UNIVERSIDADE FEDERAL FLUMINENSE

MATHEUS SOUZA D'ANDREA ALVES

COLORAÇÃO DE GRAFOS(r, ℓ)

Niterói

2018

UNIVERSIDADE FEDERAL FLUMINENSE

MATHEUS SOUZA D'ANDREA ALVES

COLORAÇÃO DE GRAFOS(r, ℓ)

Trabalho de Conclusão de Curso apresentado à Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Bacharel em Ciência da Computação.

Orientador:
Dr. Uéverton dos Santos Souza

Niterói

2018

MATHEUS SOUZA D'ANDREA ALVES

Coloração de $Grafos(r, \ell)$

Trabalho de Conclusão de Curso apresentado à Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Bacharel em Ciência da Computação.

Aprovada em xx/xx/2018.

BANCA EXAMINADORA

Dr. Uéverton dos Santos Souza - Orientador, UFF
(Presidente)

Drª. Raquel Bravo - Avaliadora, UFF

Drª. Loana Tito Nogueira - Avaliadora, UFF

Niterói

2018

Resumo

Um problema clássico na literatura é o problema de coloração própria de um grafo, isto é, encontrar uma $q - coloração$ para um grafo G tal que todo vértice $v \in V(G)$ não possua nenhum vizinho da mesma cor e q seja mínimo. Esse problema é conhecido ser NP-Difícil para grafos gerais. O trabalho a seguir tem como proposta desvendar e catalogar a complexidade clássica e parametrizada de tal problema para a classe de Grafos(r, ℓ), i.e. grafos particionáveis em r conjuntos independentes e ℓ cliques; Identificando as características que tornam o problema difícil e a relação do problema de coloração com outros problemas, quando abordado pela perspectiva parametrizada.

Palavras-chave: Complexidade parametrizada. Grafos(r, ℓ). Partição de grafos. Coloração de Grafos

Abstract

A classical problem in the literature is the problem of proper coloring a graph, i.e. to find a $q - coloring$ for a graph G such that every vertex $v \in V(G)$ does not have any neighbor of the same color and q is the smallest possible number, a problem known to be NP-Hard for a general graphs. The following work attempts to uncover and catalog the parametrized complexity of such problem for the class of graphs(r, ℓ), i.e. partitionable graphs in r independent sets and ℓ cliques; Identifying the characteristics that make the problem hard and the relation of the stated problem to other problems when approached by the parameterized perspective.

Keywords: Parametrized Complexity. Graph(r, ℓ). Graph Partitioning. Graph Coloring.

Lista de Figuras

Lista de Tabelas

Conteúdo

Capítulo 1

Introdução

Uma das principais motivações do estudo de classes de grafos é o fato de que diversos problemas, que são difíceis para grafos em geral, tornam-se tratáveis quando restritos a classes especiais de grafos. Assim, busca-se delimitar a partir de que ponto um determinado problema pode ser resolvido de forma eficiente. Particularmente, o problema de particionamento em grafos tem despertado muito interesse devido às pesquisas de grafos perfeito e também pela procura de algoritmos eficientes para o reconhecimento de determinadas classes de grafos. O problema de partição de grafos pode ser descrito como tendo por objetivo particionar o conjunto dos vértices de um grafo em subconjuntos $V_1, V_2, \ldots, V_k$ onde $V_1 \cup V_2 \cup \ldots \cup V_k = V$ e $V_i \cap V_j = \emptyset$, $i \neq j$, $1 \leq i \leq k$ e $1 \leq j \leq k$, exigindo-se, porém, algumas propriedades sobre estes subconjuntos de vértices. Estas propriedades podem ser *internas*, como por exemplo exigir que os vértices de cada subconjunto V_i sejam dois-a-dois adjacentes (isto é, V_i seja uma clique) ou dois-a-dois não-adjacentes (isto é, V_i seja um conjunto independente), ou *externas*, onde as exigências são feitas sobre os pares de $V_i \times V_j$, podem ser adjacentes ou não-adjacentes entre si.

Em outra perspectiva o problema de coloração de vértices de um grafo também é de grande interesse, tendo suas aplicações em diversas áreas; *Pattern matching*, escalonamento em esportes e no trânsito, e resolução de problemas de *Sudoku* são alguns exemplos de problemas onde a coloração de grafos pode ser empregada[10]. O problema de coloração mínima dos vértices de um grafo chamado também de coloração própria de um grafo é descrito como a atribuição de cores em um grafo G de forma que seja possível atribuir a cada um dos vértices de G uma entre k cores de forma que, dado quaisquer dois vértices vizinhos em G eles não compartilhem uma mesma cor e k seja o menor número onde tal restrição é atingida.

O intuito deste trabalho é o de desvendar e classificar a dificuldade do problema de coloração em Grafos(r, ℓ). Ao nos aprofundar na investigação mostraremos ainda a proficuidade dos tamanhos das partições para a extração de um algoritmo FPT.

O capítulo sobre análise computacional clássica se propõe a demonstrar uma dicotomia *Polinomial / NP-Completo* para o problema abordando a quantidade de partições. Uma vez definida tal dicotomia o capitúlo de análise parametrizada busca através de parâmetros encontrar algoritmos tratáveis por parâmetro fixo para solucionar o problema em questão. Concluiremos o escrito sumarizando nossas descobertas e esclarecendo as repercursões de nosso trabalho e os novos desafios que nos pode trazer.

As seções mostradas a seguir se propõem a esclarecer as estruturas, problemas e ferramental utilizado durante o trabalho.

1.1 Estruturas básicas

Um Grafo G é uma estrutura que contém um conjunto de vértices $V(G) = \{v_1, v_2, ..., v_n\}$ e um conjunto de arestas $E(G) = \{(v, j)\}|\{v, j\} \subset V(G)$, dizemos que o vértice v pertence ao grafo G se $v \in V(G)$, e que existe uma aresta entre v e u se $(u, v) \in E(G)$, nesse trabalho usaremos apenas grafos não direcionados, dessa forma $(u, v) \equiv (v, u)$. Um Grafo dito Grafo(r, ℓ) ou abreviadamente $G(r, \ell)$ é qualquer grafo pertencente à classe dos grafos que podem ser particionados em r conjuntos independentes e ℓ cliques.

Um conjunto independente $r \subseteq G$ é uma partição de G tal que: $\{v, u\} \subset V(r) \implies \nexists(v, u) \in E(r)$. Já uma clique $\ell \subseteq G$ é uma partição de G tal que: $\{v, u\} \subset V(\ell) \implies \exists(v, u) \in E(\ell)$. Dizemos que um grafo G possui uma bipartição quando todos os seus vértices podem ser divididos entre dois conjuntos independentes disjuntos.

1.2 Problemas abordados

Definição 1. *Coloração mínima de Grafos.*

Entrada:	*um Grafo G e um inteiro k.*
Pergunta:	*É possível atribuir a cada vértice pertencente à G uma entre k cores de tal forma que dado quaisquer dois vértices adjacentes eles tenham cores distintas e k seja o mínimo de cores possível?*

Definição 2. *Lista coloração de Grafos.*

Entrada:	*Uma paleta de cores P e um Grafo G onde todo $v \in V(G)$ pode ser colorido com um subconjunto $P(v) \subset P$.*
Pergunta:	*É possível escolher uma cor dentro das de $P(v)$ para todo vértice v de forma que dado quaisquer dois vértices adjacentes eles tenham cores distintas?*

1.3 Complexidade computacional

Um algoritmo de tempo polinomial é definido como um algoritmo cuja sua função de complexidade de tempo é $\mathcal{O}(p(n))$, para alguma função polinomial p, onde n é usado para denotar o tamanho da entrada.

Um problema Π pertence à classe *P* se e somente se Π pode ser solucionado em tempo polinomial por algum algoritmo determinístico. Um problema Π pertence à classe *NP* se e somente se para um dado certificado há um algoritmo que verifica sua validade em tempo polinomial.

Dados dois problemas Π e Π' dizemos que $\Pi \propto \Pi'$ (Π se reduz à Π' em tempo polinomial) se existe um algoritmo capaz de construir uma instância J de Π' a partir de uma instância I de Π em tempo polinomial, tal que a partir de uma resposta para J uma resposta para I possa ser construída em tempo polinomial.

1.3.1 NP-Completude

Um problema Π' é dito $NP - Difícil$ se todo problema $\Pi \in NP$ se reduz à Π'. Se $\Pi' \in NP$ então Π' é $NP - Completo$.

1.4 Complexidade parametrizada

1.4.1 Tratabilidade Parametrizada

Definição 3. *Dado um problema Π e um conjunto de aspectos de Π chamado $S = \{s_1, s_2, s_3, ..., s_n\}$ denotamos por $\Pi(S)$ o problema Π parametrizado por S.*

Definição 4. *Dado um problema parametrizado $\Pi(S)$ dizemos que o mesmo é FPT (*Fixed parameter tractable *(Tratável por parâmetro fixo)) se existe um algoritmo capaz de resolver* Π *em* $\mathcal{O}(f(S) \times n^c)$ *onde* $f(S)$ *é uma função arbitrária e* c *uma função* $\mathcal{O}(1)$.

1.4.2 Intratabilidade Parametrizada

Esta seção irá sumarizar as definições de $W - hierarquia$ estabelecida por Downey et al.[3], para tanto observe as seguintes definições.

Definição 5. *Sejam* $\Pi(k)$ *e* $\Pi'(k')$ *onde* $k' \leq g(k)$. *Chamamos de FPT-redução de* $\Pi(k)$ *para* $\Pi'(k')$ *é uma transformação* R *quando:*

- $\forall x, x \in \Pi(k) \iff R(k) \in \Pi'(k')$
- R *é computável por um FPT-Algoritmo, com relação a* k

Definição 6. *Satisfabilidade Ponderada em circuitos de entrelaçamento* t *e profundidade* h $WCS(t,h)$.

Entrada:	*um circuito de decisão* C *com entrelaçamento* t *e profundidade* h.
Pergunta:	C *possui uma atribuição satisfazível?*

Usando as definições 5 e 6 definiremos então a pertinência de um problema à classe $W[t]$.

Definição 7. *Um problema parametrizado* $\Pi(k)$ *pertence a classe* $W[t]$ *se e somente se existe uma FPT-Redução de tal problema para* $WCS(t,h)$ *para algum* h *constante. Logo devido a transitividade de FPT-Redução, se existe uma FPT-Redução de qualquer problema* $\Pi'(k')$ *para* $\Pi(k)$ *então* $\Pi(k) \in W[t]$

Capítulo 2

Análise clássica para coloração em Grafos(r, ℓ)

O problema de coloração aplicado a Grafos(r, ℓ) é de fácil solução para algumas especificações, por exemplo um Grafo vazio, que é um Grafo(0,0) pode ser colorido com 0 cores, um Grafo disperso i.e um Grafo(1,0) é colorível com apenas uma cor, já que não existem arestas nesse grafo.

Já um Grafo completo, ou seja um Grafo(0,1), é colorível com K cores onde K é a quantidade de vértices nesse grafo completo, em um Grafo split que é um Grafo(1,1) essa regra se repete, já que cada vértice do conjunto independente pode ser colorível com alguma cor já presente na clique.

$r \backslash l$	0	1	2	3	4	...	n
0	P	P	?	?	?	...	?
1	P	P	?	?	?	...	?
2	P	?	?	?	?	...	?
3	?	?	?	?	?	...	?
4	?	?	?	?	?	...	?
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	?
n	?	?	?	?	?	...	?

Tabela 2.1: 1ª Dicotomia P/NPc parcial do problema de coloração em Grafos(r, ℓ)

E por fim, Grafos bipartidos são coloridos com 2 cores uma cor para cada conjunto independente.

Sabemos então que coloração é de solução polinomial para grafos completos, dispersos, split e para grafos bipartidos. Assim sendo, temos como ponto de partida para a exploração futura da complexidade de Grafos de cardinalidade superiores a Tabela 2.1, a ser preenchida de acordo com os seguintes resultados.

Teorema 1. *Coloração de Grafo(0,2) é Polinomial.*

Demonstração. Um Grafo(0,2) G, também chamado de grafo co-bipartido, é um grafo separável em 2 cliques em que todo vértice faz parte de alguma das cliques. A partir da literatura [1] sabemos que um grafo co-bipartido é perfeito, isso é, seu número cromático é igual ao de sua clique máxima, observe que conhecer a clique máxima é equivalente a conhecer o conjunto independente máximo do complemento de G, que por sua vez é resolvível encontrando a menor cobertura por vértices do mesmo, mostrado ser resolvível em tempo polinomial [9, 2]. Portanto ao encontrar a cobertura por vértices mínima para o complemento de G encontramos a clique máxima de G, e como G é um grafo perfeito, sabemos resolver o problema de coloração em tempo polinomial. □

Teorema 2. *Coloração de Grafo(3,0) é Polinomial.*

Demonstração. Tendo um Grafo G da classe (3,0) como entrada para o problema de coloração sabemos então que o grafo pode ser colorido com 3 cores, resta saber se 3 é o número mínimo de cores que pode ser usado, portanto devemos verificar se G é bipartido (colorível com duas cores) ou um grafo sem arestas (colorível com uma cor), como ambas verificações são polinomiais podemos afirmar que coloração de Grafo(3,0) é resolvível de forma polinomial. □

Teorema 3. *Coloração de Grafo(4,0) é NP-Completo.*

Demonstração. Sabemos que todo grafo planar é 4-colorível, e que alguns Grafos(4,0) são planares, portanto sabemos que para qualquer Grafo G $\in$ subconjunto de planares de Grafos(4,0), sua quantidade máxima de cores é 4, nos resta saber se 4 também é sua quantidade mínima, porém 3-coloração de planar é NP-Completo logo descobrir a coloração mínima de G é NP-Completo e consequentemente coloração de Grafos(4,0) é NP-Completo □

É importante notar aqui que, todo $Grafo(r, \ell)$ é simultaneamente um $Grafo(r, \ell+1)$ já que podemos formar uma nova clique trivial utilizando qualquer vértice, e um $Grafo(r+1, \ell)$ já que podemos formar um novo conjunto independente trivial a partir de qualquer vértice, portanto se o problema de coloração é NP-Completo para um $Grafo(r, \ell)$ então ele é NP-Completo para qualquer $Grafo(r+1, \ell)$ ou $Grafo(r, \ell+1)$.

Esses resultados nos levam à preencher a Dicotomia P/NPc da forma mostrada na Tabela 2.2

$r \backslash l$	0	1	2	3	4	...	n
0	P	P	P	?	?	...	?
1	P	P	?	?	?	...	?
2	P	?	?	?	?	...	?
3	P	?	?	?	?	...	?
4	NPc	NPc	NPc	NPc	NPc	...	NPc
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	NPc
n	NPc	NPc	NPc	NPc	NPc	...	NPc

Tabela 2.2: 2ª Dicotomia P/NPc parcial do problema de coloração em Grafos(r, ℓ)

Ainda nos falta mostrar a complexidade para alguns casos de fronteira, que necessitam de uma demonstração mais complexa.

Iremos demonstrar abaixo a complexidade para tais casos utilizando o seguinte teorema.

Teorema 4. *O problema de lista coloração para um grafo $G(r, \ell)$, se reduz ao problema de coloração própria de um grafo $(r, \ell + 1)$.*

Demonstração. Para a demonstração é preciso mostrar que

- Se um grafo G(r, ℓ) possui uma lista coloração própria então H_G é k-colorível para k do tamanho da paleta C (1)
- Se H_G é k-colorível então G possui uma lista coloração própria (2)

(1):

Usaremos a seguinte construção:
Considere G um grafo(r, ℓ) e que para cada vértice $v \in V(G)$ exista uma lista de cores S_v referente a esse vértice, cada lista contém pelo menos uma cor da paleta $C = \{c_1, c_2, c_3, ..., c_k\}$, sendo G uma instância sim para o problema de lista coloração, criemos uma clique K onde cada vértice $k \in V(K)$ representa uma cor presente em C. Seja $H_G = G \cup K$ para todo vértice $v \in G$ e todo vértice $u_i \in K$ adicione uma aresta (u_i, v) à H_G se e somente se v não possui a cor c_i em sua lista coloração em G

Podemos então generalizar da seguinte forma, dado um grafo(r, ℓ) G onde cada vértice de G possui uma lista de possíveis cores então o grafo H_G obtido pela construção anterior possui uma k-coloração.

Note que a clique K possui exatamente k vértices, consequentemente para colorirmos K precisaremos de k cores, sem perda de generalidade assumimos que u_1 será colorido com c_1, u_2 com c_2 e assim por diante.

Por construção uma aresta de u_i só existe para v_a em H_G se e somente se, v_a não possui c_i em sua lista de cores, portanto a coloração atribuída à K não conflita com a com a lista coloração de G, e portanto para todo vértice pertencente a G podemos lhe atribuir a mesma cor que lhe foi atribuída no problema de lista coloração, obtendo uma coloração própria mínima para H_G

(2):
Suponha que o grafo H_G possua uma k-coloração própria, onde k é o número de cores nas listas de G

Seja K a maior clique presente em H_G, por construção H_G é colorível com k cores onde k é a cardinalidade de K, observe que a remoção de K não afeta a coloração de $H_G - K$

Como H_G é k-colorível e a clique K possui k vértices todas as cores de tal k-coloração estão presentes em K. Sem perda de generalidade podemos assumir que as cores $c_1, c_2, ..., c_k$ estão atribuídas aos vértices $u_1, u_2, ..., u_k$ pertencentes à K

Por construção de H_G todo par (v, u_i) onde $v \in H_G - K$ e $u_i \in K$ é não adjacente se e somente se o vértice v não possui c_i em sua lista coloração no grafo G

Logo a k-coloração atribuídas aos vértices em $H_G - K$ formam uma coloração para G onde todo vértice em $V(G)$ possui uma cor de sua lista. Portanto G é uma instância sim de lista coloração □

Portanto utilizando o teorema 4, derivamos os seguintes corolários:

Corolário 1. *O problema de coloração é NP-Completo para Grafos*$(1, 2)$.

Demonstração. Segue da NP-Completude de lista coloração em Grafos$(1, 1)$, demonstrado por Jansen et al.[7]. □

Corolário 2. *O problema de coloração é NP-Completo em Grafos*$(2, 1)$.

Demonstração. Segue da NP-Completude de lista coloração em grafos bipartido demonstrado por Fellows et al.[4]. □

Teorema 5. *Lista coloração é NP-Completo para Grafos(0,2).*

Demonstração. Para essa demonstração nos basearemos em um resultado obtido por Jansen[6]. A demonstração se baseia em realizar uma redução do problema 3-SAT restrito para lista coloração de co-bipartido i.e. Grafo(0,2). Suponha o problema 3-SAT com as seguintes restrições:

- cada cláusula c_i contém dois ou três terminais.
- cada terminal ou sua negação aparece no máximo em 3 cláusulas

Construiremos agora uma instância de lista coloração da seguinte forma:
Para cada terminal j crie seis vértices: $a_j^{(1)}$, $a_j^{(2)}$, $a_j^{(3)}$; $b_j^{(1)}$, $b_j^{(2)}$, $3_j^{(3)}$. Atribuindo a cada um uma lista de cores da seguinte forma:
$a_j^{(k)} <= \{x_j^{(k)}, \overline{x_j}^{(k)}\}$; $b_j^{(k)} <= \{\overline{x_j}^{(k)}, x_j^{((k \pmod 3))+1)}\}$
Definimos como A o conjunto de todos os $a_j^{(k)}$ e B o conjunto de todos os $b_j^{(k)}$ e construímos uma clique com os vértices de A e B. Observe que só existem duas maneiras de se colorir este grafo:

- (1) $f(a_j^{(k)}) = x_j^{(k)} => b_j^{(k)} = \overline{x_j}^{(k)}$
- (2) $f(a_j^{(k)}) = \overline{x_j}^{(k)} => b_j^{(k)} = x_j^{((k \pmod 3))+1)}$

Agora, para cada cláusula definimos um vértice c_i e sua lista de cores da seguinte forma: para cada literal j ou sua negação $\overline{j}$ presente na cláusula adicionamos à lista de c_i o $x_j^{(k)}$ onde k é o índice de ocorrência do literal ou de sua negação.

Por exemplo, suponha o seguinte 3-SAT:

$(p \vee q \vee r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg r \vee s)$

suas cláusulas seriam traduzidas para

- c_1 com lista: $\{p^1, q^1, r^1\}$
- c_2 com lista: $\{\overline{p}^2, q^2, r^2\}$
- c_3 com lista: $\{\overline{p}^3, \overline{r}^3, s^1\}$

Seja C o conjunto contendo todos os c_i criamos uma clique com $C \cup A$. Nosso grafo tem portanto a seguinte configuração(considere x' como $\overline{x}$):

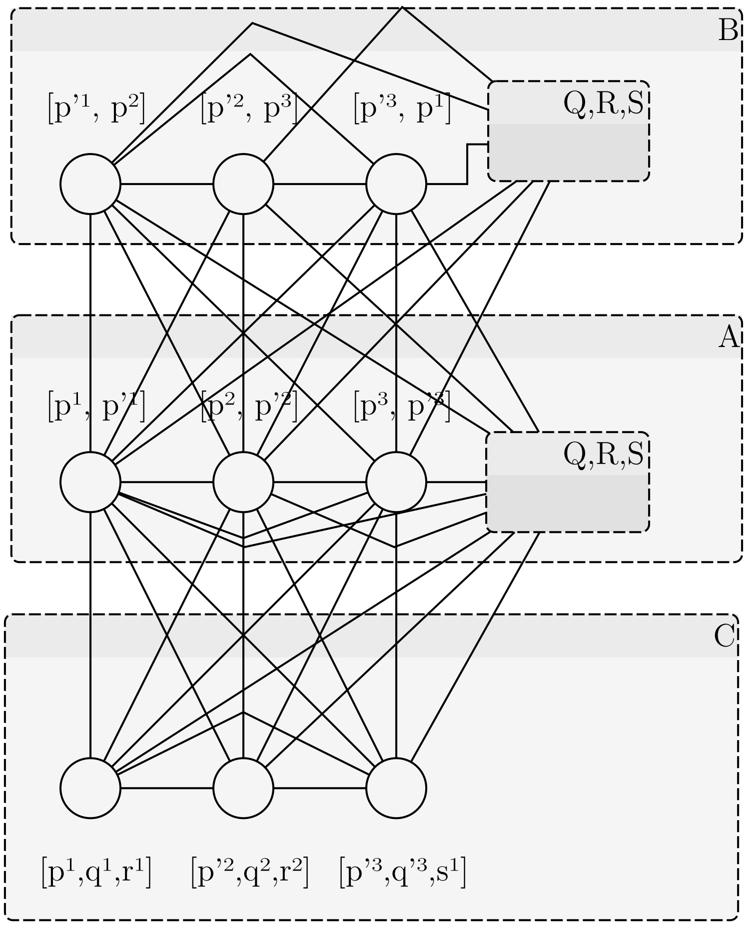

Figura 2.1: Grafo G: Transformação de 3-SAT em co-bipartido com foco na cláusula P

Suponha a cláusula p, se p é verdadeiro então $a_p^{(1)}, a_p^{(2)}, a_p^{(3)}$ será colorido com p'^1, p'^2, p'^3, permitindo que a cor p^x possa, e que a cor p'^x não possa ser escolhidas para colorir uma cláusula.

De tal forma, podemos facilmente notar que, expandindo a explicação anterior para os outros terminais uma resposta sim para o problema 3-SAT restrito nos leva a uma solução do problema de lista coloração em co-bipartido por exclusão das cores nas listas disponíveis. Para a volta a existência de uma lista coloração válida para o co-bipartido mostra uma solução para o 3-SAT restrito correspondente simplesmente descobrindo a representação em valor de terminal das cores escolhidas para as cláusulas.

□

Corolário 3. *O problema de coloração é NP-Completo em Grafos*$(0, 3)$.

Demonstração. Se da pela prova dos Teoremas 4 e 5. □

Portanto podemos agora completar nossa tabela com:

$r \backslash l$	0	1	2	3	4	...	n
0	P	P	P	NPc	NPc	...	NPc
1	P	P	NPc	NPc	NPc	...	NPc
2	P	NPc	NPc	NPc	NPc	...	NPc
3	P	NPc	NPc	NPc	NPc	...	NPc
4	NPc	NPc	NPc	NPc	NPc	...	NPc
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	NPc
n	NPc	NPc	NPc	NPc	NPc	...	NPc

Tabela 2.3: Dicotomia P/NPc do problema de coloração em Grafos(r, ℓ)

Capítulo 3

Análise parametrizada para coloração em Grafos(2,1)

Tendo mostrado a complexidade clássica nos é interessante agora que elucidemos quais características dos grafos(r, ℓ) se mostram propícias a abordagem parametrizada, a cardinalidade de suas partições se mostrou uma interessante candidata. Decidimos abordar a classe dos grafos(2,1), já que a mesma é uma das classes onde o problema é NP-Completo com o menor número possível de partições.

Um Grafo(2,1) é um grafo particionado em 2 conjuntos independentes e 1 clique, portanto ele nos entrega 3 naturais candidatos a parametrização, o tamanho da clique ℓ, o tamanho do menor conjunto independente r_1 e o tamanho do maior conjunto independente r_2.

3.1 Parametrização pelo tamanho do menor conjunto independente

Em [4] Fellows (et. al) mostrou que o problema de lista coloração é $W[1] - difícil$ parametrizado pela cobertura de vértices através da transformação do problema da clique multicolorida parametrizada pelo tamanho da clique para tal, nos aproveitaremos dessa transformação para mostrar a dificuldade do problema.

Definição 8. *Clique multicolorida.*

Entrada:	*Um Grafo G com uma k-coloração própria.*
Pergunta:	*Existe em G uma clique que contenha todas as k cores?*

Teorema 6. *Coloração em Grafos(2,1) é $W[1] - difícil$ quando parametrizado pelo tamanho do menor conjunto independente.*

Demonstração. Observe a seguinte construção.

O problema da clique multicolorida é conhecidamente $W[1] - difícil$[4].

Portanto suponha tal G proposto ao problema de clique multicolorida, temos como intenção montar um problema de lista coloração em um grafo G' a partir dele, para tanto seguimos os seguintes passos:

- Para cada cor i presente em G cria-se em G' um vértice v_i (os chamaremos de vértices-cor).
- Para cada vértice u em G colorido com a cor i, adicionamos à lista do vértice-cor v_i em G' uma cor c_u relacionada a esse vértice (as chamaremos de cores-vértice).
- Para cada aresta $e(x, y) \notin E(G)$ onde $x, y \in V(G)$ cria-se em G' um vértice z_e adjacente ao vértice-cor v_i onde i representa as cores de x e y, a lista coloração de z_e será formada por c_x e c_y.

É notável que já que formamos um grafo bipartido, a cobertura por vértices é limitado por k. Perceba agora que se G possui uma clique multicolorida podemos facilmente colorir G' da seguinte forma:

Ao vértice-cor v_i atribua a cor-vértice c_u onde u é o vértice colorido com a cor i em G. Dessa forma todos os vértices z_e possuem ainda uma cor disponível para sua coloração já que ele representa uma não-aresta em G.

Para a volta observe que uma lista coloração válida em G' implica em uma clique multicolorida em G, isso se dá pois dois vértices x, y coloridos com cores diferentes em G não aparecem em uma lista de algum z_e em G' se e somente se existe uma aresta $e(x, y) \in E(G)$, portanto as cores-vértices escolhidas para os vértices v_i são uma respectivamente uma clique formadas por tais i em G. Mostramos assim que lista coloração parametrizada por cobertura por vértices é $W[1] - difícil$.

Sabemos que coloração em Grafos(2,1) é equivalente a lista coloração em um grafo bipartido, portanto nossa tentativa de parametrizar a coloração de (2,1) pelo tamanho do menor conjunto independente é equivalente a parametrizar lista-coloração em bipartidos pelo tamanho da cobertura por vértices, mostrando assim que coloração em Grafos(2,1) parametrizada pelo tamanho do menor conjunto independente é $W[1] - difícil$.

□

3.2 Parametrização pelo tamanho do maior independente

Sabemos agora que a parametrização pelo menor independente não nos traz um algoritmo FPT, porém ao analisarmos o comportamento do problema quando parametrizado pelo maior independente vemos que a limitação do tamanho de r_2 também limita r_1; Tendo tal limitação a utilização de um método força bruta se mostra uma abordagem válida, como mostrado o seguinte teorema.

Teorema 7. *Coloração de Grafos(2,1) é FPT quando parametrizado pelo tamanho do maior conjunto independente.*

Demonstração. Para tal demonstração onde k é o tamanho de r_2, observe que são necessárias pelo menos t cores, onde t é a cardinalidade da clique para colorir tal grafo, novamente usaremos a estratégia de transformar coloração de (2,1) em lista coloração de bipartido.

Em uma lista coloração de bipartido, se um vértice possui uma lista com mais cores do que o tamanho de sua vizinhança, ele sempre terá disponível uma cor para sua coloração, podemos portanto remover esse vértice do grafo sem alterar sua coloração, ao chegarmos ao ponto onde todo vértice com tal configuração foi removido temos que t está limitado em função de k, portanto rodar um algoritmo de força bruta para encontrar a coloração se mostra FPT. □

3.3 Parametrização pelo tamanho da clique

Para a demonstração da complexidade parametrizada utilizando k igual ao tamanho da clique, nos voltamos novamente para transformação da clique em um Grafo(2,1) em listas coloração do restante bipartido, dessa forma nosso problema parametrizado original se torna um novo problema, lista coloração de bipartido parametrizado pelo tamanho da paleta de cores.

Mostraremos no entanto que essa parametrização não é proveitosa já que o problema se mostra equivalente à PreColoring Extension com limite de cores, mostrado ser NP-Completo para grafos bipartidos mesmo quando sua paleta é de tamanho três[8].

Definição 9. *PreColoring extension.*

Entrada:	*Um grafo G onde alguns vértices já possuem uma coloração definida com cores escolhidas dentre k possíveis cores.*
Pergunta:	*É possível estender a coloração já existente para todo o grafo sem que dois vértices adjacentes possuam a mesma cor?*

Teorema 8. *3-lista coloração em grafos bipartidos é NP-Completo.*

Demonstração. Suponha uma instância P do problema PreColoring Extension e G seu grafo de entrada, sabemos que G possui uma paleta C de cores de tamanho definido, e que existem $v \in V(G)$ que já estão coloridos com uma cor $c \in C$, podemos ver tal configuração como um grafo G' onde os vértices v possuem listas contendo apenas c, e os demais vértices possuem listas de tamanho $\#C$ contendo todas as cores, nos levando a um problema de lista coloração Q que tem como entrada G'.

Uma coloração possível para G implica em uma coloração possível para G', já que nos basta atribuir aos vértices em G' as mesmas cores atribuídas em G. De forma análoga, uma lista coloração possível em G' implica em uma coloração possível em G.

Assim sendo, como PreColoring está em NP e a validação de coloração é trivial, podemos dizer que $Q \in NPc$ □

Apesar do tamanho da paleta não ter se mostrado uma escolha adequada, ele levanta novos parâmetros que são interessantes para o problema de lista coloração em bipartidos, sabemos que lista coloração é polinomial se todos os vértices tem lista tamanho ou três (3-coloração), ou dois (2-coloração) ou um, e NP-Completo se tem listas de tamanho 1 à 3 [8], isso levanta duas formas de se abordar o problema

- O que acontece quando o número de vértices com listas de tamanho 1 e 2 varia?
- O que acontece quando o número de vértices com listas de tamanho 3 varia?

Mostraremos nas próximas seções como se dão tais comportamentos e como eles se relacionam a coloração de Grafos(r, ℓ)

3.4 Parametrizado pela vizinhança da clique

Nos focaremos nessa seção em grafos(2,1) cuja a clique tenha tamanho 3, já mostrada ser o menor necessário para que o problema de PreColoring Extension seja NP-Completo mostrado no teorema 8 e em [8, 11]. Portanto um vértice que é vizinho da clique tem necessariamente uma lista contendo uma ou duas cores, já que um vértice não pertencente a clique que tenha lista de tamanho zero deveria fazer parte da clique, em contrapartida um vértice com lista tamanho 3 é um vértice não vizinho a clique.

Mostraremos que mesmo quando parametrizado pela quantidade de vértices com listas de tamanho um, dois, ou um e dois o problema é para-Np-completo. Para tanto é necessário encontrar uma instância do problema já parametrizado cuja solução permanece igualmente difícil.

Portanto essa seção será dividida em três casos, um contendo vértices de listas tamanho um, outro contendo vértices com listas de tamanhos dois, e finalmente contendo listas de tamanho um e dois.

3.4.1 Vértices com listas de tamanho um

É importante ressaltar que os seguintes teoremas estabelecem a base para a resolução do problema envolvendo os vizinhos da clique.

Teorema 9. *Seis vértices com lista de tamanho um são suficientes para que lista coloração em bipartido seja NP-completo.*

Demonstração. Sabemos que em nosso problema temos dois conjuntos independentes, r_1 e r_2, também é verdade que exceto pelos citados seis vértices todos os outros tem listas de tamanho três, os vértices de r_1 podem estar ligados arbitrariamente aos de r_2. Observe a disposição de tais vértices na figura 3.1, chamaremos tal esquema de $\mathcal{N}$.

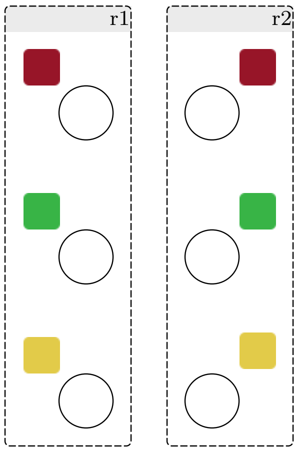

Figura 3.1: Esquema de vizinhança formado por 6 vértices com distintas listas tamanho 1.

Agora, através do gadget da figura 3.2 mostraremos como a presença de vértices com distintas listas de tamanho um influencia na coloração de sua vizinhança comum.

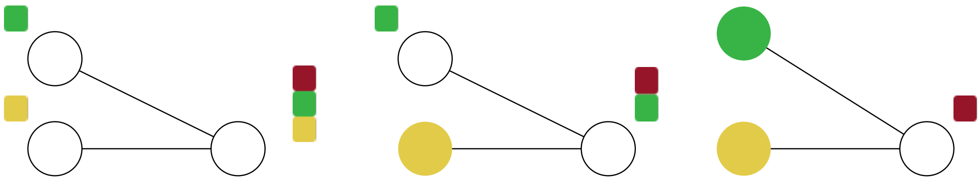

Figura 3.2: Gadget com vértices de lista um reproduzindo vértice de lista um em vértice de lista três

Com o gadget apresentado é possível conduzir à escolha de uma cor para um vértice com lista tamanho três dado que a única coloração possível para o gadget é a coloração aonde a cor desejada ocorre.

Utilizando esse gadget é possível transformar uma instância de PreColoring Extension em bipartidos em uma de lista coloração em bipartido, basta para tanto pegarmos os vértices do Grafo G que estão pré-coloridos e criar um vértice equivalente em um grafo G' com listas tamanho três, os ligando aos vértices de $\mathcal{N}$ e mantendo a bipartição de forma a excitar a cor que este vértice possuia em G, os demais vértices são mapeados para vértices com listas tamanho três, mantendo sua vizinhança equivalente a em G.

Assim se a lista coloração for possível em G' basta escolher as mesmas cores para colorir G, o mesmo se aplica a contraparte da demonstração. Dessa forma mostramos que com seis vértices com distintas listas tamanho um, encontrar uma lista coloração para tal grafo é Np-Completo.

□

Interessantemente o problema é de trivial solução quando o número de vértices com listas tamanho um é zero, já que se torna o problema de 3-coloração em bipartidos, mas NP-Completo com 6 vértices, queremos portanto encontrar o menor número de vértices no qual o problema é NP-Completo, e consequentemente para-NP-Completo para nossa parametrização.

Teorema 10. *Lista coloração em bipartido é de solução trivial quando há apenas um vértice de lista tamanho um.*

Demonstração. Sabemos que além do vértice citado todos os outros vértices têm listas de tamanho três dessa forma basta que o conjunto independente no qual tal vértice está inserido seja colorido com a única cor escolhida para o vértice e o conjunto independente sobrante pode ser colorido com qualquer cor. □

Teorema 11. *Lista coloração em bipartido é de solução linear quando existem dois vértices de lista tamanho um.*

Demonstração. Para essa demonstração é necessária a observação em que existem duas possíveis configurações para essa instância:

- Ambos os vértices pertencem ao mesmo conjunto independente.
- Os vértices pertencem a conjuntos distintos.

No primeiro caso a estratégia usada no teorema 10 pode ser adaptada para a solução. Para tanto basta colorir tais vértices com suas cores disponíveis e o conjunto independente ao qual pertencem com a cor de algum deles, e o conjunto sobressalente com a cor restante.

No segundo caso, a coloração também é simples. Se tais vértices tem cores distintas basta colorir seus respectivos conjuntos com a mesma cor. Se não, como temos três cores podemos colorir os vértices com a cor 1, um conjunto com a cor 2 e os demais vértices com a cor 3. □

Teorema 12. *Três vértices com lista de tamanho um são suficientes e necessários para que lista coloração em bipartido seja NP-completo.*

Demonstração. Mostraremos aqui como que três vértices são suficientes para que o problema seja NP-Completo, esse resultado se dá pois é possível reproduzir a estrutura do teorema 9 utilizando os ditos 3 vértices, para tanto usaremos o gadget mostrado na figura 3.2.

Usando tal gadget, dois vértices v com lista um em um vértice u com lista três são capazes de reproduzir um vértice de lista um através da retirada da lista de u as únicas possíveis cores para v, sendo assim tendo três vértices de distintas listas tamanho um, é possível obter seis vértices de lista um (com três listas distintas de cada lado) e reproduzir a estratégia mostrada no teorema 9, que nos mostra a NP-Completude desse problema.

□

Dado os resultados apresentados nessa seção mostramos portanto que o número de vértices vizinhos a dois dos três vértices pertencentes a clique não é um parâmetro viável para uma solução FPT.

3.4.2 Vértices com listas de tamanho dois

Mostraremos nessa seção que vértices com listas tamanho dois não são suficientes para que um algoritmo FPT seja extraído.

Como já visto o problema é de trivial solução quando todos os vértices tem listas de tamanho três, portanto precisamos ainda encontrar qual número de vértices de tamanho dois onde o problema se mantém NP-Completo.

Teorema 13. *Seis vértices com listas tamanho 2 são necessários e suficientes para que lista-coloração em bipartido seja NP-Completo.*

Demonstração. Para mostrarmos que qualquer número de vértices abaixo de 6 é insuficiente, mostraremos que a menos que existam pelo menos 3 vértices com distintas listas tamanho dois em cada conjunto independente, a coloração é simples de ser feita.

Se um conjunto independente contém apenas dois vértices com listas tamanho dois, podemos afirmar que todos os vértices nesse conjunto compartilham uma cor em suas listas, podendo colorir tal conjunto com essa cor, todos os outros vértices ainda têm pelo menos uma cor disponível para sua coloração podendo ser colorido com ela. É fácil notar que o argumento se estende para o caso onde alguma lista se repete dentro de qualquer r, já que dessa forma existe uma cor em comum entre todas as listas de tal r.

Para completar nossa demonstração basta portanto, encontrar uma configuração onde o problema de lista coloração permanece NP-Completo.

Para tanto nos é interessante agora a vizinhança entre os vértices com lista dois, iremos isolar as instâncias em alguns casos.

- Vizinhança de algum vértice é tamanho um:
  Nesse caso podemos notar que independentemente do vértice $v \in r_1$ e seu vizinho $u \in r_2$, eles sempre compartilharão uma cor, colorimos o vértice u com tal cor, além disso como conhecemos a vizinhança de v sabemos que nenhum outro vértice é vizinho deste, podemos então colorir os restantes vértices de r_2 com a cor de v, dessa forma uma das três cores ainda resta e podemos a usar para colorir o restante do r_1, independente das ligações entre os demais vértices de r_1 e r_2.

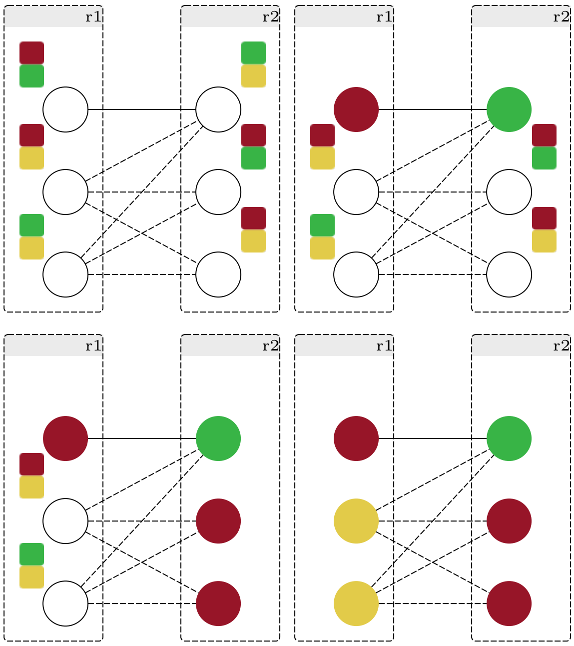

Figura 3.3: Demonstração de coloração para vizinhança de tamanho um.

- Um vértice têm vizinhança tamanho dois, e compartilha uma cor com ambos os vizinhos:
  Aqui podemos executar o seguinte algoritmo, sem perda de generalidade pinte os vértices vizinhos à $v \in r1$ com a cor compartilhada, novamente por conhecermos a vizinhança do vértice v podemos pintar ele e os demais vértices de r_2 com a cor restante, dessa forma ainda nos resta uma cor para colorir os demais vértices de r_1.

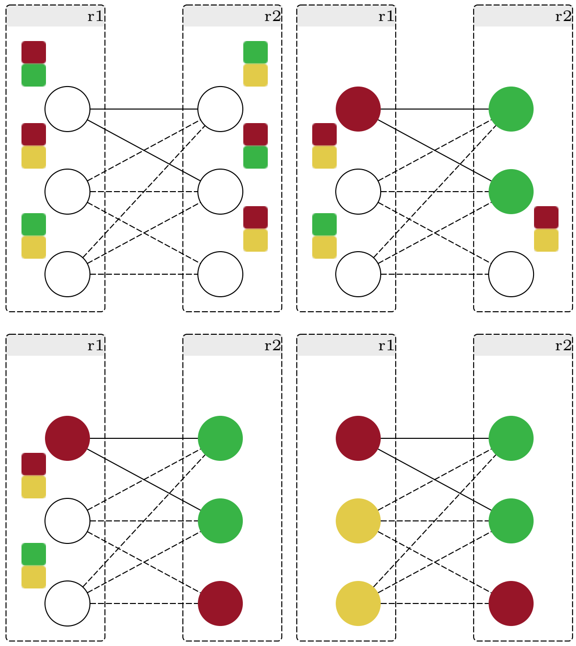

Figura 3.4: Demonstração de coloração para vizinhança de tamanho dois com cores compartilhadas.

- Todo vértice tem vizinhança de tamanho dois e nenhuma vizinhança possui uma cor em comum:

  Observe que esse caso tem suas restrições derivadas dos casos acima, aqui mostraremos como PreColoring extension se reduz a esse caso, mostrando finalmente sua Np-Completude.

  As restrições impostas a esse caso nos levam a uma única possível estrutura Γ onde suas duas possíveis colorações são intercambiáveis, vide figura 3.5. Portanto se mostra verdade que podemos excitar uma cor qualquer em outro vértice de lista tamanho três se o ligarmos a dois dos três vértices presentes no independente oposto sem ferir a bipartição do grafo.

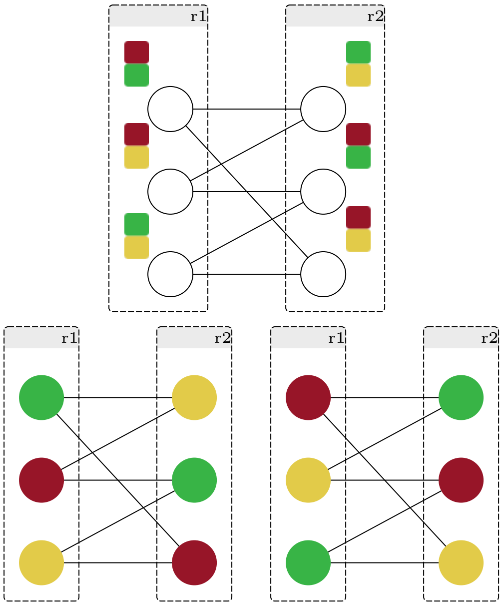

Figura 3.5: Estrutura Γ e suas possíveis colorações.

Assim sendo para reduzir um problema de PreColoring extension em bipartido Π ao nosso problema Π' basta que para todo vértice pré-colorido $v_{c_j} \in V(\Pi)|c_j \in C$ cria-se um paralelo $u \in V(\Pi')$ com lista de tamanho três e liga-o aos vértices de Γ a fim de excitar sua cor da seguinte forma:

Escolha sem perda de generalidade um vértice $v \in r1$ pré-colorido com a cor c_j, observe que Γ possui dois conjuntos de ligações capazes de excitar c_j em u utilizando o gadget da Figura 3.2, escolha um desses conjuntos. Portanto $\forall v_{c_j}|j \neq i$ basta realizar a ligação com Γ respeitando o conjunto de ligações já escolhido.

Os demais vértices de $V(\Pi)$ são construídos mantendo suas respectivas vizinhanças em Π' e com lista de tamanho três, é de pouca dificuldade compreender como uma resposta para Π' implica em uma resposta para Π, pois se Π' é lista colorível, então Π é colorível respeitando a pré-coloração, já que a coloração de Γ é indiferente para os vértices não vizinhos à ela e sempre respeita a coloração de sua vizinhança, para a demonstração da volta basta escolher as mesmas cores escolhidas em Π para seus respectivos em Π' que implicará em uma coloração para Γ inofensiva ao resultado.

$\square$

É importante notar que para o acontecimento de haver vértices com listas tamanho um e dois, basta notar que podemos remover aqueles que contém listas de tamanho um, que propagará a remoção de sua cor às listas de vértices vizinhos, e ao realizar isso iterativamente, acabaremos com um caso em que todos os vértices terão ou listas de tamanho dois ou três, caindo em algum dos casos supracitados.

3.5 Parametrizado pela quantidade de vértices não vizinhos a clique

Como visto na seção anterior, os vértices que não são vizinhos a clique, quando transformados em vértices do problema de lista coloração se transformam em vértices com listas tamanho três, portanto nosso desejo é resolver lista coloração em bipartidos com listas de tamanho um a três parametrizado pela quantidade de vértices com lista de tamanho três, a solução deriva do seguinte teorema.

Teorema 14. *Lista coloração em bipartidos com listas de tamanho um a três é FPT quando parametrizado pela quantidade de vértices com lista de tamanho três.*

Demonstração. Dado que temos k vértices com 3 escolhas cada é possível montar um algoritmo de busca em árvore de altura limitada de tamanho 3^k, e então executar o algoritmo linear proposto em [11] obtendo um algoritmo $\mathcal{O}(3^k n)$

□

Capítulo 4

Conclusão

Apresentamos aqui o desenvolvimento e especificações da dificuldade do problema de coloração em grafos(r, ℓ). ao nos aprofundar na investigação percebemos ainda a inesperada inaptidão da maior parte dos parâmetros relacionados ao tamanho das partições para a extração de um algoritmo FPT. Nas seções seguintes sumarizaremos nossos resultados e mostraremos consequências e questionamentos relacionados.

4.1 Resultados e consequências

Obtivemos no capítulo 2 uma dicotomia P/NPc para o problema de coloração em grafos(r, ℓ), é conhecido que o problema de coloração em um grafo G pode ser visto como um problema de clique cover em seu complemento G'[5], dessa forma podemos estender a dicotomia P/NPc da tabela 2.3 para o problema de clique cover em grafos(r, ℓ) simplesmente trocando as linhas pelas colunas da tabela, dessa forma obtemos a seguinte dicotomia P/NPc:

$r \backslash l$	0	1	2	3	4	$\dots$	n
0	P	P	P	P	NPc	$\dots$	NPc
1	P	P	NPc	NPc	NPc	$\dots$	NPc
2	P	NPc	NPc	NPc	NPc	$\dots$	NPc
3	NPc	NPc	NPc	NPc	NPc	$\dots$	NPc
4	NPc	NPc	NPc	NPc	NPc	$\dots$	NPc
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	NPc
n	NPc	NPc	NPc	NPc	NPc	$\dots$	NPc

Tabela 4.1: Dicotomia P/NPc do problema de clique cover em Grafos(r, ℓ)

Já no capítulo 3 exploramos com profundidade o comportamento do problema em Grafos$(2,1)$ Obtendo dessa forma 2 algoritmos FPT, uma demonstração de $W[1]-dificuldade$ e algumas de para-NP-completude, tendo colateralmente mostrado resultados clássicos e parametrizados para o problema de lista coloração em bipartidos.

4.2 Trabalhos futuros

Alguns questionamentos levantados durante a produção deste trabalho que tomaram forma de possíveis trabalhos futuros foram:

- Existe uma relação entre as parametrizações da coloração de Grafos$(2,1)$ e clique cover de Grafos$(1,2)$?
- Quais são as características que afetam parametrizações em Grafos(r,ℓ) diferentes de Grafos$(2,1)$?
- Existe algum parâmetro que seja possível extrair um algoritmo FPT para qualquer r ou ℓ?

Referências

[1] Bollobás, B. *Modern Graph Theory.* 1998, p. 165–166.

[2] Chartrand, G., Zhang, P. *A First Course in Graph Theory.* 2012, p. 189–190.

[3] Downey, R. G., Fellows, M. R., Regan, K. W. Parameterized circuit complexity and the W hierarchy.

[4] Fellows, M., Fomin, F. V., Lokshtanov, D., Rosamond, F., Saurabh, S., Szeider, S., Thomassen, C. On the Complexity of Some Colorful Problems Parameterized by Treewidth.

[5] Garey, M. R., Johnson, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* 1979.

[6] Jansen, K. Complexity Results for the Optimum Cost Chromatic Partition Problem.

[7] Jansen, K., Scheffler, P. Generalized coloring for tree-like graphs.

[8] Kratochvil, J. Precoloring Extension With Fixed Color Bound.

[9] König, D. *"Gráfok és mátrixok". Matematikai és Fizikai Lapok.* 1931, p. 116–119.

[10] Lewis, R. *A Guide to Graph Colouring: Algorithms and Applications.* 2015.

[11] M.Hujter, ZS.Tuza. PreColoring Extension. II. Graph classes related to bipartide graphs.