



Some (in)tractable Parameterizations of Coloring and List-Coloring

Pranav Arora¹, Aritra Banik¹, Vijay Kumar Paliwal^{1(✉)},
and Venkatesh Raman²

¹ Indian Institute of Technology Jodhpur, Jodhpur, India
arorapranav53@gmail.com, aritrabanik@gmail.com, getvijaypaliwal@gmail.com

² The Institute of Mathematical Sciences, HBNI, Chennai, India
vraman@imsc.res.in

Abstract. Graph Coloring and its generalization list coloring are fundamental graph optimization problems with various applications. Most versions of the problems are hard in several paradigms including approximation and parameterized complexity. We consider a few versions of the problems that are polynomial time solvable, and try to extend the notion of feasible algorithms by parameterizing suitably in the paradigm of parameterized complexity. More specifically,

- It is known that given a planar graph with any list of size 5 for each vertex, there is a proper coloring of the graph such that each vertex gets its color from its list. We show that if the graph is k vertices away from a planar graph, then deciding whether such a coloring exists is PARA-NP-hard when parameterized by k , i.e. it is NP-hard for even constant values of k . It is known that any graph with maximum degree 3 is 3-colorable unless the graph is a 4-clique. We show that if the graph is k vertices away from a maximum degree 3 graph, then determining whether it is 3-colorable is PARA-NP-hard when parameterized by k .
- It is known that if each vertex has a list of size 2, then the list coloring which asks whether there is a coloring respecting the lists is polynomial time solvable. We show that if only k vertices have lists of size more than 2, then the problem becomes W[1]-hard.
- It is known that determining whether a graph on n vertices is $n - k$ colorable, is fixed-parameter tractable on k . We consider the list coloring variation of it where each vertex has a list of size $n - k$ and we ask whether the graph has a coloring respecting the lists of colors. We show that the problem has an XP algorithm, i.e. an algorithm with runtime $n^{O(k)}$. At least this shows that the problem cannot be PARA-NP-hard unless $P = NP$. We leave open the question whether the problem is fixed-parameter tractable.
- Finally, it is known that 2-LIST COLORING is polynomial time solvable. If there is no such coloring, then we address the following natural question: are there k vertices or edges whose removal results in a feasible coloring. We show that these versions are fixed-parameter tractable when parameterized by k . These generalize the odd cycle transversal problem and edge-bipartization problem which are well-studied problems particularly in parameterized complexity.

1 Introduction

1.1 Problems and Motivation

The graph coloring problem is one of the fundamental combinatorial optimization problems with applications in scheduling, register allocation, pattern matching and many other active research areas. Given a graph $G = (V, E)$, the graph coloring problem is a way to assign colors/labels to vertices of a graph such that no two adjacent vertices share the same color. Such a coloring is also known as a *proper* coloring. The smallest number of colors needed to color a graph G is called its chromatic number, and is denoted by $\chi(G)$. Determining whether a graph is 3-colorable is NP-hard [7] while the 2-coloring problem has a linear time algorithm. It is even hard to approximate the chromatic number in polynomial time. The 3-coloring problem remains NP-complete even on 4-regular planar graphs [6]. There are some generalizations and variations of ordinary graph colorings which are motivated by practical applications. For example, sometimes a set of feasible colors $L(v)$ is attached with each vertex $v \in V$ and we may require that the vertex v be colored from a color from $L(v)$. This takes us to the *list coloring* problem, and clearly this is a generalization of standard graph coloring, and hence is considerably harder.

LIST COLORING PROBLEM

Input: A graph $G = (V, E)$ and a LIST of $|V|$ many set of colors, $L(v)$ for each $v \in V$

Question: Find an assignment of colors $c : V \rightarrow \cup_{v \in V} L(v)$ such that for any vertex v , $c(v) \in L(v)$ and for any two adjacent vertices u and v , $c(v) \neq c(u)$

A LIST is ℓ -REGULAR if each set contains exactly ℓ colors. ℓ -REGULAR LIST COLORING problem is to decide whether $G = (V, E)$ has a coloring that respects L , where L is ℓ -REGULAR. A graph G is ℓ -CHOOSABLE if for every ℓ -regular LIST L of G , there exists a coloring which respects L .

CHOOSABILITY PROBLEM

Input: A graph $G = (V, E)$ and an integer ℓ

Question: Determine whether G is ℓ -CHOOSABLE

Cai [2] in one of the earliest papers on parameterizations of graph coloring studied various parameterizations. He showed surprisingly that it is NP-hard to determine whether a graph that is two vertices away from a bipartite graph is 3-colorable. A graph is k vertices away from a graph satisfying a property (say planar or bipartite), if there are k vertices in the graph whose removal results in a graph satisfying the property. We say that such a graph is planar $+kv$ if it has k vertices whose deletion results in a planar graph. We consider such parameterizations in this paper for graph coloring and list coloring and give hardness and FPT (fixed-parameter tractable) results.

We begin with the notions of parameterized complexity before we explain our results.

1.2 Parameterized Complexity

The goal of Parameterized Complexity is to find ways of solving NP-hard problems more efficiently than brute force: here the aim is to restrict the combinatorial explosion to a parameter that is hopefully much smaller than the input size. A *parameterization* of a problem is assigning a positive integer parameter k to each input instance. Formally we say that a parameterized problem is *fixed-parameter tractable* (FPT) if there is an algorithm \mathcal{A} (called a *fixed parameter algorithm*), a computable function f , and a constant c such that, given problem instance (x, k) , \mathcal{A} correctly solves the problem in time bounded by $f(k) \cdot |x, k|^c$, where x is the input and k is the parameter [4]. There is also an accompanying theory of parameterized intractability using which one can identify parameterized problems that are unlikely to admit FPT algorithms. These are essentially proved by showing that the problem is W-hard.

A parameterized problem is called *slice-wise polynomial* (XP) if there exists an algorithm \mathcal{A} , two computable functions f, g such that, given problem instance (x, k) , the algorithm \mathcal{A} correctly solves the problem in time bounded by $f(k) \cdot |x, k|^{g(k)}$, where x is the input and k is the parameter [4]. The complexity class containing all slice-wise polynomial problems is called XP. We say that a parameterized problem is *para-NP-hard* if the problem is NP-hard for some fixed constant value of the parameter. *para-NP-hard* problems are not in XP unless $P = NP$.

1.3 Our Results

We start with parameterization of simple polynomial time solvable cases of COLORING and LIST COLORING. Thomassen [15] showed that every planar graph is 5-choosable. Therefore, by definition of choosability, planar graph is always a *Yes* instance for 5-REGULAR LIST COLORING. Our first result is that 5-REGULAR LIST COLORING is *para-NP-hard* for planar $+kv$ graphs.

Garey et al. [7] showed that COLORING is polynomial time solvable for graphs with maximum degree 3, actually all such graphs other than 4-clique are 3-colorable. Now an interesting question is, if we are given a graph which has a set of k vertices whose deletion makes the graph maximum degree 3, how hard is COLORING on this graph. First we define the problem formally. We denote graph of maximum degree d by G_d .

$G_3 + kv$ COLORING

Parameter: k

Input: A graph $G = (V, E)$ with a set $S \subseteq V$, $|S| = k$ where $G \setminus S$ is maximum degree 3 graph.

Question: Determine whether G is 3-colorable.

We show this also to be *para-NP-hard*.

In case of general graphs no results are known for LIST COLORING. Observe that for general graph k -REGULAR LIST COLORING is *para-NP-hard* as the problem is a generalization of k -coloring. It is known that if we ask whether the graph can be properly colored with $(n - k)$ colors, then the problem is FPT [4]. We consider a similar variation on LIST COLORING.

$(n - k)$ -REGULAR LIST COLORING

Parameter: k

Input: A graph $G = (V, E)$ and a $(n - k)$ -REGULAR LIST L , where $n = |V|$

Question: Determine whether there exists a feasible coloring of G with respect to L

We show that $(n - k)$ -REGULAR LIST COLORING is in XP which is the most technical part of the paper. It remains as an open problem whether the problem is fixed-parameter tractable.

It is known [5] that 2-REGULAR LIST COLORING is polynomial time solvable for general graphs. As we cannot see a formal proof, we give it here for completeness. We have the following theorem.

Theorem 1. 2-REGULAR LIST COLORING is polynomial time solvable for general graphs.

Proof. Let $G = (V, E)$ be a graph and L be a 2-regular list assignment for G . This problem can be reduced to 2-CNF SAT which is polynomial time solvable [1]. For each $u \in V$, define a variable x_u . x_u is set to true if the first color from its list is assigned to vertex u , otherwise x_u is set to false. For each edge $(u, v) \in E$, add clauses depending on the common colors in $L(u)$ and $L(v)$. No clauses need to be added if there are no common colors. There are four cases:

1. First color of $L(u)$ and first color of $L(v)$ is same. Add clause $(\overline{x_u} \vee \overline{x_v})$.
2. First color of $L(u)$ and second color of $L(v)$ is same. Add clause $(\overline{x_u} \vee x_v)$.
3. Second color of $L(u)$ and first color of $L(v)$ is same. Add clause $(x_u \vee \overline{x_v})$.
4. Second color of $L(u)$ and second color of $L(v)$ is same. Add clause $(x_u \vee x_v)$.

These clauses ensure that adjacent vertices do not get the same color because if any two adjacent vertices get the same color then at least one of the clauses will be falsified. Suppose that there exists an assignment that satisfies the 2-CNF SAT formula. If x_u is true, then u is assigned the first color in its list, otherwise it is assigned the second color. Conversely, if there exists a coloring of G with the given L , then x_u is set to true if it is assigned the first color from its list, otherwise it is set to false. Now, since no two adjacent vertices are assigned same color, none of the clauses will be falsified. \square

Two natural questions are: (1) if a graph $G = (V, E)$ is not colorable with respect to a 2-regular list L then is it possible to color at least k vertices of the graph respecting L ? This problem is known to be W[1]-hard [9]. In this paper we address the second natural question (which is sometimes called ‘parameteric dual’ defined below).

Vertex Parameterized 2-REGULAR LIST COLORING

Parameter: k

Input: A graph $G = (V, E)$ with a 2-regular list assignment L .

Question: Determine whether it is possible to color $n - k$ vertices of G with respect to L .

This is the same as asking whether we can delete k vertices and obtain a list coloring of the graph. If the lists (of size 2) of each vertex is the same, then note

that this problem is the same as asking whether there are k vertices whose deletion results in a bipartite graph, which is the well-studied odd cycle transversal (OCT) problem [11, 14]. We show that Vertex Parameterized 2-REGULAR LIST COLORING, which is a generalization of OCT, is also FPT.

We also consider a closely related problem that is whether we can delete k edges such that resultant graph can be colored satisfying the given 2-regular list assignment. We define the problem formally as follows:

Edge Parameterized 2-REGULAR LIST COLORING **Parameter:** k
Input: A graph $G = (V, E)$ with a 2-regular list assignment L .
Question: Can we delete at most k edges such that the resultant graph can be colored respecting L ?

This problem is generalization of the well known edge bipartization problem. We show that this problem is also FPT.

As mentioned earlier, 2-REGULAR LIST COLORING is polynomial time solvable. In a 2-REGULAR LIST COLORING instance, all vertices have lists of colors of size exactly 2. Another parameterization we consider is that if only k vertices have lists of colors of size more than 2, then how does the complexity of LIST COLORING change. We first define the problem formally.

Parameterized 2-REGULAR LIST COLORING **Parameter:** k
Input: A graph $G = (V, E)$ and, a set $S \subseteq V$, $|S| \leq k$ and a list assignment L such that the size of lists of colors of vertices in $V \setminus S$ is at most 2.
Question: Determine whether it is possible to color the graph respecting L

We show that Parameterized 2-REGULAR LIST COLORING is $W[1]$ -hard. On the other hand, if the maximum size of lists of colors for the vertices in S is m , then we show that if m bounded by a constant or is also a parameter, then the problem becomes FPT.

1.4 Organization of the Paper

In Sect. 2 we show the PARA-NP-hardness results on planar $+kv$ graphs and $G_3 + kv$ graphs. Next we show that Parameterized 2-REGULAR LIST COLORING is $W[1]$ -hard and that it becomes FPT with stronger parameterization or constant upper bound on the size of lists of colors. In Sect. 3 we present the XP algorithm for $(n - k)$ -REGULAR LIST COLORING. In Sect. 4 we show that the Vertex Parameterized 2-REGULAR LIST COLORING and Edge Parameterized 2-REGULAR LIST COLORING are FPT.

1.5 Related Work

Besides the results of Cai [2] mentioned in the introduction, there are other works on parameterizing the vertex coloring problems on graphs which are close to some special families. Formally if F is a graph family, then $F + kv$ is another family of graphs, which have only k vertices whose deletion results into F graph.

Now for example, COLORING problem, when parameterized by k , is W[1]-hard for chordal + kv graphs, interval + kv graphs and complete + kv graphs [12].

It is known that not every graph is k -CHOOSABLE. Therefore a natural question is, given a graph G and a LIST can we find a coloring in polynomial time. In a recent paper [5] Dabrowski et al. have addressed many such LIST COLORING problems.

2 Hardness Results

Theorem 2. 5-REGULAR LIST COLORING is PARA-NP-hard on planar + kv graphs.

Proof. We will prove that 5-REGULAR LIST COLORING is PARA-NP-hard on planar + $1v$ graphs. It is known that 4-REGULAR LIST COLORING is NP-complete for planar graphs even if every list contains four colors from $\{1, 2, 3, 4, 5\}$ [5]. We will show a polynomial reduction from this problem to our problem.

Suppose we are given an instance $G = (V, E)$ with the 4-regular list assignment. We create 5 copies of this instance and call them G_1, G_2, G_3, G_4 and G_5 . Now we add a new color 6 in lists of all vertices of G_1 . Similarly, we add colors 7, 8, 9 and 10 to all vertices of G_2, G_3, G_4 and G_5 respectively. Now we add a special vertex s with list of colors $\{6, 7, 8, 9, 10\}$. We make this vertex s adjacent to all vertices in $G_i \forall 1 \leq i \leq 5$. The resultant graph is planar + $1v$ graph because each G_i is planar and union of G_i s is also a planar graph. Also each vertex in the new instance has 5 colors in the lists. Hence this coloring instance is 5-REGULAR LIST COLORING instance on planar + $1v$ graph. We show that the new instance is colorable if and only if the original instance is an Yes instance.

If the original instance is a Yes instance, then each copy of the original vertices can be given the same colors and the vertex s can be given any color from its list. Conversely, if the new instance can be list colored, then we can color the original instance as follows: Say the vertex s is colored with color 6, then no vertex in G_1 can get color 6, so we can color the original instance from the colors that corresponding vertices in G_1 are colored from. Similarly, if s were colored from color 7, 8, 9 or 10, we would have considered G_2, G_3, G_4 or G_5 respectively.

Therefore 5-REGULAR LIST COLORING is NP-hard on planar + $1v$ graph. \square

Theorem 3. $G_3 + kv$ COLORING is PARA-NP-hard.

Proof. We will show that 3-COLORING is NP-hard on $G_3 + 3v$ graph.

It is known that LIST COLORING problem is NP-complete for 3-regular planar bipartite graphs that have list assignment in which each list is one of $\{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}$ and all neighbours of each vertex with three colors in its list have two colors in their lists [3]. We will show a polynomial time reduction from this problem. Say we are given an instance $G = (V, E)$ of this problem. We add three vertices a, b, c . We will add edge (a, v) for all $v \in V$ which have $\{2, 3\}$ in the list. We add edge (b, v) for all $v \in V$ which have $\{1, 3\}$ in the list. Similarly we add edge (c, v) for all $v \in V$ which have $\{1, 2\}$ in the list. Since the

given instance was a 3-regular planar bipartite graph, the resultant graph will be a $G_3 + 3v$ graph. Now we claim that this new graph is 3-colorable if and only if the given instance is a Yes instance. If the original instance can be colored respecting the list assignment, then we can color vertices in V with the same colors and a with 1 because any neighbor of a will not have gotten color 1, since they had $\{2, 3\}$ in their lists. Similarly we can color b with 2 and c with color 3.

Conversely, if the new instance is 3-colorable, then we can color the original instance respecting the given list assignment as follows: Say vertex a gets color i and so b and c cannot get the same color. Now say the set of vertices that get color i is V_a , then we will color them with color 1 in original instance. Similarly, we can color vertices in V_b with color 2 and vertices in V_c with color 3.

Therefore, 3-COLORING is NP-hard on $G_3 + 3v$ graph which proves PARANP-hardness of COLORING on $G_3 + kv$ graphs. \square

Theorem 4. *Parameterized 2-REGULAR LIST COLORING is W[1]-hard.*

Proof. We will show a parameterized reduction from k -INDEPENDENT SET problem which is known to be W[1]-hard [4]. We are given an instance $G = (V, E)$. Now the problem is to find whether there exists an independent set of size at least k . With each vertex $v_i \in V$, we associate a list of colors $\{i, 0\}$. Now, we add a clique of size k , in which each vertex has list of colors $\{1, 2, 3, \dots, n\}$. Also, we make each vertex of this clique adjacent to all vertices of V . It can be observed that the new instance is an instance of original problem which we wanted to prove W[1]-hard because only k vertices have lists of size more than 2. Now we prove the claim that there exists an independent set of size at least k if and only if this graph can be colored respecting the list assignment.

Let $S \subseteq V$ be an independent set of size k in G , then the vertices in S can be colored from color 0 and therefore the vertices of clique can use the k colors corresponding to the second colors in the lists of those k vertices in S . The remaining vertices of $V \setminus S$ can be colored from the first colors in their respective lists. So the new instance can be colored.

Conversely, say the new instance can be colored respecting the list assignment. Since the vertices in clique must use k colors out of $\{1, 2, 3, \dots, n\}$, so at least k vertices in V must have been colored with the color 0 which is the desired independent set of size at least k . Hence, Parameterized 2-REGULAR LIST COLORING is W[1]-hard. \square

Now, we will show that if the size of lists of colors for these k vertices is also bounded by m and m is also a parameter, then the problem becomes FPT.

Theorem 5. *Parameterized 2-REGULAR LIST COLORING is FPT with parameter (k, m) , where m is the maximum size of list of colors for the vertices in S .*

Proof. Consider the vertices in S . There are at most m^k ways to color these vertices. We try all these possibilities in at most m^k iterations and in each iteration, for the remaining vertices, the problem becomes 2-LIST COLORING

which is polynomial time solvable as shown in previous section. So if m is a constant or parameter, the algorithm is FPT algorithm and therefore the problem is FPT. \square

3 $(n - k)$ -REGULAR LIST COLORING Is in XP

It is known that determining whether the vertices of a graph can be colored properly with $n - k$ colors is fixed-parameter tractable when parameterized by k [4]. The exact parameterized complexity class of $(n - k)$ -REGULAR LIST COLORING is still unknown. In this section we present, an XP algorithm for this problem.

Let $G = (V, E)$ be any graph and L be a $(n - k)$ -regular list assignment such that $L(v)$ denotes the list of colors corresponding to the vertex v . We begin by observing the following rule which can be applied to the graph repeatedly until no longer possible.

Reduction Rule 5. *Delete any vertex with degree less than $(n - k)$.*

Observation 6. *Reduction Rule 5 is safe and can be implemented in polynomial time.*

Proof. Consider a vertex v which has degree d_v less than $(n - k)$. Since each vertex has $(n - k)$ colors in its list, vertex v also has $(n - k)$ colors. Coloring requires that no adjacent vertices get the same color, so v cannot be colored with the colors that its neighbours are colored from. Since v has less than $(n - k)$ neighbours, so at least one color will always be left for v to be colored. Therefore it can be observed that $G \setminus v$ can be colored respecting the list assignment if and only if G can be colored too. Checking the degrees of vertices and deleting a vertex from the graph are well known to be implemented in polynomial time. \square

The following more general claim follows from the proof of correctness of the reduction rule above, as we can greedily color the vertices.

Lemma 1. *If the size of the list of every vertex is more than its degree then there exists a list coloring respecting the lists.*

We keep applying Reduction Rule 5 till it is no longer applicable and hence from now onwards we assume that every vertex has degree at least $(n - k)$.

We need the following simple observation which we use in our algorithm.

Lemma 2. *LIST COLORING is polynomial time solvable on a clique.*

Proof. We are given a clique C of size n and a list assignment L of colors. Since, in a clique, no two vertices can be assigned same color, therefore at most one vertex can be colored by any color. We create a bipartite graph, with all the vertices of C in one part, and a set of vertices, one corresponding to each distinct color in the lists of vertices in C , in the other part. We add an edge between a vertex v which corresponds to a vertex in C and a vertex x which corresponds

to a color x , if and only if the color x appears in the list of v . Now, it can be observed that the size of maximum matching gives us the maximum number of vertices in C that can be colored satisfying L . Finding maximum matching in a bipartite graph is polynomial time solvable [8]. Therefore LIST COLORING is polynomial time solvable on cliques. \square

Lemma 3. *If there exists a set of k colors using which it is possible to color at least $2k$ vertices of G respecting L , then there is a feasible coloring for G with respect to L .*

Proof. Let C_k be the set of k colors, from which it is possible to color s vertices, where $s \geq 2k$. Now, we assume that we will not use any color of C_k to color any other vertex. So we can delete these k colors from the lists of other vertices. Also we can delete those s vertices from the graph, because we will not use their colors in the remaining graph. Now the size of the list of colors of any vertex will decrease by at most k . Hence each vertex will still have at least $n - 2k$ colors in its list. Since we are left with at most $n - s$ vertices in the graph and $s \geq 2k$, each vertex can have degree at most $n - 2k - 1$ and has at least $n - 2k$ colors in its list. So from Lemma 1, there is a feasible coloring respecting L . \square

Lemma 4. *If there does not exist a set of k colors using which it is possible to color more than or equal to $2k$ vertices of G respecting L , then, in time $n^{O(k)}$, we can determine whether G has a feasible coloring with respect to L or not.*

Proof. Let R be the set of colors that are used by more than one vertex in the optimal coloring if it exists. Since there does not exist any set of k colors using which at least $2k$ vertices can be colored, $|R| < k$. Let C be the union of the lists of colors of all vertices in V . Clearly $|C| < n(n - k)$, and hence the number of all possible subsets of C of size k is less than n^{2k} . Therefore, in time $n^{O(k)}$, we can guess a set S of colors of size k such that $R \subseteq S$. Now, we will guess all possible sets of vertices that can be colored from the colors in S .

We denote the set of vertices that have color i in their lists by V_i . It can be observed that the size of any V_i can be at most n . Also, since the degree of each vertex is at least $(n - k)$, the total number of independent sets in any V_i is at most $n \cdot 2^k$ (as a vertex is non-adjacent to at most k other vertices). Therefore the total possible sets of vertices, that can be colored from the colors in S , is at most $(n \cdot 2^k)^k$. Since, we are sure that colors other than those in S don't color more than one vertex in the graph, the remaining vertices get a unique color. We can remove the colors of S from the lists of the remaining vertices and make them all adjacent to each other (as no pair of them will be colored with the same color anyway). Thus the problem now reduces to list coloring a clique which can be solved in polynomial time using Lemma 2. It can be observed that we have maintained the overall complexity upper bounded by $n^{O(k)}$. \square

Theorem 7. *In time $n^{O(k)}$ given $G = (V, E)$ and $(n - k)$ regular list L , algorithm 1 determines whether there exist a color assignment respecting L .*

Proof. Let the total number of colors be r . Now the idea is to build r graphs, one corresponding to each color, which will be induced from the vertices which have that corresponding color in their list. Now, the task is simply to find sets of independent vertices from the graphs corresponding to each color such that the union covers all vertices. Since we know that each vertex has degree at least $(n - k)$, there can be at most $n \cdot 2^k$ independent sets in each graph. So we iterate over all possibilities of choosing k colors and then the vertices that can be colored from them (by choosing independent sets from their corresponding graphs) in $n^{O(k)}$ time. We now have two possibilities.

1. Suppose, we can choose k colors such that we can color at least $2k$ vertices from those colors, then we can return Yes (from Lemma 3).
2. In the other case, there cannot be more than k colors which contribute to more than 1 vertex in the optimum solution. Now from Lemma 4, we keep guessing k colors and assume that those colors which contribute to more than 1 color in the optimum solution will be covered in some iteration. In that case, since the remaining colors contribute at most one vertex, finding the size of the maximum matching in the constructed graph provides the solution.

Observe that the run time of the Algorithm is dominated by line number 13 and 15 which are $n^{O(k)}$. □

4 Deleting k Vertices or Edges to Satisfy 2-REGULAR LIST COLORING

Theorem 8. *Vertex Parameterized 2-REGULAR LIST COLORING is FPT.*

Proof. We modify the proof of Theorem 1 for this. Let $G = (V, E)$ be a graph and L be a 2-regular list for G . We reduce our problem to All-but- k 2-SAT. All-but- k 2-SAT is to determine whether in given a 2-CNF formula, it is possible to remove at most k clauses so that the resulting 2-CNF formula is satisfiable. It is known that All-but- k 2-SAT is FPT [10, 13]. From G we create an equivalent 2-CNF formula \mathcal{F} as follows.

For each $u \in V$, we make two variables x_u and y_u . So, we have $2|V|$ variables. We will set x_u to true if the vertex u is assigned the first color in its list and y_u to true and x_u to false if vertex u is assigned the second color in its list. Now we construct a 2-CNF formula by adding two types of clauses.

- **Type 1:** For each vertex $u \in V$, we add the clause $(\overline{x_u} \vee \overline{y_u})$. Also for each edge (u, v) , we add clauses such that both vertices do not get the same color. For example, if the list for vertex u is $\{a, b\}$ and that for vertex v is $\{b, c\}$, then we add clause $(\overline{y_u} \vee \overline{x_v})$.

After adding the clauses corresponding to all vertices and edges, we repeat each of these Type 1 clauses $k + 1$ times.

- **Type 2:** We add clauses $(x_u \vee y_u)$ for all $u \in V$.

Algorithm 1. $(n - k)$ -REGULAR LIST COLORING

```

1: Input : Graph  $G = (V, E)$  and  $(n - k)$  regular list assignment  $L$  such that  $L(v)$ 
   denotes the list of colors corresponding to the vertex  $v$ .
2: Parameter :  $k$ .
3: Output : Yes if  $G$  can be colored respecting the given list assignment, No
   otherwise.
4: Recursively delete vertices of degree less than  $n - k$ .
5:  $R \leftarrow \cup_{v \in V} L(v)$ 
6:  $r \leftarrow |R|$ 
7: for  $i \leftarrow 1$  to  $r$  do
8:    $V_i \leftarrow \{v \in V \mid i \in L(v)\}$ 
9:    $E_i \leftarrow \{(u, v) \in E \mid u \in V_i \text{ and } v \in V_i\}$ 
10:   $C_i \leftarrow (V_i, E_i)$ 
11: end for
12:  $C \leftarrow \bigcup_r^{i=1} C_i$ 
13: for each  $S \subseteq C$  such that  $|S| = k$  do
14:   Let  $S$  be  $\{S_1, S_2, S_3, \dots, S_k\}$ 
15:   for each possible combination  $\{T_1, T_2, T_3, \dots, T_k\}$  where  $T_i \subseteq V(S_i) \forall 1 \leq i \leq k$ 
   and  $T_i$  is independent in  $S_i$  do
16:      $T \leftarrow \bigcup_k^{i=1} T_i$ 
17:     if  $|T| \geq 2k$  then
18:       return Yes;
19:     else
20:        $G' \leftarrow G \setminus T$ 
21:        $C_M \leftarrow$  New vertices corresponding to colors in  $G'$ 
22:        $V_M \leftarrow V(G') \cup C_M$ 
23:       construct  $G_M$  from vertices  $V_M$  with adding edges between the vertex  $v$ ,
   which corresponds to a vertex from  $G'$ , and a vertex corresponding to color  $c$  if and
   only if  $c \in L(v)$ 
24:        $M \leftarrow$  maximum matching in  $G_M$ 
25:       if  $|T| + |M| \geq n$  then
26:         return Yes;
27:     end for
28: end for
return No

```

The resultant formula is the equivalent 2-CNF formula. Now, we claim that $\exists S \subseteq V$, $|S| \leq k$, such that the graph induced from $V - S$ is list colorable if and only if at least $(m - k)$ clauses of the resultant expression can be satisfied, where m is the total number of clauses. We can note that $m \leq (k + 1)(2|E|) + |V| + |V|$. Suppose that there exists an assignment of variables that satisfies at least $m - k$ clauses i.e. at most k clauses are falsified by this assignment. Now, these k clauses can be Type 2 only, because each of Type 1 clauses appears $k + 1$ times. These k clauses will be corresponding to k vertices in V . Since the rest of all the clauses in SAT expression can be satisfied, so in our graph, we can delete those k vertices, to color the remaining $n - k$ vertices. Conversely suppose that there exists a $S \subseteq V$, $|S| \leq k$, such that induced graph from $V \setminus S$ can be colored. We can

assign *false* to both x_u and y_u for each such vertex $u \in S$. So at most k Type 2 clauses will be falsified. It can easily be seen that the rest all clauses can be satisfied.

Therefore the FPT reduction is proven. Since we know that All-but- k 2-SAT is FPT, so this problem is also FPT. \square

As mentioned before, this result generalizes the result that the odd cycle transversal problem is fixed-parameter tractable. There is a related edge version (called edge bipartization) which asks whether we can delete k edges to make a graph bipartite. An analogous list-coloring version is whether we can delete k edges from a given graph to satisfy a 2-regular list coloring. As in the case of edge bipartization [16], we give a parameter preserving reduction from the edge version to the vertex version to show

Theorem 9. *Given a 2-regular list coloring instance, determining whether there are k edges from the graph whose removal results in a list 2-coloring of the graph is fixed-parameter tractable.*

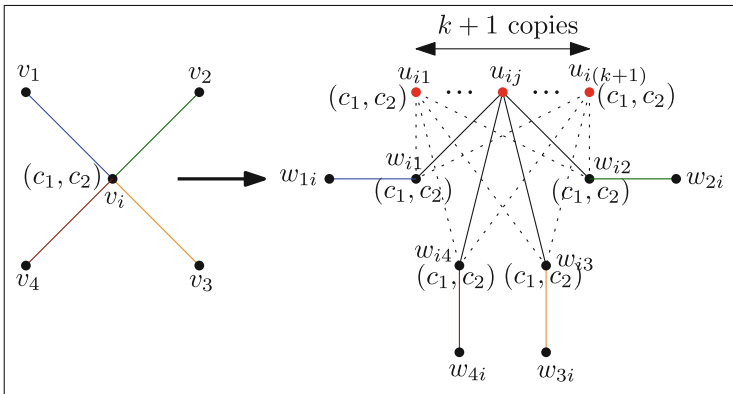


Fig. 1. Example reduction from G_1 to G_2

Proof. The proof goes along the lines of the parameter preserving reduction from edge bipartization to odd cycle transversal. Given an instance $G_1 = (V_1, E_1)$ of Edge Parameterized 2-REGULAR LIST COLORING problem, we will reduce to equivalent instance $G_2 = (V_2, E_2)$ of Vertex Parameterized 2-REGULAR LIST COLORING problem. From G_1 we construct G_2 as follows. V_2 consists of two types of vertices:

Type 1: $V_I = \{u_{ij} \mid v_i \in V_1, 1 \leq j \leq k + 1\}$

Type 2: $V_{II} = \{w_{ij}, w_{ji} \mid \forall (v_i, v_j) \in E_1\}$

For a sample reduction please refer to Fig. 1. Now we set $V_2 = V_I \cup V_{II}$. Note that we have named the vertices in V_I by a u and those in V_{II} by a w . We can call vertices in V_I by u vertices and those in V_{II} by w vertices. The edges in E_2 also consist of two sets:

Type 1: $E_I = \{ \{u_{ij}, w_{im}\} \mid v_i \in V_1, 1 \leq j \leq k + 1, (v_i, v_m) \in E_1 \}$

Type 2: $E_{II} = \{ \{w_{ab}, w_{ba}\} \mid e_l = \{v_a, v_b\} \in E_1 \}$

Now we set $E_2 = E_I \cup E_{II}$. We give the list assignment of colors in G_2 in the following manner. List of colors for each u_{ij} is same as the list of v_i in G_1 . Also list for w_{ij} is same as the list of v_i in G_1 . Construction of equivalent Vertex Parameterized 2-REGULAR LIST COLORING instance is complete now. Now we show that the edge parameterized instance G_1 can be colored if and only if vertex parameterized instance G_2 can be colored.

First assume there exists a set of k edges E_k in G_1 such that after deleting E_k from G_1 , remaining graph can be colored. Then, for each edge $e_l = (v_a, v_b) \in E_k$, we can delete any one of w_{ab} or w_{ba} . This step deletes exactly k vertices. Now, in remaining G_2 , say the list of colors for the vertex u_{ij} is $\{c_1, c_2\}$ which is same as of v_i in G_1 . If v_i is colored with c_1 , then we color u_{ij} with c_2 and if v_i is colored with c_2 , then we color u_{ij} with c_1 . For each vertex w_{ij} , we can color it with the same color as of v_i . It can be observed that there won't be any violations in the coloring.

Now for other way proof, assume there exists a set $S_k \subset V_2$ of at most k vertices whose deletion makes graph G_2 colorable. It can be observed that for each i , the set $\{u_{ij} \mid 1 \leq j \leq k + 1\}$ will be colored with same color, because otherwise there will be no color left for some w vertex. Also it should be noted that deleting any u vertex does not help in coloring, because in that case we need to delete $k + 1$ copies to get rid of the constraint caused by it. So, all vertices in S_k are w vertices. Now, we can delete the k edges in graph G_1 which correspond to the k vertices in S_k . It can then be observed that the vertex v_i in G_1 can be colored from the color which is not used by u_{ij} and this doesn't violate any coloring constraint. \square

5 Conclusions

We have filled up some gaps in the parameterizations of coloring and list-coloring and shown FPT results and hardness results for some natural parameterization. The main open problem we leave with is whether $(n - k)$ -REGULAR LIST COLORING is fixed-parameter tractable.

References

1. Aspvall, B., Plass, M.F., Tarjan, R.E.: A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Inf. Process. Lett.* **8**(3), 121–123 (1979)
2. Cai, L.: Parameterized complexity of vertex colouring. *Discrete Appl. Math.* **127**(3), 415–429 (2003)
3. Chudnovsky, M.: Coloring graphs with forbidden induced subgraphs. *Proc. ICM* **4**, 291–302 (2014)
4. Cygan, M., Fomin, F.V., Kowalik, L., Lokshantov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: *Parameterized Algorithms*. Springer, Cham (2015). <https://doi.org/10.1007/978-3-319-21275-3>

5. Dabrowski, K.K., Dross, F., Johnson, M., Paulusma, D.: Filling the complexity gaps for colouring planar and bounded degree graphs. In: Lipták, Z., Smyth, W.F. (eds.) IWOCA 2015. LNCS, vol. 9538, pp. 100–111. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-29516-9_9
6. Dailey, D.P.: Uniqueness of colorability and colorability of planar 4-regular graphs are NP-complete. *Discrete Math.* **30**(3), 289–293 (1980)
7. Garey, M.R., Johnson, D.S., Stockmeyer, L.J.: Some simplified NP-complete graph problems. *Theor. Comput. Sci.* **1**(3), 237–267 (1976)
8. Hopcroft, J.E., Karp, R.M.: An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Comput.* **2**(4), 225–231 (1973)
9. Khot, S., Raman, V.: Parameterized complexity of finding subgraphs with hereditary properties. *Theor. Comput. Sci.* **289**(2), 997–1008 (2002)
10. Lokshtanov, D., Narayanaswamy, N.S., Raman, V., Ramanujan, M.S., Saurabh, S.: Faster parameterized algorithms using linear programming. *ACM Trans. Algorithms* **11**(2), 15:1–15:31 (2014)
11. Lokshtanov, D., Saurabh, S., Sikdar, S.: Simpler parameterized algorithm for OCT. In: Fiala, J., Kratochvíl, J., Miller, M. (eds.) IWOCA 2009. LNCS, vol. 5874, pp. 380–384. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10217-2_37
12. Paulusma, D.: Open problems on graph coloring for special graph classes. In: Mayr, E.W. (ed.) WG 2015. LNCS, vol. 9224, pp. 16–30. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53174-7_2
13. Razgon, I., O’Sullivan, B.: Almost 2-SAT is fixed-parameter tractable. *J. Comput. Syst. Sci.* **75**(8), 435–450 (2009)
14. Reed, B.A., Smith, K., Vetta, A.: Finding odd cycle transversals. *Oper. Res. Lett.* **32**(4), 299–301 (2004)
15. Thomassen, C.: Every planar graph is 5-choosable. *J. Comb. Theory Ser. B* **62**(1), 180–181 (1994)
16. Wernicke, S.: On the algorithmic tractability of single nucleotide polymorphism (SNP) analysis and related problems. Ph.D. thesis. Universität Tübingen, Tübingen (2003)