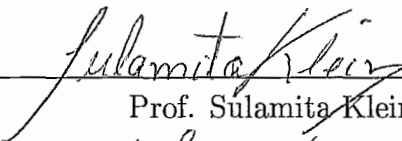


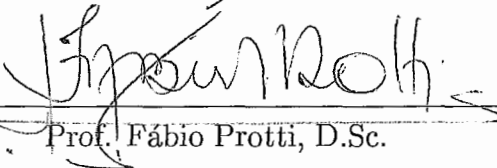
MINIMIZAÇÃO DE MODELOS DE GRAFOS 2-DIR PUROS

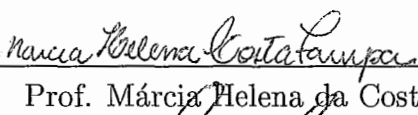
Débora Cristina Alves Rego

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:


Prof. Sulamita Klein, D.Sc


Prof. Fábio Protti, D.Sc.


Prof. Márcia Helena da Costa Fampa, D.Sc.


Prof. Carlos Alberto de Jesus Martinhon, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

OUTUBRO DE 2001

REGO, DÉBORA CRISTINA ALVES

Minimização de modelos de grafos 2-DIR
PUROS [Rio de Janeiro] 2001

X, 70 p. 29,7 cm (COPPE/UFRJ, M.Sc.,
Engenharia de Sistemas e Computação, 2001)

Tese – Universidade Federal do Rio de Ja-
neiro, COPPE

1 - Grafos de Interseção

2 - Grafos 2-DIR PUROS

3 - Grafos de Interseção em Grade

4 - Programação Linear Inteira

5 - Programação Mista

I. COPPE/UFRJ II. Título (série)

Ao meu pai (In Memoriam)

“ E eis o meu segredo, um segredo muito simples. É só com o coração que se pode ver direito; o essencial é invisível aos olhos.”

(*O Pequeno Príncipe* , Saint-Exupéry)

Agradecimentos

A Deus.

A minha querida mãe e a minha caríssima irmã Daniela, pelo amor e incentivo e que se constituíram em apoio essencial durante todos os momentos de minha vida.

A todos os meus parentes, que, de perto ou de longe, de forma manifesta ou silenciosa, estiveram a me desejar sucesso nesta empreitada.

Aos meus orientadores Sulamita Klein e Fábio Protti, pela generosidade, atenção, pelos sábios conselhos e pela oportunidade de conviver com eles, aprendendo a partir de seus exemplos.

Aos professores Carlos Alberto de Jesus Martinhon e Márcia Fampa pela leitura cuidadosa desta tese. Em particular, a professora Márcia pela valiosa contribuição no Capítulo 5, auxiliando-me a construir o modelo de programação inteira mista.

A professora Claudina Izepe Rodrigues, exemplo de profissional e de amiga, meu agradecimento e minha homenagem.

Agradeço em especial a Carlile, que desde da graduação me apoia e incentiva nos caminhos da pesquisa, pelo carinho, amizade e respeito incondicional.

Agradeço as minhas amigas Loana e Rosa, por todas as risadas e momentos de descontração. Ao amigo Marcelo Massaharu pela amizade que nos acompanha desde a época da escola.

Ao amigo Tiberius pela paciente ajuda no XPress e ao amigo Flávio pela contribuição na fase de edição da tese, pois as dúvidas de Latex eram inesgotáveis.

A todos os amigos, próximos e distantes, recentes e de longas datas. A tantos outros que não pude mencionar, que colaboraram direta ou indiretamente, seja com atitudes, com uma palavra amiga, com um sorriso, com um pensamento ou uma prece. Muito obrigada!

A todos os colegas de estudos, funcionários e professores do Programa.

A CAPES pela bolsa de estudos concedidas durante o mestrado.

A Universidade Estadual de Campinas e a Universidade Federal do Rio de Janeiro.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

MINIMIZAÇÃO DE MODELOS DE GRAFOS 2-DIR PUROS

Débora Cristina Alves Rego

Outubro/2001

Orientadores: Sulamita Klein e Fábio Protti

Programa: Engenharia de Sistemas e Computação

Nesta tese consideramos a classe dos grafos 2-DIR PUROS, também conhecidos na literatura como *grafos de interseção em grade*. Sejam I e J famílias finitas de segmentos horizontais e verticais no plano, onde segmentos na mesma direção não se interceptam. O grafo de interseção G de $I \cup J$ é conhecido como grafo 2-DIR PURO. É claro que G é um grafo bipartido $G = (L, N)$, onde os conjuntos L e N correspondem aos segmentos de I e J , respectivamente.

O primeiro resultado desta tese é um Teorema de Caracterização para grafos 2-DIR PUROS através de uma ordenação apropriada dos vértices em cada conjunto da bipartição. Esta caracterização induz a definição de modelo de *grade*: um modelo de interseção de segmentos sobre a grade formada pelas interseções das retas $x = i$, $i = 1, \dots, s$ e $y = k$, $k = 1, \dots, r$. Mostramos que todo grafo 2-DIR PURO $G = (L, N)$ de grau mínimo 2 com $|L| = r$ e $|N| = s$ tem um modelo sobre a grade $s \times r$.

A seguir, tratamos do problema de minimização de modelos de grade, isto é, obtenção de modelos onde a soma dos comprimentos dos segmentos que representam os vértices é mínima. Mostramos que este problema, na sua versão de decisão, é NP-completo.

Finalmente, apresentaremos um modelo de programação inteira mista que reconhece se um determinado grafo G bipartido pertence ou não à classe de grafos 2-DIR PUROS e minimiza seu modelo de grade, se existir.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

MINIMIZING MODELS OF PURE-2-DIR GRAPHS

Débora Cristina Alves Rego

October/2001

Advisors: Sulamita Klein and Fábio Protti

Department: Computing and Systems Engineering

This thesis considers the class of PURE-2-DIR graphs, also known in the literature as *grid intersection graphs*. Let I and J be finite families of horizontal and vertical linear segments in the plane, such that no two horizontal or vertical segments intersect. The intersection graph G of $I \cup J$ is called a PURE-2-DIR graph. It is clear that G is a bipartite graph $G = (L, N)$, where the sets L and N correspond to the segments in I and J , respectively.

The first result of this thesis is a Characterization Theorem for PURE-2-DIR graphs in terms of orderings in each partition of the vertex set. This characterization induces the definition of *grid model*: a intersection model of segments placed on the grid formed by the intersections of the lines $x = i$, $i = 1, \dots, s$ e $y = k$, $k = 1, \dots, r$. It is shown that every PURE-2-DIR $G = (L, N)$ of minimum degree two with $|L| = r$ and $|N| = s$ has a model on the grid $s \times r$.

Next, we deal with the problem of minimizing grid models, that is, obtaining models where the sum of the lengths of the segments representing the vertices is minimum. It is shown that this problem, in its decision version, is NP-complete.

Finally, it is presented a mixed integer programming model that recognizes whether a given bipartite graph G belongs or not to the classe of PURE-2-DIR graphs, and minimizes its grid model, if it exists.

Sumário

| | | |
|----------|--|-----------|
| 1 | Introdução | 1 |
| 1.1 | Definições e Notações Básicas | 4 |
| 1.1.1 | Grafos | 4 |
| 1.1.2 | Grafos de Interseção | 7 |
| 1.1.3 | Classes de Complexidade | 8 |
| 1.1.4 | Programação Linear | 10 |
| 2 | Classes de Grafos de Interseção de Segmentos Lineares | 15 |
| 2.1 | Classes de Grafos de Interseção | 15 |
| 2.1.1 | O Problema INEQ | 20 |
| 3 | A Classe de Grafos 2-DIR PUROS | 22 |
| 3.1 | Primeiras Considerações | 22 |
| 3.1.1 | Modelos de Grafos 2-DIR PUROS | 24 |
| 3.2 | Uma Caracterização de Grafos 2-DIR PUROS | 26 |
| 3.3 | Restringindo o Universo de Grafos 2-DIR PUROS | 32 |

| | | |
|----------|---|-----------|
| 4 | Definindo o Problema de Otimização | 34 |
| 4.1 | Prova de NP-Compleitude | 38 |
| 5 | Um Modelo de Programação Linear Inteira Para Resolver o Problema GRADE | 40 |
| 5.1 | Modelagem das Famílias | 43 |
| 5.1.1 | Família $\mathcal{F}(1)$ | 44 |
| 5.1.2 | Família $\mathcal{F}(2)$ | 46 |
| 5.1.3 | Família $\mathcal{F}(3)$ | 52 |
| 5.1.4 | Família $\mathcal{F}(4)$ | 54 |
| 5.2 | Corretude da Formulação | 56 |
| 5.3 | Número de Restrições e Variáveis na Formulação | 64 |
| 6 | Conclusões | 66 |

Lista de Figuras

| | | |
|-----|--|----|
| 2.1 | Representações 2-DIR e 3-DIR PURO do K_3 | 18 |
| 2.2 | Representação 2-DIR PURO do $K_{3,3}$ | 18 |
| 2.3 | Inclusões entre as classes | 19 |
| 3.1 | Exemplo de grafo bipartido que não pertence à classe 2-DIR PURO | 24 |
| 3.2 | Grafo G | 31 |
| 3.3 | Modelo 2-DIR PURO do grafo G | 31 |
| 5.1 | Grafo G | 41 |
| 5.2 | Segmento associado ao vértice $v \in L$ | 43 |
| 5.3 | Segmento associado ao vértice $w \in N$ | 44 |
| 5.4 | Família $\mathcal{F}(1)$ | 45 |
| 5.5 | Família $\mathcal{F}(2)$ | 46 |
| 5.6 | Situação $\alpha(v, w)$ | 47 |
| 5.7 | Situação $\alpha(w, v)$ | 47 |
| 5.8 | Situação $\beta(v, w)$ | 50 |

| | |
|--|----|
| 5.9 Situação $\beta(w, v)$ | 51 |
| 5.10 Situações $\gamma(v, w)$ e $\gamma(w, v)$ | 53 |
| 5.11 Situações $\tau(v, w)$ e $\tau(w, v)$ | 56 |
| 5.12 Esquema da demonstração | 61 |

Capítulo 1

Introdução

Nesta tese consideramos a classe dos grafos 2-DIR PUROS [13], também conhecidos na literatura como *grafos de interseção em grade* [9].

Sejam I e J famílias finitas de segmentos horizontais e verticais no plano, onde segmentos na mesma direção não se interceptam. O grafo de interseção G de $I \cup J$ é conhecido como grafo 2-DIR PURO. É claro que G é um grafo bipartido $G = (L, N)$, onde os conjuntos L e N correspondem aos segmentos de I e J , respectivamente.

Nesta tese apresentaremos uma caracterização de grafos 2-DIR PUROS através de uma ordenação especial dos vértices de L e N . Esta caracterização induz a definição de modelo de *grade*, como veremos.

A classe de grafos 2-DIR PUROS é uma importante subclasse dos grafos de interseção de segmentos lineares no plano, os quais têm aplicações em áreas como otimização de tráfego aéreo [6], modelos VLSI e geometria com-

putacional [4]. O artigo [13] de Kratochvíl e Matoušek contém um estudo aprofundado da teoria dos grafos 2-DIR PUROS.

Segue breve histórico da classe dos grafos 2-DIR PUROS. Kratochvíl [11] provou que o problema de reconhecimento dos grafos 2-DIR PUROS é NP-completo. Hartman, Newman e Ziv [9] provaram que todo grafo planar bipartido é um grafo 2-DIR PURO. Além disso, apresentaram famílias infinitas de grafos bipartidos que não são grafos 2-DIR PUROS. Em um trabalho posterior, Bellantoni, Hartman, Przytycka e Whitesides [1] provaram que todo grafo bipartido de *boxicity* 2 (veja a definição de *boxicity* na Seção 3.1) é um grafo 2-DIR PURO.

Este trabalho está organizado da seguinte maneira. No Capítulo 1 estabeleceremos as notações básicas da teoria dos grafos utilizadas ao longo da tese, e introduziremos grafos de interseção e classes de complexidade. Faremos também uma breve seção sobre programação linear.

No Capítulo 2, trataremos das classes de grafos de interseção de segmentos lineares no plano. Definiremos as classes e exibiremos os principais resultados conhecidos na literatura referentes a cada uma delas. Exibiremos as relações de inclusão existentes entre tais classes. É importante salientar que, neste capítulo, o objeto de estudo desta tese - a classe de grafos 2-DIR PUROS - será introduzido.

No Capítulo 3, será feito um estudo mais aprofundado da classe de grafos

2-DIR PUROS. O t3pico principal deste cap3tulo ser3a o Teorema de Caracteriza3o para esta classe de grafos. Tal teorema 3 uma das mais importantes contribu33es desta tese. Ao longo deste cap3tulo, outros resultados ser3o obtidos, como o Teorema 3.5 - que permitir3 restringir nosso estudo a grafos bipartidos de grau m3nimo maior que 1. Daremos tamb3m a defini3o de *modelo de grade*. E concluiremos que se G 3 um grafo 2-DIR PURO, existe um modelo para G representado sobre a grade de pontos com coordenadas inteiras.

No Cap3tulo 4 trataremos de um problema de otimiza3o onde procuramos minimizar os modelos de grade para G . Denominamos este problema de GRADE. Definiremos $\text{comprimento}(R(G))$, isto 3, uma medida do tamanho do modelo R de um grafo G , e a partir disto definiremos $\text{opt}(G)$ como o valor m3nimo de $\text{comprimento}(R(G))$ dentre todos os modelos de grade de G . Exibiremos um limite inferior e um limite superior para $\text{opt}(G)$. Mostraremos que o problema GRADE 3 NP-completo.

No Cap3tulo 5 iremos propor uma formula3o de programa3o inteira mista para resolver o problema GRADE, provaremos sua corretude e mostraremos o n3mero de restri33es e vari3veis na formula3o

Finalmente, no Cap3tulo 6 faremos nossas conclus3es e sugest3es de trabalhos futuros.

1.1 Definições e Notações Básicas

Vamos agora estabelecer as definições e notações que serão utilizadas ao longo da tese.

1.1.1 Grafos

Um *grafo* é um par ordenado $G = (V, E)$ onde V é um conjunto finito não vazio de *vértices* e E é um conjunto de pares não ordenados de vértices distintos, chamados *arestas*. Utilizaremos a notação $n = |V|$ e $m = |E|$ para denotar a cardinalidade de V e E , respectivamente.

Um par $(v, w) \notin E$, onde $v, w \in V$, é chamado *não-aresta*.

Um grafo G é dito *trivial* se $|V| = 1$.

Um vértice v é *adjacente* a outro vértice w em G se $(v, w) \in E$. Neste caso, dizemos também que v e w são *vizinhos* ou estão *ligados* em G , e que a aresta $e = (v, w)$ é *incidente* a v e w , ou que tem *extremos* v e w . Denotamos por $N(v)$ o conjunto de vértices adjacentes a v em G .

O *grau* de um vértice v é a cardinalidade do conjunto $N(v)$ e é denotado por $\delta(v)$. Denotamos por $\delta(G)$ o menor valor $\delta(v)$ para $v \in V$.

O *tamanho* de um grafo G é igual a $|V| + |E|$.

O *complemento* de um grafo G , denotado por \overline{G} , é o grafo que tem o

mesmo conjunto de vértices de G e tal que dois vértices são vizinhos em \overline{G} se e somente se não são vizinhos em G .

Um grafo $H = (V_H, E_H)$ é um *subgrafo* de um grafo $G = (V, E)$ se $V_H \subseteq V$ e $E_H \subseteq E$. Dado um conjunto de vértices $X \subseteq V$, $X \neq \emptyset$, o *subgrafo induzido* por X , denotado por $G[X]$, é o subgrafo H de G tal que $X = V_H$ e E_H é o conjunto das arestas de G que têm ambos os extremos em X .

Dois grafos G e H são *isomorfos* quando existe uma bijeção entre V e V_H que preserva adjacência. Nesse caso, escrevemos $G \simeq H$.

Um *caminho* num grafo G é uma sequência $P = v_1 v_2 \dots v_k$ onde os v_i 's são vértices (dois a dois distintos), e $(v_i, v_{i+1}) \in E$, $1 \leq i \leq k - 1$. Uma *corda* em P é uma aresta que liga dois vértices não consecutivos de P . Um *caminho induzido* é um caminho sem cordas. Denotamos por P_k o caminho induzido por k vértices.

Um *ciclo* num grafo G é uma sequência $C = v_1 v_2 \dots v_k v_{k+1}$ onde $v_1 v_2 \dots v_k$ é um caminho, $v_1 = v_{k+1}$ e $k \geq 3$. O número k é o *comprimento* de C . Um ciclo é dito *par* se tem comprimento par, caso contrário é dito *ímpar*. Uma *corda* em C é qualquer corda do caminho $v_1 v_2 \dots v_k$. Um ciclo é um *ciclo induzido* se não possui cordas. Denotamos por C_k o ciclo induzido por k vértices. O grafo C_3 é também chamado *triângulo*.

Um grafo G é *conexo* se para todo par de vértices distintos v e w de G existe um caminho de v a w . Caso contrário, G é dito *desconexo*. Um

componente conexo de G é um subgrafo maximal conexo de G . Se H_1 e H_2 são componentes conexos distintos de um grafo desconexo G , dizemos que os grafos formados por H_1 e H_2 são *isolados* um do outro. Em princípio, todos os grafos desta tese são conexos, a menos de menção explícita em contrário, quando for o caso.

Um grafo G é *completo* se quaisquer dois vértices distintos de G são adjacentes. Denotamos por K_n o grafo completo com n vértices.

Um grafo G é dito *bipartido* com bipartição (X, Y) do conjunto de vértices V de G , quando seu conjunto de vértices puder ser particionado em dois subconjuntos X e Y ($X \cup Y = V, X \cap Y = \emptyset, |X|, |Y| \neq \emptyset$) tais que toda aresta de G tem um extremo em X e outro em Y . Denotamos por $G = (X, Y)$. Um grafo é dito *bipartido completo* se todo vértice de X é adjacente a todo vértice de Y . Denotamos por $K_{m,n}$ o grafo bipartido completo, onde m e n ($m, n \geq 1$) representam, respectivamente, as cardinalidades de X e Y .

Pode-se mostrar que G é bipartido se, e somente se, G não contém ciclos ímpares. Para demonstração deste fato, veja [19].

Um conjunto de vértices I de um grafo G é um *conjunto independente* se $G[I]$ é um grafo sem arestas, o qual chamamos *grafo nulo*.

1.1.2 Grafos de Interseção

Seja $\mathcal{F} = (S_1, S_2, \dots, S_n)$ uma família de conjuntos não vazios. O grafo de interseção de \mathcal{F} é obtido representando cada conjunto $S_i \in \mathcal{F}$ por um vértice i , e conectando dois vértices i, j por uma aresta $e = (i, j)$ se, e somente se, para seus conjuntos correspondentes $S_i, S_j \in \mathcal{F}$, $S_i \cap S_j \neq \emptyset$. Isto é, tomamos um vértice representante para cada conjunto da família, e incluímos uma aresta entre dois vértices se, e somente se, os conjuntos correspondentes se interceptam. Vale a pena observar que qualquer grafo é um grafo de interseção de alguma família de conjuntos [17].

Um grafo G é um *grafo de intervalo* se seus vértices podem ser associados a intervalos da reta de modo que dois intervalos se interceptam se, e somente se, os vértices correspondentes são adjacentes. Em outras palavras, um grafo de intervalo é o grafo de interseção de intervalos na reta real. Quando todos os intervalos têm o mesmo comprimento, dizemos que o grafo é um *grafo de indiferença*.

Muitas vezes, por simplicidade e clareza, utilizaremos letras maiúsculas para designar uma classe de grafos: por INTERVALO, por exemplo, entende-se o conjunto de todos os grafos de intervalo.

1.1.3 Classes de Complexidade

Um problema algorítmico π consiste de um conjunto D de todas as possíveis entradas para o problema, chamado *conjunto de instâncias*, e de uma *questão* Q sobre estas instâncias. Resolver um problema algorítmico é desenvolver um algoritmo cuja entrada é uma instância do problema e cuja saída é uma resposta à questão do problema.

Um problema é dito de *decisão* quando a questão exige uma resposta do tipo SIM ou NÃO. Como exemplo, seja π o seguinte problema: “Dado um grafo G , reconhecer se G é um grafo bipartido.” O conjunto de instâncias de π é obviamente o conjunto de todos os grafos. O problema π pode ser assim esquematizado:

Instância genérica de π : um grafo G .

Questão: G é bipartido?

Fica evidente que o problema π acima é um problema de decisão, em particular, o *problema de reconhecimento*. Resolver π significa elaborar um algoritmo de reconhecimento de grafos bipartidos.

Dizemos que um algoritmo é *polinomial* quando sua complexidade de tempo (medida do número de passos que o algoritmo efetua) é uma função polinomial no tamanho da sua entrada. Os problemas de decisão para os quais existem algoritmos polinomiais constituem a classe P. Tais problemas são chamados *polinomiais*.

Um problema de decisão é *não-determinístico polinomial* quando qualquer instância que produz resposta SIM possui um *certificado* sucinto, isto é, uma comprovação de que a resposta é de fato SIM verificável em tempo polinomial no tamanho da instância. Esta classe de problemas de decisão é a classe NP.

Sejam $\pi_1(D_1, Q_1)$ e $\pi_2(D_2, Q_2)$ dois problemas de decisão. Uma *transformação* ou *redução polinomial* de π_1 em π_2 é uma função $f : D_1 \rightarrow D_2$ tal que as seguintes condições são satisfeitas:

- f pode ser computada em tempo polinomial;
- para toda instância $I \in D_1$, tem-se que I produz resposta SIM para π_1 se e somente se $f(I)$ produz resposta SIM para π_2 .

Um problema de decisão π pertence à classe *NP-completo* quando as seguintes condições são satisfeitas:

- (i) $\pi \in \text{NP}$;
- (ii) para todo problema $\pi' \in \text{NP}$ existe uma transformação polinomial de π' em π . (Notação: $\pi' \propto \pi$)

Um problema pertencente à classe *NP-completo* é chamado *NP-completo*. Para provar que um certo problema π é NP-completo, basta mostrar que

$\pi \in \text{NP}$ e que existe uma transformação de um problema NP-completo π' em π .

Caso somente a condição (ii) da definição de NP-completo seja considerada (não importando se (i) é ou não satisfeita), o problema π é denominado *NP-difícil*. Conseqüentemente, a “dificuldade” de um problema NP-difícil não é menor do que a de um NP-completo.

Existem problemas que são resolvidos em tempo polinomial, e outros que são resolvidos em *espaço polinomial*. Segue que todo problema resolvido em tempo polinomial é resolvido em espaço polinomial. Mas a recíproca não é verdadeira.

A classe *PSPACE* é constituída dos problemas de decisão que são resolvidos em espaço polinomial em função do tamanho da entrada do problema.

Como fonte de referências para esta seção, indicamos [7].

1.1.4 Programação Linear

A programação linear visa encontrar a melhor solução para problemas representados por expressões lineares. Isto consiste na maximização ou minimização de uma função linear, denominada *função objetivo*, respeitando-se um sistema linear de igualdades ou desigualdades que recebem o nome de *restrições* do modelo. Essas restrições do modelo determinam uma região à qual denominamos *conjunto das soluções viáveis*. Se existe um ponto que

satisfaz todas as restrições do problema de programação linear, denominamos tal ponto de *solução viável*. Se um problema não possui solução viável, dizemos que o problema de programação linear é *inviável*. A melhor das soluções viáveis, isto é, aquela que maximiza ou minimiza a função objetivo denomina-se *solução ótima*.

O problema de programação linear será chamado daqui por diante de *PPL*.

O desenvolvimento de um método (ou algoritmo) que determine uma solução de um PPL torna necessário a redução do problema a uma forma tal que permita a aplicação direta deste algoritmo. No caso da programação linear, podemos utilizar algoritmos como o *simplex* e o *método de pontos interiores*.

Um PPL é um problema que pode ser expresso da seguinte forma:

$$\text{Maximizar} \quad z = \sum_{j=1}^n c_j x_j \quad (1.1)$$

$$\text{sujeito a :} \quad \sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, 2, \dots, m \quad (1.2)$$

$$x_j \geq 0, \quad j = 1, 2, \dots, n \quad (1.3)$$

Caso o algoritmo utilizado na resolução do modelo seja o *simplex*, é conveniente alterar o modelo para o formato padrão, que será apresentado a seguir.

Em 1.1, 1.2 e 1.3, c_j , a_{ij} e b_i são dados (números reais) e x_j representa

uma variável de decisão, para $i = 1, 2, \dots, m$ e $j = 1, 2, \dots, n$. A função linear a ser maximizada em 1.1 é denominada *função objetivo*. As restrições 1.3 são conhecidas como *restrições de positividade*.

Cada restrição i de 1.2 pode ser substituída por uma igualdade acrescentando-se uma *variável de folga* $x_{n+i} \geq 0$. Ou seja:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i = \begin{cases} \sum_{j=1}^n a_{ij}x_j + x_{n+i} = b_i, \\ x_{n+i} \geq 0 \end{cases}$$

Sendo um *PPL* com restrições de igualdades e desigualdades, poderemos acrescentar variáveis de folga às desigualdades não triviais, passando dessa maneira a trabalhar com restrições de igualdades e desigualdades triviais. Isto é, as desigualdades podem ser transformadas em igualdades através da introdução de variáveis de folga. Temos então:

$$(PPL) \text{ Maximizar } z = \sum_{j=1}^n c_j x_j \quad (1.4)$$

$$\text{sujeito a : } \sum_{j=1}^n a_{ij} x_j = b_i, \quad i = 1, 2, \dots, m \quad (1.5)$$

$$x_j \geq 0, \quad j = 1, 2, \dots, n. \quad (1.6)$$

que poderá ser ainda apresentado sob a forma abaixo:

$$(PPL) \text{ Maximizar } z = cx \quad (1.7)$$

$$\text{sujeito a : } Ax = b \quad (1.8)$$

$$x \geq 0 \quad (1.9)$$

Onde $c = (c_1 c_2 \dots c_n)$, $x^T = (x_1 x_2 \dots x_n)$, $b^T = (b_1 b_2 \dots b_m)$ e $A = (a_1 a_2 \dots a_n)$, isto é, $c^T \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ e $a_j \in \mathbb{R}^m$ e $A \in \mathbb{R}^{m \times n}$.

Este é o *formato padrão* da programação linear, isto é, qualquer problema de programação linear pode ser formulado como acima.

Os problemas de programação linear estão casados em pares: cada problema corresponde a um problema *dual* (se D é o dual de P, P é o dual de D). Um problema e o seu dual estão intimamente relacionados: ou ambos são inviáveis, ou um é inviável e o outro é ilimitado, ou ambos têm solução. Esse fato é a essência do teorema da dualidade.

Considerando PPL (1.1), associaremos a cada restrição de (1.2) a variável $u_i \geq 0, i = 1, 2, \dots, n$. Segue a formulação do dual:

$$(D) \text{ Minimizar } d = \sum_{i=1}^n b_i u_i \quad (1.10)$$

$$\text{sujeito a : } \sum_{i=1}^n a_{ij} u_i \geq c_j, \quad j = 1, 2, \dots, n \quad (1.11)$$

$$u_i \geq 0, \quad i = 1, 2, \dots, m. \quad (1.12)$$

Um problema de programação linear, onde algumas ou todas as suas variáveis estão restritas a valores inteiros é denominado de *problema de programação linear inteira (PPLI)*. No caso em que algumas variáveis são inteiras mas não todas, podemos chamar este problema de *problema de programação inteira mista (PPIM)*.

Um PPLI pode ser formulado da seguinte maneira:

$$(PPLI) \text{ Maximizar } z = c^T x \quad (1.13)$$

$$\text{sujeito a: } Ax = b \quad (1.14)$$

$$x \geq 0 \quad (1.15)$$

$$x \text{ inteiro} \quad (1.16)$$

Onde $c^T \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ e $a_j \in \mathbb{R}^m$ e $A \in \mathbb{R}^{m \times n}$.

Quando as variáveis estão restritas aos valores inteiros 0 ou 1 em um problema de programação linear, dizemos que temos *problema de programação linear binário*.

Para maiores detalhes ver [2, 14, 15, 20].

Capítulo 2

Classes de Grafos de Interseção de Segmentos Lineares

Grafos de interseção de segmentos lineares (a classe SEG) e de outros objetos geométricos simples têm recebido atenção especial nos últimos anos. As principais dificuldades são: reconhecer se um dado grafo G pertence a uma determinada classe, e obter representações compactas de grafos no plano.

2.1 Classes de Grafos de Interseção

Nesta seção vamos definir as classes de grafos de interseção de segmentos lineares no plano.

Seja \mathcal{C} uma classe de conjuntos (objetos geométricos, neste caso). A classe dos “Grafos de Interseção de \mathcal{C} ” é denotada por $GI(\mathcal{C})$, a qual será a classe de todos os grafos isomorfos aos grafos da forma $G = (V, E)$, onde $V \subseteq \mathcal{C}$ e $e = (u, v) \in E$ se, e somente se, $u \cap v \neq \emptyset$.

Nós usamos uma representação de um dado grafo G , que associa um vértice $v \in V$ a um elemento de \mathcal{C} . Então, se $R(G)$ é uma representação de G e u um vértice de G , o objeto correspondente a u na representação é denotado por $R(u)$.

Consideremos as seguintes classes:

$$\text{CURVAS} = GI(\{\text{todas as curvas simples}\})$$

$$\text{CONV} = GI(\{\text{todos os conjuntos convexos}\})$$

$$k\text{-SEG} = GI(\{\text{todas as retas fragmentadas em não mais do que } k \text{ segmentos } \}), k \in \mathbb{N}, k \geq 1$$

Em particular, quando consideramos $k = 1$ temos a seguinte classe:

$$\text{SEG} = GI(\{\text{A classe dos grafos de interseção de segmentos lineares no plano}\}) = 1\text{-SEG}$$

$$k\text{-DIR} = GI(\{\text{A classe dos grafos de interseção de segmentos lineares paralelos em não mais do que } k \text{ direções}\}), k \in \mathbb{N}, k \geq 1$$

Observamos que a classe CURVAS é conhecida na literatura como a classe dos grafos STRING. Esta classe é a classe mais geral de grafos de interseção de

objetos geométricos, sendo introduzida por Sinden [18] em conexão com *RC-circuits*. Grafos STRING são amplamente considerados por vários autores [5, 8, 12]. Kratochvíl [10] mostrou que o reconhecimento de grafos STRING é NP-difícil.

Parece relativamente natural o surgimento da classe SEG. Usamos o termo “representação SEG de G ” em vez de “representação de G por segmentos”, e similarmente em outros casos. Se considerarmos segmentos lineares como curvas do plano, podemos concluir a seguinte inclusão natural $GI(SEG) \subseteq GI(STRING)$. Temos também que $SEG = \bigcup_{k=1}^{\infty} k\text{-DIR}$, onde $k \in \mathbb{N}$, fato demonstrado em [13].

É importante salientar que, para todo k , o reconhecimento de grafos k -DIR pertence a NP. Além disso, Kratochvíl and Matoušek [13] demonstraram que o reconhecimento de grafos SEG está em PSPACE.

Outras classes interessantes de grafos de interseção surgem quando colocamos restrições adicionais na posição relativa dos objetos associados aos vértices. Expressamos isto através da notação $GI(\mathcal{C}) \mid R$, que significará os grafos de $GI(\mathcal{C})$ que têm uma representação satisfazendo a restrição R . Assim sendo:

$$k\text{-CRUZ} = \text{CURVAS} \mid \{ \text{todas as curvas se cruzam duas a duas em} \\ \text{não mais do que } k \text{ pontos} \}.$$

k -DIR PURO = k -DIR | { segmentos paralelos são disjuntos dois a dois }.

Kratochvíl [11] também demonstrou que o reconhecimento de grafos 2-DIR e 2-DIR PUROS são problemas NP-completos.

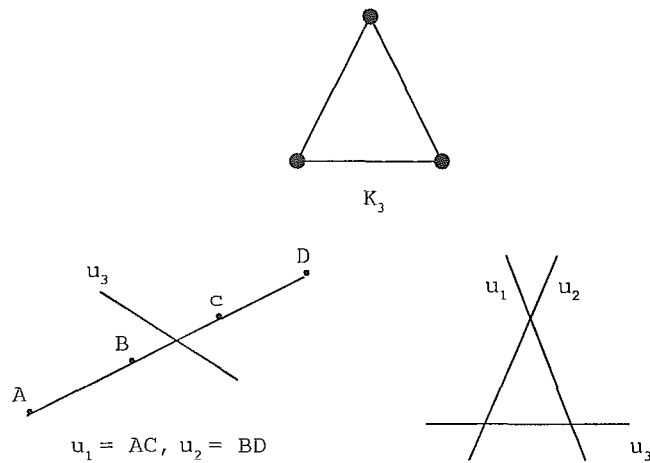


Figura 2.1: Representações 2-DIR e 3-DIR PURO do K_3

Na Figura 2.1 temos uma representação 2-DIR e outra 3-DIR PURO para o K_3 , enquanto que na Figura 2.2 temos uma representação 2-DIR PURO para o $K_{3,3}$.

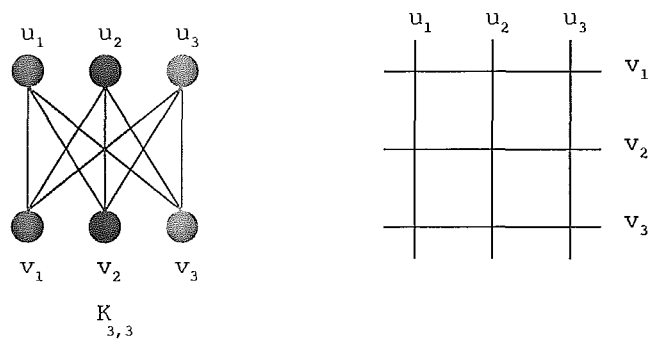


Figura 2.2: Representação 2-DIR PURO do $K_{3,3}$

Na Figura 2.3 exibimos as inclusões entre as classes mencionadas acima. As setas denotam as inclusões. Todas as inclusões são estritas.

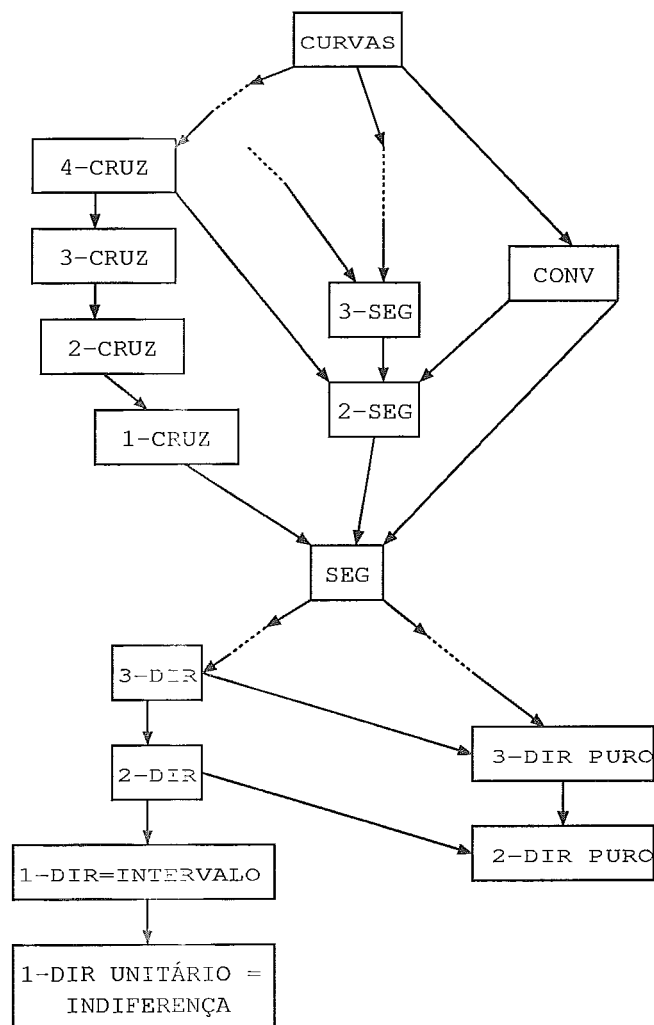


Figura 2.3: Inclusões entre as classes

O objeto de estudo desta tese é a classe de grafos 2-DIR PUROS. Isto é, grafos que têm uma representação por segmentos lineares paralelos em no máximo 2 direções tal que todos os segmentos paralelos são dois a dois

disjuntos. Tal classe é uma subclasse dos grafos bipartidos (fato este que iremos demonstrar a seguir no Teorema 3.1).

Não podemos deixar de mencionar que os grafos 2-DIR PUROS são também chamados de *Grid Intersection Graphs* [9].

2.1.1 O Problema INEQ

Seja A uma classe de grafos de interseção de objetos geométricos. Considere o seguinte problema de decisão:

ENTRADA: Um grafo G .

QUESTÃO: G pertence à classe A ?

Para resolvermos o maior número possível de problemas do tipo acima, veremos que eles podem ser formulados algebricamente e solucionados através de procedimentos que decidem se certas desigualdades polinomiais têm solução ou não.

No Capítulo 5 apresentaremos uma modelagem para o reconhecimento de grafos 2-DIR PUROS, utilizando as idéias a seguir.

Se Σ é um conjunto de polinômios com coeficientes inteiros, definimos o “tamanho” de Σ como $\|\Sigma\| = m + n + d + \log(C)$, onde m é o número de polinômios, n é o número de variáveis destes polinômios, d é o grau máximo dos polinômios e C é o máximo dos valores absolutos dos coeficientes. For-

malmente, o problema INEQ será o seguinte:

ENTRADA: Polinômios $p_1(x_1, \dots, x_n), \dots, p_m(x_1, \dots, x_n)$ com n variáveis e com coeficientes inteiros, e um inteiro k ($1 \leq k \leq m$). O tamanho da entrada é medido como $\| \{p_1, \dots, p_m\} \|$.

QUESTÃO: O sistema de desigualdades:

$$p_i(x_1, \dots, x_n) > 0 \quad i = 1, \dots, k$$

$$p_i(x_1, \dots, x_n) \geq 0 \quad i = k + 1, \dots, m \quad \text{tem solução em } \mathbb{R}^n?$$

O problema STRICT INEQ surge do problema INEQ, com a restrição de que todas as desigualdades sejam estritas, isto é, $k = m$. Sabemos que INEQ pertence a PSPACE. Veja [16] para mais referências sobre este assunto. INEQ é NP-difícil de acordo com Canny [3] e seus métodos implicam que STRICT INEQ é NP-difícil também.

Capítulo 3

A Classe de Grafos 2-DIR PUROS

Neste capítulo abordaremos de modo mais aprofundado a classe 2-DIR PURO. O principal resultado deste capítulo consiste em uma caracterização para esta classe de grafos, sendo esta uma das contribuições desta tese.

3.1 Primeiras Considerações

A priori, sabemos que se um grafo não é bipartido, não poderá pertencer à classe 2-DIR PURO, fato garantido pelo teorema abaixo.

Teorema 3.1 *Todo grafo 2-DIR PURO é bipartido.*

Necessitamos de um critério simples que nos permita fazer tal verificação, e o lema seguinte responde a esta questão.

Lema 3.2 *Ciclos ímpares não pertencem à classe 2-DIR PURO.*

Demonstração: Suponha por absurdo que o ciclo ímpar de $2n - 1$ vértices, denotado por C_{2n-1} , pertença à classe 2-DIR PURO. Escreva C_{2n-1} como a seguinte seqüência de vértices: $v_1, v_2, \dots, v_{2n-1}, v_1$. As arestas deste ciclo são as seguintes:

$$E' = \{ (v_1, v_2), (v_2, v_3), \dots, (v_{2n-2}, v_{2n-1}), (v_{2n-1}, v_1) \}$$

Isto quer dizer que segmentos que representam vértices consecutivos se interceptam. Logo, os segmentos (que representam os vértices) se interceptam se, e somente se, tais vértices têm índices com paridades distintas, donde os segmentos que representam os vértices v_1 e v_{2n-1} estão na mesma direção. Isto gera uma contradição, pois eles deveriam se interceptar. Concluímos que um ciclo ímpar não pode estar na classe 2-DIR PURO. Com isso demonstramos o Lema. □

O Lema acima auxilia na demonstração do Teorema 3.1.

Demonstração do Teorema 3.1: Suponha que G não seja um grafo bipartido. Então existe pelo menos um subgrafo induzido de G que é um ciclo ímpar. Logo, pelo Lema 3.2, G possui uma estrutura que não pode ser representada por segmentos em duas direções onde os segmentos paralelos são disjuntos. Assim sendo, o grafo G não pertence à classe 2-DIR PURO.

□

A recíproca deste Teorema não é verdadeira: existem grafos bipartidos que não são grafos 2-DIR PUROS. No entanto, todo grafo bipartido planar é 2-DIR PURO, a demonstração deste resultado pode ser obtida em [9], assim como famílias infinitas de grafos bipartidos que não são 2-DIR PUROS. Podemos comprovar através de argumentos combinatórios que o grafo da Figura 3.1 não é 2-DIR PURO, apesar de ser bipartido.

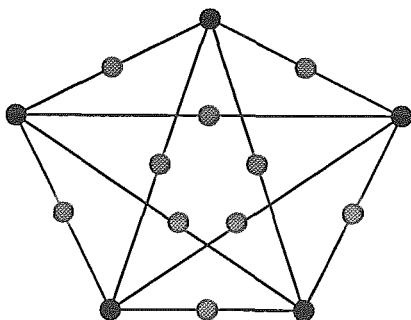


Figura 3.1: Exemplo de grafo bipartido que não pertence à classe 2-DIR PURO

Um resultado mais forte afirma que todo grafo bipartido com *boxicity* 2 é 2-DIR PURO [1] (Um grafo tem *boxicity* k se k é o menor inteiro tal que G é um grafo de interseção de paralelepípedos k -dimensionais - onde os lados dos paralelepípedos são paralelos ao eixos coordenados.)

3.1.1 Modelos de Grafos 2-DIR PUROS

Seja $G = (L, N)$ um grafo bipartido 2-DIR PURO e seja $R(G)$ a representação (modelo) de G . Assumiremos, sem perda de generalidade, que $R(G)$ tem as seguintes propriedades:

- (a) Os segmentos em uma direção são paralelos ao eixo- x , e os segmentos em outra direção são paralelos ao eixo- y ;
- (b) Os segmentos verticais (horizontais) não se interceptam dois a dois;
- (c) Todos os segmentos têm comprimento no mínimo 1(un).

Em outras palavras, G é um grafo 2-DIR PURO se, e somente se, existe um modelo $R(G)$ de G que satisfaz (a), (b) e (c).

Seja k_v o comprimento de $R(v)$. Assumimos que $R(v)$ é um segmento horizontal (respect. vertical) se $v \in L$ (respect. $v \in N$). Então, as propriedades acima podem ser reescritas do seguinte modo:

- (a) Se $v \in L$, então $R(v)$ é da forma $[(x_v, y_v), (x_v + k_v, y_v)]$; se $v \in N$, então $R(v)$ é da forma $[(x_v, y_v), (x_v, y_v + k_v)]$;
- (b) Se $v, w \in L$, então $y_v \neq y_w$; se $v, w \in N$, então $x_v \neq x_w$;
- (c) $k_v \geq 1$, para todo $v \in V$.

Vamos definir o *espaçamento* entre dois segmentos **disjuntos** $R(v)$ e $R(w)$ em $R(G)$ da seguinte maneira: Seja $d_{x,y}$ a menor distância entre um ponto de $R(x)$ e a reta que suporta $R(y)$. Então, o espaçamento entre $R(v)$ e $R(w)$ é o menor valor estritamente positivo entre $d_{v,w}$ e $d_{w,v}$ (Observe que pelo menos um dentre os valores $d_{v,w}$ e $d_{w,v}$ é não nulo.)

Agora, definamos *espaçamento de um modelo* $R(G)$ de G como sendo o menor inteiro d tal que d é o espaçamento entre dois segmentos disjuntos no modelo. Poderemos assumir também sem perda de generalidade que $R(G)$ possui a seguinte propriedade:

(d) $d = 1$.

Segue o seguinte teorema:

Teorema 3.3 *G é um grafo 2-DIR PURO se, e somente se, G tem um modelo $R(G)$ satisfazendo as propriedades (a), (b), (c), (d).*

Demonstração:

(\Leftarrow) : Trivial

(\Rightarrow) : Dado um modelo qualquer de G , podemos transformá-lo num modelo $R(G)$ satisfazendo (a), (b), (c) e (d) através de convenientes operações de rotação, translação e re-escala de segmentos. \square

3.2 Uma Caracterização de Grafos 2-DIR PUROS

Nesta seção exibiremos uma caracterização combinatória de grafos 2-DIR PUROS, através de uma ordenação especial de seus vértices.

Iniciaremos com algumas definições:

Definição: Seja $G = (L, N)$ um grafo bipartido conexo com pelo menos dois vértices. Sejam v_1, v_2, \dots, v_r e u_1, u_2, \dots, u_s ordenações dos vértices de L e N , respectivamente. Então, dizemos que :

$$v_i < v_j \quad (u_i < u_j) \text{ se, e somente se, } i < j,$$

$$v_i > v_j \quad (u_i > u_j) \text{ se, e somente se, } i > j,$$

$$v_i \geq v_j \quad (u_i \geq u_j) \text{ se, e somente se, } i \geq j,$$

$$v_i \leq v_j \quad (u_i \leq u_j) \text{ se, e somente se, } i \leq j.$$

Além disso, definimos, relativamente às ordenações acima:

$$v_i^+ = \max\{u_k \mid u_k \in N(v_i)\} \quad u_i^+ = \max\{v_k \mid v_k \in N(u_i)\}$$

$$v_i^- = \min\{u_k \mid u_k \in N(v_i)\} \quad u_i^- = \min\{v_k \mid v_k \in N(u_i)\}$$

Teorema 3.4 *Seja $G = (L, N)$ um grafo bipartido conexo com pelo menos dois vértices. O grafo G é 2-DIR PURO se, e somente, se existem ordenações v_1, v_2, \dots, v_r e u_1, u_2, \dots, u_s dos vértices de L e N , respectivamente, tais que valem as seguintes condições:*

$$I : u_k \notin N(v_i) \text{ e } v_i^- \leq u_k \leq v_i^+ \implies u_k^+ < v_i \text{ ou } u_k^- > v_i$$

$$II : v_k \notin N(u_i) \text{ e } u_i^- \leq v_k \leq u_i^+ \implies v_k^+ < u_i \text{ ou } v_k^- > u_i$$

$$III : u_k \in N(v_i) \implies v_i^- \leq u_k \leq v_i^+ \text{ e } u_k^- \leq v_i \leq u_k^+$$

Demonstração:

(\implies) : Suponha que $G = (L, N)$ seja um grafo 2-DIR PURO. Então existe um modelo de segmentos de reta em duas direções (horizontal e vertical) que representa G .

Atribua os segmentos horizontais aos vértices v_1, v_2, \dots, v_r de L , de modo que v_1 corresponda ao segmento de menor ordenada, v_2 ao segmento de segunda menor ordenada, e assim por diante.

De forma análoga, atribua os segmentos verticais aos vértices u_1, u_2, \dots, u_s de N , de modo que u_1 corresponda ao segmento de menor abcissa, u_2 ao segmento de segunda menor abcissa etc.

Considere as ordenações v_1, v_2, \dots, v_r dos vértices de L e u_1, u_2, \dots, u_s dos vértices de N .

É fácil ver, geometricamente, que estas ordenações assim construídas satisfazem as condições (I) e (II).

Mostraremos inicialmente a validade da condição (I).

De fato, considere um par de segmentos que não se interceptam representado pelos vértices u_k e v_i , onde $v_i^- \leq u_k \leq v_i^+$. Isto, geometricamente, significa que u_k ou está inteiramente contido no semi-plano inferior à reta que suporta v_i ou está inteiramente contido no semi-plano superior a esta reta. Logo, $u_k^+ < v_i$ ou $u_k^- > v_i$, respectivamente.

A condição (II) se verifica de modo análogo.

Resta-nos provar a condição (III). Se $u_k \in N(v_i)$, os segmentos que representam os vértices u_k e v_i se interceptam, e isto claramente implica que $v_i^- \leq u_k \leq v_i^+$ e $u_k^- \leq v_i \leq u_k^+$.

(\Leftarrow) : A seguinte construção gera um modelo de segmentos de reta em duas direções para G . Sejam x_i^1 e x_i^2 os índices de v_i^- e v_i^+ na ordenação, respectivamente. De modo análogo, sejam y_k^1 e y_k^2 os índices de u_k^- e u_k^+ na ordenação, respectivamente. Note que quando o grau de v_i (u_k) é igual a um, temos que $x_i^1 = x_i^2$ ($y_k^1 = y_k^2$).

Construção:

- Se o grau de v_i é 1, então atribua a v_i o segmento de reta $((x_i^1 - \frac{1}{2}, i), (x_i^2 + \frac{1}{2}, i))$; caso contrário, se o grau de v_i é maior que 1, então atribua a v_i o segmento de reta $((x_i^1, i), (x_i^2, i))$.
- Se o grau de u_k é 1, então atribua a u_k o segmento de reta $((k, y_k^1 - \frac{1}{2}), (k, y_k^2 + \frac{1}{2}))$; caso contrário, se o grau de u_k é maior que 1, então atribua a u_k o segmento de reta $((k, y_k^1), (k, y_k^2))$.

Observe que os segmentos correspondentes aos vértices de L (de N) são dois a dois disjuntos na construção.

Tome agora dois vértices v_i e u_k tais que $u_k \in N(v_i)$. Vamos mostrar que os segmentos correspondentes a v_i e u_k se interceptam no modelo. Pela

condição (III), temos que $v_i^- \leq u_k \leq v_i^+$ e $u_k^- \leq v_i \leq u_k^+$. Logo, vem que $x_i^1 - \frac{1}{2} < x_i^1 \leq k \leq x_i^2 < x_i^2 + \frac{1}{2}$ e $y_k^1 - \frac{1}{2} < y_k^1 \leq i \leq y_k^2 < y_k^2 + \frac{1}{2}$. Isto é, os segmentos correspondentes a v_i e u_k realmente se interceptam.

Completando a demonstração do teorema, tome dois vértices v_i e u_k tais que $u_k \notin N(v_i)$. Temos que verificar que os segmentos correspondentes a v_i e u_k não se interceptam no modelo. Se $u_k < v_i^-$ ou $u_k > v_i^+$, então $k < x_i^1$ ou $k > x_i^2$, isto é, os segmentos correspondentes a v_i e u_k de fato não se interceptam. Se $v_i^- \leq u_k \leq v_i^+$, então a condição (I) nos assegura que $u_k^+ < v_i$ ou $u_k^- > v_i$. Logo, $y_k^2 < i$ ou $y_k^1 > i$, donde pode-se novamente concluir que os segmentos correspondentes a v_i e u_k de fato não se interceptam.

Se v_i e u_k são tais que $v_i \notin N(u_k)$ segue analogamente que os segmentos correspondentes a v_i e u_k não se interceptam. (Use a condição (II).) Logo, G é 2-DIR PURO. □

Exemplo:

Considere a seguinte ordenação para os vértices de L e N , respectivamente: $\{4, 2, 1, 3\}$ e $\{7, 5, 6, 8\}$.

De fato, as ordenações dadas verificam as condições do teorema, logo existe um modelo 2-DIR PURO para G . Para desenharmos tal modelo, devemos fazê-lo na grade 4×4 , e ligarmos v_i^- a v_i^+ e u_k^- a u_k^+ , para todos $v_i, u_k \in V$. Segue a Figura 3.3.

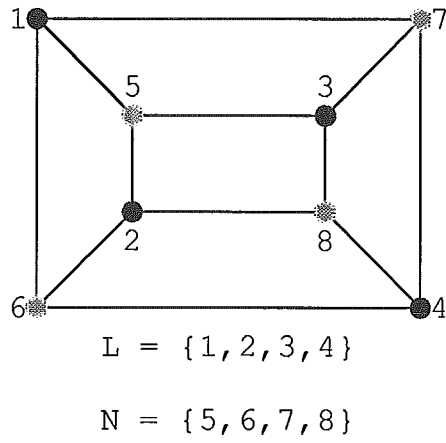


Figura 3.2: Grafo G

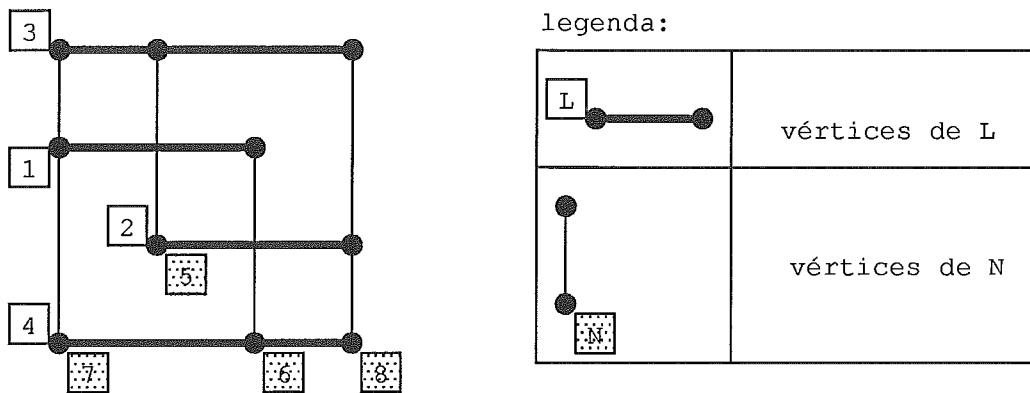


Figura 3.3: Modelo 2-DIR PURO do grafo G

Observação: O teorema acima não fornece um algoritmo polinomial para reconhecer se um determinado grafo G bipartido pertence à classe 2- DIR PURO.

3.3 Restringindo o Universo de Grafos 2-DIR PUROS

Dado um grafo bipartido conexo G , definimos o *grafo reduzido* G' de G da seguinte forma:

1. *Inicialize* $G' := G$
2. *Enquanto* G' possui um vértice v de grau 1 *faça* $G' := G' - v$

Observe que, no final do algoritmo acima, G' pode ser eventualmente um grafo com um único vértice. Por exemplo, se G é uma árvore, vértices de grau 1 são sucessivamente retirados até que reste apenas um vértice (que possui grau zero). Mas, neste caso especial, G' é trivialmente um grafo 2-DIR PURO.

O teorema a seguir permite-nos considerar apenas grafos bipartidos conexos de grau mínimo maior que 1.

Teorema 3.5 *Seja G um grafo bipartido conexo. Então, G é 2-DIR PURO se, e somente se, G' é 2-DIR PURO.*

Demonstração:

(\implies) : Trivial, pois G' é subgrafo induzido de G , e todo subgrafo induzido de um grafo 2-DIR PURO também é 2-DIR PURO.

(\impliedby) : A partir de um modelo de G' , construímos sucessivamente modelos com mais segmentos, incluindo a cada passo um novo segmento que intercepta apenas um dos já existentes no passo anterior, até obtermos um modelo para G . □

De agora em diante, assumiremos que os grafos não possuem vértices de grau inferior a 2. Observe, neste caso, que a construção feita na demonstração do Teorema 3.4 produz um modelo onde os extremos dos segmentos têm coordenadas inteiras (i, k) tais que $1 \leq i \leq s$ e $1 \leq k \leq r$.

Denominemos *grade* $s \times r$ ($r, s > 1$) ao conjunto de pontos formados pelas interseções das retas $x = i$, $i = 1, 2, \dots, s$, e $y = k$, $k = 1, 2, \dots, r$.

Podemos então inferir o seguinte corolário:

Corolário 3.6 *Seja $G = (L, N)$ um grafo 2-DIR PURO com $|L| = r > 1$ e $|N| = s > 1$. Então, existe um modelo de G representado sobre a grade $s \times r$.* □

Observe que no modelo acima o espaçamento é igual a 1.

Capítulo 4

Definindo o Problema de Otimização

No capítulo anterior caracterizamos os grafos 2-DIR PUROS, e vimos que quando $G = (L, N)$ é um grafo 2-DIR PURO com $\delta(G) > 1$, com $|L| = r$ e $|N| = s$, então G possui um modelo sobre a grade $s \times r$. Denominemos este modelo de *modelo de grade*.

Neste capítulo definiremos um problema de otimização onde procuramos minimizar o somatório dos comprimentos dos segmentos lineares em duas direções dois-a-dois disjuntos, que representam os vértices de G .

Seja $G = (L, N)$ um grafo 2-DIR PURO onde $\delta(G) > 1$, $|L| = r$ e $|N| = s$. Seja $R(G)$ um modelo de grade de G . Definimos:

$$\text{comprimento}(R(G)) = \sum_{v \in V} k_v$$

Recordemos que k_v é o comprimento do segmento associado ao vértice v

no modelo $R(G)$. Portanto $\text{comprimento}(R(G))$ é uma medida do “tamanho” do modelo $R(G)$, em termos da soma total dos comprimentos dos segmentos do modelo.

Definimos $\text{opt}(G)$ como o valor mínimo $\text{comprimento}(R(G))$ dentre todos os modelos de grade $R(G)$ para G :

$$\text{opt}(G) = \min\{\text{comprimento}(R(G)) \mid R(G) \text{ é um modelo de grade para } G\}$$

Por exemplo, na Figura 3.3, o modelo R apresentado tem comprimento 20. Além disso, pode-se mostrar que $\text{opt}(G) = 20$, isto é, o modelo “é o melhor possível”.

O Lema 4.1 fornece o valor de $\text{opt}(G)$ quando G é um grafo bipartido completo:

Lema 4.1 *Assuma $r > 1$ e $s > 1$. Então $\text{opt}(K_{r,s}) = 2rs - r - s$.*

Demonstração: É fácil ver que a única forma de construir um modelo do grafo $K_{r,s}$ sobre a grade $s \times r$ é fazer com que todos os segmentos horizontais iniciem na abcissa $x = 1$ e terminem na abcissa $x = s$, e todos os segmentos verticais iniciem na ordenada $y = 1$ e terminem na ordenada $y = r$. E o somatório dos comprimentos destes segmentos é $r(s-1) + s(r-1) = 2rs - r - s$.

□

O Lema 4.2 fornece um limite inferior e um limite superior para $opt(G)$:

Lema 4.2 *Seja $G = (L, N)$ um grafo 2-DIR PURO tal que $\delta(G) > 1$, $|L| = r > 1$ e $|N| = s > 1$. Então $4 \leq 2m - r - s \leq opt(G) \leq 2rs - r - s$.*

Demonstração: O limite inferior decorre do fato que, em qualquer modelo de grade de G , $k_v \geq \delta(v) - 1$ para todo $v \in V$:

$$\sum_{v \in V} k_v \geq \sum_{v \in V} \delta(v) - 1 = \sum_{v \in V} \delta(v) - \sum_{v \in V} 1 = 2m - n = 2m - r - s$$

Como supomos que $\delta(G) > 1$, isto implica que o valor mínimo de $2m - r - s$ é 4.

Com relação ao limite superior, este decorre do Lema 4.1, uma vez que $opt(G)$ atinge seu valor máximo quando G é o grafo $K_{r,s}$. □

O Lema 4.3 fornece um limite superior para $opt(G)$ em função de n .

Lema 4.3 *Seja G um grafo 2-DIR PURO com $\delta(G) > 1$. Então $opt(G) \leq 2rs - r - s \leq \lfloor \frac{n^2}{2} \rfloor - n$.*

Demonstração: Como G é um grafo com $n = r + s$ vértices, temos $opt(G) \leq opt(K_{r,s})$.

Considere a função f definida por $f(r, s) = 2rs - r - s$, onde $r, s > 1$ e $r + s = n$. De acordo com o Lema 4.2, precisamos encontrar os valores de r e s que maximizam esta função.

Com um simples argumento baseado no Critério da Segunda Derivada, concluímos que f é máxima quando $r = \frac{n}{2}$ e $s = \frac{n}{2}$.

Seguem os cálculos:

Dado que $r + s = n$, façamos uma substituição da variável r por $n - s$. Logo, $f(r, s) = f(n - s, s) = (n - s)(s - 1) + s(n - s - 1) = -2s^2 + 2ns - n$.

Denominaremos $g(s) = f(n - s, s) = -2s^2 + 2ns - n$.

A primeira derivada da aplicação g é: $g'(s) = -4s + 2n$. Tal derivada se anula no ponto $s = \frac{n}{2}$; este ponto é máximo, pois $g''(\frac{n}{2}) = -4 < 0$. Segue, portanto, $r = \frac{n}{2}$.

Como n é inteiro, podemos assumir sem perda de generalidade que: $r = \lfloor \frac{n}{2} \rfloor$ e $s = \lceil \frac{n}{2} \rceil$.

Utilizando o Lema 4.2, $2rs - r - s = \lfloor \frac{n^2}{2} \rfloor - n$. □

No próximo capítulo desta tese exibiremos uma estratégia para resolução do modelo de Programação Linear Inteira, que além de reconhecer se um determinado grafo G bipartido pertence à classe 2-DIR PURO, exhibe uma representação “mínima”, caso exista.

4.1 Prova de NP-Compleitude

O problema π de reconhecer se existe um modelo 2-DIR PURO de um grafo G , com já vimos, é um problema NP-completo [11]. Aproveitaremos este fato para mostrar que o problema central desta tese, descrito abaixo, também é NP-completo.

Considere o seguinte problema GRADE:

Dados (GRADE): $G(L, N)$ bipartido com $|L| = r$, $|N| = s$ e $\delta(G) > 1$, M inteiro positivo.

Decisão (GRADE): Existe um modelo de grade $R(G)$ para G , onde $\sum k_v \leq M$? Em outras palavras, $opt(G) \leq M$?

Teorema 4.4 *O problema GRADE é NP-completo.*

Demonstração:

Temos que verificar inicialmente que GRADE pertence à classe NP. De fato pertence, pois um certificado para resposta SIM de GRADE é um modelo de grade de G com $\sum k_v \leq M$. Tal certificado pode ser reconhecido facilmente em tempo polinomial no tamanho da entrada de GRADE.

Para completarmos a demonstração, vamos verificar que $\pi \propto$ GRADE.

Os dados de π consistem em um grafo G bipartido qualquer. Transforma-

mos facilmente estes dados em uma instância do problema GRADE fazendo $M = \lfloor \frac{n^2}{2} \rfloor - n$, que é um limite superior para $opt(G)$.

Resta demonstrar que G é 2-DIR PURO se, e somente se, $opt(G) \leq M$.

(\implies): De fato, se G tem um modelo 2-DIR PURO e $\delta(G) > 1$, segue do Corolário 3.6, que existe um modelo de grade para G . Vimos anteriormente que $\sum k_v \leq M$.

(\impliedby): A recíproca é trivialmente verdadeira.

Logo, GRADE é NP-completo. □

Capítulo 5

Um Modelo de Programação Linear Inteira Para Resolver o Problema GRADE

Neste capítulo iremos propor uma formulação de Programação Inteira Mista (PIM) para resolver o problema GRADE. Esta formulação parte de uma particularização do problema INEQ descrito no Capítulo 2 e dos resultados obtidos nos Capítulos 3 e 4.

A entrada de GRADE refere-se a um grafo bipartido G qualquer com $\delta(G) > 1$. A partir de G define-se um PPIM, de tal maneira que:

- Se o PPIM não apresenta solução viável, então G não é um grafo 2-DIR PURO.
- Se o PPIM apresenta solução viável, então G é um grafo 2-DIR PURO, e além disso o PPIM encontra um modelo de grade $R(G)$ para G tal

que a soma dos comprimentos dos segmentos em $R(G)$ é mínima, isto é, $\text{comprimento}(R(G)) = \text{opt}(G)$.

Vamos utilizar a Figura 5.1 para classificar as arestas e não-arestas de G em Famílias. Logo a seguir serão feitas as devidas generalizações para um grafo qualquer.

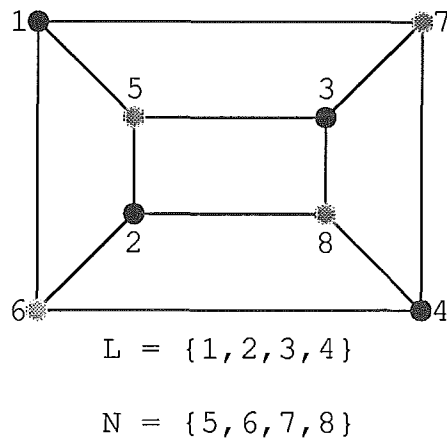


Figura 5.1: Grafo G

A Família 1, denotada por $\mathcal{F}(1)$, consiste de todas as arestas $(v, w) \in E$ com $v \in L$ e $w \in N$. Isto é:

- $\mathcal{F}(1) = \{(v, w) \in E \mid v \in L \text{ e } w \in N\}$

As Famílias 2, 3 e 4 são formadas por não-arestas, da seguinte forma:

- $\mathcal{F}(2) = \{(v, w) \in \bar{E} \mid v \in L \text{ e } w \in N\}$

- $\mathcal{F}(3) = \{(v, w) \in \bar{E} \mid v \in L \text{ e } w \in L\}$

- $\mathcal{F}(4) = \{(v, w) \in \bar{E} \mid v \in N \text{ e } w \in N\}$

Observamos que as Famílias $\mathcal{F}(2)$, $\mathcal{F}(3)$ e $\mathcal{F}(4)$ não podem ser desconsideradas do PPIM, pois não podemos perder a informação de quais segmentos não têm interseção entre si. Similarmente, através de $\mathcal{F}(1)$, saberemos quais segmentos se interceptam no modelo 2-DIR PURO do grafo G , caso ele exista.

Tomando o exemplo da Figura 5.1, as Famílias são definidas da seguinte forma:

| | |
|------------------|--|
| $\mathcal{F}(1)$ | $(1, 5)(1, 6)(1, 7)(2, 5)(2, 6)(2, 8)(3, 5)(3, 7)(3, 8)(4, 6)(4, 7)(4, 8)$ |
| $\mathcal{F}(2)$ | $(1, 8)(2, 7)(3, 6)(4, 5)$ |
| $\mathcal{F}(3)$ | $(1, 2)(1, 3)(1, 4)(2, 3)(2, 4)(3, 4)$ |
| $\mathcal{F}(4)$ | $(5, 6)(5, 7)(5, 8)(6, 7)(6, 8)(7, 8)$ |

Observe que estamos fazendo uma correspondência biunívoca dos vértices do grafo G com os números naturais. Isto é necessário para que possamos entrar com os dados no nosso PPIM. Então temos, por exemplo, que o vértice v_1 é representado por 1.

Outro ponto que precisamos destacar é o fato da Família $\mathcal{F}(3)$ (respectivamente, $\mathcal{F}(4)$) ser formada por combinações dois-a-dois dos vértices envolvidos na bipartição L (respectivamente, N).

Para cada vértice $v \in V$ serão introduzidas três variáveis, $x(v)$, $y(v)$ e $k(v)$. Temos que associar estas variáveis aos vértices $v \in L$ e $w \in N$. O segmento associado ao vértice $v \in L$ está representado na Figura 5.2.

Analogamente, o segmento associado ao vértice $w \in N$ está representado na Figura 5.3.

A variável $k(v)$ armazena o comprimento do segmento que corresponde ao vértice v no modelo. Objetivamos encontrar uma representação 2-DIR PURO para o grafo G , onde o somatório dos comprimentos dos segmentos é mínimo. Isto é,

$$\sum_{v \in V} k(v)$$

é mínimo.

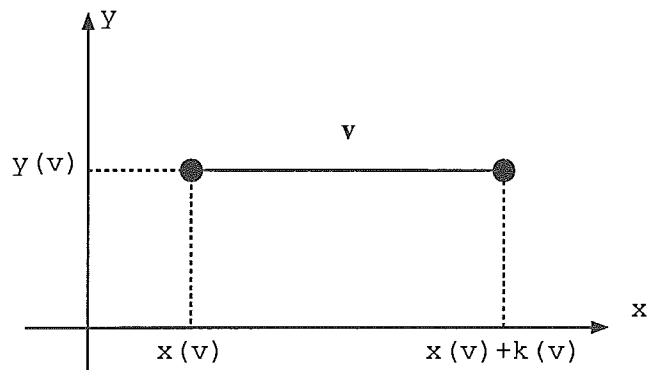


Figura 5.2: Segmento associado ao vértice $v \in L$

À continuação, veremos as inequações que descrevem cada Família.

5.1 Modelagem das Famílias

Iniciemos com a Família $\mathcal{F}(1)$.

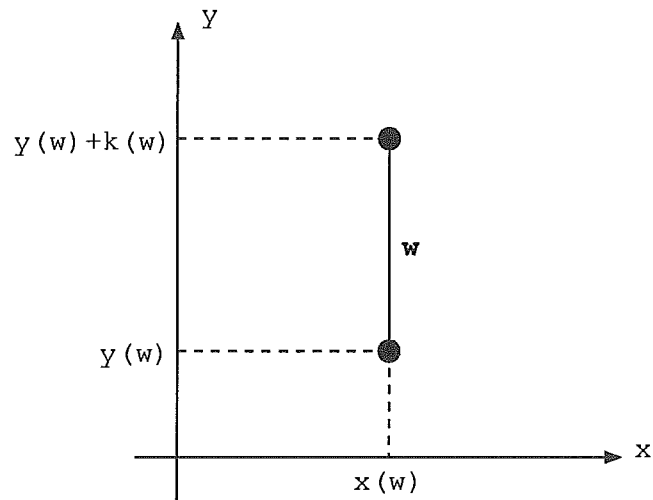


Figura 5.3: Segmento associado ao vértice $w \in N$

5.1.1 Família $\mathcal{F}(1)$

Sabemos que se $(v, w) \in E(G)$ tal que $v \in L$ e $w \in N$, esta aresta do grafo pertence a $\mathcal{F}(1)$. Caso o grafo G esteja na classe 2-DIR PURO, necessariamente, em alguma parte do modelo, vamos encontrar a configuração representada na Figura 5.4 .

Então, no eixo das abcissas acontece necessariamente a seguinte situação:

$$x(v) \leq x(w) \leq x(v) + k(v) \tag{5.1}$$

Analogamente, no eixo das ordenadas:

$$y(w) \leq y(v) \leq y(w) + k(w) \tag{5.2}$$

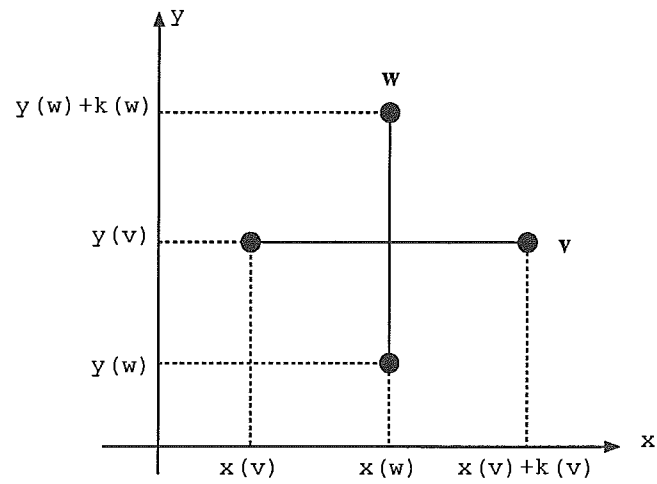


Figura 5.4: Família $\mathcal{F}(1)$

Da Inequação 5.1, seguem as inequações 5.3 e 5.4:

$$x(v) - x(w) \leq 0 \quad (5.3)$$

$$x(w) - x(v) - k(v) \leq 0 \quad (5.4)$$

Analogamente, da Inequação 5.2, seguem as inequações 5.5 e 5.6:

$$y(w) - y(v) \leq 0 \quad (5.5)$$

$$y(v) - y(w) - k(w) \leq 0 \quad (5.6)$$

Temos então que as Inequações 5.3, 5.4, 5.5 e 5.6 modelam as arestas da Família $\mathcal{F}(1)$.

5.1.2 Família $\mathcal{F}(2)$

Como observamos anteriormente, $\mathcal{F}(2)$ é composta de não-arestas tal que $(v, w) \in \bar{E}$ onde $v \in L$ e $w \in N$.

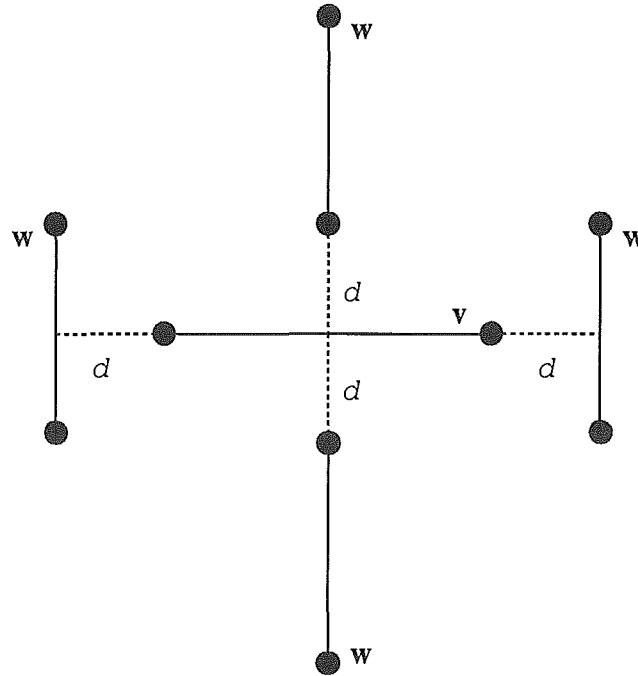


Figura 5.5: Família $\mathcal{F}(2)$

Nesta etapa será inserida mais uma restrição ao nosso problema. Aos segmentos que representam os vértices envolvidos nos pares de não-arestas, vamos forçar que eles tenham entre si no mínimo um espaçamento d . Como queremos construir um modelo de grade, vamos considerar $d = 1$. Tal situação está representada na Figura 5.5 .

Trataremos de cada caso que queremos forçar individualmente. Observe as Figuras 5.6 e 5.7.

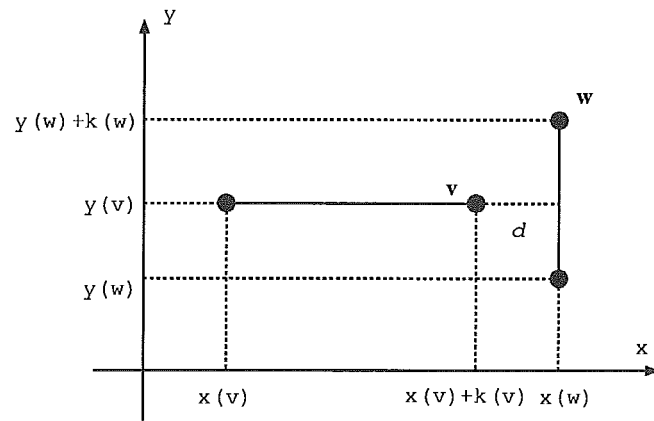


Figura 5.6: Situação $\alpha(v, w)$

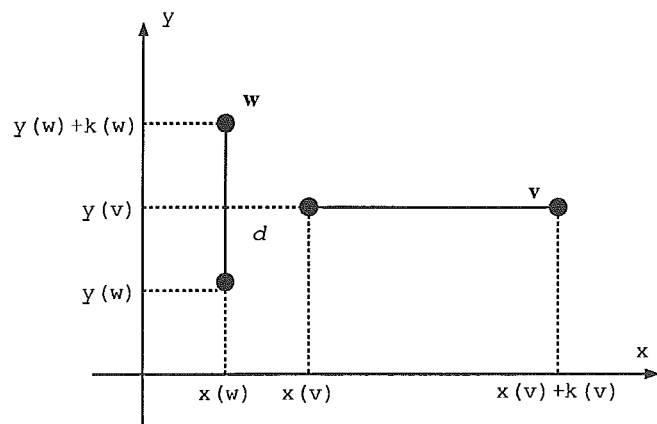


Figura 5.7: Situação $\alpha(w, v)$

Na situação referida na Figura 5.6, ao observarmos o eixo das abcissas temos a inequação:

$$x(w) - x(v) - k(v) - d \geq 0 \quad (5.7)$$

Enquanto que, na Figura 5.7, temos a seguinte configuração:

$$x(v) - x(w) - d \geq 0 \quad (5.8)$$

Estes casos são exclusivos entre si, pois o segmento que representa o vértice w só poderá estar à esquerda ou à direita do segmento que representa o vértice v . Observe que até este momento apenas atentamos para o eixo x . Mais adiante, iremos considerar o eixo y .

Como tratar as Inequações 5.7 e 5.8 em termos de restrições do PPIM? A resposta para esta questão consiste em utilizar o seguinte recurso: a inserção das variáveis binárias $\alpha(v, w)$ e $\alpha(w, v)$.

Temos que:

$$\alpha(v, w) = 0 \text{ se } x(w) - x(v) - k(v) - d < 0$$

Em termos de restrição:

$$\alpha(v, w) = \frac{x(w) - x(v) - k(v) - d}{K} \leq 1 \quad (5.9)$$

Onde K é uma constante positiva qualquer suficientemente grande, que garanta o módulo da divisão $\frac{x(w)-x(v)-k(v)-d}{K}$ entre 0 e 1. Por exemplo, podemos tomar $K \geq \max\{r, s\}$ onde $|L| = r$ e $|N| = s$.

A Inequação 5.9, da forma como foi construída, garante que a variável binária $\alpha(v, w)$ assume somente os valores 0 e 1. Isto é:

$$x(w) - x(v) - k(v) - d \geq 0 \implies \alpha(v, w) = 1 \text{ ou } 0$$

$$x(w) - x(v) - k(v) - d < 0 \implies \alpha(v, w) = 0$$

Analogamente, definimos a variável $\alpha(w, v)$:

$$\alpha(w, v) = 0 \text{ se } x(v) - x(w) - d < 0$$

Eis uma restrição necessária associada a esta variável:

$$\alpha(w, v) = \frac{x(v) - x(w) - d}{K} \leq 1 \quad (5.10)$$

Como no caso anterior é garantido que:

$$x(v) - x(w) - d \geq 0 \implies \alpha(w, v) = 1 \text{ ou } 0$$

A inequação acima confirma que:

$$y(v) - y(w) - k(w) - d \geq 0 \implies \beta(v, w) = 1 \text{ ou } 0$$

$$y(v) - y(w) - k(w) - d < 0 \implies \beta(v, w) = 0$$

E finalmente, temos a situação descrita na Figura 5.9.

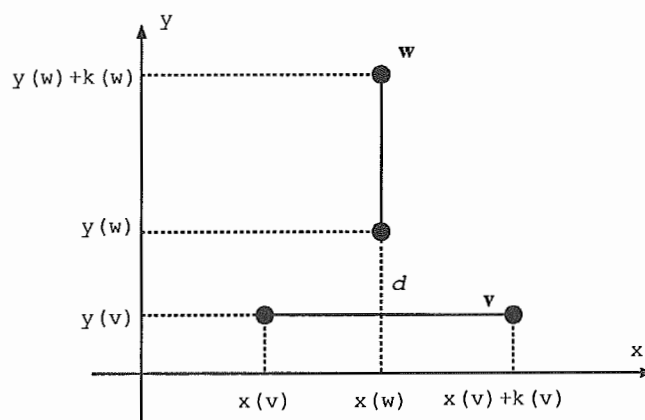


Figura 5.9: Situação $\beta(w, v)$

$$\beta(w, v) = 0 \text{ se } y(w) - y(v) - d < 0$$

A restrição do PPIM é:

$$\beta(w, v) - \frac{y(w) - y(v) - d}{K} \leq 1 \quad (5.12)$$

Isto é:

$$y(w) - y(v) - d \geq 0 \implies \beta(w, v) = 1 \text{ ou } 0$$

$$y(w) - y(v) - d < 0 \implies \beta(w, v) = 0$$

Observe que durante a construção da formulação temos que as variáveis binárias $\alpha(v, w)$ e $\alpha(w, v)$, assim como as variáveis $\beta(v, w)$ e $\beta(w, v)$, são exclusivas entre si (isto é ambas não podem assumir valor 1 simultaneamente), mas uma variável do tipo *alfa* não exclui uma variável do tipo *beta*. Logo, como queremos forçar pelo menos uma das situações descritas pelas variáveis $\alpha(v, w)$, $\alpha(w, v)$, $\beta(v, w)$ e $\beta(w, v)$ aconteça, temos que incluir uma última restrição ao conjunto de restrições da Família 2, que force pelo menos uma destas variáveis binárias assumir o valor 1.

$$\alpha(v, w) + \alpha(w, v) + \beta(v, w) + \beta(w, v) \geq 1 \quad (5.13)$$

5.1.3 Família $\mathcal{F}(3)$

A Família $\mathcal{F}(3)$ é composta de não-arestas, tal que $(v, w) \in \bar{E}$ onde $v \in L$ e $w \in L$.

Como sabemos que num modelo de grade dois segmentos horizontais disjuntos possuem ordenadas que diferem em pelo menos uma unidade, precisamos apenas controlar os valores das variáveis $y(v)$ e $y(w)$.

Na Figura 5.10, temos as situações que pretendemos forçar no eixo y . As equações do modelo que expressam tal situação são:

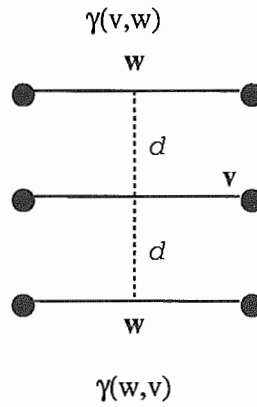


Figura 5.10: Situações $\gamma(v, w)$ e $\gamma(w, v)$

$$\gamma(v, w) = 0 \quad \text{se } y(w) - y(v) - d < 0$$

Em termos de restrição:

$$\gamma(v, w) - \frac{y(w) - y(v) - d}{K} \leq 1 \quad (5.14)$$

A Inequação 5.14 , da forma que foi construída, garante que a variável binária $\gamma(v, w)$ assume somente os valores 0 e 1. Isto é:

$$y(w) - y(v) - d \geq 0 \implies \gamma(v, w) = 1 \text{ ou } 0$$

$$y(w) - y(v) - d < 0 \implies \gamma(v, w) = 0$$

Analogamente, obtemos a variável $\gamma(w, v)$:

$$\gamma(w, v) = 0 \text{ se } y(v) - y(w) - d < 0$$

A restrição associada a esta variável é:

$$\gamma(w, v) - \frac{y(v) - y(w) - d}{K} \leq 1 \quad (5.15)$$

Como no caso anterior 5.15 garante :

$$y(v) - y(w) - d \geq 0 \implies \gamma(w, v) = 1 \text{ ou } 0$$

$$y(v) - y(w) - d < 0 \implies \gamma(w, v) = 0$$

Temos que incluir uma última restrição ao conjunto de restrições da Família 3. Precisamos garantir que uma das variáveis binárias assume o valor 1, e a outra assume o valor 0.

$$\gamma(v, w) = 1 - \gamma(w, v) \quad (5.16)$$

5.1.4 Família $\mathcal{F}(4)$

A Família $\mathcal{F}(4)$ é composta de não arestas, tal que $(v, w) \in \bar{E}$ onde $v \in N$ e $w \in N$.

Na figura 5.11, temos as situações que pretendemos forçar no eixo x . as equações do modelo que expressam tal situação são:

$$\tau(v, w) = 0 \quad \text{se } x(w) - x(v) - d < 0$$

Em termos de restrição:

$$\tau(v, w) - \frac{x(w) - x(v) - d}{K} \leq 1 \quad (5.17)$$

A Inequação 5.17 , da forma que foi construída, garante que a variável binária $\tau(v, w)$ assume somente os valores 0 e 1. Isto é:

$$x(w) - x(v) - d \geq 0 \implies \tau(v, w) = 1 \text{ ou } 0$$

$$x(w) - x(v) - d < 0 \implies \tau(v, w) = 0$$

E finalmente:

$$\tau(w, v) = 0 \quad \text{se } x(v) - x(w) - d < 0$$

A restrição associada a esta variável é:

$$\tau(w, v) - \frac{x(v) - x(w) - d}{K} \leq 1 \quad (5.18)$$

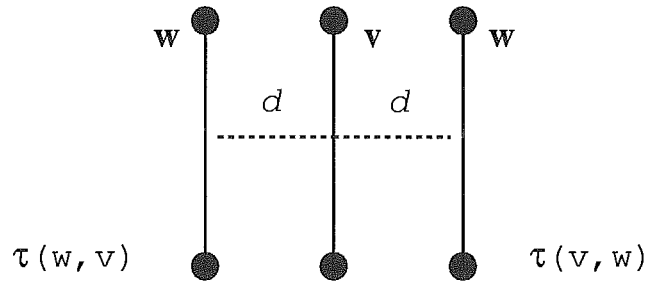


Figura 5.11: Situações $\tau(v, w)$ e $\tau(w, v)$

Como no caso anterior 5.18 garantem :

$$x(v) - x(w) - d \geq 0 \implies \tau(w, v) = 1 \text{ ou } 0$$

$$x(v) - x(w) - d < 0 \implies \tau(w, v) = 0$$

Temos que incluir uma última restrição ao conjunto de restrições da Família 4, para indicar que uma variável binária assume o valor 1 enquanto a outra assume o valor 0:

$$\tau(v, w) = 1 - \tau(w, v) \tag{5.19}$$

5.2 Corretude da Formulação

A formulação completa do PPIM para resolver o problema GRADE necessita de algumas restrições adicionais. Primeiramente:

$$d = 1 \tag{5.20}$$

Com a equação acima queremos garantir que segmentos não adjacentes tenham espaçamento no mínimo 1.

A seguir, temos duas restrições naturais:

$$1 \leq x(v) \leq s \quad \forall v \in V \tag{5.21}$$

$$1 \leq y(v) \leq r \quad \forall v \in V \tag{5.22}$$

As restrições acima garantem que o modelo está construído sobre a grade $s \times r$, onde $|L| = r$ e $|N| = s$.

Finalmente, como assumimos que o grafo de entrada G tem $\delta(G) > 1$, vem que:

$$k(v) \geq \delta(v) - 1, \forall v \in V \tag{5.23}$$

Adicionando estas novas restrições às expostas nas subseções anteriores, a formulação completa do PPIM para resolver o problema GRADE fica estabelecida da seguinte forma:

Π : Minimizar $\sum_{v \in V} k(v)$

Sujeito a:

Para todo $(v, w) \in E$ tal que $v \in L, w \in N$:

$$x(w) - x(v) \geq 0 \quad (5.3)$$

$$x(v) + k(v) - x(w) \geq 0 \quad (5.4)$$

$$y(v) - y(w) \geq 0 \quad (5.5)$$

$$y(w) + k(w) - y(v) \geq 0 \quad (5.6)$$

Para todo $(v, w) \in \bar{E}$ tal que $v \in L, w \in N$:

$$(x(w) - x(v) - k(v) - d) - K\alpha(v, w) + K \geq 0 \quad (5.9)$$

$$(x(v) - x(w) - d) - K\alpha(w, v) + K \geq 0 \quad (5.10)$$

$$(y(v) - y(w) - k(w) - d) - K\beta(v, w) + K \geq 0 \quad (5.11)$$

$$(y(w) - y(v) - d) - K\beta(w, v) + K \geq 0 \quad (5.12)$$

$$\alpha(v, w) + \alpha(w, v) + \beta(v, w) + \beta(w, v) - 1 \geq 0 \quad (5.13)$$

Para todo $(v, w) \in \bar{E}$ tal que $v \in L, w \in L$:

$$(y(w) - y(v) - d) - K\gamma(v, w) + K \geq 0 \quad (5.14)$$

$$(y(v) - y(w) - d) - K\gamma(w, v) + K \geq 0 \quad (5.15)$$

$$\gamma(v, w) = 1 - \gamma(w, v) \quad (5.16)$$

Para todo $(v, w) \in \overline{E}$ tal que $v \in N, w \in N$:

$$(x(w) - x(v) - d) - K\tau(v, w) + K \geq 0 \quad (5.17)$$

$$(x(v) - x(w) - d) + K\tau(w, v) + K \geq 0 \quad (5.18)$$

$$\tau(v, w) = 1 - \tau(w, v) \quad (5.19)$$

$$d = 1 \quad (5.20)$$

Para todo $v \in V$:

$$x(v) \geq 1 \quad (5.21)$$

$$y(v) \geq 1 \quad (5.22)$$

$$x(v) \leq s \quad (5.21)$$

$$y(v) \leq r \quad (5.22)$$

$$k(v) \geq \delta(v) - 1 \quad (5.23)$$

Para todo $(v, w) \in \overline{E}$ tal que $v \in L, w \in N$:

$$\alpha(v, w) \in \{0, 1\}$$

$$\alpha(w, v) \in \{0, 1\}$$

$$\beta(v, w) \in \{0, 1\}$$

$$\beta(w, v) \in \{0, 1\}$$

$$\gamma(v, w) \in \{0, 1\}$$

$$\gamma(w, v) \in \{0, 1\}$$

$$\tau(v, w) \in \{0, 1\}$$

$$\tau(w, v) \in \{0, 1\}$$

Este modelo de programação mista pode ser alterado para reconhecer se G é 2-DIR PURO UNITÁRIO. Para tal, basta adicionar a restrição

$$k(v) = k(w), \text{ para todo } v, w \in V \quad (5.24)$$

O teorema abaixo garante que a representação 2-DIR PURO do grafo G está representado sobre a grade de pontos inteiros.

Teorema 5.1 *Para toda solução viável do problema Π , os valores das variáveis $x(v), y(v), k(v)$, para $v \in V$, são inteiros.*

Demonstração:

Sabemos que no problema Π , formulado acima, as seguintes restrições são satisfeitas: $1 \leq x(v) \leq s$ e $|x(v) - x(w)| \geq 1$ para todo $v, w \in N$ (segue do conjunto de restrições associadas à Família 4). Mas como $|N| = s$, podemos concluir que as abcissas dos segmentos lineares relativos aos vértices de N assumem valores inteiros.

Analogamente, concluímos que as ordenadas dos segmentos lineares relativos aos vértices de L assumem valores inteiros. De fato, $1 \leq y(v) \leq r$, $|y(v) - y(w)| \geq 1$ para todo $v, w \in L$ (segue do conjunto de restrições associadas à Família 3) e $|L| = r$.

Temos ainda que verificar que as abcissas dos segmentos lineares relativos aos vértices de L assumem valores inteiros, para podermos concluir que $x(v)$ é inteiro para todo $v \in V$.

Considere $v' \in L$. Sabemos que v' possui no mínimo dois vértices adjacentes em N . Isto decorre dos seguintes fatos: $\delta(v) > 1$ para todo $v \in V$ e o grafo em questão é bipartido. Então, tomemos $w_1 \in N$ adjacente a v' tal

que w_1 tem o menor valor de abcissa dentre todos os vértices adjacentes a v' e seja $w_2 \in N$, adjacente a v' , tal que w_2 tem o maior valor de abcissa dentre todos os vértices adjacentes a v' . Mas como $w_1, w_2 \in N$ possuem ordenadas inteiras, isto é, $y(w_1), y(w_2) \in \mathbb{N}$, e o problema Π minimiza o comprimento dos segmentos, isto implica que $x(v')$ e $y(v')$ assumem valores inteiros quando $v' \in L$. Segue a Figura 5.12 :

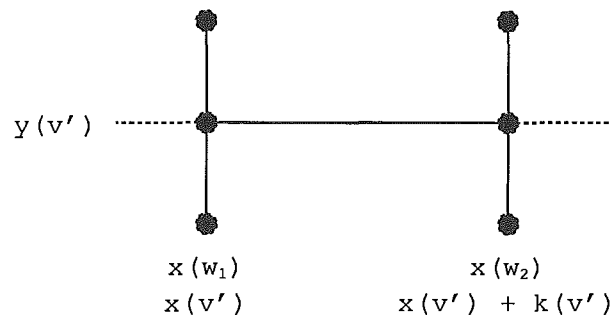


Figura 5.12: Esquema da demonstração

Verificamos que $x(v)$ e $k(v)$ são inteiros para todo $v \in V$. Resta-nos provar que $y(v)$ e $k(v)$ são inteiros para todo $v \in N$ para concluirmos que $y(v)$ e $k(v)$ são inteiros para todo $v \in V$, mas este fato decorre de um argumento semelhante ao anterior.

□

A corretude da formulação segue do teorema abaixo:

Teorema 5.2 *O problema Π apresenta solução viável se, e somente se, G é um grafo 2-DIR PURO .*

Demonstração:

(\implies): Suponha que o problema Π tenha solução viável. Sejam os valores $x(v)$, $y(v)$ e $k(v)$ associados a cada vértice $v \in V$. Dado que conhecemos os vértices de cada lado da bipartição, defina para cada vértice $v \in V$ o segmento $R(v)$ em função de $x(v)$, $y(v)$ e $k(v)$. Se $v \in L$, $R(v) = ((x(v), y(v)), (x(v) + k(v), y(v)))$ e se $v \in N$, $R(v) = ((x(v), y(v)), (x(v), y(v) + k(v)))$.

Sejam $v \in L$ e $w \in N$ tais que $(v, w) \in E$. Como as restrições da Família 1 são satisfeitas, elas forçam a interseção entre os segmentos $R(v)$ e $R(w)$ que os representam no modelo.

Analogamente, se $v \in L$ e $w \in N$ tais que $(v, w) \in \bar{E}$, sabemos que $R(v)$ e $R(w)$ não se interceptam pois as restrições da Família 2 são satisfeitas.

Por argumentos semelhantes, concluímos a não interseção dos segmentos $R(v)$ e $R(w)$ quando $v \in L$ e $w \in L$ tais que $(v, w) \in \bar{E}$ e a não interseção dos segmentos $R(v)$ e $R(w)$ quando $v \in N$ e $w \in N$ tais que $(v, w) \in \bar{E}$, pois as restrições da Família 3 e da Família 4 são satisfeitas, respectivamente, para cada caso. Logo, G é um grafo 2-DIR PURO .

(\impliedby): Suponha que o grafo G pertence à classe 2-DIR PURO. Então, G possui um modelo de grade.

Sejam $v \in L$ e $w \in N$ tais que $(v, w) \in E$. Então é claro que as restrições da Família 1 são satisfeitas.

Sejam $v \in L$ e $w \in N$ tais que $(v, w) \in \overline{E}$. Definamos:

- $\alpha(v, w) = 1$ e $\alpha(w, v) = 0$, se o segmento $R(w)$ estiver à direita do segmento $R(v)$ no modelo de grade de G ,
- $\alpha(v, w) = 0$ e $\alpha(w, v) = 1$, se o segmento $R(w)$ estiver à esquerda do segmento $R(v)$,
- $\beta(v, w) = 1$ e $\beta(w, v) = 0$, se o segmento $R(w)$ estiver abaixo do segmento $R(v)$,
- $\beta(v, w) = 0$ e $\beta(w, v) = 1$, se o segmento $R(w)$ estiver acima do segmento $R(v)$.

Se um par de variáveis α ou β estiver indefinido após a aplicação das regras acima, faça ambas assumirem o valor zero.

Com isso as restrições da Família 2 são satisfeitas.

Similarmente, sejam $v \in L$ e $w \in L$ tais que $(v, w) \in \overline{E}$. Definamos:

- $\gamma(v, w) = 1$ e $\gamma(w, v) = 0$, se o segmento $R(w)$ estiver acima do segmento $R(v)$,
- $\gamma(v, w) = 0$ e $\gamma(w, v) = 1$, se o segmento $R(w)$ estiver abaixo do segmento $R(v)$.

Logo as restrições da Família 3 são satisfeitas.

Resta-nos analisar o caso onde $v \in N$ e $w \in N$ tais que $(v, w) \in \overline{E}$. E como procedemos nos casos anteriores, definamos:

- $\tau(v, w) = 1$ e $\tau(w, v) = 0$, se o segmento $R(w)$ estiver à direita do segmento $R(v)$,
- $\tau(v, w) = 0$ e $\tau(w, v) = 1$, se o segmento $R(w)$ estiver à esquerda do segmento $R(v)$.

Desse modo as restrições da Família 4 são satisfeitas e finalmente podemos concluir que o problema II apresenta solução viável.

□

5.3 Número de Restrições e Variáveis na Formulação

Nesta seção iremos verificar que a ordem do número de restrições e variáveis da formulação completa do PPIM para resolver o problema GRADE são $O(n^2 + m)$ e $O(n^2)$, respectivamente, onde n é o número de vértices do grafo G e m o número de arestas.

Observe o quadro abaixo:

| Famílias | Nº de Restrições | | Nº de Variáveis | |
|------------------|--------------------------------------|----------|--------------------------------------|----------|
| | $\mathcal{F}(1)$ | $4m$ | $O(m)$ | $3n$ |
| $\mathcal{F}(2)$ | $5\left(\frac{n(n-1)}{2} - m\right)$ | $O(n^2)$ | $4\left(\frac{n(n-1)}{2} - m\right)$ | $O(n^2)$ |
| $\mathcal{F}(3)$ | $3\left(\frac{r(r-1)}{2}\right)$ | $O(n^2)$ | $2\left(\frac{r(r-1)}{2}\right)$ | $O(n^2)$ |
| $\mathcal{F}(4)$ | $3\left(\frac{s(s-1)}{2}\right)$ | $O(n^2)$ | $2\left(\frac{s(s-1)}{2}\right)$ | $O(n^2)$ |

Logo, podemos concluir que a formulação possui $O(n^2 + m)$ restrições e $O(n^2)$ variáveis.

Capítulo 6

Conclusões

Apresentamos agora as principais conclusões desta tese e sugestões de trabalhos futuros.

Provamos um Teorema de Caracterização de grafos 2-DIR PUROS, a partir de uma ordenação especial dos vértices do grafo. Observamos que tal teorema não induz um algoritmo polinomial.

Verificamos que todo grafo 2-DIR PURO é bipartido, e supondo que o grau mínimo é maior que 1, ele pode ser representado sobre a grade de pontos com coordenadas inteiras. A dimensão da grade é obtida através da cardinalidade de cada lado da bipartição.

Definimos o problema GRADE, verificamos que é NP-completo e exibimos a formulação completa do problema de programação inteira mista para resolvê-lo. Exibimos a corretude desta formulação, e finalmente analisamos o número de restrições e variáveis na formulação.

Ao longo do nosso trabalho, realizamos algumas implementações com vários grafos utilizando o software XPress e obtivemos resultados satisfatórios. Por exemplo, o XPress obteve o modelo da Figura 3.3 para o grafo da Figura 3.2

Em uma próxima etapa do nosso trabalho, planejamos retirar a a função objetivo da formulação, para construirmos um “testador” de grafos 2-DIR PUROS: usando o resultado do Teorema 5.2, sabemos que o sistema de restrições colocado no XPress terá solução viável se, e somente se, o grafo em questão é 2-DIR PURO. Removendo a necessidade de minimização, pode-se testar grafos bem maiores.

Outra questão interessante será utilizar técnicas mais avançadas de programação linear para acelerar a resolução do problema GRADE de minimização.

Referências Bibliográficas

- [1] S. BELLANTONI, I. BEN-ARROYO HARTMAN, T. PRZYTYCKA, and S. WHITESIDES. Grid intersection graphs and boxicity. *Discrete Mathematics*, 114:41–49, 1993.
- [2] P. F. BREGALDA, A. F. OLIVEIRA, and C. T. BORNSTEIN. *Introdução à Programação Linear*. Campus, Rio de Janeiro, 1981.
- [3] J. CANNY. Some algebraic and geometric computations in PSPACE. In *Complexity of Computer Computations*, pages 460–467. 20th ACM Symposium on Theory of Computing, 1988.
- [4] H. EDELSBRUNNER. *Algorithms in Combinatorial Geometry*. Springer-Verlag, Berlin/New York, 1987.
- [5] S. EVEN, R. E. TARJAN, and G. EHRLICH. Intersection graphs of curves in the plane. *Journal of Combinatorial Theory, Series B*, 21:8–20, 1976.

- [6] R. FONDACCI, O. GOLDSCHMIDT, and V. LETROUIT. Combinatorial issues in air traffic optimization. *Transportation Science*, 32:256–267, 1998.
- [7] M. R. GAREY and D. S. JOHNSON. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.
- [8] R. L. GRAHAM. Problem 1, open problems at 5th hungarian colloquium on combinatorics. Keszthely 1976, in A. Hajnal, ed. *Finite and Infinite Sets*, Colloquia Math. Soc. J. Bolyai.
- [9] I. BEN-ARROYO HARTMAN, I. NEWMAN, and R. ZIV. On grid intersection graphs. *Discrete Mathematics*, 87:41–52, 1991.
- [10] J. KRATOCHVÍL. String graphs II - recognizing string graphs is NP-hard. *Journal of Combinatorial Theory, Series B*, 52:67–78, 1991.
- [11] J. KRATOCHVÍL. A special planar satisfiability problem and a consequence of its np-completeness. *Discrete Applied Mathematics*, 52:233–252, 1994.
- [12] J. KRATOCHVÍL, M. GOLJAN, and P. KUCERA. *String graphs*. Academia, Prague, 1986.
- [13] J. KRATOCHVÍL and J. MATOUŠEK. Intersection graphs of segments. *Journal of Combinatorial Theory, Series B*, 62:289–315, 1994.

- [14] N. MACULAN, M. H. C. FAMPA, and P. MICHELON. *Programação Linear e Inteira. Notas, Programa de Engenharia de Sistemas e Computação*. COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil, 1998.
- [15] C. H. PAPADIMITRIOU and K. STEIGLITZ. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, New Jersey, 1982.
- [16] J. RENEGAR. On the computation complexity and geometry of the first order theory of the reals. *J. Symbolic. Comput.*, 1992.
- [17] F. S. ROBERTS. *Graphy Theory and its Applications to Problems of Society*. SIAM, Philadelphia, 1978.
- [18] F. W. SINDEN. Topology of thin film RC-circuits. *Bell System Technol. J.*, pages 1639–1662, 1966.
- [19] J. L. SZWARCFITER. *Grafos e Algoritmos Computacionais*. Campus, Rio de Janeiro, 1986.
- [20] L. A. WOLSEY. *Integer Programming*. Wiley-Interscience Publication, New Jersey, 1998.