. Thus variants of Graph Homomorphism form a rich family of problems that are more general than classical graph coloring, but does not have the full generality of CSPs.

List Coloring is a generalization of ordinary graph coloring: for each vertex v, the input contains a list $L(v)$ of allowed colors associated to v, and the task is to find a coloring where each vertex gets a color from its list. In a similar way, *List Homomorphism* is a generalization of Graph Homomorphism: given two undirected graphs G, H and a list function $L : V(G) \to 2^{V(H)}$, the task is to decide if there exists a list homomorphism $\phi : G \to H$, i.e., a homomorphism $\phi : G \to H$ such that for every $v \in V(G)$ we have $\phi(v) \in L(v)$. The List Homomorphism problem was introduced by Feder and Hell [8] and has been studied extensively [7, 11, 9, 10, 14, 17]. It is also referred to as List H-Coloring the graph G since in the special case of $H = K_c$ the problem is equivalent to list coloring where every list is a subset of $\{1, \dots, c\}$.

An active line of research on homomorphism problems is to study the complexity of the problem when the target graph is fixed. Let H be an undirected graph. The Graph Homomorphism and List Homomorphism problems with fixed target H are denoted by $\textsc{Hom}(H)$ and L-$\textsc{Hom}(H)$, respectively. A classical result of Hell and Nešetřil [16] states that $\textsc{Hom}(H)$ is polynomial-time solvable if H is bipartite and NP-complete otherwise. For the more general List Homomorphism problem, Feder et al. [9] showed that L-$\textsc{Hom}(H)$ is in P if H is a bipartite graph whose complement is a circular arc graph, and it is NP-complete otherwise. Egri et al. [7] further refined this characterization and gave a complete classification of the complexity of L-$\textsc{Hom}(H)$: they showed that the problem is complete for NP, NL, or L, or otherwise the problem is first-order definable.

In this paper, we increase the expressive power of (list) homomorphisms by allowing a bounded number of vertex deletions from the left-hand side graph G. Formally, in the DL-$\textsc{Hom}(H)$ problem we are given as input an undirected graph G, an integer k, a list function $L : V(G) \to 2^{V(H)}$ and the task is to decide if there is a *deletion set* $W \subseteq V(G)$ such that $|W| \leq k$ and the graph $G \setminus W$ has a list homomorphism to H. Let us note that DL-$\textsc{Hom}(H)$ is NP-hard already when H consists of a single isolated vertex: in this case the problem is equivalent to Vertex Cover, since removing the set W has to destroy every edge of G.

We study the parameterized complexity of DL-$\textsc{Hom}(H)$ parameterized by the number of allowed vertex deletions and the size of the target graph H. We show that DL-$\textsc{Hom}(H)$ is fixed parameter tractable (FPT) for a rich class of target graphs H. That is, we show that DL-$\textsc{Hom}(H)$ can be solved in time $f(k, |H|) \cdot n^{O(1)}$ if H is a (P_6, C_6)-free bipartite graph, where f is a computable function that depends only of k and $|H|$ (see [5, 13, 24] for more background on fixed parameter tractability). This unifies and generalizes the fixed parameter tractability of certain well-known problems in the FPT world:

- Vertex Cover asks for a set of k vertices whose deletion removes every edge. This problem is equivalent to DL-Hom(H) where H is a single vertex.
- Odd Cycle Transversal (also known as Vertex Bipartization) asks for a set of at most k vertices whose deletion makes the graph bipartite. This problem can be expressed by DL-Hom(H) when H consists of a single edge.
- In Vertex Multiway Cut parameterized by the size of the cutset and the number of terminals, a graph G is given with terminals $t_1, \ldots, t_d$, and the task is to find a set of at most k vertices whose deletion disconnects t_i and t_j for any $i \neq j$. This problem can be expressed as DL-Hom(H) when H is a matching of d edges, in the following way. Let us obtain G' by subdividing each edge of G (making it bipartite) and let the list of t_i contain the vertices of the i-th edge e_i; all the other lists contain every vertex of H. It is easy to see that the deleted vertices must separate the terminals otherwise there is no homomorphism to H and, conversely, if the terminals are separated from each other, then the component of t_i has a list homomorphism to e_i.

Note that all three problems described above are NP-hard but known to be fixed-parameter tractable [4, 5, 21, 25].

Our Results: Clearly, if L-Hom(H) is NP-complete, then DL-Hom(H) is NP-complete already for $k = 0$, hence we cannot expect it to be FPT. Therefore, by the results of Feder et al. [9], *we need to consider only the case when H is a bipartite graph* whose complement is a circular arc graph. We focus first on those graphs H for which the characterization of Egri et al. [7] showed that L-Hom(H) is not only polynomial-time solvable, but actually in logspace: these are precisely those (bipartite) graphs that exclude the path P_6 on six vertices and the cycle C_6 on six vertices as induced subgraphs. This class of graphs admits a decomposition using certain operations (see [7]), and to emphasize this decomposition, we also call this class of graphs *skew decomposable graphs*. Let us emphasize that these graphs are bipartite by definition. Note that the class of skew decomposable graphs is a strict subclass of chordal bipartite graphs (P_6 is chordal bipartite but not skew decomposable), and bipartite cographs and bipartite trivially perfect graphs are trivially skew decomposable.

Our first result is that the DL-Hom(H) problem is fixed-parameter tractable for this class of graphs.

Theorem 1. *If H is a skew decomposable bipartite graph, then* DL-Hom(H) *is FPT parameterized by solution size and $|H|$.*

Observe that the graphs considered in the examples above are all skew decomposable bipartite graphs, hence Theorem 1 is an algorithmic meta-theorem unifying the fixed-parameter tractability of Vertex Cover, Odd Cycle Transversal, and Vertex Multiway Cut parameterized by the size of the cutset and the number of terminals, and various combinations of these problems.

Theorem 1 shows that, for a particular class of graphs where L-Hom(H) is known to be polynomial-time solvable, the deletion version DL-Hom(H) is fixed-parameter tractable. We conjecture that this holds in general: whenever L-Hom(H) is polynomial-time solvable (i.e., the cases described by Feder et al. [9]), the corresponding DL-Hom(H) problem is FPT.

Conjecture 2. If H is a fixed graph whose complement is a circular arc graph, then DL-Hom(H) is FPT parameterized by solution size.

It might seem unsubstantiated to conjecture fixed-parameter tractability for every bipartite graph H whose complement is a circular arc graph, but we show that, in a technical sense, proving Conjecture 2 boils down to the fixed-parameter tractability of a single fairly natural problem. We introduce a variant of maximum ℓ-satisfiability, where the clauses of the formula are implication chains[3] $x_1 \to x_2 \to \cdots \to x_\ell$ of length at most ℓ, and the task is to make the formula satisfiable by removing at most k clauses; we call this problem *Clause Deletion ℓ-Chain-SAT (ℓ-CDCS)* (see Definition 14). We conjecture that for every fixed ℓ, this problem is FPT parameterized by k.

Conjecture 3. For every fixed $\ell \geq 1$, Clause Deletion ℓ-Chain-SAT is FPT parameterized by solution size.

We show that for every bipartite graph H whose complement is a circular arc graph, the problem DL-Hom(H) can be reduced to CDCS for some ℓ depending only on $|H|$. Somewhat more surprisingly, we are also able to show a converse statement: for every ℓ, there is a bipartite graph H_ℓ whose complement is a circular arc graph such that ℓ-CDCS can be reduced to DL-Hom(H_ℓ). That is, the two conjectures are equivalent. Therefore, in order to settle Conjecture 2, one necessarily needs to understand Conjecture 3 as well. Since the latter conjecture considers only a single problem (as opposed to an infinite family of problems parameterized by $|H|$), it is likely that connections with other satisfiability problems can be exploited, and therefore it seems that Conjecture 3 is a more promising target for future work.

Theorem 4. *Conjectures 2 and 3 are equivalent.*

Our Techniques: For our fixed-parameter tractability results, we use a combination of several techniques (some of them classical, some of them very recent) from the toolbox of parameterized complexity. Our first goal is to reduce DL-Hom(H) to the special case where each list contains vertices only from one side of one component of the (bipartite) graph H; we call this special case the "fixed side, fixed component" version. We note that the reduction to this special case in non-trivial: as the examples above illustrate, expressing Vertex Multiway Cut seems to require that the lists contain vertices from more than one component of H, and expressing Odd Cycle Transversal seems to require that the lists contain vertices from both sides of H.

We start our reduction by using the standard technique of iterative compression to obtain an instance where, besides a bounded number of precolored vertices, the graph is bipartite.

We look for obvious conflicts in this instance. Roughly speaking, if there are two precolored vertices u and v in the same component of G with colors a and b,

[3] The notation $x_1 \to x_2 \to \cdots \to x_\ell$ is a shorthand for $(x_1 \to x_2) \wedge (x_2 \to x_3) \wedge \cdots \wedge (x_{\ell-1} \to x_\ell)$.

respectively, such that either a and b are in different components of H, or a and b are in the same component of H but the parity of the distance between u and v is different from the parity of the distance between a and b, then the deletion set must contain a $u - v$ separator. We use the treewidth reduction technique of Marx et al. [22] to find a bounded-treewidth region of the graph that contains all such separators. As we know that this region contains at least one deleted vertex, every component outside this region can contain at most $k - 1$ deleted vertices. Thus we can recursively solve the problem for each such component, and collect all the information that is necessary to solve the problem for the remaining bounded-treewidth region. We are able to encode our problem as a Monadic Second Order (MSO) formula, hence we can apply Courcelle's Theorem [3] to solve the problem on the bounded-treewidth region.

Even if the instance has no obvious conflicts as described above, we might still need to delete certain vertices due to more implicit conflicts. But now we know that for each vertex v, there is at most one component C of H and one side of C that is consistent with the precolored vertices appearing in the component of v, otherwise a direct conflict between two precolored vertices would arise. This seems to be close to our goal of being able to fix a component C of H and a side of C for each vertex. However, there is a subtle detail here: if the deleted set separates a vertex v from every precolored vertex, then the precolored vertices do not force any restriction on v. Therefore, it seems that at each vertex v, we have to be prepared for two possibilities: either v is reachable from the precolored vertices, or not. Unfortunately, this prevents us from assigning each vertex to one of the sides of a single component. We get around this problem by invoking the "randomized shadow removal" technique introduced by Marx and Razgon [23] (and subsequently used in [1, 2, 18–20]) to modify the instance in such a way that we can assume that the deletion set does not separate any vertex from the precolored vertices, hence we can fix the components and the sides.

Note that the above reductions work for any bipartite graph H, and the requirement that H be skew decomposable is used only at the last reduction step: if H is a skew decomposable graph, then the *fixed side fixed component* version of the problem can be solved by appealing to the inductive construction of such graphs given by Egri et al. [7] and using bounded depth search.

If H is a bipartite graph whose complement is a circular arc graph (recall that this class strictly contains all skew decomposable graphs), then we show how to formulate the problem as an instance of ℓ-CDCS (showing that Conjecture 3 implies Conjecture 2). Let us emphasize that the reduction to ℓ-CDCS works only if the lists of the DL-Hom(H) instance have the "fixed side" property, and therefore our proof for the equivalence of the two conjectures (Theorem 4) needs the reduction machinery described above.

2 Preliminaries

Given a graph G, let $V(G)$ denote its vertices and $E(G)$ denote its edges. If $G = (U, V, E)$ is bipartite, we call U and V the *sides* of H. Let G be a graph and $W \subseteq V(G)$. Then $G[W]$ denotes the subgraph of G induced by the vertices in W. To simplify notation, we often write $G \setminus W$ instead of $G[V(G) \setminus W]$. The set $N(W)$ denotes the neighborhood of W in G, that is, the vertices of G which are not in W, but have a neighbor in W. Similarly to [22], we define two types of separators:

Definition 5. *A set S of vertices* separates *the sets of vertices A and B if no component of $G \setminus S$ contains vertices from both $A \setminus S$ and $B \setminus S$. If s and t are two distinct vertices of G, then an $s - t$ separator is a set S of vertices disjoint from $\{s, t\}$ such that s and t are in different components of $G \setminus S$.*

Definition 6. *Let G, H be graphs and L be a* list function $V(G) \to 2^{V(H)}$. *A list homomorphism ϕ from (G, L) to H (or if L is clear from the context, from G to H) is a homomorphism $\phi : G \to H$ such that $\phi(v) \in L(v)$ for every $v \in V(G)$. In other words, each vertex $v \in V(G)$ has a list $L(v)$ specifying the possible images of v. The right-hand side graph H is called the* target *graph.*

When the target graph H is fixed, we have the following problem:

L-Hom(H)
Input : A graph G and a list function $L : V(G) \to 2^{V(H)}$.
Question : Does there exist a list homomorphism from (G, L) to H?

The main problem we consider in this paper is the vertex deletion version of the L-Hom(H) problem, i.e., we ask if a set of vertices W can be deleted from G such that the remaining graph has a list homomorphism to H. Obviously, the list function is restricted to $V(G) \setminus W$, and for ease of notation, we denote this restricted list function $L|_{V(G)\setminus W}$ by $L \setminus W$. We can now ask the following formal question:

DL-Hom(H)
Input : A graph G, a list function $L : V(G) \to 2^{V(H)}$, and an integer k.
Parameters : k , $|H|$
Question : Does there exist a set $W \subseteq V(G)$ of size at most k such that there is a list homomorphism from $(G \setminus W, L \setminus W)$ to H?

Notice that if $k = 0$, then DL-Hom(H) becomes L-Hom(H). In the first part of the paper, we reduce DL-Hom(H) to a more restricted version of the problem where every list $L(v)$ contains vertices only from one component of H, and moreover, only from one side of that component (recall that we are assuming that H is bipartite). We call lists satisfying this property *fixed side fixed component.*

DL-Hom(H)-Fixed-Side-Fixed-Component, where H is bipartite (**FS-FC(H)**)
Input : A graph G, a fixed side fixed component list function $L : V(G) \to 2^{V(H)}$, and an integer k.
Parameters : k, $|H|$
Question : Does there exist a set $W \subseteq V(G)$ such that $|W| \leq k$ and $G \setminus W$ has a list homomorphism to H?

We argue that it is sufficient to solve the FS-FC(H) problem:

Theorem 7. *If the* FS-FC(H) *problem is FPT (where H is bipartite), then the* DL-Hom(H) *problem is also FPT.*

The main ideas in the reduction from DL-Hom(H) to FS-FC(H) are presented below. The proof is by induction on k, i.e., we are assuming that such a reduction is possible for $k-1$. In the full version of the paper, we solve FS-FC(H) for skew decomposable graphs, completing the proof of Theorem 1.

Theorem 8. *If H is a skew decomposable graph, then the* FS-FC(H) *problem is FPT.*

3 The Algorithm

The algorithm proving Theorem 1 is constructed through a series of reductions. We begin with applying the standard technique of *iterative compression* [25], and this is followed by some preprocessing of the *disjoint* version of the *compression* problem.

DL-Hom(H)-Disjoint-Compression
Input : A graph G_0, a list function $L : V(G_0) \to 2^{V(H)}$, an integer k, and a set $W_0 \subseteq V(G_0)$ of size at most $k+1$ such that $G_0 \setminus W_0$ has a list homomorphism to H.
Parameters : k, $|H|$
Question : Does there exist a set $W \subseteq V(G_0)$ disjoint from W_0 such that $|W| \leq k$ and $(G_0 \setminus W, L \setminus W)$ has a list homomorphism to H?

Since the techniques related to iterative compression are folklore, we just note here that any FPT algorithm for the DL-Hom(H)-Disjoint-Compression problem defined below translates into an FPT algorithm for DL-Hom(H) with an additional blowup factor of $O(2^{|W_0|}n)$ in the running time. The details of this reduction are given in the full version of the paper. In the rest of the paper, we concentrate on giving an FPT algorithm for the DL-Hom(H)-Disjoint-Compression problem.

Since the new solution W can be assumed to be disjoint from W_0, for any solution set W, we must have a partial homomorphism from $G_0[W_0]$ to H. We guess all such partial list homomorphisms γ from $G_0[W_0]$ to H, and we hope that we can find a set W such that γ can be extended to a total list homomorphism from $G_0[W]$ to H. To guess these partial homomorphisms, we simply enumerate all possible mappings from W_0 to H and check whether the given mapping is a list homomorphism from $(G_0[W_0], L|_{W_0})$ to H. If not we discard the given mapping. Observe that we need to consider only $|V(H)|^{|W_0|} \leq |V(H)|^{k+1}$ mappings. Hence, in what follows we can assume that we are given a partial list homomorphism γ from $G_0[W_0]$ to H.

Recall that we are assuming that H is bipartite. Since we have a fixed partial homomorphism γ from W_0 to H, we can propagate the consequences of this homomorphism to the lists of the vertices in the neighborhood of W_0, as follows. For every $v \in W_0$, let H_v be the component of H in which $\gamma(v)$ appears. Furthermore, let S_v be the side of H_v in which $\gamma(v)$ appears, and let $\bar{S}_v$ be the other side of H_v. For each neighbor u of v in $N(W_0)$, trim $L(u)$ as $L(u) \leftarrow L(u) \cap \bar{S}_v$. The list of each vertex in $N(W_0)$ is now contained in one of the sides of a single component of H. We say that such a list is *fixed side* and *fixed component*. Note that while doing this, some of the lists might become empty. We delete those vertices from the graph, and reduce the parameter accordingly.

Recall that $G_0 \setminus W_0$ has a list homomorphism to the bipartite graph H, and therefore $G_0 \setminus W_0$ must be bipartite. We will later make use of the homomorphism from $G_0 \setminus \{W_0 \cup N(W_0)\}$ to H, so we name this homomorphism ϕ_0. To summarize the properties of the problem we have at hand, we define it formally below. Note that we will not need the graph G_0 and the set W_0 any more, only the graph $G_0 \setminus W_0$, and the neighborhood $N(W_0)$. To simplify notation, we refer to $G_0 \setminus W_0$ and $N(W_0)$ as G and N_0, respectively.

DL-Hom(H)-Bipartite-Compression (BC(H))
Input : A bipartite graph G, a list function $L : V(G) \to 2^{V(H)}$, a set $N_0 \subseteq V(G)$, where for each $v \in N_0$, the list $L(v)$ is fixed side and fixed component, a list homomorphism ϕ_0 from $(G \setminus N_0, L \setminus N_0)$ to H, and an integer k.
Parameters : k, $|H|$
Question : Does there exist a set $W \subseteq V(G)$, such that $|W| \leq k$ and $(G \setminus W, L \setminus W)$ has a list homomorphism to H?

We define two types of *conflicts* between the vertices of N_0 (Definition 9). Our algorithm has two subroutines, one to handle the case when such a conflict is present, and one to handle the other case.

3.1 There is a Conflict

If a conflict exists, its presence allows us to invoke the treewidth reduction technique of Marx et al. [22] to split the instance into a bounded-treewidth part, and into instances having parameter value strictly less than k. After solving these instances with smaller parameter value recursively, we encode the problem in Monadic Second Order logic, and apply Courcelle's theorem [3]. We outline these ideas, as follows.

Recall that the lists of the vertices in N_0 in a BC(H) instance are fixed side fixed component.

Definition 9. *Let (G, L, N_0, ϕ_0, k) be an instance of* BC*(H). Let u and v be vertices in the same component of G. We say that u and v are in* component conflict *if $L(u)$ and $L(v)$ are subsets of vertices of different components of H. Furthermore, u and v are in* parity conflict *if u and v are not in component conflict, and either u and v belong to the same side of G but $L(u)$ is a subset of one of the sides of a component of C of H and $L(v)$ is a subset of the other side of C, or u and v belong to different sides of G but $L(u)$ and $L(v)$ are subsets of the same side of a component of H.*

The following lemma easily follows from the definitions.

Lemma 10. *Let (G, L, N_0, ϕ_0, k) be an instance of* BC*(H). If u and v are any two vertices in N_0 that are in component or parity conflict, then any solution W must contain a set S that separates the sets $\{u\}$ and $\{v\}$.*

The result we need from [22] states that all the minimal $s - t$ separators of size at most k in G can be covered by a set C inducing a bounded-treewidth subgraph of G. In fact, a stronger statement is true: this subgraph has bounded treewidth even if we introduce additional edges in order to take into account connectivity outside C. This is expressed by the operation of taking the torso:

Definition 11. *Let G be a graph and $C \subseteq V(G)$. The graph* $\mathrm{torso}(G, C)$ *has vertex set C and two vertices $a, b \in C$ are adjacent if $\{a, b\} \in E(G)$ or there is a path in G connecting a and b whose internal vertices are not in C.*

Observe that by definition, $G[C]$ is a subgraph of $\mathrm{torso}(G, C)$.

Lemma 12 ([22]). *Let s and t be two vertices of G. For some $k \geq 0$, let C_k be the union of all minimal sets of size at most k that are $s-t$ separators. There is a $O(g_1(k) \cdot (|E(G) + V(G)|))$ time algorithm that returns a set $C \supset C_k \cup \{s, t\}$ such that the treewidth of* $\mathrm{torso}(G, C)$ *is at most $g_2(k)$, for some functions g_1 and g_2 of k.*

Lemma 10 gives us a pair of vertices that must be separated, and Lemma 12 gives us a bounded-treewidth region C of the input graph in which we know that at least one vertex must be deleted.

Courcelle's Theorem gives an easy way of showing that certain problems are linear-time solvable on bounded-treewidth graphs: it states that if a problem can be formulated in MSO, then there is a linear-time algorithm for it. This theorem also holds for relational structures of bounded-treewidth instead of just graphs, a generalization we need because we introduce new relations to encode the properties of the components of $G \setminus C$.

The following lemma formalizes the above ideas to prove that the subroutine used to handle the case when a conflict exists is correct:

Lemma 13. *Let $\mathcal{A}$ be an algorithm that correctly solves* DL-HOM(H) *for input instances in which the first parameter is at most $k-1$. Suppose that the running time of $\mathcal{A}$ is $f(k-1, |H|) \cdot x^c$, where x is the size of the input, and c is a sufficiently large constant. Let I be an instance of* BC*(H) with parameter k that contains a component or parity conflict. Then I can be solved in time $f(k, |H|) \cdot x^c$ (where f is defined in the proof).*

Proof. Let $I = (G, L, N_0, \phi_0, k)$ be an instance of BC(H). Let $v, w \in N_0$ such that v and w are in component or parity conflict. Then by Lemma 10, the deletion set must contain a $v - w$ separator. Using Lemma 12, we can find a set C with the properties stated in the lemma (and note that we will also make use of the functions g_1 and g_2 in the statement of the lemma). Most importantly, C contains at least one vertex that must be removed in any solution, so the maximum number of vertices that can be removed from any connected component of $G[V(G) \setminus C]$ without exceeding the budget k is at most $k-1$. Therefore, the outline of our strategy is the following. We use $\mathcal{A}$ to solve the problem for some slightly modified versions of the components of $G[V(G) \setminus C]$, and using these solutions, we construct an MSO formula that encodes our original problem I. Furthermore, the relational structure over which this MSO formula must be evaluated has bounded treewidth, and therefore the formula can be evaluated in linear time using Courcelle's theorem. The details of the proof are deferred to the full version of the paper. □

3.2 There is no Conflict

In the case when there is no component or parity conflict, the problem FS-FC-IG(H) is the same as FS-FC(H) except that if the solution separates a vertex v from N_0, then we do not require that v is assigned to any vertex of H. We first trim the lists which allows us to reduce the BC(H) problem to the FS-FC-IG(H) problem. Then we use the "shadow removal" technique of Marx and Razgon [23] which allows us to reduce the FS-FC-IG(H) problem to the FS-FC(H) problem. Finally, we use the inductive construction of skew decomposable bipartite graphs [7] which allows us to solve the FS-FC(H) problem recursively. The details about this case are deferred to the full version of the paper.

4 Relation between DL-Hom(H) and Satisfiability Problems

Theorem 4 establishes the equivalence of DL-Hom(H) with the Clause Deletion ℓ-Chain SAT (ℓ-CDCS) problem, where H is restricted to be a graph for which L-Hom(H) is characterized as polynomial-time solvable by Feder et al. [9], that is, where H is restricted to be a bipartite graph whose complement is a circular arc graph. Here we only define the ℓ-CDCS problem, and the technical proof of Theorem 4 can be found in the full version of the paper.

Definition 14. *A* chain clause *is a conjunction of the form*

$$(x_0 \to x_1) \wedge (x_1 \to x_2) \wedge \cdots \wedge (x_{m-1} \to x_m),$$

where x_i and x_j are different variables if $i \neq j$. The length *of a chain clause is the number of variables it contains. (A chain clause of length* 1 *is a variable, and it is satisfied by both possible assignments.) To simplify notation, we denote chain clauses of the above form as*

$$x_0 \to x_1 \to \cdots \to x_m.$$

An ℓ-Chain-SAT formula consists of:

- *a set of variables V;*

– *a set of chain clauses over V such that any chain clause has length at most ℓ;*
– *a set of unary clauses (a unary clause is a variable or its negation).*

Clause Deletion ℓ-Chain-SAT (ℓ-CDCS)
Input : An ℓ-Chain-SAT formula F.
Parameter : k
Question : Does there exist a set of clauses of size at most k such that removing these clauses from F makes F satisfiable?

5 Concluding Remarks

The list homomorphism problem is a widely investigated problem in classical complexity theory. In this work, we initiated the study of this problem from the perspective of parameterized complexity: we have shown that the DL-Hom(H) is FPT for any skew decomposable graph H parameterized by the solution size and $|H|$, an algorithmic meta-result unifying the fixed parameter tractability of some well-known problems. To achieve this, we welded together a number of classical and recent techniques from the FPT toolbox in a novel way. Our research suggests many open problems, four of which are:

1. If H is a fixed bipartite graph whose complement is a circular arc graph, is DL-Hom(H) FPT parameterized by solution size? (Conjecture 2.)
2. If H is a fixed digraph such that L-Hom(H) is in logspace (such digraphs have been recently characterised in [6]), is DL-Hom(H) FPT parameterized by solution size?
3. If H is a matching consisting of n edges, is DL-Hom(H) FPT, where the parameter is only the size of the deletion set?
4. Consider DL-Hom(H) for target graphs H in which both vertices with and without loops are allowed. It is known that for such target graphs L-Hom(H) is in P if and only if H is a *bi-arc* graph [10], or equivalently, if and only if H has a *majority polymorphism.* If H is a fixed bi-arc graph, is there an FPT reduction from DL-Hom(H) to ℓ-CDCS, where ℓ depends only on $|H|$?

Note that for the first problem, we already do not know if DL-Hom(H) is FPT when H is a path on 7 vertices. (If H is a path on 6 vertices, there is a simple reduction to Almost 2-SAT once we ensure that the instance has fixed side lists.) Observe that the third problem is a generalization of the Vertex Multiway Cut problem parameterized only by the cutset. For the fourth problem, we note that the FPT reduction from DL-Hom(H) to CDCS for graphs without loops relies on the fixed side nature of the lists involved. Since the presence of loops in H makes the concept of a fixed side list meaningless, it is not clear how to achieve such a reduction.

References

1. R. H. Chitnis, M. Cygan, M. Hajiaghayi, and D. Marx. Directed subset feedback vertex set is fixed-parameter tractable. In *ICALP (1)*, 2012.
2. R. H. Chitnis, M. Hajiaghayi, and D. Marx. Fixed-parameter tractability of directed multiway cut parameterized by the size of the cutset. In *SODA*, 2012.
3. B. Courcelle. Graph rewriting: An algebraic and logic approach. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science: Volume B: Formal Models and Semantics*, pages 193–242. Elsevier, Amsterdam, 1990.
4. M. Cygan, M. Pilipczuk, M. Pilipczuk, and J. O. Wojtaszczyk. On multiway cut parameterized above lower bounds. In *IPEC*, pages 1–12, 2011.
5. R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer-Verlag.
6. L. Egri, P. Hell, B. Larose, and A. Rafiey. An L vs. NL dichotomy for the digraph list homomorphism problem. 2013. Manuscript in preparation.
7. L. Egri, A. A. Krokhin, B. Larose, and P. Tesson. The complexity of the list homomorphism problem for graphs. *Theory of Computing Systems*, 51(2), 2012.
8. T. Feder and P. Hell. List homomorphisms to reflexive graphs. *J. Comb. Theory, Ser. B*, 72(2):236–250, 1998.
9. T. Feder, P. Hell, and J. Huang. List homomorphisms and circular arc graphs. *Combinatorica*, 19(4):487–505, 1999.
10. T. Feder, P. Hell, and J. Huang. Bi-arc graphs and the complexity of list homomorphisms. *Journal of Graph Theory*, 42(1):61–80, 2003.
11. T. Feder, P. Hell, and J. Huang. List homomorphisms of graphs with bounded degrees. *Discrete Mathematics*, 307:386–392, 2007.
12. T. Feder and M. Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1):57–104, 1998.
13. J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, 2006.
14. G. Gutin, A. Rafiey, and A. Yeo. Minimum cost and list homomorphisms to semicomplete digraphs. *Discrete Applied Mathematics*, 154:890–897, 2006.
15. P. Hell and J. Nešetřil. *Graphs and homomorphisms*. Oxford University Press.
16. P. Hell and J. Nešetřil. On the complexity of H-coloring. *Journal of Combinatorial Theory, Series B*, 48:92–110, 1990.
17. P. Hell and A. Rafiey. The dichotomy of list homomorphisms for digraphs. In *SODA*, pages 1703–1713, 2011.
18. S. Kratsch, M. Pilipczuk, M. Pilipczuk, and M. Wahlström. Fixed-parameter tractability of multicut in directed acyclic graphs. In *ICALP (1)*, 2012.
19. D. Lokshtanov and D. Marx. Clustering with local restrictions. *Inf. Comput.*, 222:278–292, 2013.
20. D. Lokshtanov and M. S. Ramanujan. Parameterized tractability of multiway cut with parity constraints. In *ICALP (1)*, pages 750–761, 2012.
21. D. Marx. Parameterized graph separation problems. *Theor. Comput. Sci.*, 351(3):394–406, 2006.
22. D. Marx, B. O'Sullivan, and I. Razgon. Finding small separators in linear time via treewidth reduction. *CoRR*, abs/1110.4765, 2011.
23. D. Marx and I. Razgon. Fixed-parameter tractability of multicut parameterized by the size of the cutset. In *STOC*, pages 469–478, 2011.
24. R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.
25. B. A. Reed, K. Smith, and A. Vetta. Finding odd cycle transversals. *Oper. Res. Lett.*, 32(4):299–301, 2004.