Alcom-FT Technical Report Series
ALCOMFT-TR-02-105

(H, C, K)-Coloring: Fast, Easy, and Hard Cases*

Josep Díaz† Maria Serna† Dimitrios M. Thilikos†

Abstract

We define a variant of the H-coloring problem by fixing the number of preimages of a subset C of the vertices of H, thus allowing parameterization. We provide sufficient conditions to guarantee that the problem can be solved in $O(kn + f(k, H))$ steps where f is a function depending only on the number k of fixed preimages and the graph H, and in $O(n^{k+\alpha})$ steps where α is a constant independent of k. Finally, we prove that whenever the non parameterized vertices induce in G a graph that is bipartite and loopless the problem is NP-complete.

1 Introduction

For a given input graph G, let us consider the set of labelings that are given as homomorphisms to a fixed graph H. Recall that given graphs H and G, an homomorphism maps the vertices in G to vertices in H in such a way that adjacent vertices are mapped to adjacent ones.

For a fixed graph H, the *H-coloring problem* checks whether, there exists an homomorphism from an input graph G to H. The name of H-coloring comes from the fact that each vertex in H can be thought as a "color" that can be assigned to v. In fact, if H is the complete graph on c nodes, the set of H-colorings of G coincides with the set of proper c coloring of G. An interesting generalization of the H-coloring problem is the *list* H-coloring problem where each vertex of G is given together with a list of vertices of H, the vertices where it can be mapped. For results related to the complexity of the H-coloring, see [HN90], and for the complexity of the list H-coloring problem, see [FH98, FHH99].

In recent times, the H-coloring problem has received a lot of attention, from the structural combinatorics point of view as well as from the algorithmic point of view. For instance, when H is bipartite or it has a loop the H-coloring problem can be trivially solved in polynomial time, but in the case that H is loopless and not-bipartite, the problem is known to be NP-complete [HN90] (see also [GHN00] for the complexity of the H-coloring

*The work of all the authors was supported by the IST Program of the EU under contract number IST-99-14186 (ALCOM-FT) and by the Spanish CICYT TIC-2000-1970-CE. The work of the last author was partially supported by the Ministry of Education and Culture of Spain, Grant number MEC-DGES SB98 0K148809.

†Departament de Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya. Campus Nord Mòdul C6. c/ Jordi Girona Salgado 1-3, 08034, Barcelona, Spain. E-mails: {diaz, mjserna, sedthilk}@lsi.upc.es

problem for bounded degree graphs). The classical approach to cope with the hardness of the H-coloring problem is to consider subproblems for which G is restricted. The most general result in this direction was given recently in [DST01] where a $O(nh^{k+1}\min\{k,h\})$ time algorithm is given in the case when G is a partial k-tree, where $n = |V(G)|$ and $h = |V(H)|$ (see also [TP97]).

In this paper, we define a natural version of the H-coloring problem that allows parameterization. In particular, we set up a weighting K of a subset C of $V(G)$ with non negative integers and, we say that an input graph G has a (H,C,K)-coloring if there exists an H-homomorphism $\chi : V(G) \to V(H)$ such that the number of the preimages of any weighted vertex is equal to its weight. We call the triple (H,C,K) a *partial weighted assignment*. If we additionally assign to each vertex of G a list of permissible images, we have a more general version of the (H,C,K)-coloring problem, the *list* (H,C,K)*-coloring* problem.

A well known and popular interpretation of the various versions of the H-coloring problem is based on a model where the vertices of H are processors and the edges of H represent communication links between them. An instance of the H-coloring problem represents the problem where we have an input graph G where each vertex represents a job to be processed and each edge represents a communication demand between two jobs. A solution to the H-coloring problem is an assignment of jobs to the processors satisfying all the communication demands [FH98]. If, for some subset C of processors, there is a set of restrictions on the number of jobs to be assigned, represented by a weighting function K, then we have an instance of the (H,C,K)-problem. If, additionally, we demand that each job can be processed by only a list of processors, we have an instance of the list H-coloring and if again we restrict the number of jobs that may be processed by some of the processors, we have an instance of the list (H,C,K)-coloring problem.

In the (H,C,K)-coloring problem, we can consider the images in K to be fixed constants, having a parameterization of the H-coloring problem. That way, we initiate an alternative approach to study the complexity of the problem. The independent set problem and the vertex cover problem are examples of parameterized problems expressed in this way (see Figure 1).

In general, many NP-complete problems can be associated with one or more parameters. In many cases, when the parameter is considered to be a fixed constant, the problem becomes polynomially solvable and the question then is whether it is possible to find an algorithm of complexity $O(f(k)n^{\alpha})$, where α is a constant. In such a case, the problem is called *Fixed Parameter Tractable* (FPT). On the other hand, not all the parameterizations yield an FPT or even a polynomial time solution. A characteristic example of this behaviour is the coloring problem, where the parameter k is the number of the used colors: It is in P when $k \leq 2$ and it is NP-complete when $k \geq 3$. In this paper we will attempt a classification of the partial weighted assignments (H,C,K), where the integers in K are fixed constants. We will give properties for (H,C,K) that guarantee that the corresponding problem is in FPT, in P, or NP-complete.

Our main results are three theorems. We first prove that there exists a $O(kn+f(k,h))$ step algorithm for solving the (H,C,K)-coloring problem, when the non-weighted vertices

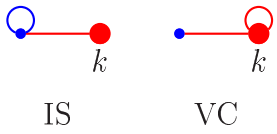

Figure 1: The graphs (H, C, K) for the parameterized independent set and vertex cover problems. The fat vertices represent the labeled vertices of H

of H form an independent set, where k is the total weight of the vertices in C. This algorithm is the main subroutine for our next result, a sufficient condition, to be satisfied by the connected components of H, for the (H, C, K)-coloring problem to be in FPT. Our second result is a condition for the (H, C, K)-coloring problem to have a polynomial time algorithm. In particular, we provide a $O(n^{k+\alpha})$ time algorithm for the case where the non parameterized vertices induce a graph for which the list H-coloring is in P. Our third result is a sufficient condition for the parameterized H-coloring problem to remain an NP-complete problem. This situation corresponds to the case when the non parameterized vertices induce a loopless and non-bipartite graph. All those results depend only on properties of the graph induced by the non-weighted vertices.

We also show that for a any graph F for which the F-coloring is in P but the list F-coloring is NP-complete, it is possible to connect additional bounded vertices in such a way that the resulting (H, C, K)-coloring problem is in P. Moreover, we provide some examples of NP-complete (H, C, K)-coloring problems verifying that the list $(H - C)$-coloring is NP-complete and the (H, C, K)-coloring is also NP-complete.

Finally, we prove that a similar classification exists for the list (H, C, K)-coloring problem. In particular, we show that the problem is in FPT when the non-weighted vertices of H form an independent set. Moreover, we prove that the list (H, C, K)-coloring problem is NP-complete if the list $H - C$-coloring problem is NP-complete. Otherwise the problem is in P. Thus the dichotomy results for the list H-coloring can be extended to the list (H, C, K)-coloring.

In Section 2, we give some basic definitions as well as a formal description of the problems introduced in this paper. In Section 3, we present the algorithm showing the fixed parameter tractability, as well as the theorem supporting its correctness. In Section 4, we present the cases where the parameterization accepts a polynomial solution. The NP-completeness proof is presented in Section 5. Section 6 presents the extensions to list H-colorings. Finally, in Section 7 we conclude with some remarks and open problems.

2 Preliminaires

All the graphs in this paper are undirected, without multiedges, but can have loops. Following the terminology in [FH98, FHH99] we say that a graph is *reflexive* if all its vertices have a loop, a graph is *irreflexive* when none of its vertices have a loop, and a graph is *general* if it some looped vertices and some unlooped ones. Making an abuse of notation, we denote a loop at vertx v as $\{v, v\}$. Given a graph G, $V(G)$ denotes its vertex set and $E(G)$ its edege set. If $S \subseteq V(G)$, we call the graph $(S, E(G) \cap \{\{x, y\} \mid x, y \in S\})$ the *subgraph of* G *induced by* S and we denote it by $G[S]$. We denote by $\gamma(G)$ the number of connected components of G. We also use the notation $G - S$ for the graph $G[V(G) - S]$. The neighborhood of a vertex v in graph G is the set of vertices in G that are adjacent to v in G and we denote it as $N_G(v)$. Unless otherwise mentioned, we assume that all the bipartite graphs in this paper are irreflexive. K_n^r denotes the reflexive clique with n vertices and C_n the irreflexive cycle with n vertices.

Given two graphs G and G', with no vertices in common, $G \cup G'$ denotes its *disjoint union*, the graph $G \cup G' = (V(G) \cup V(G'), E(G) \cup E(G'))$, and its *connected union* $G \oplus G'$ is the graph $G \oplus G' = (V(G) \cup V(G'), E(G) \cup E(G') \cup \{\{u, v\} \mid u \in V(G), v \in V(G')\}$.

For any function $\varphi : A \to B$ and any subset $C \subseteq A$, we define the *restriction of* φ *to* C as $\varphi|_C = \{(a, b) \in \varphi \mid a \in C\}$. Given two functions $\varphi : A \to \mathbb{N}$ and $\psi : A' \to \mathbb{N}$, with $A' \cap A = \emptyset$, the *disjoint union* of φ and ψ, $\varphi \cup \psi : A \cap A' \to \mathbb{N}$ is defined as:

$$(\varphi \cup \psi)(x) = \begin{cases} \varphi(x) & \text{if } x \in A, \\ \psi(x) & \text{if } x \in A'. \end{cases}$$

In the case that $A = A'$ we define $\varphi + \psi$ as $(\varphi + \psi)(x) = \varphi(x) + \psi(x)$, for any $x \in A$. Given two functions $\varphi, \psi : A \to \mathbb{N}$ we say that $\phi \leq \psi$ if, for any $x \in A$, $\phi(x) \leq \psi(x)$.

The triple (H, C, K), where H is a graph, $C \subseteq V(H)$ and $K : C \to \mathbb{N}$, is called a *partial weighted assignment* on H. All the vertices of C are the *weighted vertices* of H. For each partial weighted assignment (H, C, K), we use the notation $h = |V(H)|$, $r = |C|$ and $k = \sum_{c \in C} K(c)$. We call a partial weighted assignment (H, C, K) *positive* if, for any $c \in C$, $K(c) > 0$, in a *non-zero* partial weighted assignment there is some $c \in C$ for which $K(c) > 0$. A partial weighted assignment (H, C, K) is a *weighted extension* of a graph F if $H - C = F$. Finally, for any $S \subseteq C$, we define $(H, C, K)[S] = (H - (C - S), S, K|_S)$.

Given a partial weighted assignment (H, C, K), we say that a mapping $\chi : V(G) \to V(H)$ is an (H, C, K)*-coloring* of G if χ is an H-coloring of G such that for all $c \in C$, we have $|\chi^{-1}(c)| = K(c)$. If in the above definition we replace "$=$" with "$\leq$", we say that χ is a $(H, C, \leq K)$*-coloring* of G.

For fixed H, C, and K, given an input graph G, the (H, C, K)*-coloring* problem checks whether there exists an (H, C, K)-coloring of G. In the same way, the $(H, C, \leq K)$*-coloring* problem checks whether G has an $(H, C, \leq K)$-coloring.

Notice that the existence of a (H, C, K)-coloring implies the existence of a $(H, C, \leq K)$-coloring. However, the existence of an $(H, C, \leq K)$-coloring of G does not necessarily imply the existence of a (H, C, K)-coloring of G. In Figure 1 we give two examples where the (H, C, K)-coloring expreses the independent set and the vertex cover problem. In whatever concerns the parameterization of the above problems the parameters will be the integers in the images of K.

We say that two partial weighted assignments (H, C, K) and (H', C', K') are *equivalent* and we denote it as $(H, C, K) \sim (H', C', K')$, if, for any graph G, G has a (H, C, K)-coloring iff G has a (H', C', K')-coloring. In the same way we call them $\leq$*-equivalent* and we denote it as $(H, C, K) \sim_{\leq} (H', C', K')$, if, for any graph G, G has a $(H, C, \leq K)$-coloring iff G has a $(H', C', \leq K')$-coloring.

For a fixed graph H and a given a graph G, a (H, G)-*list* is any function $L : V(G) \to 2^{V(H)}$ mapping any vertex of G to a subset of $V(H)$. For any vertex $v \in V(G)$, the *list* of v is the set $L(v) \subseteq V(H)$. The *list H-coloring* problem asks whether, given a graph G and a (H, G)-list L, there is an H-coloring χ of G so that for every $u \in V(G)$ we have $\chi(u) \in L(u)$. This problem can be parameterized in the same way as the H-coloring to define the *list (H, C, K)-coloring* and the *list $(H, C, \leq K)$-coloring* problems.

Notice that in contraposition with the H-coloring problem, the (H, C, K)-coloring problem, in general, cannot be solved by considering the solution of subproblems in which both involved graphs are connected. The next two lemmas, whose proof is starightforward will be used later on.

Lemma 2.1. Let (H, C, K_i), $i = 1, \dots, t$, be partially weighted assignements of a fixed graph H and let G be a graph with connected components G_i, $i = 1, \dots, t$. If, for any i, $1 \leq i \leq s$, we have a (H, C, K_i)-coloring χ_i of G_i, then $\chi^* = \chi_1 \cup \cdots \cup \chi_t$ is a $(H, C, K_1 + \cdots + K_t)$-coloring of G.

Lemma 2.2. Let (H, C, K) and (H', C', K') be two partially weighted assignements of two disjoint graphs H and H', let G and G' be two disjoint graphs, let ρ be an (H, C, K)-coloring of G and let χ' be an (H', C', K')-coloring of G'. Then $\chi \cup \chi'$ is a $(H \cup H', C \cup C', K \cup K')$-coloring of $G \cup G'$.

Before we proceed we need a technical lemma, consider the following knapsack like problem. Let R be a finite set, and let $\mathcal{F}$ be the set of functions from R to $\mathbb{N}$.

PROBLEM (*):
Instance: Given a sequence $\mathcal{A} = (A_i)_{1 \leq i \leq s}$ of sets of non-zero functions in $\mathcal{F}$, and a function $g \in \mathcal{F}$, such that, for any $f \in \cup_{1 \leq i \leq s} A_i$, we have that $f \leq g$.

Question: Is there is a set $J \subseteq \{1, \dots, s\}$, and a sequence of functions $(f_j)_{j \in J}$ such that, for any $j \in J$, $f_j \in A_j$ and $\sum_{j \in J} f_j = g$?

Lemma 2.3. Let R be a finite set, and let $\mathcal{F}$ be the set of functions from R to $\mathbb{N}$. Given a sequence $\mathcal{A} = (A_i)_{1 \leq i \leq s}$ of sets of non-zero functions in $\mathcal{F}$, and a function $g \in \mathcal{F}$, such that, for any $f \in \cup_{1 \leq i \leq s} A_i$, we have that $f \leq g$. We can determine the set $\mathcal{S}$ containing any possible solution of the (*) problem in $O(k(s\ k^r)^{k+1})$ steps and, moreoever, $|\mathcal{S}| \leq (s\ k^r)^{k+1}$, where $r = |R|$ and $k = \sum_{x \in R} g(x)$.

Proof. Notice that in any solution to (*) $|J| \leq k$. Furthermore, the same function cannot appear more than k times in a solution. Let us define a set that contains the set $\mathcal{S}$.

Let $\mathcal{F}_{\leq g} = \{f \in \mathcal{F} \mid f \leq g\}$, clearly $|\mathcal{F}_{\leq g}| \leq k^r$. Notice that $\mathcal{S}$ is contained in the set $\mathcal{E}$ of sequences, with length k or less, of elements from $\{1, \dots, s\} \times \mathcal{F}_{\leq g}$. Therefore, we have $|\mathcal{S}| \leq (s\ k^r)^{k+1}$

We can obtain the set $\mathcal{S}$ by checking, for each element in $\mathcal{E}$, if the sequence is a solution. To do so, for each pair (i, f) in the list, we check wheteher $f \in A_i$, and finally, check that the sum of all the functions is g. This can be done in k additional steps per list. $\square$

3 Cases in FPT

In this section we will give a sufficient condition for the parameterized (H, C, K)-coloring problem to be in FPT. The basic component of the general algorithm will be a procedure that outputs, if it exists, an (H, C, K)-coloring of an input graph G, in the case that C is a vertex cover of H, in $O(kn + f(k, h))$ steps. The main idea of the proof is to create an equivalent problem whose size does not depend in the size of G.

Proposition 3.1. Let (H, C, K) be a partial weighted assignment such that $E(H - C) = \emptyset$. Then the Find-Coloring algorithm checks whether an input graph G has an (H, C, K)-coloring and, if yes, outputs the corresponding mapping, in $O(kn + h^{k^2+2k+(k+1)\cdot 2^k})$ time.

Algorithm Find-Coloring(G, H, C, K).

Input: Two graphs G, H, and a partial weighted assignment (H, C, K) on H
such that $E(H - C) = \emptyset$.
Output: If exists, an (H, C, K)-coloring of G and,
if not, a report that `‘‘no` (H, C, K)`-coloring of` G `exists’’`.

1: Let R_1 be the set of vertices in G of degree $> k$.
2: If $|R_1| > k$ or $|E(G)| > kn$ then return `‘‘no` (H, C, K)`-coloring of` G `exists’’`.
3: Set $G' = G[V(G) - R_1]$.
4: Let R_2 be the non isolated vertices in G' and let R_3 be the isolated vertices in G'.
5: If $|R_2| > k^2 + k$ then return `‘‘no` (H, C, K)`-coloring of` G `exists’’` and stop.
6: Set up a partition $\mathcal{R} = (P_1, \ldots, P_q)$ of R_3 where
q is the number of different neighborhoods of vertices in R_3 and
for all $v, u \in R_3$ $(\exists_{1 \leq i \leq q}\ \{v, u\} \in P_i \Leftrightarrow N_G(v) = N_G(u)$.
7: Let $Q = \emptyset$.
8: For $i = 1, \ldots, q$,
if $|P_i| \leq k + 1$, then set $F_i = P_i$,
otherwise let F_i be any subset of P_i where $|F_i| = k + 1$.
Set $Q = Q \cup F_i$ and $P_i = P_i - F_i$.
9: Let $\mathcal{H}$ be the set of all the (H, C, K)-colorings of $G[R_1 \cup R_2 \cup Q]$.
10: If $|\mathcal{H}| = \emptyset$ then output `‘‘no` (H, C, K)`-coloring of` G `exists’’` and stop.
11: Let χ be an (H, C, K)-coloring in $\mathcal{H}$.
12: For any $i = 1, \ldots, q$ such that $P_i \neq \emptyset$,
Let u be a vertex of F_i such that $\chi(u) \in V(H) - C$.
For any $w \in P_i$, set $\chi'(w) = \chi(u)$.
13: Output $\chi \cup \chi'$ as an (H, C, K)-coloring of G.

Figure 2: Algorithm Find-coloring

Proof. Let us prove the correctness of the Find-Coloring algorithm (see figure 2).

Observe first that, for any (H, C, K)-coloring χ of G, $\chi^{-1}(C)$ is a vertex cover of G. This implies that, for any vertex $v \in \chi^{-1}(V(H) - C)$, $N_G(v) \subseteq \chi^{-1}(C)$, thus $|E(G)| \leq |\chi^{-1}(C)| = kn$. Moreoever, as $|\chi^{-1}(C)| = k$ we have that,for any $v \in \chi^{-1}(V(H) - C)$, $d_G(v) \leq k$. Therefore, in any (H, C, K)-coloring of G, any $v \in V(G)$ with $d_G(v) > k$ must be mapped to some vertex in C. So, the claim in Step **2** is correct.

To justify Step **5**, let $G' = G[V(G) - R_1]$. By definition of R_1, we have that, for all $v \in V(G')$, $d_G(v) \leq k$, and moreoever, at most k of them can be mapped to C. As C is a vertex cover of H, all the vertices in G' that are not mapped to C, are either isolated or connected to vertices in G'. Therefore, all non-isolated vertices of G' are in $\chi^{-1}(C) \cup N_G(\chi^{-1}(C))$ and, if there is a (H, C, K)-coloring of G, there are at most $k^2 + k$ non isolated vertices in G'.

Notice that Step **6** creates a partition $\mathcal{R} = (P_1, \ldots, P_q)$ of R_3 such that, for any $u, v \in R_3$, $\{u, v\} \in P_i$ iff $N_G(u) = N_G(v)$. For each P_i with size larger than $k+1$, the sampling in Step **8**, produces a set F_i with $k+1$ vertices. Therefore, as there are at most 2^k different neighborhoods, we have $|Q| \leq (k+1)2^k$. The sampling guaranties that, for any F_i with $F_i \neq P_i$, any (H, C, K)-colorings of $G[R_1 \cup R_2 \cup Q]$ maps at least one element of F_i to a vertex in $V(H) - C$. So, we can extended any such a coloring to an (H, C, K)-coloring of G, as it is done in Step **12**.

To finish the correcteness proof we have to prove that if G has an (H, C, K)-coloring then $\mathcal{H}$ is non-empty. Suppose now that $\tilde{\chi}$ is a (H, C, K)-coloring of G. For any i, $1 \leq i \leq q$, let $N_i = \tilde{\chi}^{-1}(C) \cap (P_i)$ and $M_i = P_i - N_i$. Notice that when $|P_i| > k$, $M_i \neq \emptyset$ and $|N_i| \leq k$ As all the vertices in P_i have the same neighborhood, we can obtain a new (H, C, K)-coloring of G by interchanging the images of two vertices in P_i. If we do so in such a way that every vertex in $N_i - F_i$ is replaced by vertices in F_i, we obtain a mapping whose restriction to $G[R_1 \cup R_2 \cup Q]$ is an (H, C, K)-coloring.

To verify the complexity. Step **1**, **2** and **3** take $O(kn)$ time. Steps **4** and **5** have a cost $O(n)$. Step **6** can be accomplished in $O(kn + 2^k)$ steps. Step **8** needs $O((k+1) \cdot 2^k)$ time. Notice that, as $|Q| \leq (k+1) \cdot 2^k$, $|V(G[R_1 \cup R_2 \cup Q])| \leq k^2 + 2k + (k+1) \cdot 2^k$ Step **9** can be done in $O(h^{k^2+2k+(k+1)\cdot 2^k})$ steps. The vertices of Q that map to $V(H) - C$ can be obtained in $O((k+1) \cdot 2^k)$ steps. Therefore, Step **12** can be done in $O(n)$ steps and the result follows. $\square$

A partial weighted assignment (H, C, K) is said to be *compact* when each connected component H' of H satisfies one of the following:

(1) $E(H' - C) = \emptyset$,

(2) $H'[C]$ is a non-empty reflexive clique with all its vertices adjacent with one looped vertex of $H' - C$, or

(3) $V(H') \cap C = \emptyset$ and H' contains at least one looped vertex.

Notice that the three conditions are exclusive. We call a connected component of H that satisfies (1) a *1-component*. We define analogously the *2-components* and the *3-components*

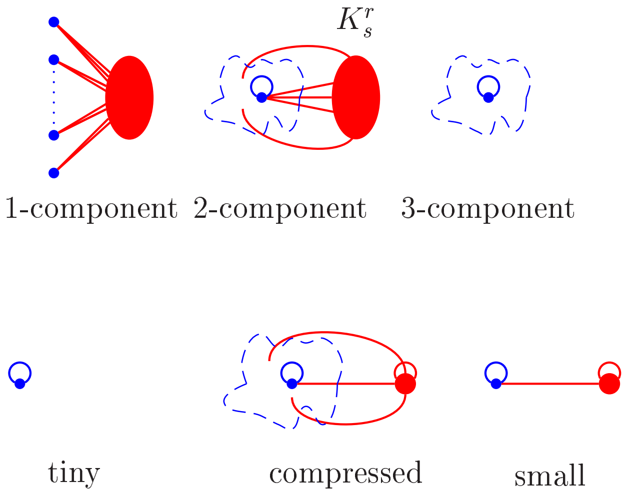

Figure 3: Components for a compact partial weighted assignment

of H. We call a 2-component H' of H *compressed* if $|V(H') \cap C| = 1$, and we call it *small* if it is compressed and $|V(H') - C| = 1$. We call a 3-component of H *tiny* if it consists only of a single looped non-weighted vertex. Figure 3 presents the different component types.

A compact partial weighted assignment (H, C, K) is said to be

- *compressed* if either all the 2-components of H are compressed or H has no 2-components,
- *small* if either all the 2-components of H are small or H has no 2-components,
- *tiny* if H if either all of its 3-components are tiny or H has no 3-components.

In order to prove the fixed parameter tractability for compact partial weighted assignments, we will reduce it to an equivalent problem with a simplifyed form. This will be achieved through a series of lemmas. The first results establishes the equivalence with positive partial weighted assignments.

Lemma 3.1. For any compact partial weighted assignment (H, C, K) there exist an equivalent one that is compact and positive. Moreover, such an assignment can be computed in $O(rh)$ steps.

Proof. Consider the set $S = \{c \in C \mid K(c) \neq 0\}$, as it is imposible to map vertices in G to vertices outside S, $(H, C, K) \sim (H, C, K)[S]$. Furthermore, as we are only removing parameterized vertices, $(H, C, K)[S]$ is a compact partial weighted assignment. $\square$

The next equivalence allows us to consider only compressed compact partial weighted assignments.

Lemma 3.2. For any compact partial weighted assignment (H, C, K), there exists an equivalent one that is compressed and compact. Moreover, such an assignment can be computed in $O(rh)$ steps.

Proof. To prove the claim, we show that for any compact partial weighted assignment (H, C, K), where H contains $m \geq 1$ non-compressed 2-components, there exists an equivalent compact partial weighted assignment (H', C', K') such that H contains $m - 1$ non-compressed 2-components.

Let H' be a non-compressed 2-component of H and set $D = C \cap V(H_i)$. Let $a_{\text{new}} \notin V(H)$ be a new vertex, and let, $k_D = \sum_{d \in D} K(d)$. We define (H', C', K') in the following way:

$$
\begin{aligned}
&V(H') = V(H) - D \cup \{a_{\text{new}}\},\\
&E(H') = E(H[V(H) - D]) \cup \{\{a, a_{\text{new}}\} \mid a \in N_H(D) - D\} \cup \{\{a_{\text{new}}, a_{\text{new}}\}\},\\
&C' = C - D \cap \{a_{\text{new}}\}, \text{ and}\\
&K' \text{ is such that } K'|_{C-D} = K \text{ and } K'(a_{\text{new}}) = k_D.
\end{aligned}
$$

Notice that (H', C', K') is a compact partial weighted assignment that can be constructed in $O(r|V(H')|)$ steps. Let us show that $(H', C', K') \sim (H, C, K)$. Given a graph G, if χ is a (H, C, K)-coloring of G, we define $\chi' : V(G) \to V(H')$ as

$$
\chi'(v) = \begin{cases} \chi(v) & \text{if } \chi(v) \notin D, \\ a_{\text{new}} & \text{otherwise.} \end{cases}
$$

Clearly χ' is a (H', C', K')-coloring of G.

On the contrary, given a (H', C', K')-coloring χ' of G, then $|{\chi'}^{-1}(a_{\text{new}})| = k_D$. Therefore, we can define a function $\sigma : {\chi'}^{-1}(a_{\text{new}}) \to D$ such that, for all $d \in D$, we have $|\sigma^{-1}(d)| = K(d)$, and define $\chi : V(G) \to V(H)$ as:

$$
\chi(v) = \begin{cases} \chi'(v) & \text{if } \chi'(v) \neq a_{\text{new}}, \\ \sigma(v) & \text{otherwise.} \end{cases}
$$

Notice that χ is a (H, C, K)-coloring of G. As (H', C', K') has $m - 1$ non-compressed 2-components, this completes the proof of the lemma. □

The next result establishes that we need to consider only small 2-components.

Lemma 3.3. For any compressed compact partial weighted assignment (H, C, K) there exists an equivalent one that is small and compact. Moreover, such an assignment can be computed in $O(h)$ steps.

Proof. Again we will show that for any compressed compact partial weighted assignment (H, C, K) with $m \geq 1$ non-small 2-components, there exists an equivalent compressed compact partial weighted assignment $(\tilde{H}, C', K')$ with $m - 1$ non-small 2-components.

Suppose that H' is a compressed 2-component of H such that $C \cap V(H') = \{c\}$ and c is adjacent in H' to a looped non-weighted vertex a. Let $\tilde{H}$ be the graph obtained from H by replacing H' with $H'' = H[\{b, c\}]$. Then $(\tilde{H}, C, K)$ is compressed and compact, has one non-small component less and can be constructed in $O(V(H'))$ steps.

Let us show that the two partial weighted assignments are equivalent. Notice that any $(\tilde{H}, C, K)$-coloring of G is also a (H, C, K)-coloring of G, because H'' is a subgraph of H' containing all the parameterized vertices in H'.

Let χ be an (H, C, K)-coloring of G, and consider the function $\rho : V(H) \to V(\tilde{H})$ such that $\rho(a) = a$ and $\rho(c) = b$ if $c \neq a$. Notice that ρ is a $\tilde{H}$-coloring of H, and the function $\rho \circ \chi : V(G) \to V(H')$ is a $(\tilde{H}, C, K)$-coloring of G.

As $(\tilde{H}, C, K)$ contains $m - 1$ non-small 2-components, the lemma follows. ☐

Now we will see that for the 3-components it is enough to consider only tiny components.

Lemma 3.4. For any compact parameter assignment (H, C, K) there exists a tiny compact parameter assignment equivalent to (H, C, K) with only one 3-component. Moreover, such an assignment can be computed in $O(|V(H)|)$ steps.

Proof. Let H' be a 3-component of H that is not tiny, and let a be a looped vertices in H'. We set $\tilde{H} = H - V(H') \cup (\{a\}, \{\{a, a\}\})$ and observe that $(\tilde{H}, C, K)$ is compact, can be constructed in $O(|V(H')|)$ steps, and has one non-tiny component less.

Given a graph G, if χ is a (H, C, K)-coloring of G, then so is $\chi' : V(G) \to V(H)$ where

$$\chi'(v) = \begin{cases} \chi(v) & \text{if } \chi(v) \notin V(H'), \\ a & \text{otherwise.} \end{cases}$$

Furthermore, as a is a looped vertex, χ' is also a $(\tilde{H}, C, K)$-coloring of G. Finally, as $\tilde{H}$ is a subgraph of H, then any $(\tilde{H}, C, K)$-coloring of G is also a (H, C, K)-coloring of G. Therefore $(H, C, K) \sim (\tilde{H}, C, K)$.

Now we can assume that all the 3-components in (H, C, K) are tiny 3-components. Let V' be the set of vertices in the 3-components of H, let a_{new} be a new vertex, and consider the graph $\tilde{H} = (H - V') \cup \{(\{a_{\text{new}}\}, \{\{a_{\text{new}}, a_{\text{new}}\}\})$. Then $(H, C, K) \sim (\tilde{H}, C, K)$, and this last assignment has only one tiny 3-component. ☐

Finally, we can further reduce the (H, C, K) coloring problem to one in which we have either small or tiny components, but not both.

Lemma 3.5. For any small and tiny compact partial weighted assignment (H, C, K), there exists a small and tiny partial weighted assignment $(\tilde{H}, \tilde{C}, \tilde{K})$ that is equivalent to (H, C, K) and such that $\tilde{H}$ does not contain both 2-components and 3-components. Moreover, such a partial weighted assignment can be constructed in $O(h)$ steps.

Proof. Assume that H has 2 and 3-componnets, by the previous lemma we can assume that H has only one tiny 3-component H'. Let b be a looped non-weighted vertex in one of the 2-components of H. We define the graph $\tilde{H} = H[V(H) - V(H')]$ and we observe that $(\tilde{H}, C, K)$ is compact, can be constructed in $O(|V(H)|)$ steps, and has no 3-components. Given a graph G, any $(\tilde{H}, C, K)$-coloring of G is also a (H, C, K)-coloring. If χ is a (H, C, K)-coloring of G, define $\chi' : V(G) \to V(H')$ by

$$\chi'(v) = \begin{cases} \chi(v) & \text{if } \chi(v) \notin V', \\ b & \text{otherwise.} \end{cases}$$

Clearly, χ' is a $(\tilde{H}, C, K)$-coloring of G. Thus, $(H, C, K) \sim (\tilde{H}, C, K)$ and the lemma follows. □

Now we can characterize the existence of an (H, C, K)-coloring for any compact small partial weigthed assignment.

Lemma 3.6. Let (H, C, K) be a partial weighted assignment that contains only small 2-components. Then a graph G has a (H, C, K)-coloring iff there exists a partition $P = \{\mathcal{G}_c\}_{c \in C}$ of the connected components of G such that, for any $c \in C$, $|V(\mathcal{G}_c)| \geq K(c)$.

Proof. Assume that $P = \{\mathcal{G}_c\}_{c \in C}$ is such a partition, then for each $c \in C$, we can choose a set S_c of k_c vertices of $\mathcal{G}_c$. Now, as all the components in H are small 2-componnets, any vertex $c \in C$ has an associated, looped vertex, denoted as b_c. Define $\chi_c : V(\mathcal{G}_c) \to V(H)$ such that

$$\chi_c(v) = \begin{cases} c & \text{if } v \in S_c, \\ b_c & \text{if } v \notin S_c. \end{cases}$$

As all the vertices of H are looped, the mapping $\chi = \cup_{i=1,\dots,r} \chi_i$ is an (H, C, K)-coloring of G.

Let now χ be a (H, C, K)-coloring of G. For any $c \in C$, define $\mathcal{G}_c$ as the gset of connected components of G that contain vertices in $\chi^{-1}(c)$. Clearly, the sum of the vertices in the components in $\mathcal{G}_c$ is at least $|\chi^{-1}(c)| = K(c)$. □

Lemma 3.7. For any small partial weighted assignment (H, C, K), there exists an algorithm checking, in $O(|C|^{\gamma(G)} \cdot \max\{\gamma(G), k\})$ steps, whether a graph G has a (H, C, K)-coloring.

Proof. For every connected component of G, there are $|C|^{\gamma(G)}$ possible partitions of the connected components of G into $|C|$ sets. Therefore, it is enought to check that the criterion of Lemma 3.6 is satisfied for some of them. But each check can be performed in $O(\max\{\gamma(G), k\})$ steps and the lemma follows. □

In the proof of the main fixed parameter tractability result we differenciate the case in which G has "many components"from the case in which G has "few components". In the first case, we use the Find-Coloring algorithm to select a "small" number of them that will be mapped to the 1-components, and we check if the "many" components left are enough to cover the other components. In the second case, the number of components of

G that can be "mapped" to components of H does not depend on the size of G, and we check the existence of an (H, C, K)-coloring using a combination of exhaustive search and application of the Find-Coloring algorithm.

Theorem 3.1. For any compact partial weighted assignment (H, C, K), the (H, C, K)-coloring problem is in FPT.

Proof. Let (H, C, K) be a compact partial weighted assignment, according to Lemmas 3.1–3.5, we can assume that (H, C, K) is a positive compact partial weighted assignment, where H may contain 1-components and either small 2-components or tiny 3-components but not both. Furthermore, if H has 3-componnets, it has only one tiny 3-component. As (H, C, K) is positive, we have that $r = |C| \leq k$.

The set C can be partitioned into two sets: C_1 containing the weighted vertices in the 1-components, and C_2 containing the weighted vertices in the small 2-components. Notice that if there are any 2-components, all of them are small we will name them by its unique weighted vertex, we use the notation $\{H'_c\}$, for $c \in C_2$, to denote the 2-components. Furthermore, each $c \in C_2$ has an associated, looped, non-weighted vertex b_c. For sake of simplicity we use the notation $K_1 = K|_{C_1}$, $K_2 = K|_{C_2}$, $r_1 = |C_1|$ and $r_2 = |C_2|$.

The graph H can be decomposed as the disjoint union of two graphs, H_1 and H_2, where H_1 is the disjoint union of the 1-componnets of H, and H_2 contains the remaining components.

Let G be a graph, if H has only 3-components then we can trivially compute an (H, C, K)-coloring of G in linear time. If H has only 1-components, we can directly apply Proposition 3.1. Therefore, we just need to give an algorithm for the remaining three cases.

Case 1. H has only 2-components. We consider two subcases:

Case 1.1. $\gamma(G) \geq k|C_2|$.
According to Lemma 3.6, we have to construct a $|C_2|$-partition P of the connected components of G such that any set $\mathcal{G}_c$ in P contains components of G with total number of vertices at least $K(c)$. This partition can be constructed in $O(n)$ steps, by a round-robin procedure assigning components to partitions.

Case 1.2. $\gamma(G) < k|C_2|$.
As $|C| \leq k$, the problem can be solved in $O(k^{k^2+2})$ steps by Lemma 3.7

Before we proceed to the analysis of the two remaining cases we define some additional notation. We partition the connected components of G in two blocks $\mathcal{G}_1$ and $\mathcal{G}_2$ such that $\mathcal{G}_1 = (G_1, \dots, G_m)$ contains all the G components that have a $(H_1, C_1, \leq K_1)$-coloring.

For each $1 \leq i \leq m$ we define a set A_i containing all the positive weighted assignments $K' \leq K_1$ for which G_i has a (H_1, C_1, K')-coloring. Notice that, by Lemma 2.1, the assignments in A_i represent the ways in which G_i can contribute towards covering the weighted vertices of the 1-components of H.

Notice that the sequence $(A_i)_{1 \leq i \leq m}$ can be computed in $O(kn + \gamma(G)k^k h^{k^2+2k+(k+1)2^k})$ steps by running the algorithm Find-Coloring for each one of the connected components of G. Observe that Find-Coloring needs to be called $O(\gamma(G)k^{|C_1|})$ times).

Notice that no assignment can be used more than k times, therefore, we can consider only the first k copies of any set in $(A_i)_{1\leq i\leq m}$. Formally, letting $J = \{j|1 \leq j \leq m \text{ and } |\{j < i \mid A_i = A_j\}| \geq k\}$, this set is $(A_i)_{i\notin J}$. This prunnig leaves a multiset formed with at most k copies of any subset of assignments $K' \leq K_1$. Therefore, $|J| \leq k^{k^2+k}$.

The combination of assignments of different componnents that guarantee a (H_1, C_1, K_1)-coloring for the 1-componnents of G can be obtained solving problem (*) with input $(A_i)_{i\notin J}$ and K_1. By Lemma 2.3 we can determine, in $O(k^{k^3+3k^2+2k+1})$ steps, the set $\mathcal{S}$ containing any possible solution of this instance of the (*) problem. Furthermore, $|\mathcal{S}| \leq k^{k^3+3k^2+2k)}$.

We now proceed with the two remaining cases of our analysis.

Case 2. H has 1- and 2-components.
Let $J' \subseteq \{1, \ldots, m\}$ be the 1-components selected in a solution to the (*) problem and, let f_i be the selected function form A_i. We know that, for any $i \in J'$, G_i has an (H_1, C_1, f_i)-coloring, let χ_i be one such a coloring. Then, by Lemma 2.1. $\chi = \cup_{i\in J'}\chi_i$ is a (H_1, C_1, K_1)-coloring of the graph G' formed by the disjoint union of the components $(G_i)_{i\in J'}$. Now we consider two subcases:

Case 2.1. $\gamma(G) \geq k(r - r_1 + 1)$.
In this case the graph $G^+ = G[V(G) - V(G')]$ has at least $k(r - r_1 + 1) - k = (r - r_1)k$ connected components and it is trivial to construct an $(r-r_1)$-partition P of the connected components of G^+ such that, for any set $\mathcal{G}_i \in P$, the total number of vertices is at least k. As, for any $c \in C_2$, we have that $K(c) \leq k$, Lemma 3.6 implies that G^+ has a (H_2, C_2, K_2)-coloring. Furthermore, an (H_2, C_2, K_2)-coloring of G^+, χ^+, can be constructed in $O(n)$ steps. Finally, setting $\chi = \chi' \cup \chi^+$, by Lemma 2.2, χ is a (H, C, K)-coloring of G.

Case 2.2. $\gamma(G) < k(r - r_1 + 1) \leq kr$.
It is enough to check whether there exist a selection J' in $\mathcal{S}$ such that $G^+ = G[V(G)-V(G')]$ has a (H_2, C_2, K_2)-coloring χ^+. Notice that if this is the case, Lemma 2.2 implies that $\chi' \cup \chi^+$ is a (H, C, K)-coloring of G. As $\gamma(G) \leq kr$, Lemma 3.7 implies that this last check can be done in $O(r^{\gamma(G')} \cdot \max\{\gamma(G', k\}) = O(k^{k^2+2})$ steps. Therefore, if χ^+, exists, it can be found in $O(k^{3k^2+k+2})$ steps.

Case 3. H has 1- and 3-components.
We proceed as in Case 2.1 to obtain an (H_1, C_1, K_1)-coloring χ' of G', then we define $\chi^+ : V(G) - V(G') \to V(H) - V(H_1)$ such that $\forall v \in V(G) - V(G', \chi^+(v) = c$, where c is the looped vertex in the 3-component of H. Then $\chi' \cup \chi^+$ is an (H, C, K)-coloring of G. The time complexity of this case is dominated by the one of Case 2. ☐

Therefore, the overall time complexity of deciding the (H, C, K)-coloring for a compact assignment is bounded by $O(n(k + h)) + \gamma(G)f(k, h))$.

Observe that by running $O(k^r)$ times an algorithm for the (H, C, K)-coloring we also have an algorithm for the $(H, C, \leq K)$-coloring problem.

Corollary 3.1. If (H, C, K) is compact, then the $(H, C, \leq K)$-coloring problem is in FPT.

4 Cases in P

In this section we show some cases of the (H, C, K)-coloring problem that can be solved by a polynomial time algorithm with time bound $O(n^{k+\alpha})$, where α is a constant independent of k.

Our first result relates the decision problem with the *list H-coloring* problem. In particular, we provide a polynomial time algorithm for the case in which the list H-coloring problem can be solved on the graph induced by the non-weighted vertices.

Theorem 4.1. For any K, the (H, C, K)-coloring problem can be solved in $O(n^{k+\alpha})$ steps whenever the list $(H - C)$-coloring can be solved in $O(n^\alpha)$ steps.

Proof. The basis of the algorithm is the construction of a list $(H - C)$-coloring instance for each partial assignment of vertices to the parameterized nodes that constitute a $G[C]$-coloring. As the number of such partial assignments is at most n^k, we obtain the desired result.

Assume that we have a $H[C]$-coloring ρ of some $D \subseteq V(G)$, with $|D| \leq k$. We consider the graph $G' = G[V(G) - D]$ and associate to each vertex $v \in V(G')$ the list

$$L(v) = \begin{cases} \bigcap_{N_G(v) \cap D} (N_H(\rho(u)) - C), & \text{if } N_G(v) \cap D \neq \emptyset, \\ V(H - C) & \text{otherwise.} \end{cases}$$

Notice that ρ can be completed to form a (H, C, K)-coloring χ of G, so that $\chi|_{V(G')}$ is compatible with the list L. Furthermore, if χ' is a list $(H - C)$-coloring of (G', L), then $\rho \cup \chi'$ is an (H, C, K)-coloring for G. $\square$

The previous theorem includes the case when $H - C$ is a reflexive interval graph [FH98], the case when $H - C$ is an irreflexive graph whose complement is a circular arc graph of clique covering number two [FHH99] and the case in which H is a *general* bi-arc graph [Hel01].

The above proof can be extended to show tractability of a weighted extension of any graph F for which the list F-coloring is still NP-complete, but the F-coloring is in P.

Theorem 4.2. Let F_1 be either a bipartite graph with or without loops or a graph with at least one loop, and let F_2 be any graph. Then, for any K, the $(F_1 \oplus F_2, V(F_2), K)$-coloring can be solved in polynomial time for any partial weighted assignment K.

Proof. As we have all the edges between $V(F_1)$ and $V(F_2)$, the lists associated to each vertex are just all the vertices in F_1. Therefore, we have to solve a F_1-coloring problem, which can be solved in polynomial time. $\square$

Finally we mention that any weighted extension of a graph F for which the F-coloring is in P, is also polynomially solvable in the trivial case when $K = (0, \dots, 0)$.

5 NP-complete cases

In this section we will prove first that if $H - C$ is not bipartite and does not contain a loop then the $(H, C, \leq K)$-coloring problem is NP-complete, then we will show that some weighted extensions of bipartite graphs with or without looped vertices are also NP-complete. We will consider only the cases where (H, C, K) is positive, as by Lemma 3.1 these cases imply the general case. All the problems considered here belong trivially to the class NP, therefore, in the proofs, we will show only the reductions proving its NP-hardness.

Let (H, C, K) be a partial weighted assignment and let c be a vertex in C. We say that (H, C, K) is *c-reducible* if H is has an $(H - \{c\})$-coloring mapping c to a vertex in $V(H) - C$. We call the vertex c a *reducible* vertex of (H, C, K). We say that (H, C, K) is *reducible* if it is c-reducible for some $c \in C$.

The following lemma indicates that for any reducible partial weighted assignment there exists an irreducible equivalent partial weighted assignment.

Lemma 5.1. If (H, C, K) is c-reducible then $(H, C, \leq K) \sim_{\leq} (H, C, \leq K)[V(H) - \{c\}]$.

Proof. Let $H' = H - \{c\}$. As c is a reducible vertex, there exists some H'-coloring φ of H such that $\varphi(c) = a \in V(H) - C$. As φ is an H'-coloring, it should map each vertex $b \in N_H(c)$, to one of the neighbors of a. Therefore, $\varphi(N_H(c)) \subseteq N_{H'}(a)$.

We now define a mapping $\rho : V(H) \to V(H')$ by setting $\rho(c) = a$ and $\rho(b) = b$, for any $b \in V(H) - \{c\}$. Then ρ is a H'-coloring of H, because the only edges that are not mapped to themselves are those having c as endpoint but, such edges are also mapped by ρ to edges.

Given a graph G, let χ be a $(H, C, \leq K)$-coloring of G. As ρ is a H'-coloring of H and the composition of homomorphisms is an homomorphism, we have that $\chi' = \rho \circ \chi$ is an H'-coloring of G. Also, by the construction of χ' and the fact that $\chi'(c) \notin C$, for any $b \in C - \{c\}$, $|{\chi'}^{-1}(b)| \leq K(b)$. Therefore χ' is an $(H, C, \leq K)[V(H) - \{c\}]$-coloring for G.

Suppose now that χ is an $(H, C, \leq K)[V(H) - \{c\}]$-coloring for G. As $H' = H - \{c\}$, χ is also an $(H, C, \leq K)$-coloring of G, and the lemma follows. $\square$

Therefore we can remove all the reducible weighted vertices while maintaining the problem equivalence.

Lemma 5.2. For any partial weighted assignment (H, C, K), where the $(H - C)$-coloring problem is NP-complete, the $(H, C, \leq K)$-coloring problem is also NP-complete.

Proof. From Lemma 5.1 we can assume that (H, C, K) is a non reducible partial weighted assignment. We set $H' = H - C$ and, we will reduce the H'-coloring problem to the $(H, C, \leq K)$-coloring problem.

Define an auxiliary graph (see Figure 4). Let $\tilde{H}$ be the graph constructed from H' such that each vertex $c \in C$ is replaced by a set of vertices $V_c = \{v_1^c, \ldots, v_{K(c)}^c\}$, any edge connecting two vertices in $V(H) - C$ is maintained, an edge connecting a vertex $a \in V(H) - C$ to a vertex $c \in C$ is replaced by the set $\{\{a, v_i^c\} | 1 \leq i \leq K(c)\}$, and an edge connecting a two vertices $c, d \in C$ is replaced by the set $\{\{v_i^c, v_j^d\} | 1 \leq i \leq K(c), 1 \leq$

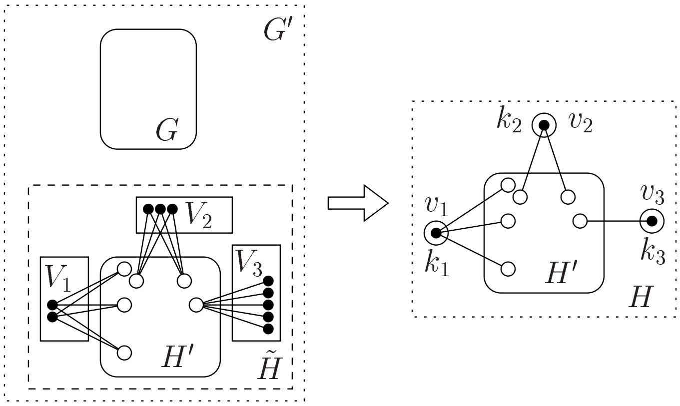

Figure 4: The reduction of Lemma 5.2

$j \leq K(d)\}$. $\tilde{H}$ is $(H, C, \leq K)$-colorable by projecting all the copies to the corresponding vertex. Moreoever, it verifies that for any $(H, C, \leq K)$-coloring ρ of $\tilde{H}$, for any $c \in C$, $|\rho^{-1}(c)| = K(c)$.

Given a graph G, we consider the graph $G' = G \cap \tilde{H}$. We claim that G has an H'-coloring iff G' has a $(H, C, \leq K)$-coloring. On one direction, the union of a $(H, C, \leq K)$-coloring of $\tilde{H}$ and a H'-coloring of G gives a $(H, C, \leq K)$-coloring of G'. On the contrary, let χ be a $(H, C, \leq K)$-coloring of G'. As $\chi|_{V(\tilde{H})}$ is a $(H, C, \leq K)$-coloring of H', all the vertices of G must be mapped to $V(H')$ but then $\chi|_{V(G)}$ is a H' coloring of G. □

Using Lemma 5.2 and the result of P. Hell and J. Nešetřil, on the NP-completeness of H-coloring, we conclude to the following.

Theorem 5.1. The $(H, C, \leq K)$-coloring problem is NP-complete if $H - C$ is not bipartite and does not contain a loop.

Notice that the existence of a polynomial algorithm to solve the (H, C, K)-coloring problem implies a polynomial algorithm for the $(H, C, \leq K)$-coloring problem. Therefore, we have the following corollary.

Corollary 5.1. The (H, C, K)-coloring problem is NP-complete if $H - C$ is not bipartite and does not contain a loop.

We will prove that the condition of Theorem 5.1 is not a necessary condition. In particular we will prove that the (H, C, K)-coloring remains NP-compete for certain weighted extensions of graphs F that are either bipartite or that contain at least one looped vertex. Notice that in the case that $H - C$ contains a looped vertex the $(H, C, \leq K)$-coloring is trivially in P.

The NP-completeness proof in [FHH99] for the list H-coloring on bipartite graphs that contain a chordless cycle of length $2k - 2$ for $k > 3$ is a reduction from the k-coloring problem. The reduction transforms a given graph G into a new graph G', and a list.

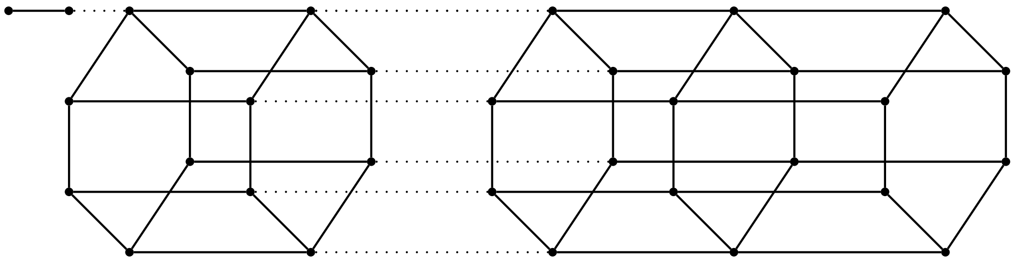

Figure 5: The gadget from [FHH99] replacing an edge (x, y), C_0 is common to all the edges

The graph G' is obtained from G by replacing each edge by the gadget given in figure 5. Making the cycle C_0 common to all the edges. The list L fixes a unique possible image of the vertices in C_0 in the chordless cycle in H. The remainig vertices get list $V(H)$.

Theorem 5.2 ([FHH99]). Let H be a bipartite graph that contain a chordless cycle of length $2k-2$, for some $k \geq 3$. Then, given a graph G, G has a k-coloring iff (G', L) has a list H-coloring.

We can achieve the same result replacing the additional list by an adequate parametrized extension (H', C, K) of H. To explain this idea, let us define a a *rigid* extension of a cycle. Given a cycle C_α, we define a graph $\tilde{C}_\alpha$ by adding a set E_α with α additional weighted vertices, each new vertex corresponds to an edge in C_α and it is connected to the end-points of the associated edge (see figure 6). Rigid extensions of reflexive or irreflexive cycles verify the following.

Lemma 5.3. Let C_α be a cycle with $\alpha \geq 4$ vertices, let $\tilde{C}_\alpha$ be its rigid extension and let 1_{E_α} the functiont constant 1 on E_α. Then any $(\tilde{C}_\alpha, E\alpha, 1_{E_\alpha})$-coloring of $\tilde{C}_k$, preserves the cyclical ordering of the vertices in C_k.

Proof. Observe that the triangles must be mapped to triangles, but each triangle in C_α can get at most one triangle. □

Let H be a bipartite graph that contain a chordless cycle of length $k \geq 4$. We reduce the k-coloring problem to the (H', C, K)-coloring problem, where H' is obtained from H making a rigid extension of the chordless cycle, setting C to the vertices added and makimg $K = 1_C$. The reduction is a modification of the construction in [FHH99]: replace C_0 by its rigid extension. The graph G' is obtained from G by replacing each edge by the gadget given in Figure 7. Finally make $\tilde{C}_0$ common to all the edges. From Lemma 5.3 we have that the vertices in $\tilde{C}_0$ must be mapped to the unique triangles in H', and From Lemma 5.3 they must preserve the cyclic ordering. Therefore, from Theorem 5.2 we have the following

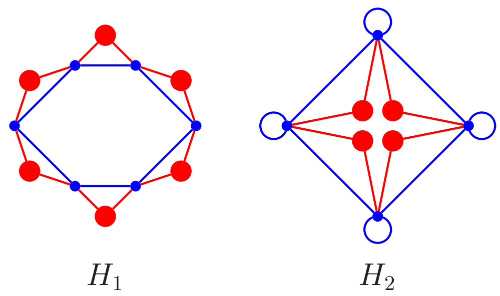

Figure 6: The rigid cycle extensions for irreflexive and reflexive cycles

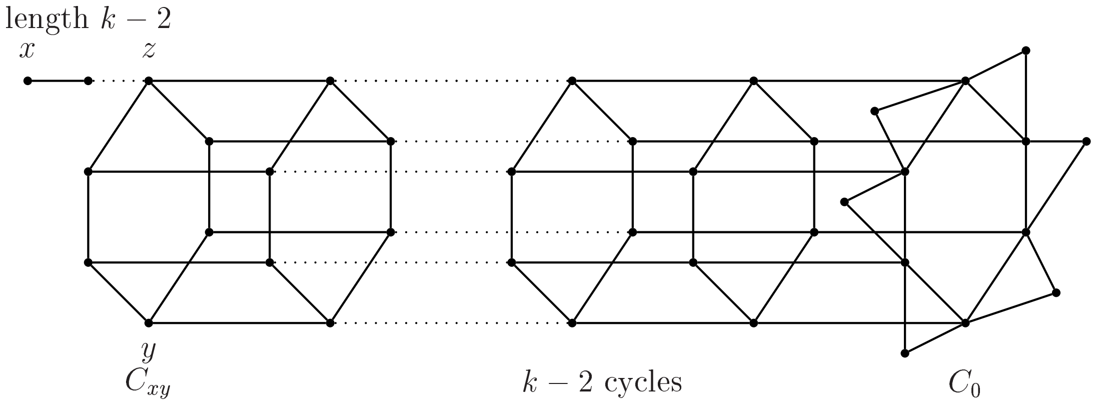

Figure 7: The gadget replacing an edge (x, y), C_0 is common to all the edges

Theorem 5.3. If F is a bipartite graph that contains a chordless cycle of length greater than four, then F has a weigthed extension that is NP-complete.

We can work in the same way with the construction given as reduction in [FH98] for the case that H is a bipartite reflexive graph that contains a chordless cycle with length 4 or more. Therefore, we have

Theorem 5.4. If F is a reflexive bipartite graph that contains a chordless cycle of length greater than four, then F has a weigthed extension that is NP-complete.

Finally notice that, by Theorem 4.2, the graphs verifying the conditions in Theorems 5.3 and 5.4, have another weighted extension so that the corresponding parameterized coloring problem is in P.

6 Results on list (H, C, K)-coloring

Notice that, given a (H, G)-list and provided that $E(H - C) = \emptyset$, Algorithm Find-Coloring can be easily adapted so to compute a list (H, C, K)-coloring of G. All the steps except Step **6** are exactly the same. In Step **6**, the unique difference is that we define $\mathcal{R} = (P_1, \dots, P_q)$ in such a way that, for all $v, u \in R_3$, there exists i, $1 \leq i \leq q$, such that

$$\{v, u\} \in P_i \Leftrightarrow (N_G(v) = N_G(u)) \text{ and } (L(v) = L(u))).$$

This means that we partition the vertices of R_3 not only with respect to their neighbors but also with respect to the sets of images that they may have, according to the (H, G)-list. The only difference between the correcteness of Find-Coloring algorithm, as proved in Proposition 3.1, and the correcteness proof for this extension of Find-Coloring is that now $|Q| \leq (k+1) \cdot 2^k \cdot 2^h$. We conclude to the following extension of Proposition 3.1.

Theorem 6.1. If (H, C, K) is a partial weighted assignment where $E(H - C) = \emptyset$, then there exists an $O(kn + h^{k^2+2k+(k+1)\cdot 2^{k+h}})$ time algorithm checking whether an input graph G with n vertices has a list (H, C, K)-coloring and, if yes, outputs the corresponding mapping.

Theorems 4.1 and 5.1 can be generalized for the list (H, C, K)-coloring as follows.

Theorem 6.2. For any K, the list (H, C, K)-coloring problem can be solved in $n^{k+\alpha}$ steps whenever the list $(H - C)$-coloring can be solved in $O(n^\alpha)$ steps.

Proof. The proof of this theorem is a direct extension of the proof of Theorem 4.1, taking care of the (H, G)-list L, when constructing the list $(H - C)$-coloring instance.

Assume that the input is a (H, G) list L, and that we have a list $G[C]$-coloring χ for some $D \subseteq V(G)$, with at most k vertices and the list induced by L, to C. We consider the graph $G' = G[V(G) - D]$ and associate to each vertex $v \in V(G')$ the list

$$L'(v) = \begin{cases} (L(v) - D) \cap \bigcap_{N_G(v) \cap D} (N_H(\varphi(u)) - C), & \text{if } N_G(v) \cap D \neq \emptyset, \\ (L(v) - D) \cap V(H - C) & \text{otherwise.} \end{cases}$$

Then (G', L') has a list $(H - C)$-coloring if and only if the coloring χ can be extended to a list (H, C, K) coloring of (G, L). $\square$

Theorem 6.3. The list $(H, C, \leq K)$-coloring problem is NP-complete if the list $(H - C)$-coloring is NP-complete.

Proof. Notice that the $(H, C, \leq K)$-coloring problem can be reduced to the list $(H, C, \leq K)$-coloring problem in the following way: Associate to a given graph G the list L, defined as $L(u) = V(H)$, for any $u \in V(G)$. $\square$

7 Remarks and open problems

Theorems 6.2 and 6.3 constitute a dichotomy for the list (H, C, K) coloring to be in P. The results in Theorems 4.1 and 5.1 are sharp in the following sense: If F is a graph where the F-coloring is NP-complete then for any weighted extension (H, C, K) of F the (H, C, K)-coloring problem is also NP-complete. On the other hand, if the F-coloring is in P then *there exist* a weighted extension (H, C, K) of F so that the (H, C, K)-coloring is in P. Moreover the results in Theorem 4.1 and 5.1 are also sharp in the sense that if F is a graph where the list F-coloring is in P then for any weighted extension (H, C, K) of F, the (H, C, K)-coloring is also in P and on the other hand, we provided some examples where the F-list coloring is NP-complete and F has a weighted extension (H, C, K) such that (H, C, K)-coloring is also NP-complete.

The above observations argue that there are graphs that for different weighted extensions produce parameterized coloring problems with different complexities. It seems to be a hard problem to achieve a dichotomy distinguishing those parameterized assignments (H, C, K) of a given graph F for which the (H, C, K)-coloring problem is NP-complete or in P. However we conjecture the following.

Conjecture 1. For any graph F such that the list F-coloring is NP-complete, there is a weighted extension (H, C, K) of F such that the (H, C, K)-coloring is also NP-complete.

All of our results on NP-completeness indicate that this frontier depends not only on $H - C$ but also on the structure imposed by the weighted vertices. However, we observe that in all our complexity results – positive or negative – this dichotomy does not depend on the choice of the numbers in K when K is positive.

We now fix our attention on whether there are cases more general than those of Theorem 3.1 where (H, C, K)-coloring is in FPT. In particular we conjecture that the condition of Theorem 3.1 is also necessary.

Conjecture 2. For any partial weighted assignment (H, C, K), if (H, C, K) is compact then the (H, C, K)-coloring problem, with the numbers in K as parameters, is in FPT, otherwise it is $\mathsf{W}[1]$-hard.

In support of our dichotomy conjecture we recall that the parameterized independent set problem, that is known to be $\mathsf{W}[1]$-complete [DF99] falls in this area.

We mention that our second conjecture is quite general to express several open problems in parameterized complexity such as

1. (Parameter: k) Does G contain an independent set S, where $|S| = k$, and such that $G[V(G) - S]$ is bipartite? (This problem can be seen as a parameterization of 3-coloring where some color should be used exactly k times.)

2. (Parameter: k) Is there a set $S \subseteq V(G)$, where $|S| = k$ and $G[V(G) - S]$ is bipartite?

3. (Parameters: k, l) Does G contain $K_{k,l}$ as subgraph?

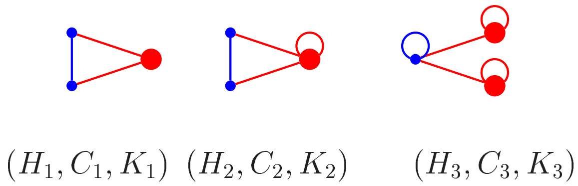

Figure 8: Three weighted extensions conjectured to be W[1]-hard when K is parameterized

This problems correspond to the the parameterized colorings given in figure 8. The (H_3, C_2, K_3)-coloring problem is equivalent with asking if the complement of G contains K_{k_1,k_2} as a subgraph.

References

[DF99] R. G. Downey and M. R. Fellows. *Parameterized complexity*. Springer-Verlag, New York, 1999.

[DST01] Josep Díaz, Maria Serna, and Dimitrios M. Thilikos. Evaluating H-colorings of graphs with bounded decomposability. In *Proceedings of The 7th Annual International Computing and Combinatorics Conference, COCOON 2001*, 2001.

[FH98] Tomas Feder and Pavol Hell. List homomorphisms to reflexive graphs. *Journal of Combinatorial Theory (series B)*, 72(2):236–250, 1998.

[FHH99] Tomas Feder, Pavol Hell, and Jing Huang. List homomorphisms and circular arc graphs. *Combinatorica*, 19(4):487–505, 1999.

[GHN00] Anna Galluccio, Pavol Hell, and Jaroslav Nešetřil. The complexity of H-colouring of bounded degree graphs. *Discrete Mathematics*, 222(1-3):101–109, 2000.

[Hel01] Pavol Hell. Personal communication. 2001.

[HN90] P. Hell and J. Nešetřil. On the complexity of H-coloring. *Journal of Combinatorial Theory (series B)*, 48:92–110, 1990.

[TP97] 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.