



UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA

**GRAFOS ROTULADOS, GRAFOS GRACIOSOS E PROBLEMAS DE COLORAÇÃO
ESPECIAIS**

VICTÓRIA PATRÍCIA SILVA AIRES

Julho de 2015

Manaus - AM

VICTÓRIA PATRÍCIA SILVA AIRES

GRAFOS ROTULADOS, GRAFOS GRACIOSOS E PROBLEMAS DE COLORAÇÃO
ESPECIAIS

Relatório final apresentado ao Programa Institucional de Bolsas de Iniciação Científica (PIBIC), como parte dos requisitos necessários para o projeto de pesquisa.

Orientador(a): Rosiane de Freitas Rodrigues, D.Sc.

Julho de 2015

Manaus - AM

Sumário

Lista de Figuras	c
1 Introdução	1
1.1 Contexto	1
1.2 Motivação e objetivos	1
1.3 Metodologia de desenvolvimento do trabalho	2
1.4 Cronograma	3
1.5 Organização do restante deste relatório	3
2 Referencial teórico	5
2.1 Teoria dos Grafos	5
2.1.1 Conceitos básicos	5
2.1.2 Algumas propriedades	6
2.1.3 Classes especiais de grafos	9
2.2 Complexidade Computacional	13
2.2.1 Um pouco sobre complexidade computacional	13
2.2.2 O problema P vs NP	13
3 Grafos rotulados	15
3.1 Grafos rotulados	15
3.1.1 Origem	15
3.1.2 Algumas classes de grafos rotulados	16
3.2 Coloração em grafos	18
3.3 Grafos graciosos	18
3.3.1 Definição	18
3.3.2 Grafos graciosos e alguns teoremas	19

<i>SUMÁRIO</i>	b
4 Problemas de coloração em grafos	20
4.1 Coloração clássica de vértices	20
4.1.1 Colorações mínimas	20
4.1.2 Coloração de mapas	21
4.2 T-coloring	21
4.2.1 Problema de alocação de canais	21
4.2.2 Definição de T-coloring	22
4.2.3 Propriedades do T-Coloring	23
4.3 Distance coloring	24
5 Resultados	27
5.1 Detalhamento de teoremas	27
5.1.1 T-Coloring	27
5.1.2 Distance coloring	29
5.2 Algoritmos	32
5.2.1 Ordem arbitrária de vértices	33
5.2.2 Ordem decrescente do grau dos vértices	34
5.2.3 Ordem pelo grau de saturação	36
5.2.4 Testes computacionais	38
6 Considerações finais	41
Referências	42

Lista de Figuras

2.1	Um exemplo de grafo $G = (V, E)$ com 4 vértices.	6
2.2	Um grafo com laços: o vértice C está ligado consigo mesmo, constituindo um laço.	6
2.3	Um multigrafo: os vértices A e B compartilham 2 arestas (paralelas).	7
2.4	Um grafo simples com 4 vértices e 4 arestas.	7
2.5	Um grafo desconexo composto de 2 componentes conexas: um grafo 4-ciclo e um grafo 3-ciclo.	8
2.6	Um grafo conexo com 7 vértices e 9 arestas.	8
2.7	Um grafo direcionado: toda aresta possui uma direção, ou seja, sai de um vértice e chega em outro.	9
2.8	Um grafo completo K_4 (contém 4 vértices e todas as possíveis arestas).	9
2.9	Um grafo nulo (não contém arestas).	10
2.10	Um grafo regular de grau 3, onde todos os vértices possuem grau 3.	10
2.11	Um grafo ciclo de tamanho 3, C_3	10
2.12	Um grafo caminho P_2 , de tamanho 2.	11
2.13	Uma árvore (grupo conexo e acíclico).	11
2.14	Um grafo bipartido $B_{2,4}$	12
2.15	Um grafo bipartido completo $K_{2,4}$	12
2.16	Um exemplo de grafo rotulado, onde as arestas possuem rótulos (números naturais) associados.	12
3.1	Um exemplo de grafo harmonioso. Aqui, $k = 6$, portanto o rótulo de cada aresta xy é igual a $(x + y) \pmod k$. Deste modo, a aresta (3,5), por exemplo, recebeu o valor 2 como rótulo, pois $(3 + 5) \pmod 6 = 8 \pmod 6 = 2$. De modo análogo, todas as arestas receberam um rótulo.	16

3.2	Um grafo mágico, onde a soma dos rótulos dos vértices com o rótulo da aresta que os relacionam é constante, independente do vértice escolhido.	16
3.3	Quadrado mágico 3x3 modelado em teoria dos grafos, resultando em um grafo mágico. Em a), há o grid ainda não preenchido. Em b), três soluções que atendem às restrições do quadrado mágico e seus respectivos modelos como grafos mágicos.	17
3.4	Exemplos de grafos graciosos. Para cada um deles, nota-se que os rótulos das arestas formam uma sequência crescente, e para cada aresta xy , o rótulo da mesma é igual a $ f(x) - f(y) $, sendo f um valor atribuído aos vértices.	18
4.1	Exemplo de alocação de canais em estações de redes móveis.	22
4.2	Um exemplo de grafo colorido com T-coloração. Os números em vermelho representam as cores atribuídas aos vértices, considerando um conjunto $T = \{1,2\}$	23
4.3	Exemplos de grafos 2-coloríveis, os quais admitem solução para ECDGP.	25
4.4	Exemplos de grafos com ciclos ímpares, os quais não admitem solução em ECDGP. Isso está representado pelos vértices em branco, que não podem receber uma cor seguindo as restrições do ECDGP.	26
5.1	Figura ilustrativa ao teorema 6. Em a) está o grafo ainda não colorido. Em b), foi realizada uma coloração clássica, onde o número ao lado de cada vértice representa a cor atribuída a esse vértice. Em c), levando em conta um conjunto proibido T , o mesmo grafo recebeu uma T-coloração. Assim, nota-se que ambas as colorações utilizaram o mesmo número de cores, como demonstrado pelo teorema.	28

- 5.2 Exemplo da execução do algoritmo usado no teorema 9 em uma árvore. Em a) está um grafo que deve ser colorido segundo as regras do ECDGP. Nesta situação, todos os vértices estão desmarcados. Em b), o vértice 2 foi escolhido para receber a cor 1 e foi marcado como visitado. A seguir, escolhe-se um vértice não visitado adjacente a algum vértice marcado v_k e atribui a ele a cor $x(v_j) = d_{jk} + x_{v_k}$. Após, marca-se o vértice v_j como visitado. Assim, em c), d), e) e f), os vértices recebem suas respectivas cores. Em g) está o resultado final: uma coloração que respeita o ECDGP, porém usando 5 cores, o que não é uma solução ótima. 31
- 5.3 Exemplo de coloração escolhendo vértices sequencialmente. Em a), têm-se o grafo a ser colorido. Em b), o primeiro vértice recebe a cor 1. Em c), o vértice 2 recebe a menor cor possível (cor 5). O procedimento se repete em d) e e) atribuindo sempre a menor cor possível ao vértice que está sendo colorido. Isto termina em f), quando o vértice 5 recebe a cor 16 e o grafo está totalmente colorido. 33
- 5.4 Exemplo de coloração ordenando os vértices pelo grau. Em a), têm-se o grafo inicial, ainda sem nenhuma cor. Em b), ocorre um empate: os vértices 2 e 3 são candidatos à coloração, pois possuem o maior grau. O vértice 3 é escolhido por ter o maior índice, recebendo a cor 1. Em c), o vértice 2 recebe a cor 3. Em d), novamente há um empate, agora entre os vértices 1, 4 e 5. De modo análogo à situação anterior, o vértice 5 é escolhido por ser o de maior índice e recebe a cor 3. Em d), o vértice 4 recebe a cor 9 e, finalmente, em f) o vértice 1 recebe a cor 7, finalizando a coloração. 35

5.5 Exemplo de coloração ordenando os vértices pelo grau de saturação. Em a), têm-se o grafo inicial, ainda sem nenhuma cor. Em b), o vértice de maior grau recebe a cor 1 (no caso, o vértice 3, por ter índice maior que o vértice 2). Em c), há empate ao determinar quem possui maior grau de saturação. O vértice 2 é escolhido por ter o maior grau dentre os vértices empatados, recebendo assim a cor 3. Em d), o vértice de maior saturação (vértice 1) recebe a cor 7. Já em e) ocorre empate novamente. Como ambos possuem o mesmo grau, o vértice 5 é escolhido por ter maior índice, recebendo a cor 3. Por fim, o vértice 4 recebe a cor 9, em f), encerrando o processo. 37

Capítulo 1

Introdução

Este trabalho envolve a investigação de conhecidos problemas e algoritmos da área de Teoria dos Grafos, com enfoque no estudo da classe de grafos rotulados, em especial os grafos graciosos e alguns problemas de coloração especiais.

1.1 Contexto

O presente projeto de pesquisa é a continuação do projeto envolvendo grafos mágicos e jogos matemáticos, do Programa de Iniciação Científica Júnior 2013/2014, do CNPq/U-FAM (projeto PIBJR- E0002, sobre Grafos Mágicos) do qual a aluna foi bolsista. No PIBIC-Jr., a ênfase foi o estudo do quadrado mágico e seu modelo em Teoria dos Grafos, os chamados grafos mágicos. Estes grafos pertencem à classe dos grafos rotulados. Sendo assim, o objetivo deste projeto é um estudo mais abrangente desta classe de grafos, com foco nos grafos graciosos e problemas de coloração especiais.

1.2 Motivação e objetivos

Os problemas de coloração em grafos são muito importantes para a Computação e Matemática, pois a aparente simplicidade dos modelos utilizados e a robustez de representação, propriedades e restrições tornam-os ideais para o estudo de problemas clássicos, além de numerosas aplicações práticas.

Este projeto de pesquisa tem ênfase nos problemas de coloração envolvendo restrições sobre os vértices, arestas ou cores a serem usadas. Um exemplo, que será mais ampla-

mente abordado por este trabalho, é a classe dos grafos graciosos. Considerando um grafo com vértices e arestas rotulados com números naturais, de modo que os rótulos das arestas recebem a diferença dos rótulos dos vértices em que incidem. Quando as arestas rotuladas determinam uma sequência de 1 a n (onde n é o número de arestas), o grafo é denominado gracioso.

Os principais objetivos deste projeto são:

- Adaptar modelos e algoritmos existentes para o problema clássico de coloração em vértices, nas variações em estudo e para os grafos rotulados e grafos graciosos.
- Estudar os problemas de coloração relacionados à alocação de canais em redes móveis sem fio e geometria de distâncias.
- Validar propriedades na tentativa de se propor caracterizações para os problemas em investigação.
- Organizar e divulgar os resultados da pesquisa sob a forma de relatórios e artigos científicos.
- Compartilhar os resultados obtidos com a comunidade de algoritmos e combinatória em eventos nacionais e internacionais.

1.3 Metodologia de desenvolvimento do trabalho

A metodologia de desenvolvimento inclui os seguintes itens:

- Aquisição de material, revisão da literatura e realização de disciplinas para adquirir o ferramental teórico necessário para o desenvolvimento do projeto de pesquisa.
- Participação de reuniões periódicas com a orientadora e com o grupo de pesquisa de Otimização, Algoritmos e Complexidade Computacional.
- Estudo da classe de grafos rotulados e sub-classes de interesse, como os grafos graciosos, bem como de variações de problemas de colorações com restrições, de tal forma a contribuir com a teoria envolvida através da estruturação da teoria relacionada, divulgação, e na tentativa de geração de novos conceitos, propriedades, caracterizações e algoritmos.

- Estudo dos problemas de coloração relacionados à alocação de canais em redes móveis sem fio e à geometria de distâncias, assuntos de interesse do grupo de Otimização, Algoritmos e Complexidade Computacional. Estudar as propriedades validadas até então na tentativa de contribuir com novas caracterizações.
- Elaboração de resumos do projeto de pesquisa com documentação do embasamento teórico obtido e inclusão dos resultados parciais gerados, para a elaboração dos relatórios parcial e final, bem como apresentações.
- Divulgação da pesquisa através de seminários, participação em eventos nacionais e internacionais de relevância e publicação em periódicos importantes para a área.
- Elaboração da tese partindo dos resultados obtidos e registrados durante a pesquisa.

1.4 Cronograma

O cronograma de atividades encontra-se na Tabela 1.1, localizada na página seguinte, na orientação horizontal para facilitar a visualização. A atividade “Divulgação dos resultados (participação em eventos e publicação em revistas científicas)” ainda está em andamento, sendo o próprio Congresso de Iniciação Científica da UFAM parte dos planos, além de outros eventos.

1.5 Organização do restante deste relatório

O trabalho está organizado de acordo com a seguinte estrutura:

- o Capítulo 2 contém o referencial teórico para o trabalho;
- o Capítulo 3 aborda a classe de grafos rotulados;
- o Capítulo 4 apresenta problemas de coloração especiais, como a T-coloração e outros;
- o Capítulo 5 traz os resultados do projeto;
- o Capítulo 6 possui as considerações finais deste relatório parcial.

Tabela 1.1: Cronograma de atividades do PIBIC de agosto de 2014 a julho de 2015. Em cinza estão destacadas as atividades realizadas. Em azul, as atividades que estão sendo concluídas.

Nº	Descrição	Ago	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul
1	Levantamento de material, revisão da literatura.	X	X	X	X	X				X	X		
2	Reuniões periódicas com o orientador e o grupo de pesquisa	X	X	X	X	X	X	X	X	X	X	X	X
3	Realização de disciplinas optativas relacionadas	X	X	X	X	X	X	X	X	X	X	X	
4	Elaboração da Apresentação da proposta do projeto de pesquisa e revisão da literatura	X	X	X	X								
5	Desenvolvimento da pesquisa (caracterizações, provas, análise complexidade, elaboração de algoritmos)	X	X	X	X	X	X	X	X	X	X		
6	Elaboração do Resumo e Relatório Parcial				X	X	X						
7	Elaboração do Resumo e Relatório Final (atividade obrigatória)									X	X	X	X
8	Preparação da Apresentação Final para o Congresso (atividade obrigatória)											X	X
9	Divulgação dos resultados (participação em eventos e publicação em revistas científicas)										X	X	X

Capítulo 2

Referencial teórico

Este capítulo apresenta os conceitos básicos para o desenvolvimento do projeto, explorando as áreas de Teoria dos Grafos e Complexidade Computacional. Como a aluna participou de um PIBIC-Jr relacionado à Teoria dos Grafos, a bagagem teórica inicial já foi exposta no relatório final de PIBIC-Jr [1]. Aqui, é feita uma representação para situar o leitor à área.

2.1 Teoria dos Grafos

2.1.1 Conceitos básicos

Um grafo $G = (V, E)$ é uma estrutura matemática composta de dois conjuntos:

- O conjunto V , dos **vértices** (do inglês, *vertex*).
- O conjunto E , das **arestas** (do inglês, *edges*).

As arestas estabelecem uma relação entre os vértices. Na Figura 2.1, abaixo, os vértices são os círculos azuis e as arestas, os segmentos de reta que os relacionam [17].

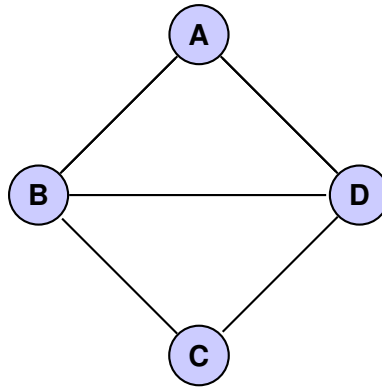


Figura 2.1: Um exemplo de grafo $G = (V, E)$ com 4 vértices.

Portanto, em um grafo é importante conhecer quais são seus vértices e como estão relacionados, isto é, quais são as arestas. Uma grande variedade de problemas pode ser modelada desta forma, desde o mapa viário entre cidades até a relação estabelecida pelos usuários de um site de relacionamentos, como uma grande rede de contatos.

2.1.2 Algumas propriedades

2.1.2.1 Grau de um vértice

Quando uma aresta está ligada a um vértice, dizemos que ela é **incidente** ao vértice. O número de vezes que uma aresta incide em um vértice determina o **grau** (do inglês, *degree*) do vértice v e é representado por $d(v)$. Por exemplo, no grafo anterior, pode-se analisar o vértice A e constatar que $d(A) = 2$.

2.1.2.2 Laços

Uma aresta pode estabelecer uma relação entre um vértice e ele mesmo. Quando isso acontece, o grafo possui um **laço** (do inglês, *loop*). Na Figura 2.2, abaixo, têm-se um laço no vértice C, que está ligado consigo mesmo.

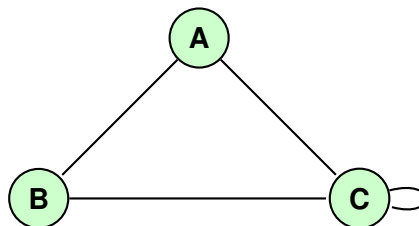


Figura 2.2: Um grafo com laços: o vértice C está ligado consigo mesmo, constituindo um laço.

2.1.2.3 Multigrafos

Quando dois vértices estão ligados por mais de uma aresta, o grafo é denominado **multigrafo** ou **grafo com arestas múltiplas**. A Figura 2.3 representa um grafo com essas características.

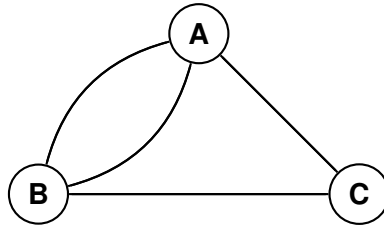


Figura 2.3: Um multigrafo: os vértices A e B compartilham 2 arestas (paralelas).

2.1.2.4 Grafo simples

Um grafo que não contém laços nem arestas múltiplas pode ser denominado um **grafo simples**. Neste trabalho, os grafos estudados serão grafos simples (Figura 2.4).

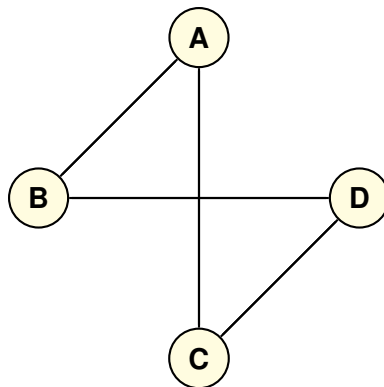


Figura 2.4: Um grafo simples com 4 vértices e 4 arestas.

2.1.2.5 Grafos conexos e desconexos

Considerando a Figura 2.5:

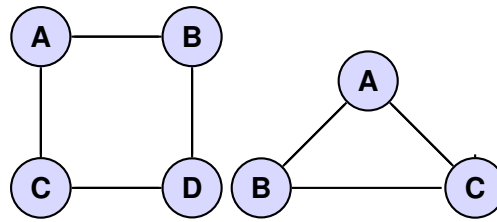


Figura 2.5: Um grafo desconexo composto de 2 componentes conexas: um grafo 4-ciclo e um grafo 3-ciclo.

Os grafos acima estão **desconectados**, pois não possuem uma relação estabelecida entre seus vértices. É o que se pode chamar de **grafo desconexo**. Quando há um caminho entre os grafos, pode-se chamá-los de **grafos conexos**, como na Figura 2.6.

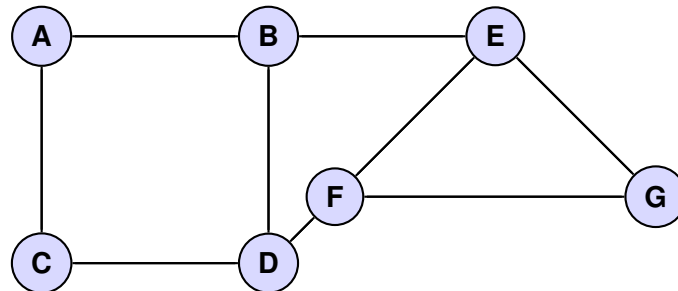


Figura 2.6: Um grafo conexo com 7 vértices e 9 arestas.

2.1.2.6 Grafos direcionados e não-direcionados

Um grafo é dito **direcionado** (Figura 2.7) quando estabelece uma ordem, uma direção em suas arestas. Só se pode percorrer o grafo seguindo as direções das arestas, que tornam-se setas. Quando não há ordem nos pares de vértices, o grafo é dito **não-direcionado**.

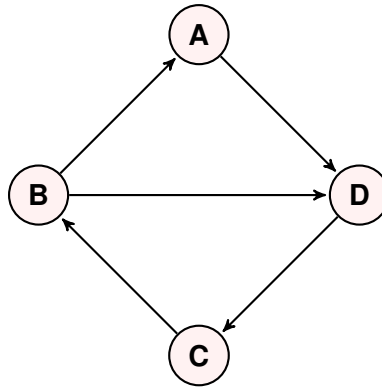


Figura 2.7: Um grafo direcionado: toda aresta possui uma direção, ou seja, sai de um vértice e chega em outro.

2.1.3 Classes especiais de grafos

Alguns tipos de grafos possuem características bem definidas e, por isso, são considerados modelos clássicos para estudo. Abaixo, têm-se alguns deles.

2.1.3.1 Grafos completos

Um **grafo completo** pode ser definido como um grafo onde todo par de vértices é ligado por uma aresta. Um grafo completo com n vértices pode ser chamado de K_n . Abaixo, na Figura 2.8, um exemplo de grafo completo.

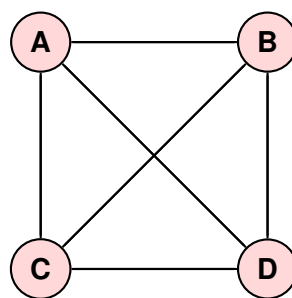


Figura 2.8: Um grafo completo K_4 (contém 4 vértices e todas as possíveis arestas).

Grafos nulos ou vazios

Um **grafo nulo** ou **vazio** é um grafo que não contém nenhuma aresta, apenas vértices (Figura 2.9).

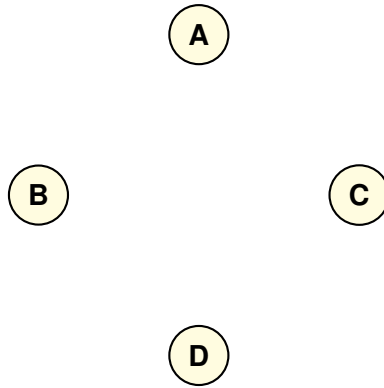


Figura 2.9: Um grafo nulo (não contém arestas).

2.1.3.2 Grafos regulares

Em um **grafo regular**, todos os vértices possuem o mesmo grau. Dizemos que o grafo é k -regular ou de grau k (Figura 2.10).

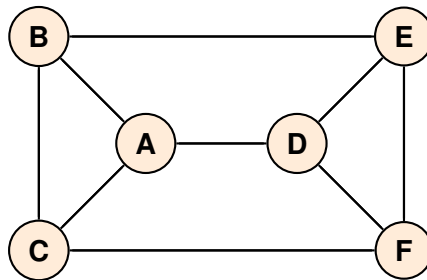


Figura 2.10: Um grafo regular de grau 3, onde todos os vértices possuem grau 3.

2.1.3.3 Ciclo

Um grafo **ciclo** é um grafo conexo regular onde todos os vértices tem grau 2. A notação é C_n , onde n é o número de vértices (Figura 2.11).

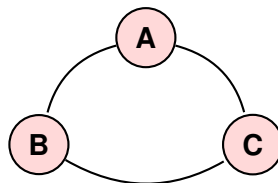


Figura 2.11: Um grafo ciclo de tamanho 3, C_3 .

2.1.3.4 Caminho

Quando uma aresta é retirada de um ciclo, têm-se o grafo **caminho**. Neste tipo de grafo, o comprimento é dado pelo número de arestas. Dessa forma, o caminho P_n é obtido ao remover uma aresta do ciclo C_{n+1} , como na Figura 2.12.

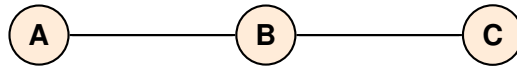


Figura 2.12: Um grafo caminho P_2 , de tamanho 2.

2.1.3.5 Árvores

Uma **árvore** é um grafo conexo sem ciclos como subgrafos (Figura 2.13). Dessa forma, consegue-se realizar buscar mais rápidas, pois os vértices estão conectados de forma econômica [17].

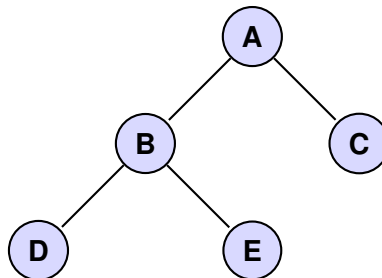


Figura 2.13: Uma árvore (grupo conexo e acíclico).

2.1.3.6 Grafos bipartidos

Um **grafo bipartido** é um grafo em que o conjunto de vértices pode ser dividido em dois subconjuntos V_1 e V_2 , de modo que cada aresta tem uma extremidade em V_1 e outra em V_2 (Figura 2.14). Dessa forma, V_1 e V_2 são ditos **subconjuntos independentes**, pois não há arestas ligando dois vértices de V_1 , nem dois vértices de V_2 .

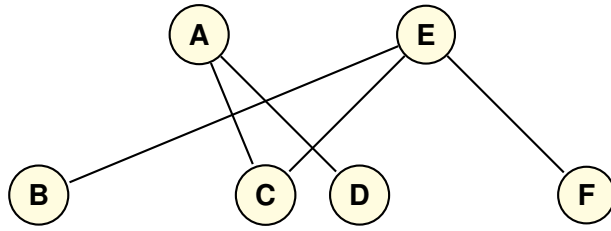


Figura 2.14: Um grafo bipartido $B_{2,4}$.

2.1.3.7 Grafos bipartidos completos

Quando, em um grafo bipartido, todos os vértices de V_1 estão ligados a todos os vértices de V_2 , ele é denominado **grafo bipartido completo** e recebe a notação $K_{p,q}$ (Figura 2.15).

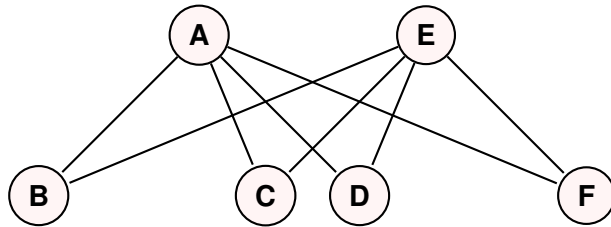


Figura 2.15: Um grafo bipartido completo $K_{2,4}$.

2.1.3.8 Grafos rotulados

Um grafo é dito **rotulado** quando recebe valores (ou rótulos) nos vértices ou arestas (Figura 2.16). Esses valores podem ser números, ou, em outra abordagem, cores. Quando os vértices e arestas possuem rótulos, o grafo é chamado **totalmente rotulado** [2].

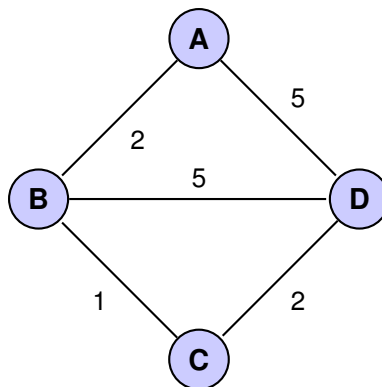


Figura 2.16: Um exemplo de grafo rotulado, onde as arestas possuem rótulos (números naturais) associados.

2.2 Complexidade Computacional

2.2.1 Um pouco sobre complexidade computacional

A complexidade computacional é a área da ciência da computação que estuda o porquê de alguns problemas serem tão difíceis de serem resolvidos por computadores [19].

A área nasceu quando os computadores deixaram de ser restritos a instituições científicas, em meados dos anos sessenta. Os programadores perceberam que, para alguns tipos de problemas, os melhores algoritmos demoravam tanto tempo para executar que não era possível resolvê-los por meios computacionais. A questão era: esse fato era decorrente da incapacidade de encontrar um algoritmo eficiente ou da dificuldade inerente do problema?

A partir disso, buscou-se medir a dificuldade de determinadas tarefas com base na quantidade de recursos computacionais necessários para computá-las e, assim, obteve-se uma classificação dos problemas que podem ser resolvidos algorítmicamente de acordo com seu grau de dificuldade.

Dessa forma, os principais objetivos da complexidade computacional são: estimar os recursos computacionais necessários para solucionar problemas algorítmicos; identificar e definir a noção de “eficiência” ou “tratabilidade” dos problemas computacionais; desenvolver métodos para classificá-los dentro de diversas classes de complexidade; comparar a eficiência de diversos modelos computacionais distintos [19].

2.2.2 O problema P vs NP

A pesquisa em complexidade computacional pode ser dividida em dois grandes grupos: o estudo da dificuldade de um problema computacional específico ou a investigação de como certos recursos computacionais e classes de problemas estão relacionados [19]. No entanto, existem problemas individuais que capturam propriedades importantes da classe a que pertencem. Logo, pode-se voltar o estudo de classes a esses problemas especiais, chamados de **NP completos**.

É nesse contexto que se insere o problema P vs NP. P é a denotação do conjunto de problemas computacionais com soluções que podem ser encontradas de forma eficiente. Já NP é a classe de problemas computacionais com soluções que podem ser verificadas de forma eficiente. A pergunta é como essas duas classes se relacionam, ou seja, se para todo problema cujas soluções são facilmente verificáveis existe um algoritmo capaz de

solucioná-lo eficientemente.

Para solucionar um problema pertencente à classe NP, é necessária uma busca exaustiva de soluções: um algoritmo gera todas as respostas possíveis para uma dada instância e depois as checa. Como resultado, não é possível obter um algoritmo final que seja eficiente. A maioria dos pesquisadores acredita que essa busca exaustiva não pode ser evitada, no entanto, até agora ninguém conseguiu comprovar esse fato matematicamente.

O problema P vs NP é muito utilizado na criptografia moderna. Serviços como a segurança da Internet e transações financeiras dependem de algumas hipóteses relacionadas a esse problema, como a dificuldade de fatorar números inteiros muito grandes. Se $P = NP$, significaria que existe uma solução para essas hipóteses, comprometendo a segurança desses serviços.

Além disso, se fosse provado que $P = NP$, aconteceriam mudanças fantásticas não apenas na área de ciência da computação, mas também em diversas outras. Um computador poderia encontrar provas formais para diversos teoremas da matemática contemporânea, por exemplo, e a área de inteligência artificial seria muito favorecida pela existência de algoritmos práticos para problemas NP.

Capítulo 3

Grafos rotulados

Este capítulo trata dos grafos rotulados, abordando mais especificamente a classe dos grafos graciosos.

3.1 Grafos rotulados

Rotulação de grafos é a atribuição de valores para os vértices ou arestas, ou ambos, sob certas condições [12]. Existem muitas técnicas de rotulação de grafos, estudadas desde a introdução do conceito nos anos 1960.

3.1.1 Origem

A introdução do conceito de grafos rotulados é, em sua maioria, atribuída a Rosa em 1967, e também a Graham e Sloane, em 1980.

Rosa [21] criou uma função f e a chamou de β -*valuation*: em um grafo G com n arestas, se f é uma função injetora dos vértices de G para o conjunto $\{0, 1, \dots, n\}$ de modo que cada aresta xy recebe um rótulo $|f(x) - f(y)|$, e todas as arestas possuem um rótulo diferente. Posteriormente, Golomb denominou este tipo de rotulação como *graciosa*, e este termo segue popular.

O objetivo principal de Rosa ao propor este tipo de rotulação era criar uma ferramenta para decompor um grafo completo em subgrafos isomorfos. Assim, tornaria-se possível atacar a conjectura de Ringel. Essa conjectura diz que, dado um número n de arestas, um grafo completo K_{2n+1} pode ser decomposto em $2n + 1$ subgrafos isomorfos.

3.1.2 Algumas classes de grafos rotulados

Os grafos recebem rótulos de acordo com certas condições, dependendo da situação e do problema que se deseja resolver. Dessa forma, diferentes restrições são atribuídas aos grafos, gerando diferentes classes de grafos. A seguir, algumas delas serão abordadas.

3.1.2.1 Grafos harmoniosos

Seja G um grafo com k arestas. Uma **rotulação harmoniosa** de G é uma função injetora dos vértices de G para um conjunto $\{0, 1, 2, \dots, k\} \pmod k$. Essa função injetora induz uma bijeção entre as arestas de G e os números $\pmod k$ da seguinte forma: ao tomar uma aresta xy , o rótulo desta aresta será $(x + y) \pmod k$, onde os valores de x e y são os rótulos dos vértices x e y [22] (Figura 3.1).

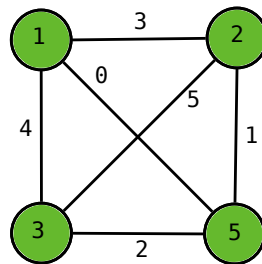


Figura 3.1: Um exemplo de grafo harmonioso. Aqui, $k = 6$, portanto o rótulo de cada aresta xy é igual a $(x + y) \pmod k$. Deste modo, a aresta $(3,5)$, por exemplo, recebeu o valor 2 como rótulo, pois $(3 + 5) \pmod 6 = 8 \pmod 6 = 2$. De modo análogo, todas as arestas receberam um rótulo.

3.1.2.2 Grafos mágicos

Um grafo é dito **mágico** se as arestas podem ser rotuladas com números reais de modo que a soma dos rótulos das arestas incidentes a um vértice é a mesma, independente da escolha do vértice [2]. A Figura 3.2 apresenta um exemplo de grafo mágico.

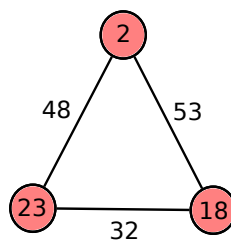


Figura 3.2: Um grafo mágico, onde a soma dos rótulos dos vértices com o rótulo da aresta que os relacionam é constante, independente do vértice escolhido.

Os grafos mágicos possuem uma relação direta com os chamados **quadrados mágicos**. Um quadrado mágico é uma tabela de números dispostos na forma de um quadrado, onde a soma dos elementos de uma linha, coluna e diagonais seja constante [3]. Ao adotar um modelo em grafos para esse jogo matemática, obtêm-se uma configuração como a seguinte, representada na Figura 3.3:

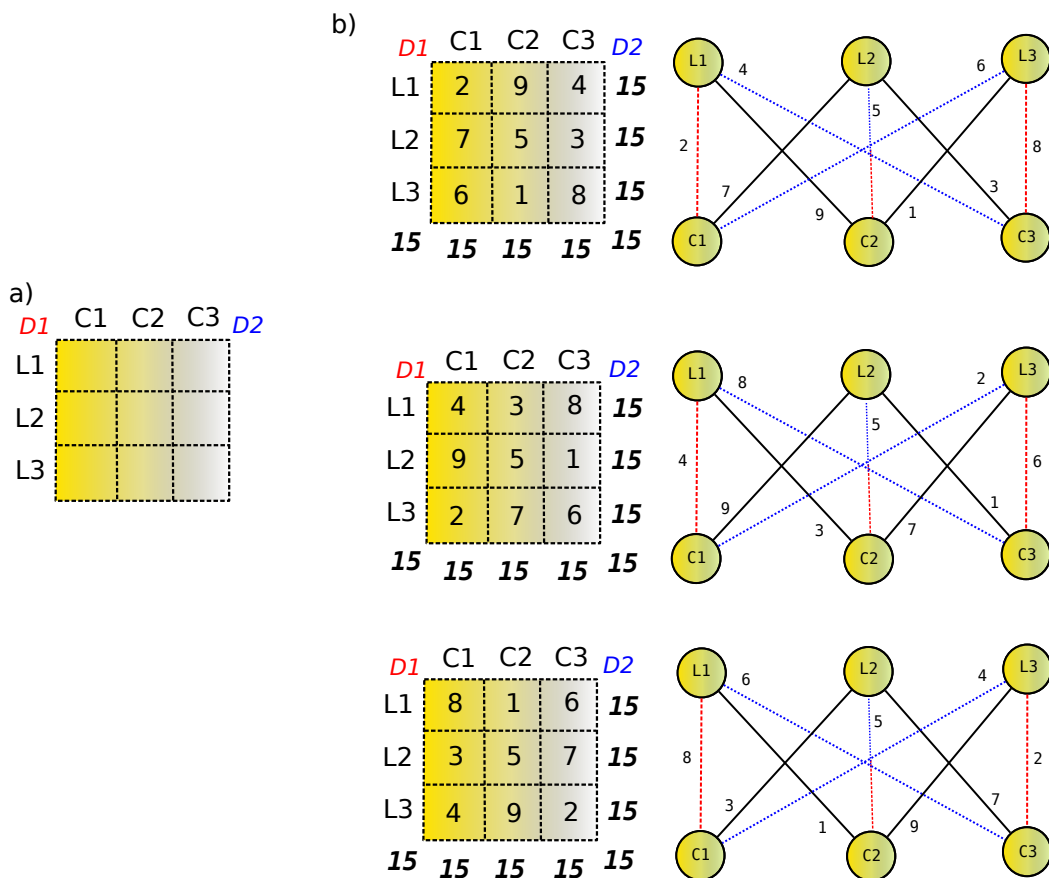


Figura 3.3: Quadrado mágico 3x3 modelado em teoria dos grafos, resultando em um grafo mágico. Em a), há o grid ainda não preenchido. Em b), três soluções que atendem às restrições do quadrado mágico e seus respectivos modelos como grafos mágicos.

Os grafos mágicos e o problema do quadrado mágico foram os temas abordados no PIBIC-Jr da presente aluna. Para mais informações, consultar [1].

3.2 Coloração em grafos

A coloração de grafos é a atribuição de cores aos vértices de um grafo, de maneira que vértices adjacentes tenham cores diferentes. Neste tipo de rotulação em grafos, as cores podem ser encaradas como números a serem atribuídos aos vértices. Como este assunto é de grande relevância ao projeto, ele será abordado mais especificamente no próximo capítulo.

3.3 Grafos graciosos

Como já mencionado anteriormente, a chamada *rotulação graciosa* foi introduzida por Rosa em 1967 [21]. Por ser a primeira classe de grafos rotulados proposta, ela recebeu uma atenção especial. Por exemplo, foram realizados muitos estudos baseados na *Conjectura da Árvore Graciosa*, provando a validade da afirmação em diferentes classes de árvores.

3.3.1 Definição

Seja G um grafo com n vértices e m arestas. Uma rotulação graciosa é um mapeamento $f: V(G) \rightarrow \{0, 1, 2, \dots, m\}$ de modo que $|f(x) - f(y)|$ é um rótulo único para cada aresta $xy \in E(G)$, onde $V(G)$ é o conjunto de vértices e $E(G)$ é o conjunto de arestas. Dessa forma, o conjunto $E(G) = \{1, 2, \dots, m\}$ [4]. A Figura 3.4 traz dois exemplos de grafos graciosos.

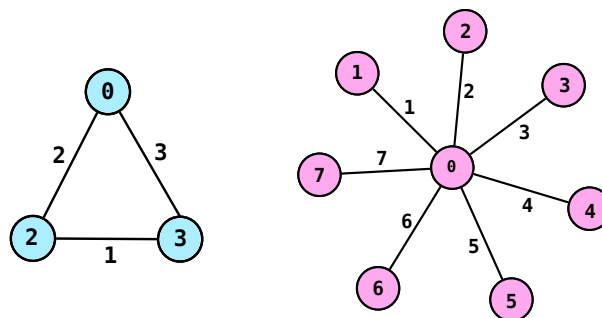


Figura 3.4: Exemplos de grafos graciosos. Para cada um deles, nota-se que os rótulos das arestas formam uma sequência crescente, e para cada aresta xy , o rótulo da mesma é igual a $|f(x) - f(y)|$, sendo f um valor atribuído aos vértices.

3.3.2 Grafos graciosos e alguns teoremas

Existem alguns teoremas e conjecturas envolvendo grafos graciosos. Estudá-los e compreender suas provas é parte importante desta iniciação científica. São eles:

Teorema 1. *Se $n > 4$, o grafo completo K_n não é gracioso.*

Teorema 2. *Se G é um grafo euleriano com m arestas de modo que $m \equiv 1$ ou $2 \pmod{4}$, então G não pode ter rotulação graciosa.*

Uma conjectura em especial é bem relevante no estudos dos grafos graciosos. Chama-se *Conjectura da Árvore Graciosa* e despertou o interesse de muitos pesquisadores em estudar a questão.

Conjectura 1. *Todas as árvores são graciosas.*

Até o momento, a validade da conjectura foi provada para algumas classes restritas de grafos. Também é interesse desta iniciação científica estudar quais são essas classes e de que forma o problema foi solucionado para elas.

Capítulo 4

Problemas de coloração em grafos

Neste capítulo, alguns problemas de coloração em grafos serão expostos. Tais problemas constituem-se em uma espécie de rotulação de grafos, onde os números podem ser encarados como cores atribuídas aos vértices seguindo algumas restrições. Aqui, serão abordadas colorações nos vértices, de acordo com restrições nas arestas.

4.1 Coloração clássica de vértices

Uma *coloração* do conjunto de vértices de um grafo é atribuir cores aos vértices de tal forma que vértices adjacentes tenham cores diferentes [11]. Assim, pode-se notar que a coloração de grafos é, na verdade, um tipo especial de rotulação de grafos: ao encarar uma cor como um número inteiro, têm-se uma rotulação de grafos tradicional.

4.1.1 Colorações mínimas

Uma coloração de vértices é *mínima* se o número de cores é o menor possível, ou seja, não há outra coloração que utilize menos cores.

Ao conceito de coloração mínima, é associado o de *número cromático* de um grafo G , que é o número mínimo de cores necessárias para colorir o grafo. É denotado por $\chi(G)$. Se $\chi(G) \leq k$, diz-se que G é colorível com k cores, ou que G é k -colorível.

4.1.2 Coloração de mapas

O problema da coloração de mapas é muito utilizado para dar um exemplo inicial de como a coloração de grafos pode ser aplicada.

Em 1852, o aluno inglês Francis Guthrie propôs o seguinte problema: quantas cores diferentes são necessárias para colorir um mapa qualquer, de tal modo que estados que tem fronteira em comum tenham cores diferentes? A questão foi amplamente divulgada por Augustus De Morgan, professor de Guthrie, e uma demonstração correta foi obtida apenas em 1976, por Kenneth Appel e Wolfgang Haken [14].

Se um mapa for visto como um grafo G , cada país pode ser um vértice e as arestas representam as fronteiras entre eles. Assim, chegou-se ao seguinte teorema, chamado de *Teorema das quatro cores*:

Teorema 3. *Se H é um grafo planar, é sempre possível colorir seus vértices com quatro cores, de modo que quaisquer dois vértices adjacentes tenham cores distintas.*

Ou seja, todo mapa é 4-colorível.

4.2 T-coloring

A coloração de grafos é um recurso muito utilizado para modelar alguns problemas. Dessa forma, diferentes tipos de coloração surgem para atender restrições variadas, de acordo com o problema. O chamado *T-coloring* surgiu como um modelo de um problema de telecomunicações chamado *alocação de canais* (em inglês, *channel assignment problem*).

4.2.1 Problema de alocação de canais

Em um mundo onde os dispositivos móveis, tais como *smartphones* e *tablets*, tornam-se cada vez mais populares entre as pessoas, torna-se uma preocupação prover um serviço bom de telefonia móvel celular (uma vez que esses dispositivos utilizam, em sua maioria, redes móveis para realizar o acesso à internet, além de chamadas e afins). Para tanto, é importante realizar um uso eficaz do espectro magnético, assim evitando possíveis problemas como interferências na rede. Uma das formas de resolver esse problema é encontrar a melhor ocupação possível do espectro. É dessa questão que trata o problema de alocação de canais [10].

A definição formal do problema é a seguinte: existe um conjunto de n antenas, onde cada antena i tem uma demanda de q_i canais, que corresponde à quantidade de chamadas daquela estação. Cada chamada precisa de um canal para ser realizada. O k -ésimo canal alocado para a i -ésima antena é denotado por f_{ki} . Assim, para todo par (i, j) de antenas que podem interferir entre si, existe uma distância de separação d_{ij} para a qual a condição $|f_{ki} - f_{mj}| \geq d_{ij}$ deve ser respeitada. Além disso, as estações não podem assumir qualquer canal dentre os disponíveis: para cada uma, existe um conjunto chamado *domínio de frequência* que contém as possibilidades de canais para as chamadas de i . A Figura 4.1, abaixo, é um exemplo de como as estações podem estar dispostas e sua representação como um grafo.

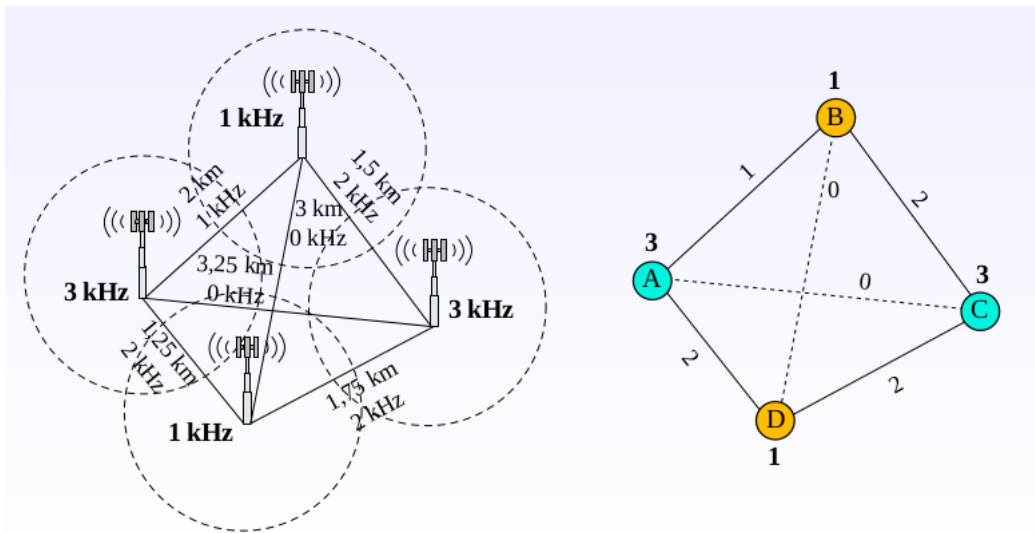


Figura 4.1: Exemplo de alocação de canais em estações de redes móveis.

4.2.2 Definição de T-coloring

O T-coloring ou T-coloração pode ser definido da seguinte forma: seja T um conjunto de inteiros não-negativos (assumindo que T contém 0). Então uma T -coloração de um grafo simples G é uma função $f(v): V(G) \rightarrow N$. Essa função atribui um inteiro positivo para cada vértice de G , de modo que se uv é uma aresta de G , então $|f(u) - f(v)| \notin T$. Se $T = \{0\}$, então têm-se apenas a coloração clássica de vértices [24]. A Figura 4.2 apresenta um grafo colorido seguindo as restrições da T-coloração.

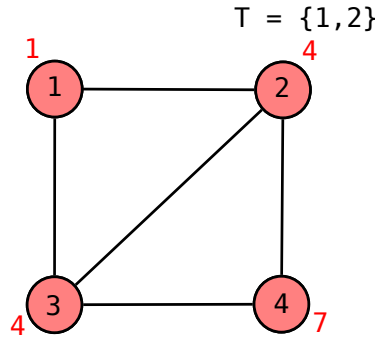


Figura 4.2: Um exemplo de grafo colorido com T-coloração. Os números em vermelho representam as cores atribuídas aos vértices, considerando um conjunto $T = \{1, 2\}$.

Simplificando o problema de alocação de canais e trazendo-o para a Teoria dos Grafos, podemos obter um grafo simples $G = (V, E)$, onde os vértices representam as estações e as arestas possuem todas as estações que interferem entre si. Além disso, há um conjunto T de distâncias que não podem ocorrer entre as estações que interferem (para evitar interferências indesejadas e outros problemas) [20]. Com essa restrição em mente, pode-se chegar à seguinte definição:

$$\{x, y\} \in E(G) \Rightarrow |f(x) - f(y)| \notin T.$$

Assim, podemos notar que o problema de alocação de canais, em uma forma mais simples, pode ser modelado como uma coloração T-coloring.

4.2.3 Propriedades do T-Coloring

O T-coloring possui algumas propriedades importantes:

1. assumindo que o grafo G possui uma coloração T-coloring. Se s é a maior cor atribuída a um vértice de G pelo T-coloring c , então \bar{c} de G , definido por

$$\bar{c}(v) = s + 1 - c(v)$$

para cada vértice v de G também é um T-coloring de G , chamado *coloração complementar* de c [15].

As propriedades seguintes podem ser encontradas em [5].

2. o T -número cromático $\chi_T(G)$ é o número mínimo de cores que podem ser usadas em um T -coloring de G ;
3. para um T -coloring c de G , o c -span $sp_T(c)$ é o valor máximo de $|c(u) - c(w)|$ considerando todos os pares u, w de vértices de G ;
4. o T -span $sp_T(G)$ de G é o mínimo c -span considerando todos os T -colorings c de G .

4.3 Distance coloring

O *distance coloring* é um modelo de coloração que está sendo proposto e estudado pela orientadora desta iniciação científica, Rosiane de Freitas, e o doutorando Bruno Raphael Dias. O distance coloring é, na verdade, um problema de coloração baseado no conceito de *geometria de distâncias*. Nesse problema, é fornecido um grafo não direcionado $G = (V, E, d)$, onde $d: E \rightarrow \mathbb{R}^+$, e deve-se encontrar a projeção $x: V \rightarrow \mathbb{R}^3$ tal que $\|x(i) - x(j)\| = d_{ij}$ para toda aresta $(i, j) \in E$.

Assim, pode-se considerar as cores que serão atribuídas aos vértices como sendo pontos na reta que serão posicionados nos vértices do grafo. Assim, chega-se às seguintes definições [6]:

Definição 1 (Problema de Coloração em Geometria de Distâncias Iguais - ECDGP).

Dado um grafo não direcionado $G = (V, E, d)$, onde $d: E \rightarrow \mathbb{Z}^+$, deve-se encontrar uma projeção $x: V \rightarrow \mathbb{R}$, tal que $|x(i) - x(j)| = d_{ij}$, para cada $(i, j) \in E$. A extensão da projeção é dada por $\max_{i \in V} x(i)$. O objetivo é encontrar a projeção de menor extensão.

Definição 2 (Problema de Coloração em Geometria de Distâncias Mínimas - MCDGP).

Dado um grafo não direcionado $G = (V, E, d)$, onde $d: E \rightarrow \mathbb{Z}^+$, deve-se encontrar uma projeção $x: V \rightarrow \mathbb{R}$ tal que, para cada $(i, j) \in E$, $|x(i) - x(j)| \geq d_{ij}$. O objetivo é encontrar a projeção de menor extensão.

Definição 3 (Problema de Coloração em Geometria de Distâncias Mistas - MxCDGP).

Dado um grafo não direcionado $G = (V, E, d, f)$, onde $d: E \rightarrow \mathbb{N}$ e $f: E \rightarrow \{0, 1\}$, deve-se encontrar uma projeção $x: V \rightarrow \mathbb{N}$ tal que, para cada $(i, j) \in E$ se, e somente se, $f(i, j) = 0$, então $|x(i) - x(j)| \geq d_{ij}$, e se, e somente se, $f(i, j) = 1$, então $|x(i) - x(j)| = d_{ij}$, onde a extensão de x deve ser a menor possível.

Até o momento, algumas propriedades foram detectadas e provadas envolvendo o ECDGP, ou seja, quando as distâncias entre os vértices são sempre iguais:

Teorema 4. *Seja $G = (V, E, d)$, onde $\forall (i, j) \in E, d_{ij} = \mu$, onde μ é uma constante e G é 2-colorível (considerando o problema de coloração clássica de vértices). Então G sempre admite uma solução para ECDGP usando duas cores.*

A Figura 4.3 traz dois exemplos de grafos 2-coloríveis e uma maneira de serem coloridos seguindo as restrições do ECDGP.

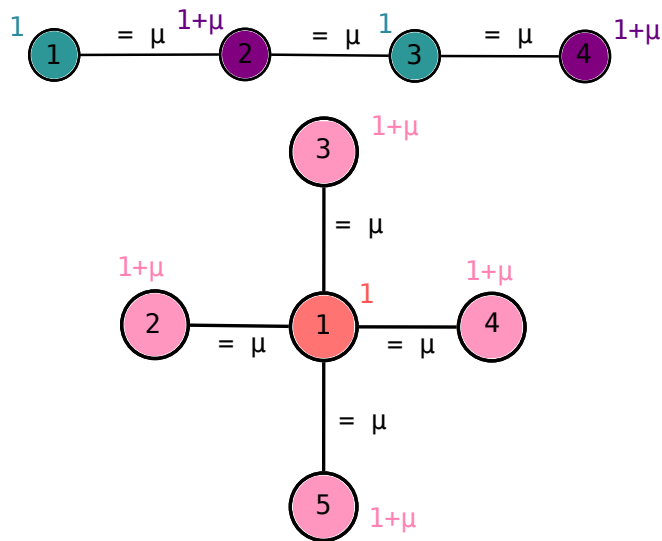


Figura 4.3: Exemplos de grafos 2-coloríveis, os quais admitem solução para ECDGP.

Teorema 5. *Seja $G = (V, E, d)$, onde $\forall (i, j) \in E, d_{ij} = \mu$ (onde μ é uma constante) um grafo contendo um ciclo ímpar de comprimento maior ou igual a 3. Então G não possui solução para ECDGP.*

A seguir, a Figura 4.4 contém dois grafos contendo ciclo ímpar de comprimento 3, exemplificando o teorema 5.

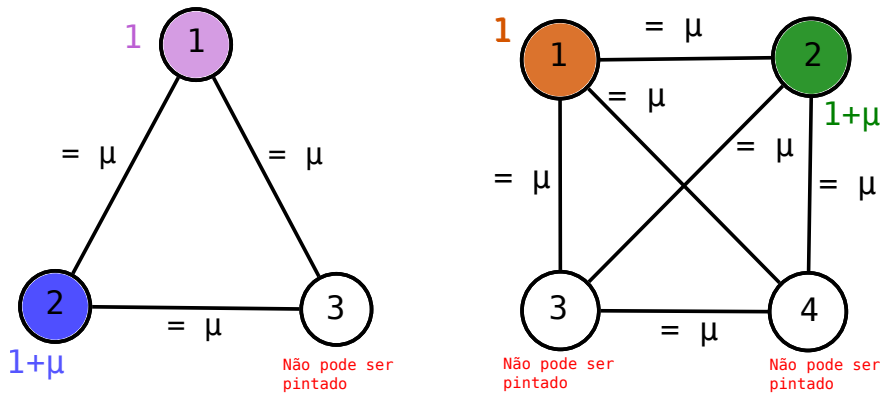


Figura 4.4: Exemplos de grafos com ciclos ímpares, os quais não admitem solução em ECDGP. Isso está representado pelos vértices em branco, que não podem receber uma cor seguindo as restrições do ECDGP.

Dessa forma, estudar a demonstração destes teoremas, bem como encontrar novas formas de prová-los e analisar a validade da coloração para outros tipos de grafos (como os graciosos, por exemplo) é o principal foco deste projeto de iniciação científica.

Capítulo 5

Resultados

Neste capítulo, os resultados do projeto serão apresentados. Dentre eles, estão detalhamento de teoremas relacionados às colorações em estudo e resultados de implementação de algoritmos.

5.1 Detalhamento de teoremas

Os problemas estudados neste projeto possuem forte bagagem teórica, como alguns teoremas que trazem conclusões importantes. A compreensão e detalhamento destes teoremas, apresentados a seguir, é um dos focos deste trabalho.

5.1.1 T-Coloring

Os autores Margaret B. Cozzens e Fred S. Roberts demonstram alguns teoremas que explicam o porquê de o T-span ser mais interessante que o T-número cromático no caso da T-coloração.

Teorema 6. *Seja G um grafo. Para cada conjunto finito de números não negativos T (incluindo 0), $\chi_T(G) = \chi(G)$.*

Prova. Uma vez que toda T-coloração de G é também um problema de coloração clássico de G , segue que $\chi(G) \leq \chi_T(G)$.

Supondo que $\chi(G) = k$ e que r é o maior inteiro em T . Seja dada uma coloração clássica k -coloring c de G usando as cores $1, 2, \dots, k$. Defina-se a função $c' : V(G) \rightarrow N$

como

$$c'(v) = (r+1)c(v)$$

para cada vértice v de G . Para cada dois vértices u e w de G ,

$$\begin{aligned} |c'(u) - c'(w)| &= |(r+1)c(u) - (r+1)c(w)| \\ &= (r+1)|c(u) - c(w)|. \end{aligned}$$

Nota-se que, na coloração clássica, as cores atribuídas a vértices adjacentes devem ser diferentes, ou seja, para cada dois vértices adjacentes u e w , $c(u) \neq c(w)$. Isso pode ser reescrito como $|c(u) - c(w)| \geq 1$. Levando isto em consideração, pode-se afirmar que, ao multiplicar o valor $|c(u) - c(w)|$ por $(r+1)$, o resultado será sempre maior ou igual a $r+1$. Logo:

$$(r+1)|c(u) - c(w)| \geq r+1$$

e então $|c'(u) - c'(w)| \notin T$. Logo, c' é uma T-coloração de G . Uma vez que k cores são usadas nesta T-coloração de c , segue que $\chi_T(G) \leq k = \chi(G)$. Portanto $\chi_T(G) = \chi(G)$. A Figura 5.1 ilustra este teorema.

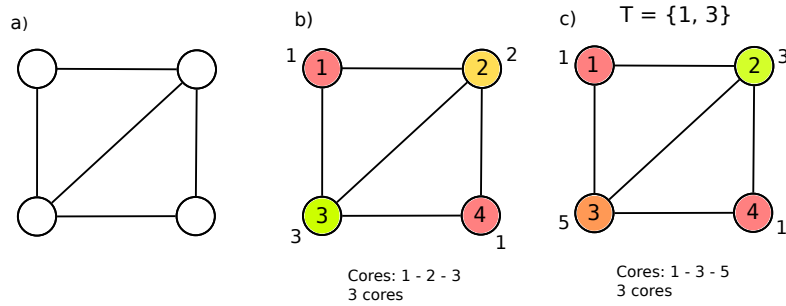


Figura 5.1: Figura ilustrativa ao teorema 6. Em a) está o grafo ainda não colorido. Em b), foi realizada uma coloração clássica, onde o número ao lado de cada vértice representa a cor atribuída a esse vértice. Em c), levando em conta um conjunto proibido T , o mesmo grafo recebeu uma T-coloração. Assim, nota-se que ambas as colorações utilizaram o mesmo número de cores, como demonstrado pelo teorema.

Teorema 7. *Seja T um conjunto finito de inteiros não negativos (incluindo 0). Se G é um grafo k -cromático com número de clique ω , então $sp_T(K_\omega) \leq sp_T(G) \leq sp_T(K_k)$, onde*

K_ω é um grafo completo de ordem ω e K_k é um grafo completo de ordem k .

Prova. Inicialmente, considera-se o seguinte: uma *clique* de G é um subgrafo completo contido em G . Uma *clique máxima* é a clique de maior cardinalidade do grafo, ou seja, que possui o maior número de vértices. O número de vértices da clique máxima de um grafo G é denotado por $\omega(G)$.

Seja c uma T-coloração de G de modo que o c -span de G é $sp_T(G)$. Uma vez que $\omega(G) = \omega$, segue que G contém um subgrafo completo H de ordem ω . Assim,

$$sp_T(K_\omega) \leq \max_{u,w \in V(H)} |c(u) - c(w)|,$$

ou seja, o $sp_T(K_\omega)$ é menor ou igual ao maior valor de $|c(u) - c(w)|$ para cada dois vértices $u, w \in V(H)$.

Mas $\max_{u,w \in V(H)} |c(u) - c(w)| = \max_{u,w \in V(G)} |c(u) - c(w)|$, ou seja, o maior valor de $|c(u) - c(w)|$ de H e G é o mesmo, para os vértices $u, w \in V(G)$.

Reescrevendo a inequação, têm-se:

$$sp_T(K_\omega) \leq \max_{u,w \in V(H)} |c(u) - c(w)| = sp_T(G).$$

Logo, têm-se que $sp_T(K_\omega) \leq sp_T(G)$.

Agora, será estabelecida a segunda inequação. Seja c uma T-coloração de K_k usando as cores r_1, r_2, \dots, r_k onde $\forall r_i, 1 \leq r_i \leq sp_T(K_k) + 1$ de modo que $r_1 < r_2 < \dots < r_k$. Uma vez que $\chi(G) = k$, também existe um k -coloring c' de G usando as cores r_1, r_2, \dots, r_k .

Já que c é uma T-coloração de K_k , segue que $|r_i - r_j| \notin T$ para cada par i, j de inteiros com $1 \leq i, j \leq k$ e $i \neq j$. Se c' utiliza as mesmas cores de c , ele também é uma T-coloração de G e $sp_T(G) \leq sp_T(K_k)$.

Portanto, chega-se a $sp_T(K_\omega) \leq sp_T(G) \leq sp_T(K_k)$.

5.1.2 Distance coloring

O problema de coloração envolvendo distâncias com restrições nas arestas, ou seja, o distance coloring, também possui alguns teoremas interessantes que foram discutidos e detalhados ao longo deste PIBIC. O foco destes teoremas é caracterizar casos de factibilidade para os problemas apresentados, ou seja, situações em que sempre haja uma solução

ou o contrário.

Os teoremas seguintes são apresentados no artigo do doutorando Bruno Dias e da orientadora do projeto, professora Rosiane de Freitas [7], e foram estudados e redetalhados para este projeto.

Teorema 8. *Um grafo G admite solução para o problema de ECDGP-Constante, ou seja, quando as distâncias entre os vértices são sempre iguais a uma constante, se e somente se G é um grafo bipartido.*

Prova. Seja G um grafo submetido ao problema ECDGP constante, onde para cada aresta $v_i v_j \in E$, a distância $d_{ij} = \varphi$, $\varphi \in \mathbb{N}$, constante. Supondo que G admite solução para o problema, seja $x(i)$ a cor atribuída ao vértice $v_i \in V$.

Escolhendo um caminho arbitrário v_1, v_2, \dots, v_k de G , não necessariamente simples. Então $|x(i) - x(j)| = \varphi$, para $|i - j| = 1$ (ou seja, vértices adjacentes). Isto implica que $x(i) = x(i + 2)$, $i = 1, 2, \dots, k - 2$. Portanto, este grafo é 2-colorível. Além disso, se o caminho contém o mesmo vértice v_i duas vezes, seus índices correspondentes são os mesmos. Isto é, o comprimento do caminho, contado deste a primeira ocorrência do vértice v_i até a segunda, é necessariamente par, significando que existe um ciclo par no grafo. Se G é um grafo 2-colorível e não possui ciclo ímpar, G é um grafo bipartido.

Reciprocamente, se G é um grafo bipartido, os vértices admitem uma coloração adequada com duas cores distintas. Atribuindo o valor $x(i)$ para os vértices da primeira cor, e o valor $\varphi + 1$ para os da segunda, então $|x(i) - x(j)| = \varphi$, para cada aresta $v_i v_j$ de G . Portanto, admite solução para o caso de ECDGP constante.

O interessante deste teorema é que este resultado reúne os teoremas 4 e 5, apresentados no capítulo anterior. O primeiro define que, se G é um grafo 2-colorível, sempre admite solução para ECDGP constante usando duas cores. O segundo aponta que, quando G contém um ciclo ímpar de comprimento maior ou igual a 3, não há solução para ECDGP constante. Ser 2-colorível e não possuir ciclo ímpar são atributos de um grafo bipartido, foco deste teorema.

Teorema 9. *Seja $G = (V, E, d)$ uma árvore, onde $\forall (i, j) \in E$, d_{ij} é um inteiro positivo arbitrário. Então G sempre admite solução para ECDGP.*

Prova. Para provar este algoritmo, será descrito um algoritmo simples para atribuição de cores que satisfará o problema ECDGP.

Inicialmente, todos os vértices estão desmarcados. Escolhendo um vértice arbitrário v_i , atribui-se qualquer valor inteiro positivo $x(v_i)$ para v_i . Então, v_i é marcado como visitado. Depois, escolhendo um vértice não marcado v_j , adjacente a algum vértice marcado v_k . Atribui-se o valor $x(v_j) = d_{jk} + x(v_k)$ e, após, marcar o vértice v_j . Repetir até que todos os vértices estejam marcados.

Este algoritmo garante uma solução para o problema ECDGP que pode ser encontrada em tempo linear. Porém, não é uma solução ótima, ou seja, a solução que use o mínimo de cores possível. A Figura 5.2 ilustra este algoritmo.

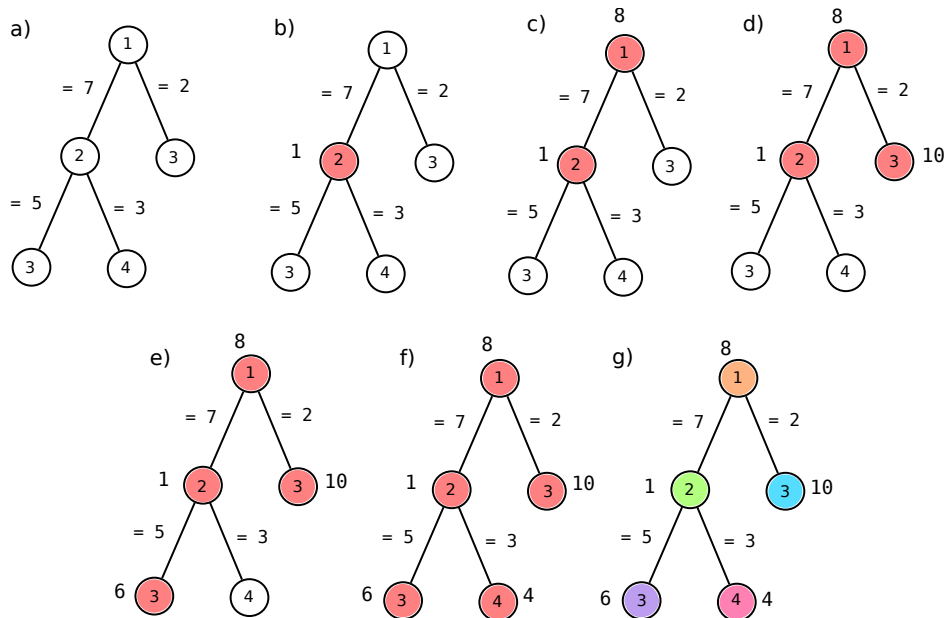


Figura 5.2: Exemplo da execução do algoritmo usado no teorema 9 em uma árvore. Em a) está um grafo que deve ser colorido segundo as regras do ECDGP. Nesta situação, todos os vértices estão desmarcados. Em b), o vértice 2 foi escolhido para receber a cor 1 e foi marcado como visitado. A seguir, escolhe-se um vértice não visitado adjacente a algum vértice marcado v_k e atribui a ele a cor $x(v_j) = d_{jk} + x(v_k)$. Após, marca-se o vértice v_j como visitado. Assim, em c), d), e) e f), os vértices recebem suas respectivas cores. Em g) está o resultado final: uma coloração que respeita o ECDGP, porém usando 5 cores, o que não é uma solução ótima.

5.2 Algoritmos

Neste projeto, foram implementados algoritmos para solucionar o problema do distance coloring. Para isso, entretanto, foram implementados antes os algoritmos relativos à coloração clássica, para ambientar a aluno à estrutura de um grafo e à estratégia heurística para resolução de problemas.

A variação escolhida para implementação foi o MCDGP, ou seja, um grafo $G = (V, E)$ deve ser colorido de tal forma que, para cada aresta $(i, j) \in E$, $|c(i) - c(j)| \geq d_{ij}$, onde d_{ij} é a distância entre os vértices (ou seja, o peso das arestas) e $c(i)$ é a cor atribuída a um vértice v_i . A seguir, as estratégias de coloração, adaptadas para o caso com distâncias, são apresentadas, além dos resultados dos testes em algumas instâncias da literatura.

O desafio resume-se no seguinte: dado um grafo, como atribuir cores aos vértices, sendo que vértices adjacentes não podem ter a mesma cor? Além disso, no caso da coloração com distâncias, vértices adjacentes devem receber cores de forma que o módulo da diferença de seus rótulos seja maior ou igual ao peso da aresta.

Para solucionar o problema, é importante determinar algum tipo de ordenação, ou seja, escolher uma sequência de vértices a serem pintados. Neste contexto, foram escolhidas três abordagens:

1. sequência arbitrária dos vértices;
2. ordenando os vértices em ordem decrescente de acordo com o grau;
3. usando o conceito de grau de saturação para uma ordenação decrescente.

A escolha da ordem em que os vértices serão coloridos é importante para a chamada *heurística gulosa*. Nesta estratégia, dado um grafo G e uma sequência de vértices $V = \{v_1, v_2, \dots, v_k\}$, para cada vértice $v_i \in V$ é atribuída a menor cor possível [18]. Este método não garante que o mínimo de cores será utilizado, sendo muito afetado pela forma com que os vértices a serem pintados são escolhidos. Por isso, é importante ter uma estratégia para a ordenação dos vértices.

Tendo estes conceitos como base, implementou-se um algoritmo para cada abordagem de escolha da ordem os vértices.

5.2.1 Ordem arbitrária de vértices

Para colorir um grafo utilizando uma sequência arbitrária para escolha dos vértices, são utilizados os seguintes passos:

1. o primeiro vértice recebe a primeira cor;
2. para os vértices restantes, considere-se o vértice atual;
3. atribui a esse vértice a menor cor possível, de modo que não inflija a regra;
4. volta para o passo 2 e repete o processo até colorir todos os vértices do grafo.

A Figura 5.3 apresenta um exemplo de coloração onde os vértices são coloridos sequencialmente, seguindo os passos descritos.

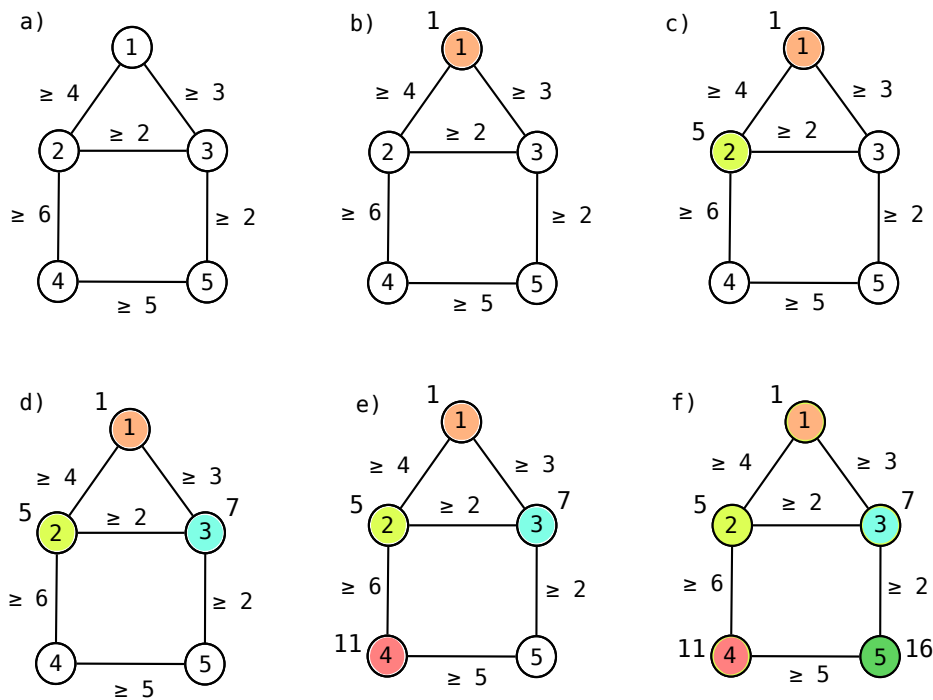


Figura 5.3: Exemplo de coloração escolhendo vértices sequencialmente. Em a), têm-se o grafo a ser colorido. Em b), o primeiro vértice recebe a cor 1. Em c), o vértice 2 recebe a menor cor possível (cor 5). O procedimento se repete em d) e e) atribuindo sempre a menor cor possível ao vértice que está sendo colorido. Isto termina em f), quando o vértice 5 recebe a cor 16 e o grafo está totalmente colorido.

O pseudocódigo deste algoritmo está descrito no Algoritmo 1. Ele tem como entrada um grafo G com n vértices e inicia colorindo o vértice de índice 1. Os próximos vértices são

escolhidos de forma sequencial (ou seja, vértice 2, depois 3, 4, ..., n). Após escolher um vértice, o mesmo recebe a menor cor possível e o processo se repete.

Algoritmo 1: Coloração gulosa com ordem arbitrária de vértices ($G = (V,E)$)

```

início
  Vértice de índice 1 recebe a cor 1;
  para cada  $v \in V(G)$  faça
     $cor \leftarrow 1$ ;
    enquanto Não atribuiu cor ao vértice v faça
      para cada  $w$  vizinho de  $v$  faça
        se  $w$  já está colorido então
          se  $|cor - c(w)| < d_{vw}$  então
            | Sinalizar que a cor não pode ser usada;
          fim se
        fim se
      fim para cada
      se  $cor$  pode ser atribuída a  $v$  então
        |  $c(v) \leftarrow cor$ ;
      fim se
      senão
        |  $cor \leftarrow cor + 1$ ;
      fim se
    fim enquanto
  fim para cada
fim

```

5.2.2 Ordem decrescente do grau dos vértices

Para garantir um melhor desempenho do método de coloração gulosa, é importante utilizar uma estratégia para escolha dos vértices. Uma delas é ordenar os vértices de acordo com o grau, desde o de maior grau até o de menor. Essa estratégia é conhecida como algoritmo de Welsh-Powell [16] e possui os passos a seguir, exemplificados na Figura 5.4.

1. Ordenar os vértices v_1, v_2, \dots, v_n em ordem decrescente de grau (para vértices de mesmo grau, aquele que tiver maior índice é escolhido como tendo maior grau);
2. o vértice de maior grau recebe a cor 1;
3. para os vértices restantes, cada vértice recebe a menor cor disponível, de modo que não haja conflito com as cores de seus vértices adjacentes;

4. o processo se repete até que todos os vértices estejam coloridos.

A Figura 5.4 apresenta um exemplo da execução destes passos em uma instância de grafo.

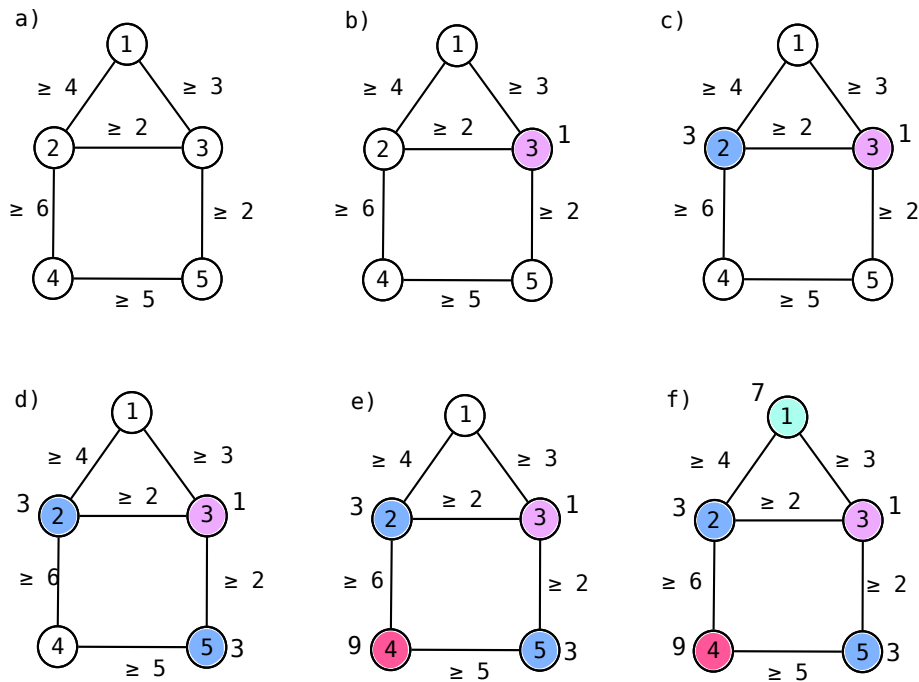


Figura 5.4: Exemplo de coloração ordenando os vértices pelo grau. Em a), têm-se o grafo inicial, ainda sem nenhuma cor. Em b), ocorre um empate: os vértices 2 e 3 são candidatos à coloração, pois possuem o maior grau. O vértice 3 é escolhido por ter o maior índice, recebendo a cor 1. Em c), o vértice 2 recebe a cor 3. Em d), novamente há um empate, agora entre os vértices 1, 4 e 5. De modo análogo à situação anterior, o vértice 5 é escolhido por ser o de maior índice e recebe a cor 3. Em d), o vértice 4 recebe a cor 9 e, finalmente, em f) o vértice 1 recebe a cor 7, finalizando a coloração.

O pseudocódigo do algoritmo de Welsh-Powell está descrito no Algoritmo 2. Tal como o anterior, ele recebe como entrada um grafo G e executa a sequência de passos descrita anteriormente: primeiro, é feita a ordenação dos vértices por ordem decrescente de grau. Em caso de empate, ou seja, quando dois vértices tiverem o mesmo grau, aquele que tiver maior índice é escolhido. Após esta nova sequência, o vértice de maior grau recebe a primeira cor. Os demais vértices recebem a menor cor possível, de modo que a regra do MCDGP seja respeitada. O processo termina quando o último vértice, ou seja, o

de menor grau, recebe uma cor.

Algoritmo 2: Coloração gulosa com ordem decrescente do grau ($G = (V,E)$)

```
início
  Os vértices são ordenados em ordem decrescente de grau;
  O vértice de maior grau recebe a cor 1;
  para cada  $v \in V(G)$  faça
     $cor \leftarrow 1$ ;
    enquanto Não atribuiu cor ao vértice v faça
      para cada  $w$  vizinho de  $v$  faça
        se  $w$  já está colorido então
          se  $|cor - c(w)| < d_{vw}$  então
            Sinalizar que a cor não pode ser usada;
          fim se
        fim se
      fim para cada
      se cor pode ser atribuída a  $v$  então
         $c(v) \leftarrow cor$ ;
      fim se
      senão
         $cor \leftarrow cor + 1$ ;
      fim se
    fim enquanto
  fim para cada
fim
```

5.2.3 Ordem pelo grau de saturação

Uma estratégia mais eficaz para ordenar os vértices é chamada DSATUR (sigla para *Degree of Saturation*). Este algoritmo heurístico de abordagem gulosa leva em conta o grau de saturação dos vértices, sendo originalmente proposto por Daniel Brélaç em 1979. O grau de saturação de um vértice é a quantidade de vértices adjacentes a ele que já receberam uma cor. A sequência de passos do DSATUR é a seguinte [13]:

1. primeiramente, os vértices são ordenados por ordem decrescente de grau;
2. o vértice de maior grau é inicialmente colorido com a cor 1;
3. na próxima iteração, seleciona-se o vértice de maior grau de saturação. Caso haja empate, escolhe-se o vértice de maior grau. Persistindo esta situação, a escolha é

feita pelo vértice de menor índice. Por fim, o vértice escolhido recebe a menor cor disponível, tal que essa cor não conflite com seus vértices adjacentes;

4. repete-se a iteração até que todos os vértices estejam coloridos.

A Figura 5.5 traz um exemplo da execução do algoritmo DSATUR em um grafo, seguindo os passos descritos anteriormente.

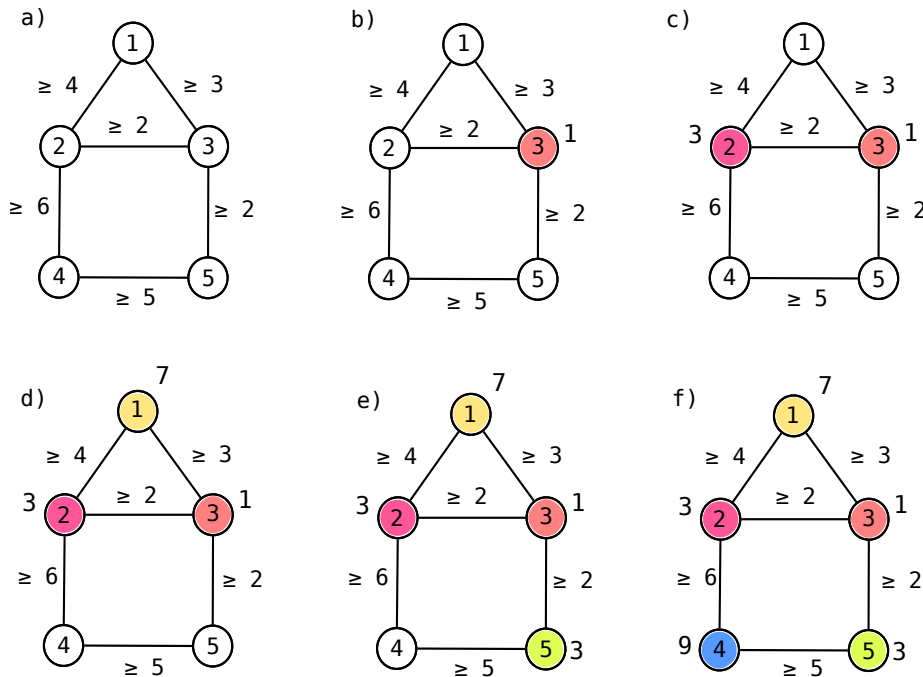


Figura 5.5: Exemplo de coloração ordenando os vértices pelo grau de saturação. Em a), têm-se o grafo inicial, ainda sem nenhuma cor. Em b), o vértice de maior grau recebe a cor 1 (no caso, o vértice 3, por ter índice maior que o vértice 2). Em c), há empate ao determinar quem possui maior grau de saturação. O vértice 2 é escolhido por ter o maior grau dentre os vértices empatados, recebendo assim a cor 3. Em d), o vértice de maior saturação (vértice 1) recebe a cor 7. Já em e) ocorre empate novamente. Como ambos possuem o mesmo grau, o vértice 5 é escolhido por ter maior índice, recendo a cor 3. Por fim, o vértice 4 recebe a cor 9, em f), encerrando o processo.

A sequência de passos descrita é apresentada no Algoritmo 3 em pseudocódigo. Tal como os outros algoritmos apresentados anteriormente, ele recebe como entrada um grafo G a ser colorido. O primeiro passo é ordenar os vértices pelo grau, de modo decrescente. Após, o vértice de maior grau recebe a primeira cor. O próximo a ser colorido é o vértice de maior grau de saturação, ou seja, aquele que é adjacente a mais vértices coloridos. Caso ocorra empate neste critério, escolhe-se o vértice de maior grau. Persistindo, aquele

que tiver maior índice é escolhido. Uma vez escolhido o vértice, é atribuída a menor cor possível a ele. Então, o grau de saturação é recalculado, outro vértice é escolhido e o procedimento segue até que o grafo esteja totalmente colorido.

Algoritmo 3: Coloração gulosa DSATUR ($G = (V,E)$)

```

início
  Ordenar os vértices  $V(G)$  em ordem decrescente de grau;
  Vértice de maior grau recebe a cor 1;
  enquanto  $\exists$  vértice descolorido faça
     $v \leftarrow$  maior grau saturacao( $G$ );
    se  $\exists$  mais de um vértice com grau de saturação igual a  $v$  então
      | Escolhe o vértice de maior grau;
    fim se
    cor  $\leftarrow$  1;
    enquanto Não atribuiu cor ao vértice  $v$  faça
      para cada  $w$  vizinho de  $v$  faça
        se  $w$  já está colorido então
          | se  $|cor - c(w)| < d_{vw}$  então
            | | Sinalizar que a cor não pode ser usada;
          fim se
        fim se
      fim para cada
      se cor pode ser atribuída a  $v$  então
        |  $c(v) \leftarrow$  cor;
      fim se
      senão
        | cor  $\leftarrow$  cor + 1;
      fim se
    fim enquanto
  fim enquanto
fim

```

5.2.4 Testes computacionais

Os algoritmos implementados para o distance coloring foram testados em instâncias da literatura que descrevem um grafo com pesos nas arestas. A implementação ocorreu em C++ e a execução, em uma máquina DELL Optiplex 9020, com processador Intel Core i7 4770-U (@ 3.4 GHz), 8 GB de memória RAM e usando sistema operacional Ubuntu Gnome 15.04. As instâncias utilizadas foram criadas por Michael Trick e estão disponíveis na página do mesmo [23]. Elas são de três tipos: GEOM n (grafos esparsos),

GEOM na e GEOM nb (onde n é o número de vértices da instância), ambos trazendo grafos densos, porém as instâncias GEOM nb precisam de menos cores. A Tabela 5.1 apresenta os resultados.

Tabela 5.1: Resultados dos testes, onde os algoritmos implementados foram aplicados a instâncias do conjunto GEOM.

Instância	Num.Vert.	Estratégia utilizada					
		Arbitrário		Decrescente		DSATUR	
		Num. cores	Tempo (ms)	Num. cores	Tempo (ms)	Num. cores	Tempo (ms)
GEOM20	20	10	0.003	10	0.01	10	0.02
GEOM20a		13	0.003	11	0.017	11	0.025
GEOM20b		13	0.002	10	0.046	12	0.02
GEOM30	30	14	0.003	17	0.023	15	0.152
GEOM30a		18	0.005	22	0.02	16	0.041
GEOM30b		20	0.002	19	0.077	17	0.033
GEOM40	40	17	0.018	21	0.071	22	0.274
GEOM40a		27	0.008	27	0.049	26	0.155
GEOM40b		27	0.013	24	0.035	24	0.067
GEOM50	50	21	0.013	24	0.032	23	0.096
GEOM50a		35	0.019	35	0.164	32	0.115
GEOM50b		34	0.013	31	0.076	30	0.093
GEOM60	60	28	0.009	28	0.115	31	0.108
GEOM60a		41	0.019	38	0.073	38	0.13
GEOM60b		41	0.03	37	0.066	36	0.136
GEOM70	70	32	0.013	32	0.059	31	0.239
GEOM70a		45	0.033	48	0.088	42	0.318
GEOM70b		46	0.048	44	0.087	41	0.181
GEOM80	80	39	0.055	34	0.142	40	0.191
GEOM80a		50	0.038	49	0.116	51	0.437
GEOM80b		53	0.039	48	0.117	47	0.364
GEOM90	90	43	0.027	39	0.122	42	0.236
GEOM90a		55	0.042	56	0.303	54	0.498
GEOM90b		61	0.053	57	0.155	58	0.388
GEOM100	100	46	0.048	47	0.104	47	0.366
GEOM100a		62	0.096	65	0.277	59	1.569
GEOM100b		67	0.216	62	0.183	63	0.519
GEOM110	110	48	0.13	51	0.132	47	0.365
GEOM110a		66	0.109	67	0.38	67	0.639
GEOM110b		75	0.079	68	0.225	62	0.701
GEOM120	120	52	0.043	58	0.137	52	0.911
GEOM120a		72	0.092	71	0.259	71	3.249
GEOM120b		82	0.1	70	0.423	73	0.787

Analisando os valores apresentados, é interessante verificar o desempenho dos algoritmos para cada instância. Naquelas que traziam grafos esparsos, as três estratégias tiveram desempenhos semelhantes. Porém, no caso dos grafos densos, o algoritmo DSA-TUR trouxe bons resultados, em sua maioria mais eficazes que os outros dois métodos. A escolha arbitrária de vértices obteve o desempenho menos satisfatório neste caso, especialmente nas instâncias *GEOM nb* , que necessitam de menos cores para serem coloridas. Estes dados reforçam que, ao adotar uma heurística gulosa, é importante ter uma estratégia para ordenação dos vértices que serão coloridos.

Capítulo 6

Considerações finais

Este relatório final apresentou o conteúdo de grafos rotulados, grafos graciosos e alguns problemas de coloração especiais em grafos. Em um primeiro momento, foram expostas informações como objetivos, metodologia e o cronograma de atividades. O capítulo seguinte trouxe o referencial teórico necessário para iniciar o estudo. A seguir, as classes de grafos de interesse para este projeto foram apresentadas, tais como os grafos rotulados e suas variações (grafos graciosos e coloração de grafos). Em especial, as definições e teoremas relacionados ao problema de coloração com distâncias foram abordadas, pois constituem um grande foco desta iniciação científica. O último capítulo trouxe os resultados do projeto, desenvolvido ao longo de agosto de 2014 – julho de 2015.

Dentre os resultados, encontram-se o detalhamento de teoremas relacionados aos problemas de coloração em estudo e a implementação de algoritmos envolvendo o problema de coloração com distâncias. Ambos foram resultados muito proveitosos, pois forneceram um ótimo contato com o tema, tanto com a parte teórica quanto a mais prática (implementacionalmente falando). Soma-se a isso o fato de que a presente bolsista está cursando o segundo ano de graduação, ou seja, foi uma experiência muito enriquecedora no sentido acadêmico.

Os planos futuros envolvem dar continuidade ao projeto, seguindo a mesma linha de estudo (grafos graciosos e problemas de coloração), buscando relacionar estes dois conceitos com mais profundidade, propondo novas caracterizações para estes problemas. Além disso, pretende-se também divulgar o projeto em eventos científicos e iniciar os preparativos para a apresentação no Congresso de Iniciação Científica da UFAM (CONIC 2015), que é o passo final para a conclusão do projeto atual.

Referências

- [1] Victória Patrícia Silva Aires, Rosiane de Freitas, and Mário Salvatierra. Grafos mágicos, jogos matemáticos e algoritmos. Relatório final de pibic jr, IComp - UFAM, 2014.
- [2] Florian Arnold. Totally magic graphs - a complete search on small graphs. Master's thesis, Clausthal University of Technology, 2012.
- [3] Karla Cristiane Arsie. Jogos sudoku e quadrado mágico. Master's thesis, Universidade Federal do Paraná, 2010.
- [4] Charles Michael Cavalier. Graceful labelings. Master's thesis, University of South Carolina, 2009.
- [5] Gary Chartrand and Ping Zhang. *Chromatic Graph Coloring*. CRC Press, 2009.
- [6] Rosiane de Freitas. Distance coloring problems, spatial properties and feasibility conditions. In *LAWCG 2014*.
- [7] Bruno Dias, Rosiane de Freitas, Nelson Maculan, Philippe Michelon, and Jayme Szwarcfiter. Distance geometry approach for special graph coloring problems. 2014.
- [8] Bruno Raphael Cardoso Dias. Otimização da alocação de canais em redes móveis sem fio celulares e cognitivas. Master's thesis, Universidade Federal do Amazonas, 2014.
- [9] Bruno Raphael Cardoso Dias. *Otimização da alocação de canais em redes sem fio*. PhD thesis, Universidade Federal do Amazonas, 2014.
- [10] Bruno Raphael Cardoso Dias, Rosiane de Freitas Rodrigues, and Nelson Maculan Filho. Alocação de canais em redes celulares sem fio: algoritmos e modelos teóricos

- em grafos e escalonamento. In *SBPO - Simpósio Brasileiro de Pesquisas Operacionais*, 2012.
- [11] P. Feofiloff, Y. Kohayakawa, and Y. Wakabayashi. *Uma Introdução Sucinta à Teoria dos Grafos*. Universidade de São Paulo, 2011.
- [12] Joseph A. Gallian. A dynamic survey of graph labeling. *The Electronic Journal of Combinatorics*, 2011.
- [13] Simone Ingrid Monteiro Gama. Sobre o problema da lista coloração em grafos, 2014.
- [14] Anamaria Gomide and Jorge Stolfi. Elementos de matemática discreta para computação, 2011.
- [15] Willian K. Hale. Frequency assignment: Theory and applications. 1980.
- [16] Anthony Henderson. *Introduction to Graph Theory*. 2014.
- [17] Samuel Jurkiewiza. Grafos - uma introdução. In *OBMEP*.
- [18] Adrian Kosowski and Krzysztof Manuszewski. Classical coloring of graphs. 2004.
- [19] Igor Carboni Oliveira. Complexidade computacional e o problema p vs np. Master's thesis, UNICAMP, 2010.
- [20] Fred S. Roberts. T-colorings of graphs: recent results and open problems. *Discrete Mathematics*, 93:229–245, 1991.
- [21] A. Rosa. On certain valuations of the vertices of a graph. *Theory of Graphs (Internat. Symposium, Rome, July 1966)*, 1967.
- [22] Dushyant Tanna. Harmonious labeling of certain graphs. *International Journal of Advanced Engineering Research and Studies*, 2013.
- [23] Michael Trick. Graph coloring and its generalizations. <http://mat.gsia.cmu.edu/COLOR03/>.
- [24] Adrian O. Waller. Some results on list t-colourings. *Discrete Mathematics*, 174:357–363, 1997.