Aspectos de complexidade parametrizada e problemas análogos em problemas de lista coloração de grafos e suas variações

Simone Gama[1], Rosiane de Freitas[1], Ueverton Souza[2]

[1]Intituto de Computação (IComp) – Universidade Federal do Amazonas (UFAM)
Manaus – AM – Brasil

[2]Intituto de Computação (IC) – Universidade Federal Fluminense (UFF)
Niterói – RJ – Brasil

{simone.gama,rosiane}@icomp.ufam.edu.br, ueverton@ic.uff.br

Abstract. *List coloring is a generalization of the classical vertex coloring problem in graphs. Such a problem has some variations, among them the (γ, μ)-coloring. In this work, we show the reducibility of the list-coloring problem, (γ, μ)-coloring and pre-coloring extended, to better provide an analysis of (γ, μ)-coloring under the parameterized complexity, where it is FTP when parametrized by vertex cover and color lists. We also present the correctness proof of a polynomial algorithm for (γ, μ)-coloring, from the literature, in complete bipartite graphs.*

Resumo. *Lista-coloração é uma generalização do problema clássico de coloração de vértices em grafos. Tal problema possui algumas variações, dentre elas a (γ, μ)-coloração. Neste trabalho, uma redutibilidade do problema da lista coloração para a (γ, μ)-coloração e pré-coloração estendida é apresentada, para melhor se prover uma análise da (γ, μ)-coloração sob o enfoque da complexidade parametrizada, onde a mesma é FTP quando parametrizada pela cobertura de vértices e listas de cores. É apresentada também a prova de corretude de um algoritmo polinomial, dado na literatura, para (γ, μ)-coloração em grafos bipartidos completos.*

1. Introdução

Seja $G = (V, E)$ um grafo simples não orientado e sem laços, onde V é o conjunto de vértices e E o conjunto de arestas. Um grafo bipartido é completo se para quaisquer dois vértices, $v_1 \in V_1$ e $v_2 \in V_2$, v_1v_2 é uma aresta em G. Uma k-coloração de G é uma atribuição de k cores ao seu conjunto de vértices tal que vértices adjacentes recebem cores diferentes. Seja G um grafo para o qual existe um conjunto associado $L(v)$ de lista de cores permitidas para cada v de G. A Lista-coloração de um grafo G é uma coloração própria de G tal que $c(v) \in L(v)$ para cada $v \in V(G)$ [Erdos et al. 1979]. A Lista-coloração também possui suas variações, dentre elas, a pré-coloração (onde dado um grafo com alguns vértices previamente coloridos, objetiva-se estender tal coloração para uma coloração própria de G) e a (γ, μ)-coloração. Neste trabalho, iremos analisar a (γ, μ)-coloração. Dado um grafo G e uma função $\gamma, \mu : V(G) \to \mathbb{N}$ tal que $\gamma(v) \leq \mu(v)$ para todo $v \in V(G)$, G é (γ, μ)-colorível se existe uma coloração própria f de G tal que $\gamma(v) \leq f(v) \leq \mu(v)$ para todo $v \in V(G)$ [Bonomo et al. 2009].

2. Redutibilidade em problema de Lista-coloração e suas variações

A noção de *problemas análogos* surgiu da possibilidade de que dois problemas aparentemente diferentes estejam bem relacionados. A definição é dada a seguir:

Definição 2.1 ([Fellows et al. 2015]). *Dois problemas de decisão Π e Π' em NP são ditos análogos se existe uma redução de tempo polinomial, f, g tal que: $\Pi \propto^f \Pi'$ e $\Pi' \propto^g \Pi$; Cada certificado $\mathcal{C}$ facilmente verificável para respostas-sim da questão "$I \in Y(\Pi)$?" implica em um certificado facilmente verificável $\mathcal{C}'$ para respostas-sim da questão "$f(I) \in Y(\Pi')$?"; Cada certificado $\mathcal{C}'$ facilmente verificável para respostas-sim da questão "$I' \in Y(\Pi')$?" implica em um certificado facilmente verificável $\mathcal{C}$ para respostas-sim da questão "$g(I') \in Y(\Pi)$?", onde $|C| = |C'|$.*

Lista-coloração, Pré-coloração e (γ, μ)-coloração são, de certa forma, semelhantes nos mais diversos aspectos. Para a Lista-coloração, a sua redutibilidade (aqui definida pela $\propto$) para outras colorações se dá da seguinte maneira: Seja G um grafo onde cada vértice $v \in V(G)$ é equipado com uma lista de cor $L(v)$. O grafo $\psi(G)$ é construído da seguinte forma:

1. Para todo vértice $v \in V(G)$ adicione v a $V(\psi(G))$; para toda aresta $e \in E(G)$, adicione e a $E(\psi(G))$;
2. Adiciona em $\psi(G)$, para cada $v \in V(G) \cap V(\psi(G))$, c vértices pendantes adjacentes à v $(w_v^1, w_v^2, \ldots, w_v^c)$, onde c é a maior cor em uma lista de G;
3. Para todo $v \in V(G)$ e todo $i \in L(v)$, remova w_v^i;
4. Para todo $w_v^i \in \psi(G)$ adicione à w_v^i rótulos $\gamma(w_v^i) = i$ e $\mu(w_v^i) = i$;
5. Para todo $v \in V(G) \cap V(\psi(G))$ adicione à v rótulos $\gamma(v) = 1$ e $\mu(v) = c$.

Seja $\mathcal{F}$ uma classe de grafos. Então: $\psi(\mathcal{F}) = \{\psi(G) \mid G = \psi(G')$ para algum $G' \in \mathcal{F}\}$. A classe $\mathcal{F}$ de grafos é fechada sob o operador ψ se $\psi(\mathcal{F}) \subseteq \mathcal{F}$.

Lema 2.1. *Os seguintes problemas são análogos quando restritos a classes* ***fechadas sob o operador*** *ψ: Lista-coloração, (γ, μ)-coloração e pré-coloração estendida.*

Prova. Observe que a pré-coloração estendida é um caso particular da (γ, μ)-coloração, que também é um caso particular da Lista-coloração. Assim, por restrição, segue-se que pré-coloração $\propto^{f_1}$ (γ, μ)-coloração $\propto^{f_2}$ Lista-coloração. Note que $\psi(G)$ é uma instância de (γ, μ)-coloração e pré-coloração estendida, porque $\psi(G)$ contém apenas listas de cores de tamanho 1 e listas de tamanho c. Por construção, G admite uma lista de cores se e somente se $\psi(G)$ admite (γ, μ)-coloração, assim pode-se ver que Lista-coloração $\propto^\psi$ (γ, μ)-coloração. Então Lista-coloração e (γ, μ)-coloração são análogos em $\mathcal{F}$. Como $\psi(G)$ contém apenas listas de tamanho 1 ou listas de tamanho c então lista coloração, (γ, μ)-coloração e pré-coloração estendida são também análogos em $\mathcal{F}$. □

Lista-coloração é $W[1]$-difícil quando parametrizado pela cobertura de vértices [Fellows et al. 2008], porém a Pré-coloração é FPT sob o mesmo parâmetro. A (γ, μ)-coloração, quando parametrizada pela cobertura de vértices e pelo tamanho de sua lista de cores, temos um algoritmo FPT, como afirmado no Teorema 2.1 a seguir:

Teorema 2.1. *(γ, μ)-coloração é FPT quando parametrizado pelo número cobertura de vértices e pelo tamanho de sua lista de cores.*

Prova. Em um grafo G equipado com listas de cores (γ, μ), considere l o tamanho da maior lista de cores em G, k o número da cobertura de vértices e S uma cobertura de vértices mínima de G ($|S| = k$). Seja $S' = V(G) \backslash S$. É possível analisar todas as possíveis maneiras de se colorir S, e em ordem sequencial verificar se cada elemento de S' possui uma cor disponível em tempo $O(l^k.n^{O(1)})$. □

3. A (γ, μ)-coloração em grafos bipartidos completos

Na literatura [Bonomo et al. 2009], é apresentado um algoritmo polinomial para (γ, μ)-coloração em um grafo bipartido completo (Teorema 3.1). Durante os estudos desse algoritmo, foram identificados algumas instâncias em que as etapas descritas nele não são suficientes.

Teorema 3.1 ([Bonomo et al. 2009])**.** *O problema da (γ, μ)-coloração em grafos bipartidos completos pode ser resolvido em tempo polinomial.*

Sendo assim, apresentamos alguns lemas que nos auxiliam a completar o algoritmo, bem como demonstrar a sua corretude. De forma resumida, o algoritmo proposto no Teorema 3.1 primeiramente estende as cores únicas dos vértices para os vértices de sua partição. As cores usadas são retiradas dos conjuntos de cores dos vértices adjacentes. Esse processo é repetido enquanto for possível. Após esse processo, para os vértices ainda não coloridos, o algoritmo determina que para os vértices de uma partição seja atribuído cores pares e para os vértices da outra partição seja atribuído cores ímpares, de acordo com suas listas de cores. Para exemplificar, considere a seguinte instância da Figura 1. Como pode ser observado em Figura 1(b), existem casos em que a última etapa do algoritmo não pode ser aplicada.

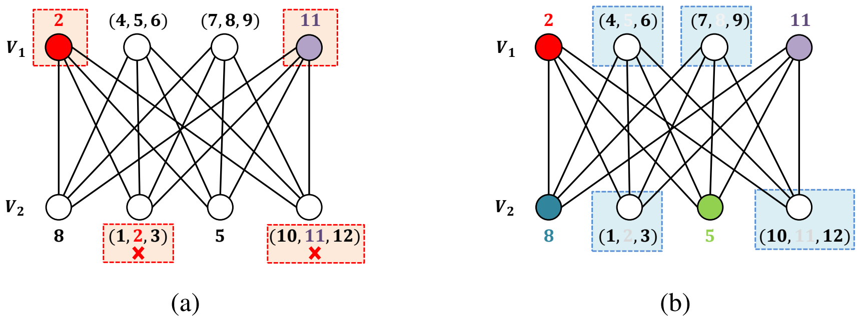

Figura 1. Em (a), na partição V_1 a cor 2 e 11 são retiradas das listas das cores de seus adjacentes. O mesmo acontece com a cor 8 e a cor 5 em V_2. Em (b), tanto em V_1 quanto em V_2 restaram listas com apenas cores pares e listas com apenas cores ímpares, não permitindo a escolha de cores exclusivamente pares (ou ímpares) para cada uma das partes.

Os lemas apresentados nesta seção servem para mostrar a pertinência das listas de cores das instâncias que não podem ser coloridas pelo algoritmo. Considere, portanto as seguintes definições: B o grafo bipartido completo **original**; B^R o grafo bipartido completo **remanescente**, após a execução do algoritmo. Diz-se que dois vértices v e w são de igual paridade quando os mesmos possuem listas de cores remanescentes pares (ou ímpares) ao mesmo tempo, bem como, vértices homogêneos (vértices com listas de cores remanescentes de uma paridade) e vértices heterogêneos (vértices com listas de cores remanescentes com paridades distintas).

Lema 3.1. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices homogêneos de igual paridade. Segue que $L(v) \cap L(w)$ não contém valores de outra paridade.*

Prova. Suponha por absurdo que exista valor $k \in L(v) \cap L(w)$ onde k possui paridade distinta. Como k possui **paridade diferente**, temos que $k \notin L^R(v) \cap L^R(w)$, ou seja, k **não** faz parte das listas de cores remanescentes. Pelo algoritmo, sabemos que algum vértice foi colorido com a cor k. Assume-se, sem perda de generalidade, que um vértice $x \in V_1$ recebeu a cor k. Logo, pelo algoritmo, v também teria recebido cor k, e portanto $v \notin V^R$. Absurdo! $\square$

Corolário 3.1. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices homogêneos de igual paridade. Segue que $|L^R(v) \cap L^R(w)| \leq 1$.*

Lema 3.2. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices homogêneos de igual paridade tal que $L^R(v) \cap L(w)^R = \{a\}$. Então $a = \gamma(i)$ e $a = \mu(j)$, onde $i \neq j$ e $i, j \in \{v, w\}$.*

Prova. Pelo Corolário 3.1, temos que: Se $\gamma(v) = \gamma(w)$, então $a + 1 \in L(v) \cap L(w)$. Absurdo; Se $\mu(v) = \mu(w)$, então $a - 1 \in L(v) \cap L(w)$. Absurdo. Sendo assim, sem perda de generalidade, assumimos que $\gamma(v) < \gamma(w)$. Se $a > \gamma(w)$, então $a > \gamma(w) > \gamma(v)$. Portanto $a - 1 \in L(v) \cap L(w)$ e isso é absurdo. Portanto, $a = \gamma(w)$. Se $a = \gamma(w) < \mu(v)$ então $a + 1 \in L(v) \cap L(w)$. Absurdo. Logo $a \geq \mu(v)$. Dado que $L(v) \cap L(w) \neq 0$, temos que $\gamma(w) = \mu(v) = a$. $\square$

Lema 3.3. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices de B^R tal que v é um vértice homogêneo e w é um vértice heterogêneo. Segue que $|L^R(v) \cap L^R(w)| \leq 1$.*

Prova. Sem perda de generalidade, assuma que v é par e suponha por absurdo que $|L^R(v) \cap L^R(w)| \geq 2$. Então existem dois inteiros $a, b \in L^R(v) \cap L^R(w)$, onde $a < b$. Logo $a + 1 \in L(v) \cap L(w)$. Dado que $a + 1$ é ímpar e v é par, pelo algoritmo, sabemos que algum vértice foi colorido com a cor $a + 1$. Se $u \in V_1$, então após colorir u, o algoritmo teria colorido v com a cor $a + 1$ e $v \notin B^R$. Analogamente, se $u \in V_2$ após colorir u o algoritmo teria colorido w com a cor $a + 1$, logo w não está em B^R. Absurdo! $\square$

Lema 3.4. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices de B^R tal que v é um vértice homogêneo e w é um vértice heterogêneo, onde $L^R(v) \cap L^R(w) = \{a\}$. Então $a = \gamma(i)$ e $a = \mu(j)$, onde $i \neq j$ e $i, j \in \{v, w\}$.*

Lema 3.5. *Seja $v \in V_1^R$ e $w \in V_2^R$ dois vértices de B^R tal que $L^R(v) \cap L^R(w) = 0$. Segue que $\mu(i) < \mu(j)$, onde $i \neq j$ e $i, j \in \{v, w\}$.*

Concluímos que existe uma ordem parcial entre as listas dos vértices remanescentes, que nos permite colorir B_r de forma que cada vértice homogêneo pode ser colorido com sua menor cor sempre.

Referências

Bonomo, F., Durán, G., and Marenco, J. (2009). Exploring the complexity boundary between coloring and list-coloring. *Annals of Operations Research*, 169(1):3.

Erdos, P., Rubin, L., and Taylor, H. (1979). Chosability in graphs. *Proceedings West Coast Conference on Combinatorics*, 26:125–157.

Fellows, M., Lokshtanov, D., Rosamond, F., Saurabh, S., and Misra, N. (2008). Graph layout problems parameterized by vertex cover. *Algorithms and Computation, Proceedings of ISAAC 2008*.

Fellows, M. R., dos Santos Souza, U., Protti, F., and da Silva, M. D. (2015). Tractability and hardness of flood-filling games on trees. *Theoretical Computer Science*, 576:102–116.