

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE MATEMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA APLICADA

**O Espectro de Grafos
Threshold e Aplicações**

por

Fernando Colman Tura

Tese submetida como requisito parcial
para a obtenção do grau de
Doutor em Matemática Aplicada

Prof. Dr. Vilmar Trevisan
Orientador

Porto Alegre, Julho de 2013.

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Colman Tura, Fernando

O Espectro de Grafos *Threshold* e Aplicações / Fernando Colman Tura.—Porto Alegre: PPGMAp da UFRGS, 2013.

95 p.: il.

Tese (doutorado) —Universidade Federal do Rio Grande do Sul, Programa de Pós-Graduação em Matemática Aplicada, Porto Alegre, 2013.

Orientador: Trevisan, Vilmar

Tese: Matemática Aplicada

Grafos *Threshold* e Teoria Espectral de Grafos

O Espectro de Grafos *Threshold* e Aplicações

por

Fernando Colman Tura

Tese submetida ao Programa de Pós-Graduação em Matemática Aplicada do Instituto de Matemática da Universidade Federal do Rio Grande do Sul, como requisito parcial para a obtenção do grau de

Doutor em Matemática Aplicada

Linha de Pesquisa: Algoritmos Numéricos e Algébricos

Orientador: Prof. Dr. Vilmar Trevisan

Banca examinadora:

Prof. Dr. Jayme Luiz Szwarcfiter
INF/UFRJ

Profa. Dra Renata Raposo Del-Vecchio
IM/UFF

Profa. Dra. Luciana Salete Buriol
INF/UFRGS

Profa. Dra. Virgínia Maria Rodrigues
PPGMAp/IM/UFRGS

Tese apresentada e aprovada em
04 de Julho de 2013.

Profa. Maria Cristina Varriale, Ph.D.
Coordenador

Conteúdo

LISTA DE ABREVIATURAS	VI
RESUMO	VII
ABSTRACT	VIII
AGRADECIMENTOS	IX
1 INTRODUÇÃO	1
2 GRAFOS <i>THRESHOLD</i>	8
2.1 Introdução	8
2.2 Cografos	11
2.3 Caracterizações	15
2.3.1 Definição	15
2.3.2 Caracterização livre de $2K_2, C_4$ e P_4	16
2.3.3 Caracterização via grafos <i>split nested</i>	20
2.4 Redução do polinômio característico	25
2.4.1 A partição equilibrada mínima	25
2.4.2 A multiplicidade dos autovalores 0 e -1	26
2.4.3 O grafo quociente	28
3 LOCALIZANDO AUTOVALORES	31
3.1 Introdução	31
3.2 Diagonalizando $A + xI$	32
3.2.1 Exemplos	38
3.2.2 Encontrando autovalores	41
3.3 Simplicidade dos autovalores	44
4 A MINIMALIDADE DO MENOR AUTOVALOR	46

4.1	Introdução	46
4.2	O espectro de $G(k, j)$	47
4.3	O Menor autovalor $\lambda_{min,n}$	50
5	ORDENANDO OS GRAFOS <i>THRESHOLD</i>	57
5.1	Introdução	57
5.2	Ordenação via índice	58
5.3	Uma família com espectro integral	61
6	O POLINÔMIO CARACTERÍSTICO	63
6.1	Introdução	63
6.2	Diagonalizando $A - \lambda I$	64
6.3	Considerações sobre a complexidade	72
7	EXTENSÃO PARA A MATRIZ DISTÂNCIA	74
7.1	Introdução	74
7.2	Diagonalizando $\Theta + xI$	76
8	CONCLUSÃO	86
	BIBLIOGRAFIA	90
	ÍNDICE	93

LISTA DE ABREVIATURAS

(b_1, b_2, \dots, b_n)	a sequência binária de um grafo <i>threshold</i>
$G(a_1, a_2, \dots, a_r)$	grafo <i>threshold</i> via partição equilibrada
π	partição equilibrada
T_G	coárvore de um grafo G
G/π	grafo quociente
K_n	grafo completo
A	matriz de adjacência
Θ	matriz distância
D	matriz diagonal
$A(G/\pi)$	matriz de adjacência do grafo quociente
$\lambda(G)$	o índice de um grafo G
$\lambda_{min,n}$	o menor autovalor mínimo
$p(\lambda)$	polinômio característico
$\alpha_{G,x}$	α -sequência

RESUMO

Nesta tese de doutorado estudamos uma classe de grafos denominada *threshold*. Iniciamos apresentando algumas caracterizações dos grafos *threshold* e definindo-os de uma forma apropriada para o nosso propósito.

Mais especificamente, estudamos o espectro dos grafos *threshold*. Para isso apresentamos alguns resultados previamente conhecidos, como por exemplo, em relação à matriz de adjacência, uma redução para o cálculo do polinômio característico e a multiplicidade dos autovalores não principais.

Desenvolvemos um algoritmo que constrói uma matriz diagonal D congruente a $A + xI$, onde A é a matriz de adjacência de um grafo *threshold*, x é um número real e I é a matriz identidade. Como aplicação, determinamos quantos autovalores de um grafo *threshold* G pertencem a um intervalo real $(a, b]$. A implementação do nosso algoritmo depende apenas da sequência binária de um grafo *threshold* e sua complexidade é de ordem $O(n)$. Tal algoritmo é facilmente adaptado para a matriz distância Θ de um grafo *threshold* G .

Nesta tese apresentamos aplicações desse algoritmo, como a simplicidade dos autovalores principais, a minimalidade do menor autovalor de um *threshold*, exibindo uma fórmula para esse menor autovalor, uma ordenação dos grafos *threshold* via índice, e uma classe infinita de grafos *threshold* com espectro integral. Além disso, apresentamos um novo algoritmo para o cálculo do polinômio característico de um grafo *threshold* G .

ABSTRACT

In this thesis we study the class of threshold graphs. We begin showing some characterizations of threshold graphs defining them in a convenient way for our purposes.

More specifically, we study the spectrum of threshold graphs. To this end, we show some previously known results about the adjacency matrix, such as the reduction for computing the characteristic polynomial and the multiplicity of non main eigenvalues.

We developed an algorithm that constructs a diagonal matrix D congruent to $A + xI$, where A represents the adjacency matrix of a threshold graph and x is a real number. As an application, we determine how many eigenvalues of a threshold graph lie in an interval $(a, b]$. The algorithm implementation depends only on the binary sequence of the threshold graph, and its complexity is of order $O(n)$. This algorithm is easily adapted for the distance matrix Θ of a threshold graph G .

We finish this dissertation showing some applications of this algorithm. We show that the main eigenvalues are simple. We also determine the class of threshold graphs which have the minimum eigenvalue among threshold graphs of order n , and a formula for this eigenvalue is given. We identify an infinite class of threshold graphs with integral spectra. And finally we obtain a simple algorithm to compute the characteristic polynomial of a threshold graph G .

AGRADECIMENTOS

Primeiramente agradeço a Deus, por ter me dado força nos momentos mais difíceis e saúde para estudar e trabalhar. Agradeço a minha família, em especial a meu pai Deomar Tura (in memoriam) que sempre me incentivou aos estudos e acreditou em mim. Agradeço à minha esposa, Mariana de Mello da Silva, por sua compreensão e paciência, e por ter ido comigo à Clemson.

Agradeço a todos os colegas da Unipampa, Campus Alegrete, por terem me concedido o afastamento integral para finalizar o meu doutorado. Agradeço aos colegas de pós-graduação em Matemática Aplicada, em especial: Israel, Eliseu e Rodrigo, que de alguma forma contribuíram com suas sugestões e opiniões, através dos nossos seminários semanais, e todos outros colegas do PPGMAp.

Agradeço à coordenadora do PPGMAp, Professora Maria Cristina Variabile, aos Professores Carlos Hoppen, Luiz Emilio Allem, João Batista de Paz Carvalho e à Professora Virgínia Maria Rodrigues, obrigado pela atenção e a disposição de todos vocês.

Agradeço ao Professor David Jacobs de Clemson University que me recebeu e me orientou durante o meu doutorado sanduíche, sempre atencioso e disposto para discutirmos matemática.

Em especial quero agradecer ao meu orientador e amigo Professor Vilmar Trevisan. Obrigado por ter me concedido essa oportunidade de trabalharmos juntos. Em muitos momentos você me motivou e me encorajou a enfrentar uma tese de doutorado.

1 INTRODUÇÃO

O trabalho desenvolvido nesta tese teve como motivação principal as relações existentes entre as propriedades espectrais de matrizes associadas a grafos com a estrutura desses grafos. O estudo que relaciona autovalores, autovetores e polinômios característicos de matrizes associadas a propriedades de grafos é o núcleo da área denominada teoria espectral de grafos.

A teoria espectral de grafos teve origem basicamente na Química. Quando certas moléculas de hidrocarbonetos foram representadas por grafos, como na Figura 1.1, verificou-se no trabalho de Hückel [24], que os autovalores dos grafos estavam relacionados com a energia de certos elétrons associados à molécula.

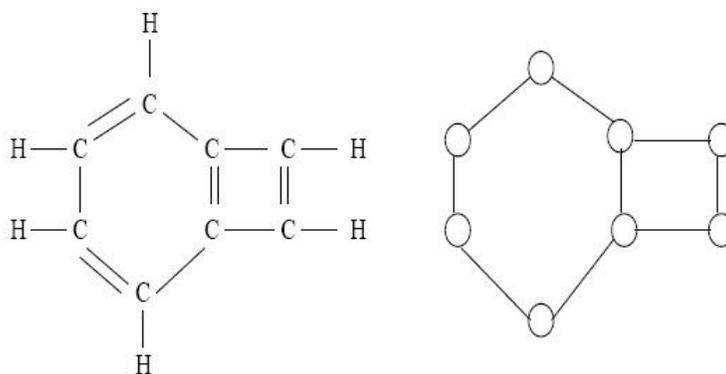


Figura 1.1: Representação de um hidrocarboneto

A teoria espectral de grafos é uma área de crescente interesse na comunidade científica internacional, o que é evidenciado pelo número de publicações nessa área, pela quantidade de conferências e pelo gabarito dos matemáticos que atualmente trabalham nela. O leitor interessado poderá consultar a página Spectral Graph Theory [34] para poder comprovar alguns destes dados.

As aplicações da teoria espectral de grafos não se limitam apenas à Química. Podemos citar diversas aplicações, como por exemplo, a que utiliza os autovalores da matriz distância de um grafo para resolver problemas de comunicação de dados [20].

De modo geral podemos dizer que o objetivo central da teoria espectral de grafos é poder descrever as propriedades de grafos a partir do seu espectro, ou seja, dos autovalores das matrizes associadas aos grafos. Por exemplo, em relação à matriz de adjacência A de um grafo $G = (V, E)$ de ordem n , definida por

$$a_{ij} = \begin{cases} 1, & \text{se } (v_i, v_j) \in E \\ 0, & \text{nos outros casos.} \end{cases}$$

A partir do polinômio característico de G , obtido por $p(\lambda) = \det(A - \lambda I)$, ou seja,

$$p(\lambda) = \lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + \dots + a_{n-1}\lambda + a_n,$$

podemos obter informações sobre o grafo, tais como:

- o número de arestas é igual a $-a_2$;
- o número de triângulos é igual a $-\frac{a_3}{2}$;
- o maior autovalor (índice) de G está entre o grau médio e o grau máximo, onde o grau de um vértice é o número de vértices adjacentes a este.

Um problema encontrado em teoria espectral de grafos e que descreve um grafo G a partir do seu espectro é o seguinte: G é bipartido se, e somente se, possui espectro simétrico em relação a zero. Por exemplo, a Figura 1.2, ilustra o grafo bipartido $K_{3,8}$ cujo espectro é $\{-\sqrt{24}, 0, 0, 0, 0, 0, 0, 0, 0, \sqrt{24}\}$.

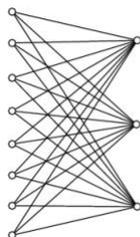


Figura 1.2: Grafo bipartido $K_{3,8}$

De maneira geral, dizemos que um grafo G é caracterizado pelo seu espectro se os grafos coespectrais (grafos com o mesmo espectro) a G são isomorfos a G . Notemos que se dois grafos são isomorfos eles têm o mesmo espectro. Idealmente gostaríamos que a recíproca fosse verdadeira, isto é, que todo grafo fosse caracterizado pelo seu espectro. No entanto, a recíproca não é verdadeira. Como exemplo, a Figura 1.3, apresenta grafos coespectrais e não isomorfos.

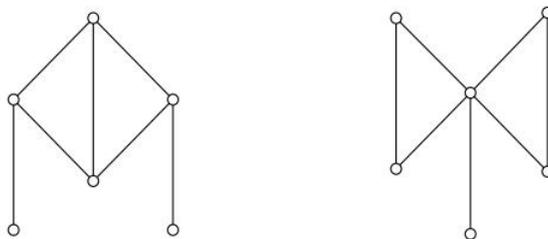


Figura 1.3: Grafos coespectrais

Neste trabalho pressupomos que o leitor tenha conhecimentos básicos em grafos e em Álgebra Linear. Apontamos algumas referências bibliográficas que solidificam ainda mais a teoria espectral de grafos, como o livro clássico de Cvetković, Doob e Sachs [15], mais recentemente temos o de Brouwer e Haemers [10] e também o de Abreu, Del-Vecchio, Vinagre e Stevanović [16].

A classe de grafos considerada neste trabalho é denominada grafos *threshold*, traduzidos por grafos de limiar. O estudo de grafos *threshold* é motivador por várias razões. A primeira razão é a importância em aplicações. Os grafos *threshold* têm várias aplicações, em psicologia, sincronização de processos paralelos, eficiência e estabilidade de estruturas de rede e outros [33, 35]. Provavelmente por este motivo, eles aparecem na literatura com vários nomes equivalentes e surgem em contextos variados.

Outra razão que torna esta classe de grafos interessante é que existem muitas formas de caracterizá-los, tais como grafos livres de $\{2K_2, C_4, P_4\}$, grafos *split nested*, como uma subclasse dos cografos, unigrafos, através de uma sequência binária, etc.

Como citamos anteriormente, existe um considerável campo de conhecimento sobre propriedades espectrais de grafos. Em relação aos cografos, que contém a classe dos grafos *threshold*, foram estudadas por Royle em [38], e recentemente por Simić, Stanić e Biyikoglu em [7]. Propriedades espectrais de grafos *threshold* foram estudadas por Sciriha e Farrugia em [39], e também por Simić, Bellardo, Marzi, Tosić em [40], e por Stanić em [41].

Nesta tese trabalhamos com apenas dois tipos de matrizes associadas aos grafos *threshold*: a matriz de adjacência e a de distância. A razão para não estudarmos a matriz laplaciana é simples, já que o espectro laplaciano de um grafo *threshold* G é facilmente determinado através da sequência dos graus dos vértices.

Considerando G um grafo *threshold* e $d_1 \geq d_2 \geq \dots \geq d_n$ a sequência de graus dos vértices, (ver, por exemplo [36]), o espectro laplaciano de G consiste nos autovalores

$$\mu_i = |\{j : d_j \geq i\}|,$$

ou seja, o espectro é determinado a partir do conjugado de sua sequência de graus. Por exemplo, a Figura 1.4 ilustra o grafo *threshold* G e a sequência de graus representada por um diagrama de Ferrers. A sequência de graus dos vértices de G , que corresponde às linhas do diagrama é igual a $[7, 6, 5, 4, 4, 3, 2, 1]$, enquanto que o espectro laplaciano de G , que corresponde às colunas do diagrama é igual a $[8, 7, 6, 5, 3, 2, 1, 0]$.

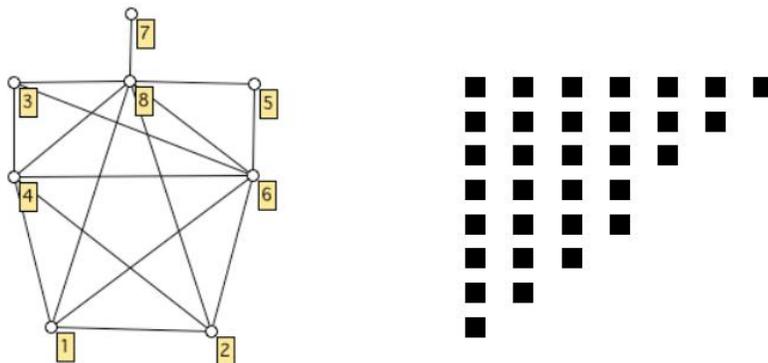


Figura 1.4: Grafo *threshold* e o diagrama de Ferrers

Existem dois pontos relevantes para o nosso algoritmo desenvolvido para matrizes associadas a grafos *threshold* que gostaríamos de enfatizar. O primeiro deles é que, embora o nosso algoritmo seja baseado em operações elementares de matrizes, na prática sua implementação não requer o armazenamento dessas matrizes, ou seja, o algoritmo pode ser implementado em dois vetores de comprimento n , de modo que sua complexidade é de ordem $O(n)$.

O segundo ponto é que, enquanto podemos esperar algoritmos de tempo linear para grafos com matrizes de adjacência esparsas (por exemplo árvores), os grafos *threshold* têm matrizes de adjacência que podem ser densas, pois estes grafos podem ter muitas arestas e, em particular, apresentar muitos ciclos, como mostra a Figura 1.6. Sob o ponto de vista numérico, anular as entradas da matriz de adjacência de um grafo com muitas arestas pode ser um problema difícil. Nesse sentido nosso algoritmo tira vantagem da estrutura da matriz de adjacência de grafos *threshold*: mesmo que ela possua muitos elementos não nulos, nosso algoritmo é eficiente para a diagonalização.

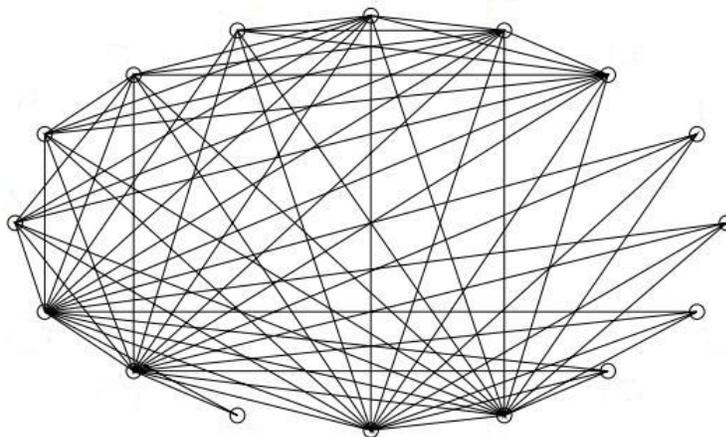


Figura 1.6: Grafo threshold

Apresentamos várias aplicações do algoritmo de diagonalização para grafos *threshold*. Dentre elas podemos destacar: determinamos o número de autovalores de G em um intervalo real $(a, b]$, o que permite localizar os autovalores de um grafo *threshold* com uma certa precisão; mostramos que todo autovalor $\lambda \neq 0, -1$ de

um grafo *threshold* é simples; determinamos a classe de grafos *threshold* que admite o menor autovalor mínimo entre todos os grafos *threshold* de ordem n ; apresentamos um algoritmo para o cálculo do polinômio característico de um grafo *threshold* G ; obtemos uma ordenação sobre os grafos *threshold* via índice e exibimos uma classe de grafos *threshold* com espectro integral.

Como mencionamos anteriormente, existem várias formas de caracterizar os grafos *threshold*. Dessa forma, um capítulo anterior aos nossos resultados é dedicado ao estudo dos grafos *threshold*. Apresentaremos três caracterizações equivalentes que consideramos importantes para o desenvolvimento do nosso trabalho: via sequência binária, grafos livres de $2K_2$, C_4 e P_4 , e grafos *split nested*. Além dessas caracterizações, ainda no mesmo capítulo, apresentaremos algumas propriedades espectrais dos grafos *threshold* em relação à matriz de adjacência, conhecidas da literatura.

Esta tese está organizada da seguinte forma.

O capítulo 2 desta tese é dedicado à apresentação dos grafos *threshold* e de algumas caracterizações equivalentes. Na seção final, apresentamos uma redução para o cálculo do polinômio característico de um grafo *threshold* em relação à matriz de adjacência, devido Sciriha e Farrugia [39], que diminui o tamanho da matriz original.

No capítulo 3 apresentamos o nosso algoritmo de diagonalização e provamos a simplicidade dos autovalores de um grafo *threshold*. Do capítulo 4 ao 6, obtemos novos resultados espectrais, a partir do algoritmo de diagonalização.

No capítulo 7, fazemos uma extensão de nossos resultados para a matriz distância. Adaptamos o nosso algoritmo de diagonalização para a matriz distância Θ de um grafo *threshold* G . De forma análoga, determinamos quantos autovalores de Θ pertencem a um intervalo real qualquer e provamos também a simplicidade dos autovalores $\lambda \neq -1, -2$.

2 GRAFOS *THRESHOLD*

Este capítulo é dedicado à apresentação dos grafos *threshold*. Definimos os grafos *threshold* de forma apropriada para o nosso propósito, e algumas caracterizações equivalentes são apresentadas. Finalizamos o capítulo, apresentando uma redução para o cálculo do polinômio característico de grafos *threshold*, em relação à matriz de adjacência de acordo com Sciriha e Farrugia [39].

2.1 Introdução

Os grafos *threshold* foram introduzidos por Chvátal e Hammer [11] e por Henderson e Zalcstein [25] nos anos 70. Uma referência mais recente sobre os grafos *threshold* é o livro de Mahadev e Peled [33]. Esse livro apresenta diversas caracterizações destes grafos e faz várias aplicações. Em relação a propriedades espectrais podemos destacar os artigos [5, 6, 39, 40, 41].

Existem muitas formas de caracterizar os grafos *threshold*, tais como grafos livres de $\{2K_2, C_4, P_4\}$, grafos *split nested*, uma subclasse dos cografos, uni-grafos, através de sequência binária, etc. Nessa tese apresentaremos três dessas caracterizações, que mostraremos serem equivalentes.

Uma caracterização dos grafos *threshold* é via sequências binárias, que adotaremos como definição. Escolhemos essa caracterização como definição porque ela permite obter um grafo *threshold* através de processo recursivo. Além disso, essa caracterização será usada para os algoritmos que apresentaremos nos próximos capítulos.

Podemos, também, caracterizar os grafos *threshold* como grafos livres de $\{2K_2, C_4, P_4\}$. A nossa motivação é dada a seguir. Consideremos as seguintes definições.

Definição 2.1.1. *Seja $G = (V, E)$ um grafo, onde V é o conjunto de vértices e E é o conjunto de arestas. Dizemos que dois vértices $u, v \in V$ são adjacentes se $\{u, v\} \in E$, e não adjacentes, caso contrário. Um subconjunto S de V é dito estável em G se os elementos em S são dois a dois não adjacentes em G .*

Definição 2.1.2. *Sejam $G = (V, E)$ um grafo, e $S \subset V = \{v_1, v_2, \dots, v_n\}$ um subconjunto de V estável em G . O vetor característico de S é o vetor (x_1, x_2, \dots, x_n) dado por:*

$$x_i = \begin{cases} 1, & \text{se } v_i \in S \\ 0, & \text{caso contrário.} \end{cases}$$

Exemplo 2.1.1. *Seja $G = K_3$ o grafo completo de 3 vértices representado pela Figura 2.1. Temos que os possíveis conjuntos estáveis em G são \emptyset , $\{u\}$, $\{v\}$, e $\{w\}$. Então os vetores característicos de S são dados, respectivamente por $(0, 0, 0)$, $(1, 0, 0)$, $(0, 1, 0)$, e $(0, 0, 1)$.*

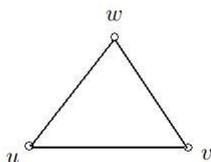


Figura 2.1: K_3

Notemos que os subconjuntos de vértices de K_3 estão numa correspondência biunívoca com os vértices de um cubo unitário no \mathbb{R}^3 de acordo com as coordenadas de seus vetores característicos. Assim temos a seguinte motivação apresentada em [19]: existe um plano que divida o espaço \mathbb{R}^3 ao meio de tal forma que em um dos lados todos os vértices do cubo correspondem aos conjuntos estáveis de G e do outro correspondem a conjuntos não estáveis? Ou equivalentemente, é possível distinguir quais subconjuntos de V são conjuntos estáveis usando uma única desigualdade linear?

Mostra-se que a resposta é positiva exatamente para os grafos que são livres de $2K_2$, C_4 e P_4 .

Em nosso exemplo, como o grafo K_3 em questão é um grafo livre de $2K_2$, C_4 e P_4 então é possível distinguir os conjuntos estáveis dos não estáveis através da seguinte desigualdade linear

$$x + y + z \leq 1,$$

onde $x, y, z \in \{0, 1\}$. Notemos que a desigualdade é satisfeita somente pelos vetores característicos dos conjuntos estáveis de K_3 . O plano que faz a separação geométrica é ilustrado na Figura 2.2.

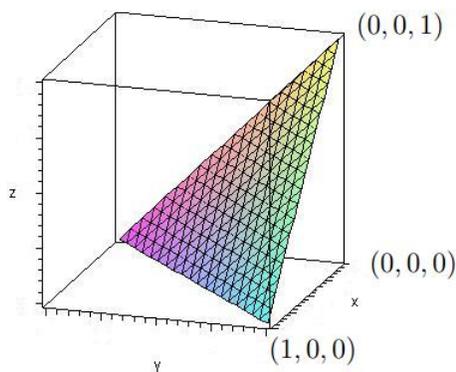


Figura 2.2: Separação do espaço \mathbb{R}^3

Uma terceira caracterização dos grafos *threshold*, é via grafos *split nested*. Dizemos que um grafo $G = (V, E)$ é *split*, se o seu conjunto de vértices V pode ser particionado em uma clique K e um conjunto estável S . Os grafos *split* são considerados uma importante classe de grafos, devido tanto a problemas relacionados à complexidade de algoritmos, quanto a problemas de reconhecimento de grafos k -coloríveis.

De fato, os grafos *threshold* G são grafos *split* com a propriedade *nested*, traduzido por vizinhança. Essa propriedade significa que além dos vértices de G formarem uma clique K e um conjunto estável S , podemos particionar K e S , em subconjuntos disjuntos tais que vértices pertencentes a K_i e S_i , admitem a mesma quantidade de vizinhos.

A importância de caracterizar grafos *threshold* via grafos *split nested* é precisamente espectral. Veremos na seção final desse capítulo, que tal estrutura fornece uma partição mínima no conjunto de vértices de G e induz um grafo denominado quociente, cujo polinômio característico está relacionado com o de G . A partir disso, podemos obter uma redução para o cálculo do polinômio característico de G .

Para obtermos tal redução do polinômio característico de G , será necessário apresentarmos a classe de grafos denominada de cografos. Dessa forma, na próxima seção faremos uma descrição breve dos cografos.

2.2 Cografos

Os cografos foram introduzidos ao mesmo tempo que os grafos *threshold* por Lerchs na década de 70. Posteriormente Corneil *et al.* [12] desenvolveram um algoritmo de complexidade linear para reconhecer tal classe.

Os cografos são obtidos através de um número finito de operações de união e junção que definiremos a seguir de acordo com [7]. Um cografo também pode ser caracterizado como um grafo livre de P_4 , o que significa que grafos *threshold* são uma subclasse dos cografos.

Antes de definirmos um cografo, consideremos as seguintes operações: Dados dois grafos disjuntos A e B , dizemos que um grafo $G = (V, E)$ é a união de A e B e escrevemos $G = A \oplus B$, se $V(G) = V(A) \cup V(B)$ e $E(G) = E(A) \cup E(B)$. Definimos o grafo junção de A e B , denotado por $G = A \otimes B$ obtido a partir de $A \oplus B$ adicionando-se todas as arestas (a, b) , com $a \in V(A)$ e $b \in V(B)$.

Definição 2.2.1. *Um cografo é obtido recursivamente a partir de K_1 e por operações de união ou junção, da seguinte forma:*

(i) K_1 é um cografo,

- (ii) Se G_1 e G_2 são cografos, então a união $G_1 \oplus G_2$ também o é, e
 (iii) Se G_1 e G_2 são cografos, então a junção $G_1 \otimes G_2$ também o é.

Definição 2.2.2. Dizemos que um subgrafo H de G é um subgrafo induzido de G se, para qualquer par de vértices x e y de H , (x, y) é uma aresta de H se e somente se (x, y) é uma aresta de G .

Os cografos admitem uma representação única através de árvores, denominada coárvores [7, 13]. Faremos aqui uma breve descrição de como obter tal árvore:

- (i) A coárvore T_G de um cografo G é uma árvore com raiz, onde cada vértice interior w ou é do tipo \oplus (correspondendo à união) ou do tipo \otimes (correspondendo à junção);
 (ii) Os vértices terminais (folhas) da coárvore T_G são do tipo $\{\bullet\}$, correspondendo aos vértices do cografo G ;
 (iii) Qualquer vértice interior w representa um subgrafo de G induzido pelos sucessores terminais de w , e é representado por G_w ;
 (iv) O sucessor direto (ou filho) de qualquer vértice interior w tem tipo diferente de w . Além disso, cada vértice não terminal tem no máximo dois sucessores não terminais;
 (v) Todos os vértices interiores de um caminho qualquer, partindo da raiz até os vértices terminais, forma um caminho alternado (\oplus, \otimes) .

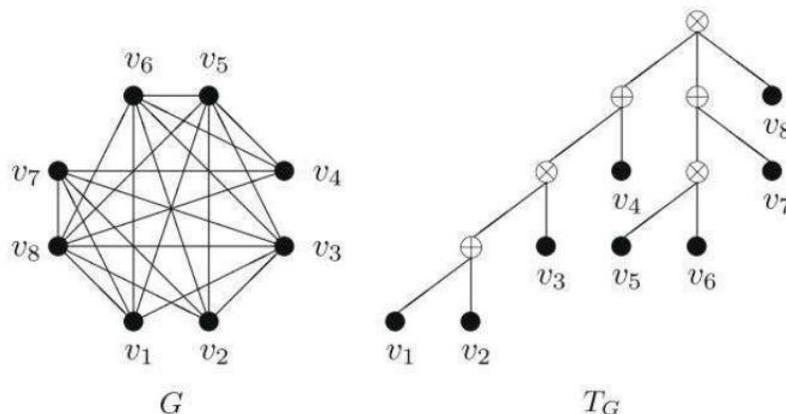


Figura 2.3: Um cografo G e a sua coárvore T_G .

A Figura 2.3 ilustra a coárvore T_G do cografo obtido recursivamente pela seguintes operações $G = \{((v_1 \oplus v_2) \otimes v_3 \oplus v_4) \otimes ((v_5 \otimes v_6) \oplus v_7)\} \otimes v_8$. Podemos observar que a raiz de T_G é representada por \otimes , se G é conexo, e por \oplus , se G for desconexo.

Lema 2.2.1. *Todo subgrafo induzido de um cografo é um cografo.*

Prova: Como qualquer subgrafo pode ser obtido removendo-se vértices um por um, é suficiente mostrar que removendo-se apenas um único vértice a partir de um cografo, ele ainda permanece um cografo. Seja $G(V, E)$ um cografo e T_G sua coárvore. O subgrafo G' induzido por $V - \{y\}$ é um cografo, se e somente se, a coárvore $T_{G'}$ pode ser associada a G' . Para a construção de $T_{G'}$, verificamos o vértice x pai de y em T_G , e consideremos os seguintes casos:

- (i) x tem mais do que dois sucessores, construímos $T_{G'}$ removendo a folha y de T_G .
- (ii) x tem exatamente dois sucessores: y e y' . Se y' é uma folha, construímos $T_{G'}$ removendo x e y de T_G e conecte y' aos pais de x . Se y' não é uma folha, remova x, y e y' a partir de T_G e conecte todos os sucessores de y' aos pais de x . Em ambos casos, $T_{G'}$ é uma coárvore do subgrafo G' , e assim estabelece que G' é um cografo. \square

Lema 2.2.2. *Seja $G = (V, E)$ um grafo. As seguintes afirmações são equivalentes:*

- (1) G não tem um subgrafo induzido isomorfo a P_4 .
- (2) O complementar de qualquer subgrafo induzido, conexo, não trivial de G é desconexo.

Prova: (2) \Rightarrow (1) Suponha que G admite um subgrafo induzido conexo isomorfo a P_4 . Como $P_4 \simeq P_4^C$, então o complementar de G admite um subgrafo induzido conexo isomorfo a P_4^C . (1) \Rightarrow (2) Provaremos por indução sobre o número de vértices. Seja G um grafo $|V(G)| = p$. Assumimos que o resultado vale para $n < p$. Considere um subgrafo de G , retirando-se um vértice, ou seja, $G' = G - \{x\}$. Se x é um vértice não adjacente em G' , então todos os subgrafos induzidos conexos não triviais de G estão em G' . Logo por hipótese de indução segue que o argumento é verdadeiro. Se x não

é um vértice isolado em G , então existe um vértice $y \in G' : (x, y) \in E(G)$. Suponha por absurdo que existe uma componente conexa C de G , tal que sua complementar \bar{C} ainda é conexa. Isso implica que $x \in C$, pois caso contrário x não seria adjacente em G . Temos então que o grafo $\bar{C} - x = \bar{C}_1 \cup \bar{C}_2$ é desconexo por hipótese de indução. Então existem $z \in V(\bar{C}_1) - \{y\}, w \in V(\bar{C}_2) : (z, w) \notin E(\bar{G}')$. Assumimos então que $y \in V(\bar{C}_1)$. Seja Q o menor caminho em \bar{G}' , unindo y a w . Então o subgrafo em \bar{G}' induzido pelas arestas $\{y, z\}, \{z, x\}$ e $\{x, w\}$ é isomorfo a P_4 .

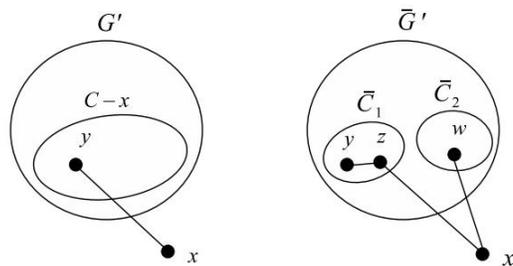


Figura 2.4: Grafos G' e \bar{G}' , respectivamente.

O teorema abaixo, caracteriza os cografos como grafos livres de P_4 , apresentado em [13].

Teorema 2.2.1. *Um grafo é um cografo se e somente se ele não admite um P_4 como um subgrafo induzido.*

Prova: Seja G um cografo. Mostraremos que G satisfaz a propriedade (2) do Lema 2.2.2. Seja H um subgrafo induzido conexo de G . Como todo subgrafo de um cografo é um cografo, o complementar de H é obtido trocando-se os vértices interiores da coárvore T_H , o que prova que H^C é desconexo. Agora suponha que G é um grafo que não admite P_4 como um subgrafo induzido. Assumimos por indução que tal afirmação vale para $n < p$. Considerando que um cografo é preservado sob o complementar, temos que verificar se G ou G^C é um cografo. Sem perda de generalidade, assumimos que G^C é desconexo. Por hipótese de indução cada uma das componentes conexas de G^C é um cografo, assim G^C é um cografo obtido através de uniões disjuntas. \square

2.3 Caracterizações

A partir de agora apresentaremos os grafos *threshold*. Para uma melhor compreensão dessa classe de grafos e dos resultados espectrais a seguir, apresentaremos três caracterizações equivalentes. Iniciaremos com a definição de grafos *threshold* que adotaremos neste trabalho.

2.3.1 Definição

Como mencionamos anteriormente, existem várias formas de caracterizar os grafos *threshold*, mas basicamente eles podem ser obtidos através de um processo recursivo que inicia com um vértice isolado e onde, a cada passo, ou um novo vértice isolado é adicionado ou um vértice adjacente a todos os vértices anteriores é adicionado (vértice *dominante*).

A partir desse processo, uma maneira conveniente de descrever um grafo *threshold* G com n vértices, é considerar uma sequência de tamanho n composta por caracteres 0 e 1. Denotaremos sempre por 0 o primeiro caractere da sequência, ele representa o primeiro vértice do grafo, de forma que 0 representa a adição de um vértice isolado e 1 a adição de um vértice dominante.

Definição 2.3.1. *Um grafo threshold $G = (V, E)$ de ordem n é definido através de uma sequência binária (b_1, b_2, \dots, b_n) , onde $b_i = 0$, representa adição de um vértice isolado e $b_i = 1$, representa adição de um vértice dominante.*

Exemplo 2.3.1. *A Figura 2.5, mostra a construção de um grafo threshold $G = (V, E)$ que começa com 3 vértices isolados, seguido de 1 vértice dominante, 4 vértices isolados e 1 vértice final dominante. A sequência binária que representa o grafo threshold G é $(0, 0, 0, 1, 0, 0, 0, 0, 1)$.*

Notemos que um grafo *threshold* G é conexo, se e somente se, $b_n = 1$.

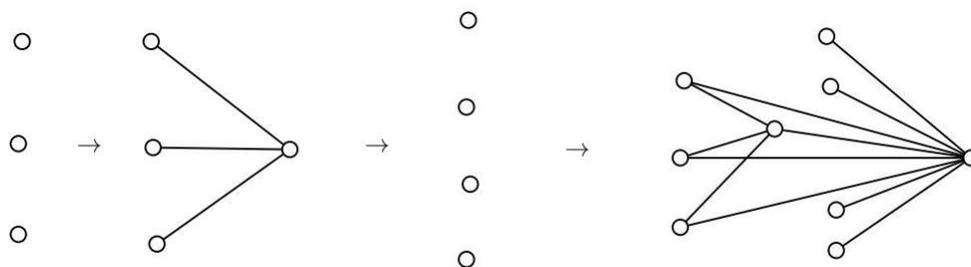


Figura 2.5: Construção de um grafo *threshold*

2.3.2 Caracterização livre de $2K_2$, C_4 e P_4

Dentre diversas propriedades estudadas em teoria de grafos, uma propriedade importante é saber determinar se um grafo pertence ou não a uma certa classe. Dessa forma uma primeira caracterização alternativa é mostrar que eles são exatamente os grafos que não admitem como subgrafos induzidos $2k_2$, C_4 e P_4 . Essa é a caracterização clássica e feita por Chvátal e Hammer [11].

Definição 2.3.2. Dada uma matriz $A = [a_{ij}]$ de ordem $m \times n$, com $a_{ij} \in \{0, 1\}$, o grafo interseção das colunas de A , $G(A)$ é o grafo obtido da seguinte maneira:

- (1) Os vértices de $G(A)$ estão em correspondência biunívoca com as colunas de A ,
- (2) Dois vértices distintos são adjacentes se e somente se as colunas correspondentes têm um produto escalar usual positivo.

Exemplo 2.3.2. A Figura 2.6 apresenta uma matriz A e o grafo interseção das colunas de A .

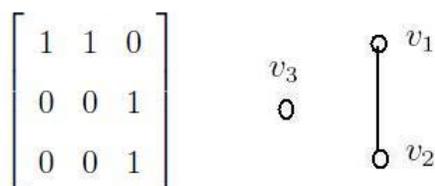


Figura 2.6: Matriz A e o grafo interseção $G(A)$.

Generalizando o problema proposto na introdução deste capítulo, consideremos $A = [a_{ij}]$ uma matriz de ordem $m \times n$, com $a_{ij} \in \{0, 1\}$. Desejamos encontrar as soluções da desigualdade

$$Ax \leq 1, \quad x \in \{0, 1\}, \quad (2.1)$$

que podem ser escritas em uma única desigualdade linear

$$c_1x_1 + c_2x_2 + \dots + c_nx_n \leq b \quad (2.2)$$

para $c_i, b \in \mathfrak{R}$. Notemos que as soluções zero-um da desigualdade (2.1) são precisamente os vetores característicos dos conjuntos estáveis em $G(A)$. Para justificarmos essa afirmação, suponhamos que $x = (x_1, \dots, x_n)$ é um vetor característico, tal que $a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n > 1$. Então o vetor característico e a i -ésima linha $(a_{i1}, a_{i2}, \dots, a_{in})$ têm ao menos duas entradas iguais a 1. Denotando por k e s essas posições, sejam v_k e v_s os vértices de $G(A)$ associados às colunas k e s da i -ésima linha de A , respectivamente. O produto escalar usual dessas colunas é positivo, implicando que os vértices v_k e v_s são adjacentes, logo não existe um conjunto estável contendo v_k e v_s . Assim, para responder o problema proposto, consideremos o seguinte resultado:

Teorema 2.3.1. *Seja um grafo $G = (V, E)$ de ordem n . Se G é um grafo threshold então existem números reais c_1, c_2, \dots, c_n e b de modo que as soluções zero-um de*

$$c_1x_1 + c_2x_2 + \dots + c_nx_n \leq b \quad (2.3)$$

são precisamente os vetores característicos dos conjuntos estáveis em G .

Prova: Provaremos por indução sobre o número de vértices. Se G tem apenas um único vértice, basta tomar $c_1 = b = 1$. Suponhamos que vale para um *threshold* G de ordem n .

Caso 1) Adicionamos um vértice dominante v . Como os conjuntos S estáveis em G não são alterados, apenas acrescenta-se um conjunto estável formado pelo próprio vértice v . Então basta atribuir a v o valor b , e deixamos os outros valores c_i e b

inalterados, obtendo-se

$$\sum_{i=1}^n c_i x_i + b x_{n+1} \leq b.$$

Caso 2) Adicionamos um vértice isolado v . Neste caso, os conjuntos S estáveis podem ser alterados. Para cada conjunto estável S , existe a possibilidade do acréscimo do conjunto estável formado por v . Então dobramos os valores dos antigos vértices, substituímos b por $2b + 1$ e atribuímos a v o valor 1, obtendo-se

$$\sum_{i=1}^n 2c_i x_i + x_{n+1} \leq 2b + 1.$$

□

Dessa forma, como uma solução x que satisfaz (2.1) é precisamente um vetor característico de algum conjunto estável de $G(A)$, segue que às restrições de (2.1) podem ser agregadas as restrições de (2.2) se e somente se $G(A)$ é um grafo *threshold*.

Lema 2.3.1. *Se G é C_4 , $2K_2$, ou P_4 então G não é um *threshold*.*

Prova: Suponhamos que um dos grafos acima é um *threshold*. Então existe uma desigualdade linear $c_1 x + c_2 y + c_3 z + c_4 w \leq b$, cujas soluções zero-um são precisamente os vetores característicos dos conjuntos estáveis de G . Como $\{x, z\}$ e $\{y, w\}$ são conjuntos estáveis comuns desses grafos logo $c_1 + c_3 \leq b$ e $c_2 + c_4 \leq b$. Por outro lado, $\{x, w\}$ e $\{y, z\}$ são conjuntos não estáveis para esses grafos, logo temos que $c_1 + c_4 > b$, e $c_2 + c_3 > b$. Claramente, essas quatro desigualdades são inconsistentes.

□

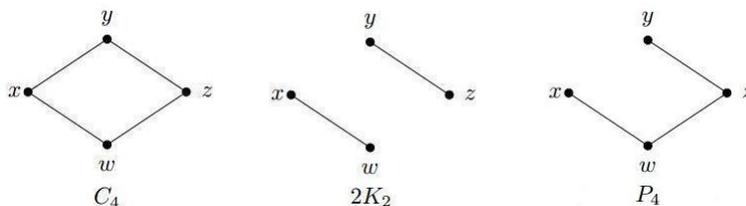


Figura 2.7: Os menores grafos não *threshold*

A Figura 2.7 ilustra os menores grafos não *threshold*: C_4 , $2K_2$ e P_4 , respectivamente.

Lema 2.3.2. *Se G é um grafo threshold, então todo subgrafo induzido de G é um threshold.*

Prova: Pelo Teorema 2.3.1, existem $c_1, c_2, \dots, c_n, b \in \mathfrak{R}$ de modo que as soluções zero-um de

$$\sum_{j=1}^n c_j x_j \leq b \quad (2.4)$$

são os vetores característicos dos conjuntos estáveis em G . Seja H um subgrafo induzido de G , com conjunto de vértices S . Denotemos por \sum^* o somatório sobre todos os subíndices j com $v_j \in S$. Assim, os conjuntos estáveis de H correspondem aos conjuntos estáveis de G que não contêm vértices de $V - S$. Afirmamos que as soluções zero-um de

$$\sum^* c_j x_j \leq b \quad (2.5)$$

são precisamente os vetores característicos dos conjuntos estáveis em H . Se existir uma solução de (2.5) que não é vetor característico de estável em H , então tal solução também satisfaz (2.4), e tal solução não é vetor característico de conjunto estável de G , pois se não é estável em H , não é estável em G . \square

Como os grafos *threshold* são livres de $2K_2, C_4, P_4$, eles são uma subclasse dos cografos e estamos agora interessados em saber que tipo de árvore representa os grafos *threshold*, então consideremos o seguinte resultado.

Teorema 2.3.2. *Se G é um grafo threshold, então cada vértice interior de T_G tem no máximo um vértice interior como um sucessor imediato.*

Prova: O resultado vale para o número de vértices $n \leq 3$. Assim suponhamos que um vértice interior possui mais de um vértice interior como sucessor. Então qualquer subgrafo induzido de G contendo ao menos 4 vértices, tem uma das seguintes representações como córvores, exibidas na Figura 2.8. Como essas córvores representam os grafos C_4 e $2K_2$, respectivamente, então G possui um subgrafo induzido isomorfo a C_4 ou a $2K_2$, contrariando a hipótese de indução.

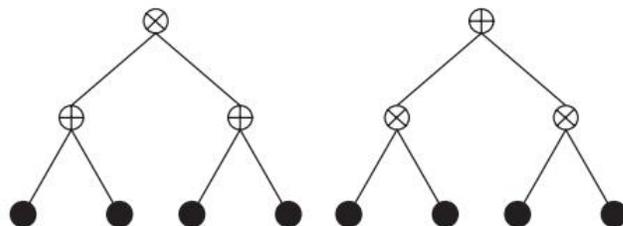


Figura 2.8: Coárvore de C_4 e $2K_2$

Lembramos que uma caterpillar é uma árvore na qual a remoção de todos os vértices terminais (vértices de grau 1) resulta num caminho. No caso em que G é um grafo *threshold*, a representação da coárvore é mais simples, conforme o resultado a seguir, apresentado em [7].

Corolário 2.3.1. *A coárvore de um grafo threshold é uma caterpillar.*

A Figura 2.9 ilustra a coárvore do grafo *threshold* $G = (0, 0, 1, 1, 0, 1, 1)$.

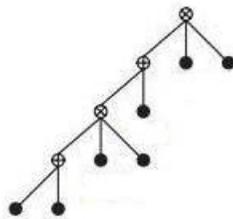


Figura 2.9: Coárvore de um *threshold*

2.3.3 Caracterização via grafos *split nested*

Como mencionamos anteriormente uma terceira caracterização dos grafos *threshold* é através dos grafos *split nested* [40]. Um grafo $G = (V, E)$ é dito *split*, se o seu conjunto de vértices V pode ser particionado em uma clique K e um conjunto

estável S . Mostraremos que os grafos *threshold* são grafos *split* com a propriedade *nested*.

Vamos mostrar que o algoritmo a seguir determina os conjuntos K e S de um grafo *threshold* $G = (V, E)$, se G não é um *threshold*, então o algoritmo encontra um subgrafo induzido de G , isomorfo a $2K_2, C_4$, ou P_4 .

Algoritmo KS

Passo 1. Determinar a sequência dos graus dos vértices de G em ordem não crescente $d = (d_1, d_2, \dots, d_n)$, $K = S = \emptyset$, e $k = n$.

Passo 2. Se $k = 1$ então d tem somente um termo, pare o algoritmo.

Se $k > 1$ sejam u e v o primeiro e último termo da sequência d , respectivamente.

Se $d(u) = |K| + k - 1$, vá para o passo 3. Faça $d(v) = |K|$, vá para o passo 4. Se $|K| < d(v) \leq d(u) < |K| + k - 1$, vá para o passo 5.

Passo 3. Faça $v_k = u$, então delete u de d , substitua K por $K \cup v_k$, k por $k - 1$ e retorne ao passo 2.

Passo 4. Faça $v_k = v$, então delete v de d , substitua S por $S \cup v_k$, k por $k - 1$ e retorne ao passo 2.

Passo 5. Faça $u_1 = u$. Encontre um vértice $u_3 \in d$ que não é adjacente a u_1 e encontre um vértice $u_2 \in d$ que é adjacente a u_3 . Encontre um vértice $u_4 \in d$ que é adjacente a u_1 mas não a u_2 . Então pare, os vértices u_1, u_2, u_3, u_4 induzem um $2K_2$ ou C_4 ou P_4 em G .

Exemplo 2.3.3. Consideremos o grafo G da Figura 2.10. Vamos usar o algoritmo KS para particionar o conjunto de vértices de V .

k	d_1	d_2	d_3	d_4	d_5	d_6	d_7	u, v	K, S
7	6	5	3	3	2	2	1	$u = v_1, v = v_7$	$K = S = \emptyset$
6	5	3	3	2	2	1		$u = v_2, v = v_7$	$K = \{v_1\}, S = \emptyset$
5	5	3	3	2	2			$u = v_2, v = v_6$	$K = \{v_1\}, S = \{v_7\}$
4	3	3	2	2				$u = v_3, v = v_6$	$K = \{v_1, v_2\}, S = \{v_7\}$
3	3	3	2					$u = v_3, v = v_5$	$K = \{v_1, v_2\}, S = \{v_6, v_7\}$
2	3	3						$u = v_3, v = v_4$	$K = \{v_1, v_2\}, S = \{v_5, v_6, v_7\}$
1	3							$u = v_4$	$K = \{v_1, v_2, v_3\}, S = \{v_5, v_6, v_7\}$
									$K = \{v_1, v_2, v_3, v_4\}, S = \{v_5, v_6, v_7\}$

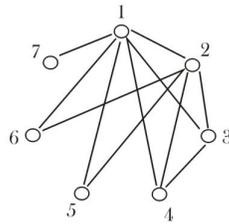


Figura 2.10: Grafo threshold G

Lema 2.3.3. Se $G = (V, E)$ é um grafo threshold então existe uma ordem nos vértices $\{v_1, v_2, \dots, v_n\}$ de G e uma partição de V em dois subconjuntos K e S tais que K é uma clique e S formam um conjunto estável em G .

Prova: Seja $G = (V, E)$ um grafo threshold com n vértices. Enumerando-se em ordem não-crescente os vértices de acordo com os seus graus $d_1 \geq d_2 \geq \dots \geq d_n \geq 0$. Sejam u, v os vértices de maior e menor grau em (d) , e k o tamanho da sequência. Como desejamos que os vértices em K formam uma clique, um candidato ao conjunto K deverá ser um vértice u que é adjacente aos elementos de K e ao resto dos elementos da sequência, ou seja, $d(u) = |K| + k - 1$. Por outro lado, um vértice em S deverá ser adjacente apenas aos elementos de K , ou seja, $d(v) = |K|$. Assim qualquer vértice $w \neq u, v$ satisfaz

$$|K| \leq d(v) \leq d(w) \leq d(u) \leq |K| + k - 1 \quad (2.6)$$

Suponhamos por absurdo que G não é um grafo *split*. Temos a seguinte situação:

$$|K| < d(v) \leq d(w) \leq d(u) < |K| + k - 1 \quad (2.7)$$

Analisaremos que tipo de subgrafo H de G , satisfaz (2.7). Pelo menos dois vértices de H não podem ser incluídos em K ou em S . Sejam u_1 e u_2 com $d(u_1) \geq d(u_2)$. Suponha que existe um vértice u_3 , adjacente a u_1 e u_2 , satisfazendo (2.7), ou seja, $|K| < d(u_2) \leq d(u_3) \leq d(u_1) < |K| + 2$. Isso implica em $d(u_1) = d(u_2) = d(u_3) = |K| + 1$. Dessa forma o vértice u_3 deveria estar em K . Então u_3 só pode ser adjacente a u_2 . Por outro lado, como $d(u_1) \geq d(u_3) \geq d(u_2)$, então deve existir um vértice u_4 adjacente a u_1 , mas não adjacente a u_3 . Dessa forma o subgrafo H de G induzido pelos vértices u_1, u_2, u_3 e u_4 induz um $2K_2$. As outras duas possibilidades de subgrafos ocorrem quando ligamos os vértices u_1 e u_2 obtendo-se um P_4 e quando ligamos u_3 e u_4 , obtendo-se um C_4 . \square

Definição 2.3.3. Dizemos que um grafo $G = (V = (K, S), E)$ é *split nested*, se ambos conjuntos K (clique) e S (estável) são particionados em h células disjuntas $K_1 \cup K_2 \cup \dots \cup K_h$ e $S_1 \cup S_2 \cup \dots \cup S_h$, de modo que todos os vértices de S_i estão ligados através de arestas a todos os vértices de $K_1 \cup K_2 \cup \dots \cup K_i$.

A Figura 2.11 ilustra a estrutura de um grafo *split nested*. O resultado a seguir mostra a equivalência das três caracterizações apresentadas.

Teorema 2.3.3. Para todo grafo $G = (V, E)$, as seguintes afirmações são equivalentes:

- (1) G é um *threshold*.
- (2) G não tem subgrafo induzido isomorfo a $C_4, 2K_2$, e P_4 .
- (3) G é um grafo *split nested* $G = (V = (K, S), E)$.

Prova: (1) \Rightarrow (2) segue dos Lemas 2.3.1 e 2.3.2.

(2) \Rightarrow (3) segue do Lema 2.3.3, que podemos particionar o conjunto de vértices V em uma clique K , e um estável S . Resta mostrar que G é *nested*. Ordenando de acordo com o grau os elementos de $S = \{u_1, u_2, \dots, u_m\}$, em ordem decrescente, definimos o

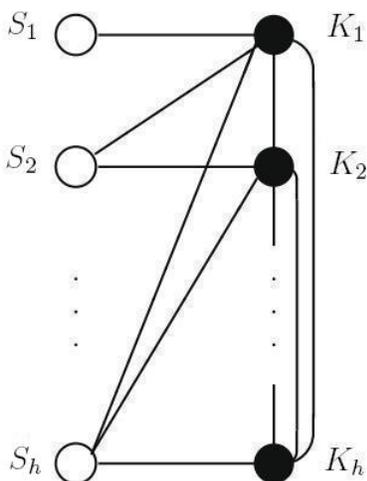


Figura 2.11: Grafo *split nested*

seguinte conjunto $N(u) = \{v \in V : v \text{ é adjacente a } u\}$. Pela construção de S , temos que $N(u_1) \supseteq N(u_2) \supseteq \dots \supseteq N(u_m)$. Definimos a partição de $S = S_1 \cup S_2 \cup \dots \cup S_h$, onde, $S_i = N(u_i) - N(u_{i+1})$. Como cada vértice de S se liga com apenas os vértices de K , a partição em S , induz uma partição em K , ou seja, $K = K_1 \cup K_2 \cup \dots \cup K_h$, como desejávamos.

(3) \Rightarrow (1) Sejam K uma clique e S um estável. Ordenamos os elementos de K e S de acordo com o grau, ou seja, $K = \{v_1, v_2, \dots, v_k\}$ $S = \{v_{k+1}, \dots, v_n\}$. Obtemos a sequência binária (b_i) de n dígitos da seguinte forma:

(i) A sequência é preenchida inicialmente pelos (k) 1's, tal que, o vértice v_k ocupa a primeira posição da sequência e o vértice v_1 a n -ésima posição ($b_1 = 0$ $b_n = 1$);

(ii) Para $2 \leq i \leq n - 1$, $b_i = 0$, se e somente se, existir $v \in S$ tal que $d(v)$ é igual a quantidade de 1's a direita de i .

(iii) Se existir $v \in S$ tal que $d(v)$ é maior que a quantidade de 1's da sequência, então ele deve ocupar a primeira posição da sequência, e o antigo b_1 deve ser igual a 1. □

2.4 Redução do polinômio característico

O objetivo desta seção é apresentar uma redução para o cálculo do polinômio característico de um grafo *threshold* em relação à matriz de adjacência de acordo com [39]. Para obtermos tal redução vamos explorar a estrutura dos grafos *split nested*, através de um grafo quociente. Iniciamos com a definição de partição equilibrada mínima.

2.4.1 A partição equilibrada mínima

Os grafos *threshold* admitem uma partição mínima sobre vértices que pode ser vista através da estrutura dos grafos *split nested*. Consideremos a seguinte definição.

Definição 2.4.1. *Seja $G = (V, E)$ um grafo. Dizemos que uma partição $\pi = V_1 \cup V_2 \cup \dots \cup V_r$ do conjunto de vértices V de G , de tamanho r , é dita equilibrada se, para cada i , e para todo $u, v \in V_i$, temos que $N(u) \cap V_j = N(v) \cap V_j$, para todo j .*

Dizemos que a partição equilibra π é mínima se r é o menor inteiro que particiona o conjunto de vértices de G .

Exemplo 2.4.1. *O grafo G com seqüência de graus $[8, 4, 2, 2, 2, 1, 1, 1, 1]$ tem partição equilibrada $\pi_1 = [\{v_1, v_2\}, \{v_3\}, \{v_4\}, \{v_5, v_6, v_7, v_8\}, \{v_9\}]$, de tamanho $r = 5$, e uma mínima $\pi_0 = [\{v_1, v_2, v_3\}, \{v_4\}, \{v_5, v_6, v_7, v_8\}, \{v_9\}]$ com $r = 4$.*

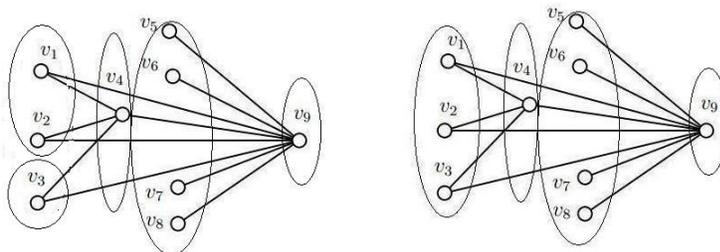


Figura 2.12: Partição equilibrada π_1 e π_0 .

Segue da definição anterior que, a partir da estrutura dos grafos *split nested* podemos obter uma partição equilibrada mínima π , sobre os vértices de G , de tamanho r , escolhendo r como sendo o número de graus distintos do grafo. Assim todas as outras partições do conjunto de vértices de G são refinamentos de π com um número maior de partes.

Dessa forma é conveniente representarmos um grafo *threshold* G com partição equilibrada mínima π de tamanho r por $G(a_1, a_2, \dots, a_r)$, onde cada inteiro positivo a_i , para $1 \leq i \leq r$, é uma clique ou estável de tamanho a_i . Tal representação também pode ser vista, a partir da co-árvore T_G de um *threshold* G , onde os vértices terminais $\{\bullet\}$ que são sucessores imediatos do vértice \otimes formam uma clique K_i , já aqueles imediatos sucessores do vértice \oplus formam um estável S_i . Por exemplo as representações do grafo *threshold* $G(2, 2, 1, 2, 1, 2)$ são exibidas na Figura 2.13.

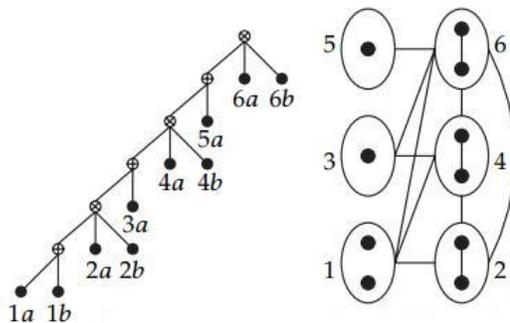


Figura 2.13: Representações do *threshold* $G(2, 2, 1, 2, 1, 2)$

2.4.2 A multiplicidade dos autovalores 0 e -1

Para obtermos a redução para o cálculo do polinômio característico de um grafo *threshold* G , precisamos obter as multiplicidades dos autovalores 0 e -1 . Para isso definiremos vértices denominados de duplicados e coduplicados (gêmeos).

Definição 2.4.2. *Seja $G = (V, E)$ um grafo *threshold*. Um par vértices de G é dito coduplicado (duplicado), se eles são adjacentes (não adjacentes) e têm exatamente os mesmos vizinhos.*

Como a matriz de adjacência depende de forma que os vértices de G são ordenados, tomamos os vértices com grau em ordem decrescente $d(v_1) \geq d(v_2) \geq \dots \geq d(v_n)$ de modo que cada linha l_i de A está associada a cada vértice $v_i \in G$.

Segue que a matriz de adjacência A de um *threshold* G , em que as linhas da matriz A correspondente aos vértices duplicados são idênticas, e para aquelas relacionadas aos vértices coduplicados k e h , a k -ésima e h -ésima linhas diferem somente nas k -ésima e h -ésima entradas. Por exemplo, o grafo *threshold* $G = (2, 2, 1, 2, 1, 2)$ tem um vértice duplicado e três vértices coduplicados. A sua matriz de adjacência A e o grafo são representados, respectivamente na Figura 2.14.

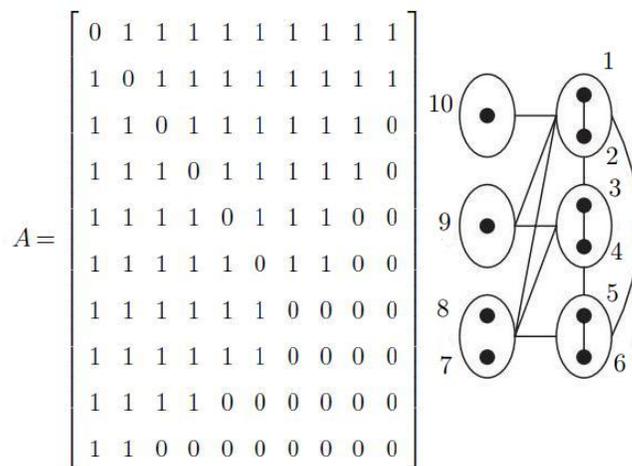


Figura 2.14: Matriz de adjacência e o grafo *threshold*.

É possível determinarmos a multiplicidade dos autovalores 0 e -1 do grafo *threshold* G , conforme dois lemas auxiliares de [38].

Lema 2.4.1. *O posto de um cografo é igual ao número de linhas distintas e não nulas da sua matriz de adjacência A .*

Lema 2.4.2. *Seja G um cografo. Então 0 é um autovalor de G se, e somente se, -1 é um autovalor do complementar G^C .*

Teorema 2.4.1. *A nulidade $\eta(A)$ de um grafo *threshold* G de ordem n é igual ao número de linhas repetidas de sua matriz de adjacência A .*

Prova: Seja A a matriz de adjacência de um grafo *threshold* G de ordem n . Como G também é um cografo segue pelo Lema 2.4.1 que o seu posto é igual ao número de linhas distintas e não nulas. Como a nulidade é obtida por $\eta(A) = n - \text{posto}$, segue o resultado. \square

O seguinte resultado mostra como obter a multiplicidade dos autovalores 0 e -1 , a partir dos vértices duplicados e coduplicados, respectivamente.

Corolário 2.4.1. *Seja $G(a_1, \dots, a_r)$ um grafo threshold com partição equilibrada mínima π de tamanho r . Quando r é par, tem-se*

(i) $\sum_{k=1}^{r/2} (a_{2k-1} - 1)$ vértices duplicados,

(ii) $\sum_{k=1}^{r/2} (a_{2k} - 1)$ vértices coduplicados.

Para r ímpar, $G(a_1, a_2, \dots, a_r)$, tem-se

(i) $\sum_{k=1}^{(r-1)/2} (a_{2k} - 1)$ vértices duplicados,

(ii) $\sum_{k=1}^{(r-1)/2} (a_{2k-1} - 1)$ vértices coduplicados.

Prova: Do Teorema 2.4.1 um vértice duplicado é obtido verificando as linhas idênticas de sua matriz de adjacência A , ou seja, a quantidade de vértices terminais repetidos em T_G que são imediato sucessores de \oplus . Do Lema 2.4.2 um vértice duplicado em G^C é um vértice coduplicado em G , portanto os vértices coduplicados de G são obtidos de vértices terminais repetidos em T_G que são imediato sucessores de \otimes . \square

2.4.3 O grafo quociente

Apresentaremos agora a redução para o cálculo do polinômio característico de um grafo *threshold* G . Tal redução é obtida através de um grafo quociente. Mais precisamente estamos interessados em obter a outra parte do espectro de um grafo *threshold*, ou seja, os autovalores $\lambda \neq 0, -1$. Consideremos a seguinte definição conforme [18].

Definição 2.4.3. Seja $G = (V, E)$ um grafo com partição equilibrada $\pi = V_1 \cup V_2 \cup \dots \cup V_r$. O grafo quociente G/π de G relativo à partição equilibrada π é um multigrafo que tem como vértices os subconjuntos de π e, para $i, j \in \{1, 2, \dots, r\}$, G/π tem q_{ij} arcos de V_i para V_j , onde $q_{ij} = N_G(v) \cap V_j$, para todo $v \in V_i$.

Exemplo 2.4.2. Considere o grafo threshold $G(2, 4, 1, 2)$, com partição equilibrada mínima igual a $\pi = [\{v_1, v_2\}, \{v_3, v_4, v_5, v_6\}, \{v_7\}, \{v_8, v_9\}]$. Os grafos threshold G e quociente G/π são representados na Figura 2.15, respectivamente.

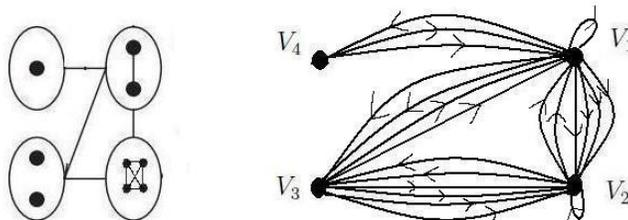


Figura 2.15: Grafos G e G/π , respectivamente.

Segue naturalmente que a definição da matriz de adjacência $A_{G/\pi}$ do grafo quociente G/π é uma matriz de ordem $r \times r$ com entradas q_{ij} .

Seja G um grafo *threshold* com partição equilibrada mínima π . O seguinte resultado relaciona os espectros do grafo quociente G/π e G , de acordo com [41].

Lema 2.4.3. Sejam $\lambda \neq 0, -1$ um autovalor de um grafo threshold $G(a_1, a_2, \dots, a_r)$ e G/π o grafo quociente. Então λ é um autovalor de G/π .

Prova: Pelo Corolário 2.4.1 a multiplicidade do autovalor 0 (respect. -1) de um grafo *threshold* $G(a_1, a_2, \dots, a_r)$ é igual a $\sum_{a_i \in S_i} (a_i - 1)$ (respect. $\sum_{a_i \in K_i} (a_i - 1)$). Assim cada vértice excluído do somatório de duplicados ou coduplicados está associado a um autovalor $\lambda \neq 0, -1$. Então $G(a_1, a_2, \dots, a_r)$ tem exatamente r autovalores distintos de 0 ou -1 . Por outro lado, como espectro do grafo quociente G/π está contido no espectro de G (Teorema 9.3.3 de [18]) e G/π tem exatamente r autovalores então G/π contém o espectro dos autovalores $\lambda \neq 0, -1$ de $G(a_1, a_2, \dots, a_r)$.

Agora apresentamos a redução para o cálculo do polinômio característico.

Teorema 2.4.2. *Seja $G(a_1, a_2, \dots, a_r)$ um grafo threshold com uma partição equilibrada mínima π de tamanho r , tendo α duplicados e β coduplicados. Seja $A_{G/\pi}$ a matriz de adjacência do grafo quociente G/π . Então, $p(G, \lambda) = \lambda^\alpha(\lambda+1)^\beta p(A_{G/\pi}, \lambda)$, onde $p(A_{G/\pi}, \lambda)$ é o polinômio característico do grafo quociente G/π .*

Exemplo 2.4.3. *Seja o grafo threshold $G(2, 4, 1, 2)$ com partição equilibrada mínima dada por $\pi = [\{v_1, v_2\}, \{v_3, v_4, v_5, v_6\}, \{v_7\}, \{v_8, v_9\}]$. Obtemos os vértices duplicados e coduplicados por $\alpha = (2 - 1) + (1 - 1) = 1$, e $\beta = (4 - 1) + (2 - 1) = 4$. Ordenando os vértices de G/π em ordem decrescente de grau, temos que a sua matriz de adjacência é dada por*

$$A_{G/\pi} = \begin{bmatrix} 1 & 4 & 2 & 1 \\ 2 & 3 & 2 & 0 \\ 2 & 4 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix}$$

De acordo com o Teorema 2.4.2, o polinômio característico de $G(2, 4, 1, 2)$ é

$$p(G, \lambda) = \lambda(\lambda + 1)^4 \cdot p(A_{G/\pi}, \lambda)$$

$$p(G, \lambda) = \lambda(\lambda + 1)^4 \cdot (\lambda^4 - 4\lambda^3 - 19\lambda^2 - 6\lambda + 16).$$

3 LOCALIZANDO AUTOVALORES

Neste capítulo apresentamos um algoritmo que determina uma matriz diagonal D congruente a $A + xI$, onde A é a matriz de adjacência de um grafo *threshold* G , x é um número real e I é a matriz identidade. Como aplicação, podemos determinar quantos autovalores de A pertencem a um intervalo qualquer. Na última seção mostramos que todo autovalor $\lambda \neq 0, -1$ de um grafo *threshold* G é simples. Os resultados deste capítulo estão no trabalho [28].

3.1 Introdução

A motivação para este capítulo é o algoritmo desenvolvido em [30] para localizar os autovalores de uma árvore T . Esse algoritmo baseia-se na diagonalização de $A + xI$, e tem várias aplicações, entre elas a verificação de que os autovalores diferentes de zero de uma caterpillar são simples.

Vale lembrar que, enquanto algoritmos de tempo linear desenvolvidos para matrizes de adjacência associadas às árvores podem ser esperados, uma vez que árvores têm matrizes de adjacência esparsas, os grafos *threshold* podem ter muitas arestas e, conseqüentemente, têm matrizes de adjacência que podem ser densas. Entretanto, o algoritmo que desenvolvemos para grafos *threshold*, embora seja baseado em operações elementares de matrizes, na prática, sua implementação só depende da seqüência binária do grafo. De fato, o nosso algoritmo pode ser implementado usando dois vetores de comprimento n e sua complexidade é $O(n)$ em tempo e espaço, onde n é o número de vértices do grafo.

Primeiramente, observamos que a construção da matriz de adjacência de um grafo *threshold* G pode ser obtida através da seqüência binária associada a G , onde a enumeração dos vértices é a mesma da seqüência binária. Por exemplo, a

Figura 3.1 apresenta a matriz de adjacência do grafo *threshold* G representado por $(0, 0, 0, 1, 1, 1, 0, 0, 1)$.

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Figura 3.1: Matriz de adjacência de um *threshold*.

Seja G um grafo *threshold* com sequência binária (b_1, \dots, b_n) , e seja $A = [a_{ij}]$ sua matriz de adjacência. É fácil ver que se $b_i = 1$, então $a_{ij} = a_{ji} = 1$, para $i < j$. E se $b_i = 0$, $a_{ij} = a_{ji} = 0$, para $i < j$.

3.2 Diagonalizando $A + xI$

O algoritmo de diagonalização para grafos *threshold* G constrói uma matriz diagonal D congruente a $A + xI$, onde A é a matriz de adjacência de G , x é um número real e I é a matriz identidade. Lembramos que duas matrizes R e S são congruentes se existe uma matriz P não singular tal que $R = P^T S P$.

Assumimos que A é de ordem n , então o algoritmo executa $n - 1$ passos e opera na matriz de baixo para cima e da direita para esquerda. Em cada passo, as linhas e colunas m e $m - 1$, onde $2 \leq m \leq n$, participam nas operações linha e coluna. A diagonalização é obtida pelo fato que, no fim de cada passo, todas as entradas da linha e coluna m serão nulas com exceção do elemento da diagonal.

Após $n - m$ passos do algoritmo, a matriz parcialmente diagonalizada é da forma exibida na Figura 3.2. O objetivo do passo seguinte é remover os v 's da linha m e coluna m . Notemos que se $v = 0$, não precisamos fazer nada.

$$c_m \leftarrow c_m - c_{m-1}$$

$$\begin{array}{r} m-1 \\ m \end{array} \begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & x & 1-x \\ 0 & \dots & 0 & 1-x & \alpha+x-2 \end{bmatrix}.$$

A maior parte dos elementos não nulos da linha e coluna m foram removidos. Agora devemos remover as duas entradas iguais a $1-x$. Assim, consideraremos três subcasos, dependendo se $\alpha+x-2 \neq 0$ e se $x=1$.

Subcaso 1a: $\alpha+x-2 \neq 0$. Então procederemos com as seguintes operações:

$$l_{m-1} \leftarrow l_{m-1} - \frac{1-x}{\alpha+x-2} l_m$$

$$c_{m-1} \leftarrow c_{m-1} - \frac{1-x}{\alpha+x-2} c_m$$

obtendo

$$\begin{array}{r} m-1 \\ m \end{array} \begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & \gamma & 0 \\ 0 & \dots & 0 & 0 & \alpha+x-2 \end{bmatrix},$$

onde $\gamma = x - \frac{(1-x)^2}{\alpha+x-2} = \frac{\alpha x - 1}{\alpha+x-2}$.

Subcaso 1b: $\alpha+x=2$ e $x=1$. Então a submatriz é da forma:

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & 1 & 0 \\ 0 & \dots & 0 & 0 & 0 \end{bmatrix},$$

ou seja, já está na forma diagonal desejada.

Subcaso 1c: $\alpha + x - 2 = 0$ e $x \neq 1$. Então a matriz é da forma:

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & x & 1-x \\ 0 & \dots & 0 & 1-x & 0 \end{bmatrix}.$$

Como $1 - x \neq 0$, para cada $i, 1 \leq i \leq m - 2$ fazemos :

$$l_i \leftarrow l_i - \frac{1}{1-x} l_m$$

$$c_i \leftarrow c_i - \frac{1}{1-x} c_m$$

Isso anula os 1's na coluna $m - 1$ e linha $m - 1$, resultando

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & x & 1-x \\ 0 & \dots & 0 & 1-x & 0 \end{bmatrix}.$$

Para remover as entradas iguais a $1 - x$, fazemos as seguintes operações:

$$l_{m-1} \leftarrow l_{m-1} + \frac{1}{2} l_m$$

$$c_{m-1} \leftarrow c_{m-1} + \frac{1}{2} c_m.$$

Com isso substituímos a diagonal x por 1, e a seguir fazemos

$$l_m \leftarrow l_m - (1 - x)l_{m-1}$$

$$c_m \leftarrow c_m - (1 - x)c_{m-1}$$

para obtermos finalmente a matriz diagonalizada

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & -(1-x)^2 \end{bmatrix}.$$

A diferença entre esse subcaso e os anteriores é que as entradas da linha e coluna $m-1$ também foram anuladas. Vamos memorizar isso fazendo a seguinte atribuição:

$$b_{m-1} \leftarrow 0.$$

Caso 2: $b_{m-1} = 0$ e $b_m = 1$. Então a submatriz tem a seguinte forma:

$$\begin{bmatrix} & & & 0 & 1 \\ & & & \vdots & \vdots \\ & & & 0 & 1 \\ 0 & \dots & 0 & x & 1 \\ 1 & \dots & 1 & 1 & \alpha \end{bmatrix}.$$

Neste caso trocamos as linhas m e $m-1$ e as colunas m e $m-1$, obtendo a matriz:

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & \alpha & 1 \\ 0 & \dots & 0 & 1 & x \end{bmatrix}.$$

Consideraremos dois subcasos, dependendo se x é zero ou não.

Subcaso 2a: $x = 0$. Então para cada i , $1 \leq i \leq m - 2$ fazemos as operações:

$$l_i \leftarrow l_i - l_m$$

$$c_i \leftarrow c_i - c_m$$

que anulam os 1 's da linha e coluna $m-1$, à esquerda e acima de α . Com as operações

$$l_{m-1} \leftarrow l_{m-1} + \frac{1-\alpha}{2} l_m$$

$$c_{m-1} \leftarrow c_{m-1} + \frac{1-\alpha}{2} c_m$$

$$l_m \leftarrow l_m - l_{m-1}$$

$$c_m \leftarrow c_m - c_{m-1}$$

obtemos a matriz

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & -1 \end{bmatrix}.$$

Subcaso 2b: $x \neq 0$. Então fazemos as operações:

$$l_{m-1} \leftarrow l_{m-1} + \frac{1}{x} l_m$$

$$c_{m-1} \leftarrow c_{m-1} + \frac{1}{x} c_m$$

obtendo a matriz:

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & \alpha - \frac{1}{x} & 0 \\ 0 & \dots & 0 & 0 & x \end{bmatrix}.$$

Como a linha e coluna $m - 1$ estão preenchidas por 1's, à esquerda e acima do elemento da diagonal, fazemos a seguinte atribuição:

$$b_{m-1} \leftarrow 1.$$

Caso 3: $b_m = 0$. Neste caso não é preciso fazer alguma operação já que a submatriz é da forma

$$\begin{bmatrix} & & & & 0 \\ & & & & \vdots \\ & & & & \vdots \\ & & & & 0 \\ \dots & & & x & 0 \\ 0 & \dots & 0 & 0 & \alpha \end{bmatrix}.$$

O algoritmo de diagonalização baseado nestas operações é apresentado na Figura 3.3. Notemos que a única informação importante é a descrição do grafo e os valores da diagonal, então ele pode assim ser implementado com dois vetores de comprimento n . Assumimos que G é representado por (b_1, b_2, \dots, b_n) , então os valores da matriz diagonal D são dados por (d_1, d_2, \dots, d_n) .

Teorema 3.2.1. *Dados G e $x \in \mathfrak{R}$, onde G é um grafo threshold com matriz de adjacência A , o algoritmo de diagonalização calcula uma matriz diagonal D congruente a $A + xI$.*

3.2.1 Exemplos

Ilustraremos o algoritmo com dois exemplos. Nosso primeiro exemplo assumimos que G é um *threshold* representado por $(0, 1, 1, 1)$ e $x = 0$. Após a inicialização o algoritmo executa três passos: $m = 4, 3, 2$.

Para $m = 4$, já que $b_3 = b_4 = 1$ e $\alpha = x = 0$, o primeiro passo o algoritmo executa o subcaso 1a e as seguintes atribuições são feitas:

$$d_3 \leftarrow \frac{\alpha x - 1}{\alpha + x - 2} = \frac{1}{2}$$

```

Algorithm Diagonalize( $G, x$ )
  initialize  $d_i \leftarrow x$ , for all  $i$ 
  for  $m = n$  to 2
     $\alpha \leftarrow d_m$ 
    if  $b_{m-1} = 1$  and  $b_m = 1$ 
      if  $\alpha + x \neq 2$  //subcase 1a
         $d_{m-1} \leftarrow \frac{\alpha x - 1}{\alpha + x - 2}$ 
         $d_m \leftarrow \alpha + x - 2$ 
      else if  $x = 1$  //subcase 1b
         $d_{m-1} \leftarrow 1$ 
         $d_m \leftarrow 0$ 
      else //subcase 1c
         $d_{m-1} \leftarrow 1$ 
         $d_m \leftarrow -(1 - x)^2$ 
         $b_{m-1} \leftarrow 0$ 
    else if  $b_{m-1} = 0$  and  $b_m = 1$ 
      if  $x = 0$  //subcase 2a
         $d_{m-1} \leftarrow 1$ 
         $d_m \leftarrow -1$ 
      else //subcase 2b
         $d_{m-1} \leftarrow \alpha - \frac{1}{x}$ 
         $d_m \leftarrow x$ 
         $b_{m-1} \leftarrow 1$ 
  end loop

```

Figura 3.3: Algoritmo de diagonalização

$$d_4 \leftarrow \alpha + x - 2 = -2.$$

Para $m = 3$, o segundo passo do algoritmo também executa o subcaso 1a, já que $b_2 = b_3 = 1$, $x = 0$, e $\alpha = \frac{1}{2}$, logo

$$d_2 \leftarrow \frac{\alpha x - 1}{\alpha + x - 2} = \frac{2}{3}$$

$$d_3 \leftarrow \alpha + x - 2 = -\frac{3}{2}.$$

Para $m = 2$, o passo final do algoritmo é executado o subcaso 2a, já que $b_1 = 0$, $b_2 = 1$, e $x = 0$. Assim,

$$d_1 \leftarrow 1$$

$$d_2 \leftarrow -1.$$

A seguinte tabela fornece os passos do algoritmo em cada passo:

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	0	0	0	0	0	0	1
1	0	1	0	1	$\frac{2}{3}$	1	-1
1	0	1	$\frac{1}{2}$	1	$-\frac{3}{2}$	1	$-\frac{3}{2}$
1	0	1	-2	1	-2	1	-2
inicial		após 1a		após 1a		após 2a	

No segundo exemplo, assumimos que G é representado por $(0, 1, 0, 1)$ e $x = \frac{\sqrt{3}+1}{2}$. Esse exemplo ilustra como o subcaso 1c pode aparecer. Ele também mostra que o caso 3 pode ocorrer, mesmo num grafo sem vértices isolados.

Após a inicialização, o primeiro passo será o subcaso 2b, já que $b_3 = 0, b_4 = 1$ e $\alpha = x \neq 0$. Note que $\frac{1}{x} = \sqrt{3} - 1$. Assim as atribuições são:

$$d_3 \leftarrow x - \frac{1}{x} = \frac{3 - \sqrt{3}}{2}$$

$$d_4 \leftarrow x = \frac{\sqrt{3} + 1}{2}$$

$$b_3 \leftarrow 1.$$

Como $b_3 = 1$, o próximo passo é o caso 1, já que $b_2 = 1$. Como $d_3 + x = 2$ e $x \neq 1$, o subcaso 1c será executado:

$$d_2 \leftarrow 1$$

$$d_3 \leftarrow -(1 - x)^2 = \frac{\sqrt{3} - 2}{2}$$

$$b_2 \leftarrow 0.$$

Finalmente, já que $b_2 = 0$, o passo final é o caso 3 e assim não é necessário alguma operação. A seguinte tabela fornece os passos do algoritmo em cada passo.

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	$\frac{\sqrt{3}+1}{2}$	0	$\frac{\sqrt{3}+1}{2}$	0	$\frac{\sqrt{3}+1}{2}$	0	$\frac{\sqrt{3}+1}{2}$
1	$\frac{\sqrt{3}+1}{2}$	1	$\frac{\sqrt{3}+1}{2}$	0	1	0	1
0	$\frac{\sqrt{3}+1}{2}$	1	$\frac{3-\sqrt{3}}{2}$	1	$\frac{\sqrt{3}-2}{2}$	1	$\frac{\sqrt{3}-2}{2}$
1	$\frac{\sqrt{3}+1}{2}$	1	$\frac{\sqrt{3}+1}{2}$	1	$\frac{\sqrt{3}+1}{2}$	1	$\frac{\sqrt{3}+1}{2}$
	inicial		após 2b		após 1c		após 3

3.2.2 Encontrando autovalores

Sejam G um grafo *threshold*, A sua matriz de adjacência e $(a, b]$ um intervalo real. Nosso objetivo é calcular o número de autovalores de G em $(a, b]$. O seguinte resultado é conhecido como a lei da Inércia de Sylvester . A prova pode ser encontrada em [9].

Teorema 3.2.2. *Duas matrizes reais, simétricas, de ordem $n \times n$ são congruentes se e somente se elas têm o mesmo número de autovalores positivos e o mesmo número de autovalores negativos.*

O seguinte resultado estabelece uma relação entre os autovalores de G com a matriz D obtida pelo algoritmo de diagonalização.

Teorema 3.2.3. *Seja D a matriz diagonal resultante da aplicação do algoritmo de diagonalização para $(G, -a)$, onde G é um grafo *threshold*. Então:*

i. O número de autovalores de G maiores do que a é o número de entradas positivas

em D .

ii. O número de autovalores de G menores do que a é o número de entradas negativas em D .

iii. A multiplicidade de um autovalor a é o número de entradas nulas na diagonal de D .

Prova: Seja A a matriz de adjacência de G , com autovalores

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \quad (3.1)$$

e sejam

$$\beta_1 \geq \beta_2 \geq \dots \geq \beta_n \quad (3.2)$$

os autovalores de $B_{-a} = A - aI$. Notemos que $\beta_i = \lambda_i - a$. Assim $\lambda_i > a$ se e somente se $\beta_i > 0$. Agora sejam

$$d_1, d_2, \dots, d_n \quad (3.3)$$

os elementos da diagonal de D . Pelo Teorema 3.2.1, D e B_{-a} são congruentes. Pelo Teorema 3.2.2, o número de valores positivos em (3.3) é exatamente o número de valores positivos em (3.2), que é o mesmo número de elementos de (3.1) que são maiores do que a . Isto prova a parte i. A parte ii. é semelhante. Agora para a parte iii., basta observar que se 0 aparece em (3.3) j vezes, então pelo Teorema 3.2.2, ele aparece em (3.2) j vezes. Logo a aparece j vezes em (3.1). \square

Corolário 3.2.1. *O número de autovalores de um grafo threshold G com matriz de adjacência A num intervalo $(a, b]$, incluindo multiplicidades, é o número de entradas positivas na diagonalização de $A - aI$, menos o número de entradas positivas na diagonalização de $A - bI$.*

Segue do corolário acima que podemos determinar o número de autovalores de um grafo *threshold* em um intervalo, chamando o algoritmo de diagonalização duas vezes, como ilustra a Figura 3.4. Se soubermos que o intervalo $(a, b]$ contém ao menos um autovalor de G , então podemos localizar o autovalor à direita,

através de um algoritmo de bissecção, exibido na Figura 3.5, onde a constante ϵ é alguma condição de precisão fixada. O algoritmo pode ser modificado, se desejarmos localizar o autovalor à esquerda.

```

Algorithm RootCount( $G, a, b$ )
   $d1 \leftarrow$  the diagonal computed by  $\text{Diagonalize}(G, -a)$ 
   $d2 \leftarrow$  the diagonal computed by  $\text{Diagonalize}(G, -b)$ 
  return  $|\{i \mid d1(i) > 0\}| - |\{i \mid d2(i) > 0\}|$ 

```

Figura 3.4: Número de autovalores em $(a, b]$.

```

Algorithm LocateRightmost( $G, a, b$ )
  do
     $\gamma \leftarrow (a + b)/2$ 
    if  $\text{RootCount}(G, \gamma, b) > 0$ 
       $a \leftarrow \gamma$ 
    else
       $b \leftarrow \gamma$ 
  while  $b - a > \epsilon$ 
  return  $\gamma$ 

```

Figura 3.5: Localizando autovalor à direita em $(a, b]$.

Exemplo 3.2.1. Consideremos o grafo threshold $G = (0, 0, 0, 1, 1)$. Para $x = 0$, o algoritmo de diagonalização fornece os seguintes valores $d = (0, 0, 1, -1, -2)$. Pelo Teorema 3.2.3, concluímos que $\lambda = 0$ é um autovalor com multiplicidade 2, e que existe um autovalor de G maior do que 0. Quando aplicamos o algoritmo ao mesmo grafo para $x = -4$, obtemos a diagonal com valores $d = (-\frac{3}{4}, -4, -4, -4, -10)$. Isso implica que todos os autovalores de G são menores do que 4. Finalmente podemos concluir que existe um único autovalor no intervalo $(0, 4]$.

3.3 Simplicidade dos autovalores

A partir desta seção apresentaremos novos resultados espectrais obtidos utilizando-se o algoritmo de diagonalização. Nossa primeira aplicação é mostrar que todo autovalor $\lambda \neq 0, -1$ de um grafo *threshold* G é um autovalor simples.

Define-se um autovalor λ como principal se a soma das coordenadas de todo autovetor associado a ele tem soma não nula. Em outras palavras, todo autovetor é não ortogonal ao vetor $(1, 1, \dots, 1)^T$. No trabalho de Sciriha e Farrugia [39], foi provado que todo autovalor de um grafo *threshold* $G, \lambda \neq 0, -1$ é um autovalor principal.

Encontrar grafos com autovalores principais é uma tarefa difícil. Sabe-se por exemplo, que grafos com apenas um autovalor principal são regulares. Cvetković em [14] propôs o problema de caracterizar grafos com exatamente $k > 1$ autovalores principais. Nesse sentido, Hagos caracterizou em [22] os grafos com exatamente dois autovalores principais.

Dizemos que um autovalor λ de um grafo G é simples se λ tem multiplicidade igual a 1. Se G é um grafo *threshold*, segue do algoritmo de diagonalização o seguinte resultado sobre os autovalores principais de G .

Teorema 3.3.1. *Em grafos threshold, a multiplicidade de um autovalor λ é um, a menos que $\lambda = 0$ ou $\lambda = -1$.*

Prova: Assumimos que λ é um autovalor. Então o algoritmo de diagonalização de $(G, -\lambda)$ produz pelo menos um zero na diagonal. Existem três maneiras de um zero aparecer na diagonal:

- (1) quando inicializamos;
- (2) quando atribuímos $d_m \leftarrow 0$ para algum $m \geq 2$;
- (3) quando atribuímos $d_{m-1} \leftarrow 0$ para $m = 2$;

Um zero pode ser produzido quando inicializamos o algoritmo somente quando $\lambda = 0$. Se a atribuição $d_m \leftarrow 0$ ocorre, para algum $m \geq 2$, uma verificação cuidadosa

do algoritmo, mostra que ela só pode ter ocorrido no subcaso 1b. Assim a diagonal inicializada com $x = 1$, implica que $\lambda = -1$. Suponha agora que $\lambda \neq 0, -1$, e que um zero foi produzido no passo final pela atribuição $d_{m-1} \leftarrow 0$ para $m = 2$. Logo somente um zero pode aparecer na diagonal, e sua multiplicidade é um. \square

Assim, temos que os autovalores principais de um grafo *threshold* G são simples.

4 A MINIMALIDADE DO MENOR AUTOVALOR

Neste capítulo utilizamos o algoritmo de diagonalização para determinar o espectro de uma subclasse dos grafos *threshold* que denotamos por $G(k, j)$. Além disso, mostramos que esta subclasse de grafos contém o grafo com o menor autovalor mínimo entre todos os grafos *threshold* de ordem n . Finalizamos o capítulo mostrando quais são os grafos que têm esta propriedade extremal.

4.1 Introdução

Existem muitos resultados na literatura sobre o maior autovalor da matriz de adjacência de um grafo G . Entretanto, pouco se conhece sobre o menor autovalor. Sabe-se que esse menor autovalor é um número negativo. Para grafos com pelo menos uma aresta, o menor autovalor é menor ou igual a -1 , sendo que a igualdade só ocorre se o grafo é completo.

Na tentativa de obter novos resultados para o menor autovalor de um grafo, pesquisadores têm explorado novas técnicas espectrais. Um trabalho recente e muito interessante tem como autores Bell, Cvetković, Rowlinson e Simić [5]. Neste artigo caracteriza-se a estrutura de grafos conexos de ordem n com m arestas, que admitem o menor autovalor entre todos os grafos de ordem n . Tal caracterização é obtida através de uma partição no conjunto de vértices do grafo, dividindo-os em vértices positivos, negativos e nulos, de acordo com o autovetor associado a esse autovalor mínimo. O resultado principal de [5] é enunciado abaixo.

Proposição 4.1.1. *Seja G um grafo conexo de ordem n com m arestas. Se G é o grafo que admite o menor autovalor mínimo entre todos os grafos conexos de mesma ordem, então ou G*

i) é um grafo bipartido, ou

ii) é uma junção de dois grafos split nested (ambos não desconexos).

Notemos que o item ii) do resultado acima é equivalente a dizer que G é uma junção de dois grafos *threshold*. Na verdade, em [5] trata-se apenas do caso ii). Em um trabalho posterior [6], os autores estudam o caso i) dos grafos bipartidos.

O que provamos neste capítulo, na última seção, é um resultado diferente dos apresentados nos trabalhos [5, 6]. Denotando por $\lambda_{\min,n}$ o menor autovalor mínimo dentre todos os grafos *threshold* G de ordem n , provamos qual é a subclasse dos *threshold* que contém o autovalor $\lambda_{\min,n}$ e determinamos quais grafos nesta subclasse atingem a minimalidade. Além disso, exibimos uma fórmula para esse autovalor mínimo. Fazemos isso usando o nosso algoritmo de diagonalização. Primeiramente, vamos determinar o espectro de uma subclasse que nos interessa.

4.2 O espectro de $G(k, j)$

Denotando por $G(k, j)$ a seguinte subclasse de grafos *threshold* com n vértices, tal que $n = k + j$, que possui representação binária da forma

$$(0, 0, \dots, 0, 1, 1, \dots, 1)$$

com $k > 0$ zeros seguidos de $j > 0$ uns. Primeiramente vamos determinar o menor autovalor de $G(k, j)$. Apesar de ser possível determinar o espectro de $G(k, j)$ usando a redução do cálculo do polinômio característico (Teorema 2.4.2) e o Corolário 2.5 de [22], podemos fazer isso usando o algoritmo de diagonalização.

Lema 4.2.1. *Sejam $G = G(k, j)$ um grafo *threshold* e $x \in \mathfrak{R}$. Se a diagonalização de (G, x) executa o subcaso 1a nas primeiras $j - 1$ iterações, então, para cada iteração $i \geq 1$, onde $m = n - i + 1$, o algoritmo faz as atribuições:*

$$d_{m-1} \leftarrow \frac{x + i}{i + 1} \tag{4.1}$$

$$d_m \leftarrow \frac{i + 1}{i}(x - 1) \tag{4.2}$$

Prova: Demonstraremos por indução em i . Para $i = 1$ o algoritmo executa uma única iteração. Neste caso temos que $d_{m-1} = \frac{xx-1}{x+x-2} = \frac{x^2-1}{2x-2} = \frac{x+1}{2}$ e $d_m = x+x-2 = \frac{2}{1}(x-1)$. Assumimos agora que o resultado vale para uma iteração i , ou seja, que as atribuições de (4.1) e (4.2) foram feitas. Então, na iteração $i+1$, o algoritmo atribuirá

$$d_{m-1} \leftarrow \frac{\frac{x+i}{i+1}x - 1}{\frac{x+i}{i+1} + x - 2} \quad (4.3)$$

$$d_m \leftarrow x + \frac{x+i}{i+1} - 2 \quad (4.4)$$

Visto que o lado direito de (4.3) pode ser escrito como:

$$\frac{x(x+i) - (i+1)}{x(i+1) + (x+i) - 2(i+1)} = \frac{(x-1)(x+i+1)}{(i+2)(x-1)} = \frac{x+i+1}{i+2}$$

o que completa a demonstração de (4.1). O lado direito de (4.4) pode ser escrito como:

$$\frac{x(i+1) + (x+i) - 2(i+1)}{i+1} = \frac{x(i+2) - (i+2)}{i+1} = \frac{(i+2)(x-1)}{i+1}$$

completando a demonstração de (4.2). \square

Lema 4.2.2. *Sejam $G = G(k, j)$ um grafo threshold e $x > 1$. Então a diagonalização de (G, x) executa o subcaso 1a nas $j-1$ primeiras iterações e atribui números positivos a todas as entradas d_m .*

Prova: É suficiente mostrar que $\alpha + x - 2 > 0$ para cada uma das $j-1$ primeiras iterações. Como $x > 1$ o algoritmo não executa o subcaso 1b. Suponha que o algoritmo executa o subcaso 1c. Logo temos que $\alpha + x - 2 = 0$. Seja j_0 o primeiro índice tal que isso ocorre. Assim, no passo anterior a j_0 o algoritmo executou o subcaso 1a, e portanto, pelo Lema 4.2.1 temos que

$$\alpha + x - 2 = \frac{x + j_0}{j_0 + 1} + x - 2 = 0 \quad (4.5)$$

De (4.5) obtemos que $x = \frac{j_0+2}{j_0+2} = 1$, contrariando a hipótese. Logo o algoritmo executa apenas o subcaso 1a. Segue que as atribuições de (4.1) e (4.2) são positivas.

\square

Lema 4.2.3. *Sejam $G = G(k, j)$ um grafo threshold e $x \in \mathfrak{R}$. Se a diagonalização de (G, x) executa o subcaso 1a nas $j-1$ primeiras iterações então a j -ésima iteração d_m é inicializada com o valor $\frac{x+j-1}{j}$.*

Prova: O resultado segue do Lema 4.2.1 e do fato que d_{m-1} na iteração $j-1$ torna-se d_m na iteração j . \square

Lema 4.2.4. *Sejam $G = G(k, j)$ um grafo threshold e $x \neq 0$. Se a diagonalização de (G, x) executa o subcaso 1a nas $j-1$ primeiras iterações então nas últimas k iterações é executado o subcaso 2b.*

Teorema 4.2.1. *Seja $G = G(k, j)$ um grafo threshold, onde $k > 1$. O menor autovalor de G é*

$$\lambda = \frac{(j-1) - \sqrt{(j-1)^2 + 4kj}}{2} \quad (4.6)$$

Prova: Seja x a quantidade positiva

$$\frac{-(j-1) + \sqrt{(j-1)^2 + 4kj}}{2} \quad (4.7)$$

Como $k > 1$, temos que $x > 1$. Então pelo Lema 4.2.2, o algoritmo de diagonalização de (G, x) executa o subcaso 1a nas primeiras $j-1$ iterações, atribuindo valores positivos a todos os d_i . Pelo Lema 4.2.3, ele atribui a d_m o valor $\frac{x+j-1}{j}$. Além disso, pelo Lema 4.2.4, o algoritmo executa o subcaso 2b para os passos seguintes. Cada iteração do subcaso 2b, subtrai de $\frac{x+j-1}{j}$ o valor $\frac{1}{x}$. Logo, após k iterações, temos

$$\frac{x+j-1}{j} - \frac{k}{x} \quad (4.8)$$

Para $-x$ ser uma cota inferior para todos os autovalores de $G(k, j)$, precisamos que (4.8) seja não negativo, ou seja,

$$x^2 + (j-1)x - kj \geq 0$$

o que é garantido quando tomamos

$$x \geq \frac{-(j-1) + \sqrt{(j-1)^2 + 4kj}}{2}.$$

\square

Corolário 4.2.1. *Seja $G = G(k, j)$ um grafo threshold, onde $k > 1$. Então o espectro de G é*

$$\text{spect}(G) = \begin{bmatrix} -1 & 0 & \frac{(j-1) - \sqrt{(j-1)^2 + 4(n-j)j}}{2} & \frac{(j-1) + \sqrt{(j-1)^2 + 4(n-j)j}}{2} \\ j-1 & k-1 & 1 & 1 \end{bmatrix},$$

onde os autovalores -1 e 0 têm multiplicidades $j-1$ e $k-1$, respectivamente.

Prova: Seja $G = G(k, j)$ com $k > 1$. Para $x = 1$, o algoritmo de diagonalização de (G, x) executa o subcaso 1b nas $j-1$ primeiras iterações e atribui $j-1$ zeros na diagonal, ou seja, -1 é um autovalor com multiplicidade $j-1$. Aplicando agora o algoritmo de diagonalização para $x = 0$, temos que o algoritmo executa o subcaso 1a nas $j-1$ primeiras iterações. Como $x = 0$, a j -ésima iteração é inicializada com $\alpha = \frac{j-1}{j}$. Além disso, já que $b_{j-1} = 0$ e $b_j = 1$, o subcaso 2a é executado. Assim, para $b_i = 0, 1 \leq i \leq k-1$, os próximos $k-1$ passos do algoritmo não são executados, ou seja, $k-1$ zeros são atribuídos na diagonal da matriz. Portanto, zero é um autovalor com multiplicidade $k-1$. Agora, sejam $\lambda(G)$ e λ , o maior e menor autovalor de $G(k, j)$, respectivamente. Desde que a soma dos autovalores é nula, temos que

$$\lambda(G) + \lambda - (j-1) = 0.$$

Segue de (4.6) que o maior autovalor de $G(k, j)$ é dado por

$$\lambda(G) = \frac{(j-1) + \sqrt{(j-1)^2 + 4kj}}{2}.$$

4.3 O Menor autovalor $\lambda_{min,n}$

Nessa seção mostraremos que os grafos *threshold* G que admitem o menor autovalor mínimo entre todos os grafos *threshold* de ordem n são do tipo $G(k, j)$. Além disso, exibimos para quais k e j o *threshold* $G(k, j)$ tem o menor autovalor mínimo.

Nessa seção consideramos G um grafo *threshold* conexo de ordem $n \geq 3$ possuindo o menor autovalor mínimo $\lambda_{min,n}$, entre todos os grafos *threshold* de ordem n . Nosso objetivo é identificar G e exibir uma fórmula para $\lambda_{min,n}$.

Seja x a quantidade positiva $-\lambda_{min,n}$. Então para qualquer grafo *threshold* G' de ordem n , quando executamos o algoritmo de diagonalização de (G', x) devemos produzir somente valores não negativos nas entradas da diagonal. Portanto é impossível o algoritmo executar o subcaso 1c e o subcaso 2a, porque nestes há atribuição de um valor negativo. Assim para $n \geq 3$ e para $x > 1$, o subcaso 1b não ocorrerá. Também a conectividade impede iniciarmos com o caso 3. Dessa forma, somente os subcasos 1a e 2b podem ocorrer na diagonalização quando tomamos $x = -\lambda_{min,n}$.

A chave para resolver esse problema da minimalidade é entendermos o comportamento das seguintes funções:

$$g(\alpha) = \frac{\alpha x - 1}{\alpha + x - 2} \quad (4.9)$$

e

$$f(\alpha) = \alpha - \frac{1}{x} \quad (4.10)$$

utilizadas nos subcasos 1a e 2b, respectivamente, onde $x \in \mathfrak{R}$ é um valor fixo. Note que g tem uma singularidade em $\alpha = 2 - x$. A Figura 4.1 exibe essas funções quando $x = 4$.

Durante a execução do algoritmo de diagonalização de (G', x) , existe uma sequência de n valores calculados da direita para esquerda

$$\alpha_{\mathbf{G}', \mathbf{x}} = (\alpha_1, \alpha_2, \dots, \alpha_{n-1}, \alpha_n = x) \quad (4.11)$$

que são temporariamente atribuídos na diagonal, que denominaremos por α -sequência. Exceto para o valor final α_1 , cada valor é renovado. Esses valores são calculados por:

$$\alpha_{i-1} = h_i(\alpha_i), \quad 2 \leq i \leq n$$

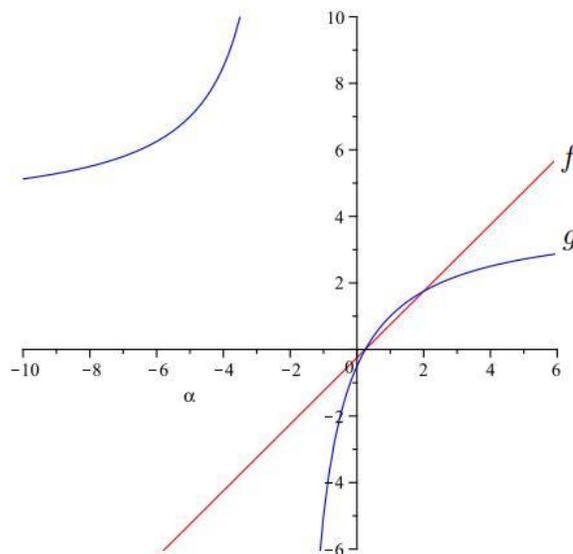


Figura 4.1: As funções f e g quando $x = 4$.

onde

$$h_i(\alpha) = \begin{cases} g(\alpha), & \text{se } b_{i-1} = 1 \\ f(\alpha), & \text{se } b_{i-1} = 0 \end{cases}$$

tal que g e f são dadas por (4.9) e (4.10). Através de uma composição de funções, obtemos

$$\alpha_1 = (h_2 \circ h_3 \circ \dots \circ h_i)(\alpha_i) \quad (4.12)$$

para $2 \leq i \leq n$. Em particular, temos

$$\alpha_1 = (h_2 \circ h_3 \circ \dots \circ h_n)(x) \quad (4.13)$$

A sequência de h_i 's depende somente dos b_i 's originais. Se para $1 < i < n$, b_i é alterado, isso afeta somente h_{i+1} , já que o subcaso 1c é evitado.

Lema 4.3.1. *Ambas as funções f e g são crescentes em $(2 - x, +\infty)$.*

Prova: Basta observar que suas derivadas $\frac{df}{d\alpha} = 1$ e $\frac{dg}{d\alpha} = \frac{(x-1)^2}{(\alpha+x-2)^2}$ são positivas.

Lema 4.3.2. *As funções f e g satisfazem as seguintes propriedades:*

- i. $f(\frac{1}{x}) = g(\frac{1}{x}) = 0$.
- ii. Para $\alpha > \frac{1}{x}$ temos $0 < f(\alpha) < \alpha$.

iii. Para $\alpha > \frac{1}{x}$ temos $0 < g(\alpha) < \alpha$.

iv. Para $\frac{1}{x} < \alpha < 2$ temos $f(\alpha) < g(\alpha)$.

v. $f(2) = g(2)$.

vi. Para $\alpha > 2$ temos $g(\alpha) < f(\alpha)$.

Prova: As propriedades *i.* e *ii.* são facilmente verificadas. Para provar *iii.* devemos mostrar que $\frac{\alpha x - 1}{\alpha + x - 2} < \alpha$. Assumindo que $\alpha + x - 2 > 0$, então obtemos que $0 < (\alpha - 1)^2$. Para verificarmos *iv.* note que $f(\alpha) = \alpha - \frac{1}{x} = \frac{\alpha x - 1}{x}$ que é menor do que $\frac{\alpha x - 1}{\alpha + x - 2}$ se e somente se $\alpha + x - 2 < x$. As propriedades *v.* e *vi.* são facilmente verificadas. \square

Lema 4.3.3. Se $\frac{1}{x} < \alpha_i < 2$ e $h_i = g$ é substituída por f , então α_1 decrescerá, desde que os α_i 's pertençam ao intervalo $(2 - x, +\infty)$.

Prova: Segue da propriedade *iv.* do Lema 4.3.2 que $\alpha'_{i-1} = f(\alpha_i) < g(\alpha_i) = \alpha_{i-1}$. Consideremos $h = h_2 \circ h_3 \circ \dots \circ h_{i-1}$ a composição restante de (4.12). Assim pelo Lema 4.3.1, h deverá ser monótona, logo

$$\alpha'_1 = h(\alpha'_{i-1}) < h(\alpha_{i-1}) = \alpha_1.$$

\square

Usando um argumento semelhante através da propriedade *vi.* do Lema 4.3.2, obtemos o seguinte resultado:

Lema 4.3.4. Se $\alpha_i > 2$ e $h_i = f$ é substituída por g , então α_1 decrescerá, desde que os α_i 's pertençam ao intervalo $(2 - x, +\infty)$.

Lema 4.3.5. Se o subcaso 1c for executado, então a diagonal final terá um valor negativo.

Isso segue da atribuição final dada que é $-(1 - x)^2$, e já que $b_{m-1} \leftarrow 0$.

Lema 4.3.6. Se uma α -seqüência contém elementos em $(-\infty, 2 - x)$, então a diagonal final terá um valor negativo.

Prova: Escolhemos o maior i de modo que $\alpha_i < 2 - x$. Se todas as funções restantes de (4.12) são f , então α_1 será negativo. Caso contrário, consideremos a primeira g aplicada em algum $\alpha < 2 - x$. O subcaso 1a, será executado e isso atribuirá $\alpha + x - 2$ na diagonal que é um número negativo. \square

Lema 4.3.7. *Sejam G um grafo threshold, $x = -\lambda_{\min, n}$ e α o primeiro elemento de $\alpha_{G, x}$ não excedendo 2, produzido pelos $j - 1$ primeiros passos da diagonalização de (G, x) . Se $\alpha < 2$ então $G = G(k, j)$. Se $\alpha = 2$, então ou $G = G(k, j)$ ou $G = G(k - 1, j + 1)$.*

Prova: Assumimos que $G = (b_1, \dots, b_n)$. Como $-x$ é um autovalor de G , sua α_G -sequência tem $\alpha_1 = 0$ e $\alpha_2 = \frac{1}{x}$. Para qualquer outro grafo *threshold* G' de ordem n , sua α -sequência deverá ter $\alpha'_1 \geq 0$. Pelo Lema 4.3.2 (ii. e iii.), $\alpha_{G, x}$ é uma sequência estritamente crescente de números variando de 0 a x . Assim existe um máximo k tal que

$$\alpha_{k+1} \leq 2 \tag{4.14}$$

Caso 1: $\alpha_{k+1} < 2$. Existem k funções aplicadas em argumentos menores que 2, e $n - k - 1$ funções aplicadas em argumentos maiores que 2. Afirmamos que $G = G(k, j)$, onde $j = n - k$. Vamos supor por contradição que para $2 \leq i \leq k, b_i = 1$. Podemos formar um novo grafo G' cuja representação é a mesma de G exceto na posição i , que é 0. Quando aplicarmos o algoritmo de diagonalização em G' , pelo Lema 4.3.5 podemos assumir que o subcaso 1c é evitado ou a diagonal terá um valor negativo. Similarmente, podemos assumir pelo Lema 4.3.6 que todos os α'_i s permanecem em $(2 - x, +\infty)$. Quando diagonalizamos G' , substituiremos $h_{i+1} = g$ por f , mas todas as outras funções de (4.13) serão as mesmas. Desde que $\frac{1}{x} = \alpha_2 < \alpha_i < 2$, pelo Lema 4.3.3, obteremos um valor negativo para α_1 , o que é uma contradição. Isto mostra que $b_i = 0$ para $i \leq k$. Afirmamos agora que $b_i = 1$ para $i > k$. Se existir $b_i = 0$, onde $k < i < n$, podemos construir um novo grafo G' tomando $b_i = 1$. A diagonalização de G' , será a mesma de G , exceto em (4.13) um $h_{i+1} = f$ será substituída por g . Pelo Lema 4.3.4 obteremos um valor negativo para α_1 , uma contradição. Portanto $G = G(k, j)$.

Caso 2: $\alpha_{k+1} = 2$. Usando o mesmo argumento acima, devemos ter $b_i = 0$ para $i < k$ e $b_i = 1$ para $i > k$. Mas pelo Lema 4.3.2 (v.) implica que $h_{k+1} = f$ e $h_{k+1} = g$ têm o mesmo efeito. Então b_k pode ser 0 ou 1, ou seja, o grafo *threshold* ou é $G = G(k, j)$ ou é $G = G(k - 1, j + 1)$. \square

Teorema 4.3.1. *Para $n \geq 3$, $G(n - \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{3} \rfloor)$ tem o menor autovalor mínimo $\lambda_{min,n}$ entre todos os grafos *threshold* de ordem n . Se $n \equiv 2 \pmod{3}$, o grafo $G(n - \lceil \frac{n}{3} \rceil, \lceil \frac{n}{3} \rceil)$ também possui o autovalor $\lambda_{min,n}$.*

Prova: É suficiente mostrar que o único α no Lema 4.3.7 ocorre quando $j = \lfloor \frac{n}{3} \rfloor$, com igualdade quando $n \equiv 2 \pmod{3}$. Seja $j = \lfloor \frac{n}{3} \rfloor$ de modo que $n = 3j + m$, $m \in \{0, 1, 2\}$. Tomando x o valor de (4.7), pelo Lema 4.2.3, após os $j - 1$ passos, o valor de α é

$$\frac{x + j - 1}{j} = \frac{\sqrt{(j - 1)^2 + 4kj} + (j - 1)}{2j}$$

Substituindo k por $n - j$, então n com $3j + m$ fornece

$$\frac{\sqrt{(j - 1)^2 + 4(2j + m)j} + (j - 1)}{2j} = \frac{\sqrt{9j^2 + (4m - 2)j + 1} + (j - 1)}{2j}$$

Assim $\alpha \leq 2$ se e somente se

$$\sqrt{9j^2 + (4m - 2)j + 1} \leq 3j + 1$$

Elevando ao quadrado e fazendo simplificações obtemos

$$4m \leq 8 \tag{4.15}$$

Isto é satisfeito para cada $m \in \{0, 1, 2\}$. Note que obtemos a igualdade quando $m = 2$. Se j é escolhido for menor do que $\lfloor \frac{n}{3} \rfloor$, então $n = 3j + m$ para $m \geq 3$. Ou seja, quando $m \geq 3$ (4.15) não é satisfeita. \square

Corolário 4.3.1. *Dentre todos os grafos *threshold* de ordem n , o menor autovalor é*

$$\lambda_{min,n} = \frac{1}{2} \left(\left(\lfloor \frac{n}{3} \rfloor - 1 \right) - \sqrt{\left(\lfloor \frac{n}{3} \rfloor - 1 \right)^2 + 4 \left(n - \lfloor \frac{n}{3} \rfloor \right) \left(\lfloor \frac{n}{3} \rfloor \right)} \right).$$

Obtemos esse valor tomando $j = \lfloor \frac{n}{3} \rfloor$ e $k = n - j$, pelo Teorema 4.2.1.

A Tabela 4.1 ilustra os $G(k, j)$ que atingem o autovalor $\lambda_{min, n}$, para alguns pequenos valores de n . Podemos mostrar que quando $n \equiv 2 \pmod 3$, $\lambda_{min, n} = -\lfloor \frac{n}{3} \rfloor - 1$. Assim os valores mostrados são inteiros e exatos para $n \equiv 2 \pmod 3$, mas aproximados para $n \not\equiv 2 \pmod 3$.

n	$\lambda_{min, n}$	graph(s)
3	-1.4142	$G(2, 1)$
4	-1.7320	$G(3, 1)$
5	-2.0000	$G(4, 1), G(3, 2)$
6	-2.3723	$G(4, 2)$
7	-2.7016	$G(5, 2)$
8	-3.0000	$G(6, 2), G(5, 3)$
9	-3.3589	$G(6, 3)$
10	-3.6904	$G(7, 3)$
11	-4.0000	$G(8, 3), G(7, 4)$
12	-4.3523	$G(8, 4)$
13	-4.6846	$G(9, 4)$
14	-5.0000	$G(10, 4), G(9, 5)$
15	-5.3485	$G(10, 5)$
16	-5.6811	$G(11, 5)$
17	-6.0000	$G(12, 5), G(11, 6)$
18	-6.3459	$G(12, 6)$
19	-6.6788	$G(13, 6)$
20	-7.0000	$G(14, 6), G(13, 7)$

Tabela 4.1: Grafos *threshold* com $\lambda_{min, n}$.

5 ORDENANDO OS GRAFOS *THRESHOLD*

Nesse capítulo estabelecemos uma ordem parcial sobre os grafos *threshold*, com o objetivo de relacionar os seus respectivos índices. Utilizamos o algoritmo de diagonalização para obter tal relação. Na seção final apresentamos uma família infinita de grafos *threshold* com espectro integral.

5.1 Introdução

Um parâmetro bastante estudado em teoria espectral de grafos é o índice ou raio espectral de um grafo. O índice de um grafo G , denotado por $\lambda(G)$ é o maior autovalor de sua matriz de adjacência. Neste capítulo mostramos que existe uma ordem parcial nos grafos *threshold* G , que está relacionada aos seus índices.

De acordo com a teoria de Perron-Frobenius, o índice de um grafo G de ordem n e m arestas, $\lambda(G) = \max\{|\lambda_i| : \lambda_i \text{ é um autovalor de } G\}$, satisfaz a desigualdade

$$\max\{d, \sqrt{\Delta}\} \leq \lambda(G) \leq \Delta,$$

onde $d = \frac{2m}{n}$ é a média dos graus dos vértices e Δ é o máximo grau de um vértice em G . Segue da desigualdade acima que $\lambda(G) > 1$.

Determinar o índice de um grafo G de ordem n é um problema considerado relevante e interessante por vários autores. Por exemplo, quando fixamos o número de vértices de G , é conhecido que o maior índice $\lambda(G)$ é atingido pelo grafo completo K_n que é um grafo *threshold*. O problema se torna muito complicado, quando fixamos o número de vértices e o número de arestas de G . Nesse caso, de acordo com Simić *et.al.* [40], ainda não se conhece, quais são os grafos conexos com o maior índice. Porém, em [40], são utilizadas técnicas de autovetores para determinar novas cotas para o índice de grafos *threshold*, fixando-se o número de vértices e arestas dos grafos.

A partir da definição de grafos *threshold* podemos facilmente definir uma ordem parcial nessa classe de grafos, ou seja, se G e H são representados por sequências binárias (b_1, b_2, \dots, b_n) e (c_1, c_2, \dots, c_n) , respectivamente, definimos $G \leq H$ se e somente se $b_i \leq c_i$ para todo i . Assim $G < H$, se e somente se existir $k > 1$ com $b_k = 0, c_k = 1$, e $b_i \leq c_i$ para $i \neq k$.

Neste capítulo, utilizando o algoritmo de diagonalização, provaremos no Teorema 5.2.1 que entre todos os grafos *threshold* G , conexos de ordem n , se $G < H$ então $\lambda(G) < \lambda(H)$. Assim, entre todos os grafos *threshold* conexos de ordem n , a árvore $K_{n-1,1}$ tem o menor índice, enquanto que o grafo completo K_n admite o maior índice.

É bem conhecido que se acrescentarmos arestas a um grafo, então o índice não diminui. Assim poderíamos argumentar que o resultado acima é verdadeiro porque se $G < H$, então H tem mais arestas do que G . De fato, quando $n = 8$, entre os 64 grafos *threshold* quase sempre, um número maior de arestas implica um índice maior. No entanto, a Tabela 5.1 ilustra que nem sempre este é o caso. Observamos que nenhum dos grafos da Tabela 5.1 estão relacionados sob a ordem parcial $<$.

grafo	índice	arestas
(01001011)	4.9837	18
(00000111)	5.0000	18
(00110011)	5.0202	18
(01111001)	5.0695	17

Tabela 5.1: índices e arestas de *thresholds* com $n = 8$

5.2 Ordenação via índice

Seja G um grafo *threshold* conexo de ordem $n > 2$, cujo índice é denotado por $\lambda(G)$. Vamos denotar por x a quantidade negativa $-\lambda(G) < -1$.

Lema 5.2.1. *Se G é um grafo threshold com índice $\lambda(G) = -x$ então a diagonalização de (G, x) produz um zero no topo da diagonal, e valores negativos abaixo.*

Prova: Como $-x$ é um autovalor de G , pelo Teorema 3.2.3 devemos obter um zero na diagonal. Como $-x$ é o maior autovalor, pelo mesmo teorema, os elementos restantes na diagonal devem ser negativos. Assim para $x \neq 0, -1$, de acordo com Teorema 3.3.1, um zero pode ser atribuído na diagonal, somente na última iteração do algoritmo, ou seja, no topo da diagonal. \square

Fazendo uma análise do algoritmo de diagonalização, quando tomamos $x = -\lambda(G)$, não é difícil de verificar que novamente recaímos nos subcasos 1a e 2b.

Lema 5.2.2. *As funções g e f definidas em (4.9) e (4.10) satisfazem as seguintes propriedades:*

i. $f(\alpha) = 0$ se e somente se $\alpha = \frac{1}{x}$.

ii. $g(\alpha) = 0$ se e somente se $\alpha = \frac{1}{x}$.

iii. Para $x < 0$ então $\alpha < \frac{1}{x} < 0$ implica que $\alpha < f(\alpha) < g(\alpha) < 0$.

Prova: As propriedades *i.* e *ii.* são facilmente verificadas. Para provar *iii.* seja $\alpha < \frac{1}{x} < 0$. Isso implica que o denominador de $g(\alpha)$ é negativo, e $x < 0$ implica que seu numerador é positivo. Assim $g(\alpha) < 0$. Para verificar $f(\alpha) < g(\alpha)$, devemos mostrar que

$$\alpha - \frac{1}{x} = \frac{\alpha x - 1}{x} < \frac{\alpha x - 1}{\alpha + x - 2}$$

que é equivalente a $\frac{1}{x} < \frac{1}{\alpha x - 2}$, que vale, já que $\alpha + x - 2 < x$. Finalmente, como $x < 0, \alpha < \alpha - \frac{1}{x} = f(\alpha)$. \square

Lema 5.2.3. *Se G é um grafo threshold com índice $\lambda(G)$ e $x = -\lambda(G)$, então a α -sequência é uma sequência estritamente decrescente de números não positivos*

$$\alpha_1 = 0 > \alpha_2 = \frac{1}{x} > \dots \alpha_{n-1} > \alpha_n = x.$$

Prova: Pelo Lema 5.2.1, $\alpha_1 = 0$. Pelo Lema 5.2.2 (partes *i.* e *ii.*) devemos ter $\alpha_2 = \frac{1}{x}$. E pelo Lema 5.2.2 (parte *iii.*), a sequência é monótona. \square

Consideremos agora G' um grafo *threshold* conexo obtido a partir de G por uma troca de um único $b_k, 1 < k < n$, de 0 para 1. Considerando a diagonalização de (G', x) , a nova α -sequência

$$\alpha_{G',x} = (\alpha'_1, \dots, \alpha'_k, \alpha_{k+1}, \dots, \alpha_n = x) \quad (5.1)$$

computada é exatamente a mesma, exceto h_{k+1} que trocará de f para g .

Lema 5.2.4. *Se $1 < k < n$, e $h_{k+1} = f$ é substituída por g , então α_1 crescerá, desde que todos os valores de (5.1) pertençam ao intervalo $(-\infty, 2 - x)$.*

Prova: Como $k > 1$, pelo Lema 5.2.3, temos $0 > \frac{1}{x} > \alpha_{k+1}$. Segue do Lema 5.2.2 (parte iii.) que

$$\alpha_k = f(\alpha_{k+1}) < g(\alpha_{k+1}) = \alpha'_k.$$

Seja $h = h_2 \circ h_3 \circ \dots \circ h_k$ a composição restante. Como f e g são contínuas e crescentes em $(-\infty, 2 - x)$, então cada h_i também será contínua e crescente em $(-\infty, 2 - x)$, logo a composição deverá satisfazer

$$\alpha_1 = h(\alpha_k) < h(\alpha'_k) = \alpha'_1.$$

□

Lema 5.2.5. *Durante a diagonalização de (G', x) , se a α -sequência não permanece em $(-\infty, 2 - x)$ então a diagonal final contém um valor positivo.*

Prova: Se existe algum $\alpha_i \in [2 - x, +\infty)$, então podemos escolher tal i como sendo o maior. Segue do Lema 5.2.3 e de (5.1), que $i \leq k$. Se o valor positivo foi atribuído após executarmos o subcaso 1c, então na iteração anterior $i + 1$, foi atribuído o valor

$$\alpha_i \leftarrow 2 - x,$$

contrariando o fato de i ser o maior. Assim o subcaso 1c não pode ocorrer. Se todas as funções de (5.1) para $i < k$ forem $h_i = f$, como $\alpha_i > 0$, então no topo da diagonal deveremos ter um valor positivo. Por outro lado, se a g é aplicada para um $\alpha_i > 2 - x$, então $\alpha_i + x - 2 > 0$ será atribuído na diagonal. □

Lema 5.2.6. *Seja G um grafo threshold conexo, representado por (b_1, \dots, b_n) , e seja G' o grafo threshold obtido pela troca de um único $b_k = 0$ por 1, para algum $1 < k < n$. Então $\lambda(G) < \lambda(G')$.*

Prova: Pelo Teorema 3.2.3, é suficiente mostrar que a diagonal após a diagonalização de (G', x) contém um número positivo. Se a sequência (5.1) permanece em $(-\infty, 2-x)$, então pelo Lema 5.2.4 $\alpha'_1 > \alpha_1 = 0$ são atribuídos. Caso contrário, pelo Lema 5.2.5 segue o resultado.

Definição 5.2.1. *A distância de Hamming entre duas sequências binárias (b_i) e (c_i) de mesmo comprimento é o número de elementos que são distintos.*

Exemplo 5.2.1. *As sequências binárias $(0, 0, 1, 1, 1)$ e $(1, 1, 0, 0, 0)$ têm distância de Hamming igual a 5.*

Teorema 5.2.1. *Para grafos threshold conexos de ordem n , se $G < H$ então $\lambda(G) < \lambda(H)$.*

Prova: Sejam G e H grafos *threshold* conexos de ordem n , representados por (b_1, \dots, b_n) e (c_1, \dots, c_n) , respectivamente. O resultado segue do Lema 5.2.6 e de uma indução sobre a distância de Hamming entre (b_1, \dots, b_n) e (c_1, \dots, c_n) . \square

Corolário 5.2.1. *Entre todos grafos threshold conexos de ordem n , a árvore $K_{n-1,1}$ tem o menor índice.*

5.3 Uma família com espectro integral

A procura de grafos com espectro integral, ou seja, grafos em que todos os autovalores são inteiros é um problema interessante em teoria espectral de grafos. Um trabalho seminal de Harary e Schwenk [23] identificou vários grafos com espectro integral. Nesta seção, identificamos uma subclasse infinita de grafos *threshold* que admite tal propriedade.

Vimos no capítulo 4, que os grafos *threshold* do tipo $G(k, j)$ têm espectro $\{\lambda_n, -1, 0, \lambda(G)\}$, onde o menor autovalor é

$$\lambda_n = \frac{(j-1) - \sqrt{(j-1)^2 + 4kj}}{2} \quad (5.2)$$

e o índice é dado por

$$\lambda(G) = \frac{(j-1) + \sqrt{(j-1)^2 + 4kj}}{2}. \quad (5.3)$$

Considerando $n = 3j + 2$, ou seja, $j = \frac{n-2}{3}$, e $k = n - j$, a partir de (5.2) obtemos o inteiro $\lambda_n = \frac{-n-1}{3} = -\lfloor \frac{n}{3} \rfloor - 1$, e (5.3) torna-se $\lambda(G) = 2\lfloor \frac{n}{3} \rfloor$. Assim temos o seguinte resultado:

Teorema 5.3.1. *Se $n \equiv 2 \pmod{3}$, então $G(n - \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{3} \rfloor)$ tem o seguinte espectro integral : $\{-\lfloor \frac{n}{3} \rfloor - 1, -1, 0, 2\lfloor \frac{n}{3} \rfloor\}$, com multiplicidades $1, \lfloor \frac{n}{3} \rfloor - 1, n - \lfloor \frac{n}{3} \rfloor - 1, 1$, respectivamente.*

Tomando $n = 3j - 1$, e $j = \frac{n+1}{3}$, e usando um argumento similar, obtemos:

Teorema 5.3.2. *Se $n \equiv 2 \pmod{3}$, então $G(n - \lceil \frac{n}{3} \rceil, \lceil \frac{n}{3} \rceil)$ tem o seguinte espectro integral : $\{-\lfloor \frac{n}{3} \rfloor - 1, -1, 0, \lfloor \frac{n}{3} \rfloor + \lceil \frac{n}{3} \rceil\}$, com multiplicidades $1, \lceil \frac{n}{3} \rceil - 1, n - \lceil \frac{n}{3} \rceil - 1, 1$, respectivamente.*

Os grafos do Teorema 5.3.1 e 5.3.2 têm o mesmo autovalor mínimo, mas admitem índices diferentes, ou seja, $\lambda(G(n - \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{3} \rfloor)) < \lambda(G(n - \lceil \frac{n}{3} \rceil, \lceil \frac{n}{3} \rceil))$, o que está de acordo com o Teorema 5.2.1.

6 O POLINÔMIO CARACTERÍSTICO

Nesse capítulo apresentamos um algoritmo para o cálculo do polinômio característico de um grafo *threshold* G , baseado no algoritmo de diagonalização. Nosso algoritmo é simples e mais prático do que o método de redução do cálculo do polinômio característico apresentada no capítulo 2.

6.1 Introdução

Um problema importante em álgebra computacional é o cálculo de polinômios característicos simbólicos de matrizes. A nossa motivação para o algoritmo que apresentaremos neste capítulo foi o trabalho de Jacobs e Trevisan [29], que apresenta um algoritmo de ordem $o(n^2)$ para o cálculo do polinômio característico de matrizes de adjacência de árvores.

O algoritmo desenvolvido para as árvores funciona de forma elegante, sem a necessidade de explicitar a matriz de adjacência, devido a uma enumeração apropriada dos vértices. No final desse processo, obtém-se a triangularização da matriz $\lambda I - A$. Dessa forma o polinômio característico é obtido multiplicando-se os valores finais atribuídos aos vértices da árvore.

O algoritmo que desenvolvemos para o cálculo do polinômio característico de grafos *threshold*, apesar de ter sido inspirado no algoritmo desenvolvido para árvores, têm diferenças em relação a esse algoritmo. Entre as diferenças está o fato que ao invés de trabalharmos com uma matriz triangular, vamos trabalhar com uma matriz diagonal. Utilizaremos as operações elementares realizadas nas matrizes para justificar a correção do algoritmo, mas é importante salientar que a implementação depende somente da sequência binária do grafo.

6.2 Diagonalizando $A - \lambda I$

Sejam λ uma variável e A a matriz de adjacência de um grafo G . Chamamos de polinômio característico do grafo G o polinômio característico de A , definido por $p(\lambda) = \det(A - \lambda I)$.

Alguns autores definem o polinômio característico como $\det(\lambda I - A)$, de qualquer maneira esses polinômios diferem apenas por um fator $(-1)^n$, onde n é a ordem da matriz de adjacência.

As raízes do polinômio característico são os autovalores de G . Vale lembrar que, como a matriz de adjacência é simétrica, os seus autovalores são números reais.

Seja G um grafo *threshold* representado através de sua sequência binária (b_1, b_2, \dots, b_n) . Assumimos que G é conexo, pois caso contrário, o polinômio característico de G é obtido por $p(\lambda) = \lambda^k \prod p_{G'}(\lambda)$, onde cada $p_{G'}(\lambda)$ é o polinômio característico de uma componente conexa de G .

O algoritmo que apresentaremos para o cálculo do polinômio característico de um grafo *threshold* conexo G , é baseado no algoritmo de diagonalização de $A + xI$, apresentado no capítulo 3. O algoritmo de diagonalização pode ser facilmente estendido para trabalhar sobre um corpo qualquer F , onde $x \in F$. Dessa forma podemos usar a diagonalização de $(G, -\lambda)$ para calcular $\det(A - \lambda I)$. Entretanto, como λ é uma variável não estamos mais trabalhando sobre um corpo, mas sim no corpo quociente $\mathbb{Q}(\lambda)$ de $\mathbb{Q}[\lambda]$.

Para uma melhor compreensão, vamos repetir os passos do algoritmo de diagonalização, adaptado para esse caso. Após $n - m$ passos do algoritmo, a diagonalização parcial é exibida na matriz abaixo, onde $\alpha_i(\lambda)$ representa o quociente

de polinômios na variável λ .

$$\begin{bmatrix} & & 0 & 1 & 0 & \cdots & 0 \\ & & \vdots & \vdots & & & \\ & & 0 & 1 & & & \\ 0 & \cdots & 0 & -\lambda & 1 & \vdots & \vdots \\ 1 & \cdots & 1 & 1 & -\lambda & 0 & 0 \\ 0 & & & 0 & \alpha_{n-m}(\lambda) & \cdots & 0 \\ \vdots & & & \vdots & & \ddots & \vdots \\ 0 & & & 0 & \cdots & 0 & \alpha_n(\lambda) \end{bmatrix}.$$

O passo seguinte é diagonalizarmos a submatriz $m \times m$. Após a diagonalização o $\det(A - \lambda I)$ é calculado através do produto dos elementos da diagonal, com o sinal ajustado. Para diagonalizarmos a submatriz $m \times m$, consideremos os seguintes casos:

Caso 1: $b_{m-1} = 1$ e $b_m = 1$. Então a matriz é da seguinte forma:

$$\begin{bmatrix} & & & 1 & 1 \\ & & & \vdots & \vdots \\ & & & 1 & 1 \\ 1 & \cdots & 1 & -\lambda & 1 \\ 1 & \cdots & 1 & 1 & \alpha(\lambda) \end{bmatrix}.$$

Consideremos as seguintes operações

$$l_m \leftarrow l_m - l_{m-1}$$

$$c_m \leftarrow c_m - c_{m-1}$$

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \cdots & 1 & -\lambda & 1 + \lambda \\ 0 & \cdots & 0 & 1 + \lambda & \alpha(\lambda) - \lambda - 2 \end{bmatrix}.$$

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & -\lambda & 1 + \lambda \\ 0 & \dots & 0 & 1 + \lambda & 0 \end{bmatrix}.$$

Fazendo as seguintes operações:

$$l_{m-1} \leftarrow l_{m-1} + \frac{1}{2}l_m$$

$$c_{m-1} \leftarrow c_{m-1} + \frac{1}{2}c_m$$

substituimos a diagonal $-\lambda$ por 1, assim basta tomar

$$l_m \leftarrow l_m - (1 + \lambda)l_{m-1}$$

$$c_m \leftarrow c_m - (1 + \lambda)c_{m-1}$$

para obtermos finalmente a matriz diagonalizada

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & -(1 + \lambda)^2 \end{bmatrix}.$$

A diferença desse subcaso em relação ao anterior é que as entradas da linha e coluna $m - 1$ também foram anuladas. Vamos memorizar isso tomando a seguinte atribuição:

$$b_{m-1} \leftarrow 0.$$

Caso 2: $b_{m-1} = 0$ e $b_m = 1$. Então a submatriz tem a seguinte forma:

$$\begin{bmatrix} & & & 0 & 1 \\ & & & \vdots & \vdots \\ & & & 0 & 1 \\ 0 & \dots & 0 & -\lambda & 1 \\ 1 & \dots & 1 & 1 & \alpha(\lambda) \end{bmatrix}.$$

Após trocarmos as linhas m e $m - 1$ e as colunas m e $m - 1$, a matriz é da forma:

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & \alpha(\lambda) & 1 \\ 0 & \dots & 0 & 1 & -\lambda \end{bmatrix}.$$

Desde que $\lambda \neq 0$, então façamos as operações:

$$l_{m-1} \leftarrow l_{m-1} + \frac{1}{\lambda} l_m$$

$$c_{m-1} \leftarrow c_{m-1} + \frac{1}{\lambda} c_m$$

obtendo a matriz da forma desejada

$$\begin{bmatrix} & & & 1 & 0 \\ & & & \vdots & \vdots \\ & & & 1 & 0 \\ 1 & \dots & 1 & \alpha(\lambda) + \frac{1}{\lambda} & 0 \\ 0 & \dots & 0 & 0 & -\lambda \end{bmatrix}.$$

Como a linha e coluna $m - 1$ foram preenchidas por 1's, devemos fazer a seguinte atribuição

$$b_{m-1} \leftarrow 1.$$

Caso 3: $b_m = 0$. Não é preciso fazer alguma operação e o algoritmo move-se para o passo seguinte.

O algoritmo do cálculo do polinômio característico de G é exibido na Figura 6.1. Notemos que a única informação importante é a descrição do grafo (b_1, b_2, \dots, b_n) e os valores da diagonal (d_1, d_2, \dots, d_n) .

```

Algorithm CharPoly( $G$ )
  initialize  $d_i \leftarrow -\lambda$ , for all  $i$ 
  for  $m = n$  to 2
     $\alpha(\lambda) \leftarrow d_m$ 
    if  $b_{m-1} = 1$  and  $b_m = 1$ 
      if  $\alpha(\lambda) \neq \lambda + 2$  //subcase 1a
         $d_{m-1} \leftarrow \frac{\alpha(\lambda)(-\lambda)-1}{\alpha(\lambda)-\lambda-2}$ 
         $d_m \leftarrow \alpha(\lambda) - \lambda - 2$ 
      else //subcase 1b
         $d_{m-1} \leftarrow 1$ 
         $d_m \leftarrow -(1 + \lambda)^2$ 
         $b_{m-1} \leftarrow 0$ 
      else if  $b_{m-1} = 0$  and  $b_m = 1$ 
         $d_{m-1} \leftarrow \alpha(\lambda) + \frac{1}{\lambda}$ 
         $d_m \leftarrow -\lambda$ 
         $b_{m-1} \leftarrow 1$ 
      end if
    end loop
   $p(\lambda) = (-1)^n \prod_{i=1}^n d_i$ 

```

Figura 6.1: Algoritmo do polinômio característico

Teorema 6.2.1. *Sejam G um grafo threshold conexo (b_1, \dots, b_n) e (d_1, \dots, d_n) os valores obtidos via diagonalização de $(G, -\lambda)$. Então $p(\lambda) = (-1)^n \prod_{i=1}^n d_i$ é o polinômio característico de G .*

Exemplo 6.2.1. *No primeiro exemplo assumimos que G é um grafo threshold dado por $(0, 1, 0, 1)$. Após inicializarmos, três passos serão executados, $m = 4, 3, 2$.*

Como $b_3 = 0$ e $b_4 = 1$ o caso 2 é executado e as seguintes atribuições são feitas:

$$d_3 \leftarrow -\lambda + \frac{1}{\lambda} = \frac{1 - \lambda^2}{\lambda}$$

$$d_4 \leftarrow -\lambda$$

$$b_3 \leftarrow 1.$$

No segundo passo do algoritmo, como $b_2 = 1, b_3 = 1$ e $\alpha(\lambda) = \frac{1-\lambda^2}{\lambda} \neq \lambda + 2$, o subcaso 1a é executado

$$d_2 \leftarrow \frac{\alpha(\lambda)(-\lambda) - 1}{\alpha(\lambda) - \lambda - 2} = \frac{\frac{1-\lambda^2}{\lambda}(-\lambda) - 1}{\frac{1-\lambda^2}{\lambda} - \lambda - 2}$$

$$d_3 \leftarrow \alpha(\lambda) - \lambda - 2 = \frac{1-\lambda^2}{\lambda} - \lambda - 2.$$

No passo final do algoritmo, como $b_1 = 0, b_2 = 1$, e $\alpha(\lambda) = d_2$, logo o caso 2 é executado novamente

$$d_1 \leftarrow \frac{\frac{1-\lambda^2}{\lambda}(-\lambda) - 1}{\frac{1-\lambda^2}{\lambda} - \lambda - 2} + \frac{1}{\lambda}$$

$$d_2 \leftarrow -\lambda$$

$$b_1 \leftarrow 1.$$

A seguinte tabela fornece os passos do algoritmo em cada estágio:

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	$-\lambda$	0	$-\lambda$	0	$-\lambda$	1	$\frac{\frac{1-\lambda^2}{\lambda}(-\lambda) - 1}{\frac{1-\lambda^2}{\lambda} - \lambda - 2} + \frac{1}{\lambda}$
1	$-\lambda$	1	$-\lambda$	1	$\frac{\frac{1-\lambda^2}{\lambda}(-\lambda) - 1}{\frac{1-\lambda^2}{\lambda} - \lambda - 2}$	1	$-\lambda$
0	$-\lambda$	1	$-\lambda + \frac{1}{\lambda}$	1	$\frac{1-\lambda^2}{\lambda} - \lambda - 2$	1	$\frac{1-\lambda^2}{\lambda} - \lambda - 2$
1	$-\lambda$	1	$-\lambda$	1	$-\lambda$	1	$-\lambda$
<i>inicial</i>		<i>após 2</i>		<i>após 1a</i>		<i>após 2</i>	

Temos que o polinômio característico é dado por

$$p(\lambda) = (-1)^4 \prod_{i=1}^4 d_i.$$

Logo segue que

$$p(\lambda) = \left(\frac{\frac{1-\lambda^2}{\lambda}(-\lambda) - 1}{\frac{1-\lambda^2}{\lambda} - \lambda - 2} + \frac{1}{\lambda} \right) (-\lambda) \left(\frac{1-\lambda^2}{\lambda} - \lambda - 2 \right) (-\lambda)$$

$$p(\lambda) = \lambda^4 - 4\lambda^2 - 2\lambda + 1.$$

Exemplo 6.2.2. Assumimos que G é um grafo threshold representado por $(0, 0, 1, 1)$.

Após inicializarmos, três passos serão executados, $m = 4, 3, 2$.

Como $b_3 = 1, b_4 = 1$, e $\alpha(\lambda) = -\lambda \neq \lambda + 2$, o subcaso 1a é executado e as atribuições são:

$$d_3 \leftarrow \frac{\alpha(\lambda)(-\lambda) - 1}{\alpha(\lambda) - \lambda - 2} = \frac{\lambda^2 - 1}{-2\lambda - 2} = \frac{\lambda - 1}{-2}$$

$$d_4 \leftarrow \alpha(\lambda) - \lambda - 2 = -2\lambda - 2.$$

No segundo passo do algoritmo como $b_2 = 0, b_3 = 1$, e $\alpha(\lambda) = \frac{\lambda-1}{-2}$, o caso 2 é executado

$$d_2 \leftarrow \alpha(\lambda) + \frac{1}{\lambda} = \frac{\lambda - 1}{-2} + \frac{1}{\lambda}$$

$$d_3 \leftarrow -\lambda$$

$$b_2 \leftarrow 1.$$

No passo final do algoritmo como $b_1 = 0, b_2 = 1$ e $\alpha(\lambda) = d_2$, o caso 2 é novamente executado

$$d_1 \leftarrow \alpha(\lambda) + \frac{1}{\lambda} = \frac{\lambda - 1}{-2} + \frac{2}{\lambda}$$

$$d_2 \leftarrow -\lambda$$

$$b_1 \leftarrow 1.$$

A seguinte tabela fornece os passos do algoritmo em cada estágio:

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	$-\lambda$	0	$-\lambda$	0	$-\lambda$	1	$\frac{\lambda-1}{-2} + \frac{2}{\lambda}$
0	$-\lambda$	0	$-\lambda$	1	$\frac{\lambda-1}{-2} + \frac{1}{\lambda}$	1	$-\lambda$
1	$-\lambda$	1	$\frac{\lambda-1}{-2}$	1	$-\lambda$	1	$-\lambda$
1	$-\lambda$	1	$-2\lambda - 2$	1	$-2\lambda - 2$	1	$-2\lambda - 2$
<i>inicial</i>		<i>após 1a</i>		<i>após 2</i>		<i>após 2</i>	

Temos que o polinômio característico é dado por

$$p(\lambda) = (-1)^4 \prod_{i=1}^4 d_i$$

Então

$$p(\lambda) = \left(\frac{\lambda-1}{-2} + \frac{2}{\lambda}\right)(-\lambda)(-\lambda)(-2\lambda-2)$$

$$p(\lambda) = \lambda^4 - 5\lambda^2 - 4\lambda.$$

6.3 Considerações sobre a complexidade

O objetivo dessa seção é apresentarmos algumas considerações gerais sobre a complexidade do algoritmo para o cálculo do polinômio característico de um grafo *threshold* G , apresentado na seção anterior desse capítulo.

No capítulo 2, foi apresentado uma redução para o cálculo do polinômio característico de grafos *threshold*, de acordo com o trabalho de Sciriha e Farrugia [39]. Tal procedimento, fornecido pelo Teorema 2.4.2, é apenas uma redução para o cálculo do polinômio característico, e não fornece explicitamente o polinômio característico.

Em relação à complexidade dessa redução, podemos perceber que em alguns casos, ela é aproximadamente igual a complexidade do problema original. Por exemplo, consideremos o grafo *threshold* G de ordem n e sequência binária igual a $(0, 1, 0, 1, \dots, 0, 1)$, tendo partição equilibrada mínima de tamanho $n - 1$. De acordo com o Teorema 2.4.2, o cálculo do polinômio característico de G , se reduz ao cálculo do polinômio característico de uma matriz de ordem $(n - 1) \times (n - 1)$.

Diferentemente da redução para o cálculo do polinômio característico de grafos *threshold* G , o algoritmo desenvolvido neste capítulo fornece de forma explícita como obter esse polinômio.

Em relação à complexidade do algoritmo desse capítulo, em linhas gerais, é de ordem $O(n)$, onde n é o número de operações polinomiais. Como cada polinômio envolvido tem grau limitado por n , se usarmos aritmética usual, multi-

plicação e divisão de polinômios são limitadas por $O(n^2)$ operações. Portanto, em uma primeira aproximação podemos dizer que nosso algoritmo tem complexidade de $O(n^3)$ operações inteiras.

Uma análise mais criteriosa certamente pode baixar a complexidade. Se usarmos a técnica de interpolação certamente podemos vislumbrar um algoritmo com $O(n^2)$ operações aritméticas.

7 EXTENSÃO PARA A MATRIZ DISTÂNCIA

Finalizamos esta tese apresentando uma extensão do algoritmo de diagonalização para a matriz distância Θ de um grafo *threshold* G . Como aplicação, podemos determinar quantos autovalores de Θ pertencem a um intervalo qualquer. Mostramos também que todo autovalor $\lambda \neq -1, -2$ da matriz distância Θ de um grafo *threshold* é simples. A técnica desenvolvida neste capítulo está no artigo [27].

7.1 Introdução

Distância em teoria de grafos é uma idéia simples, mas poderosa, sobre a qual dependem vários parâmetros, incluindo diâmetro, raio, distância média e outros.

Um caminho em um grafo G é uma sequência de vértices distintos, de modo que vértices adjacentes na sequência são adjacentes no grafo. O comprimento de um caminho é o número de arestas do caminho. Para grafos conexos, a distância entre dois vértices de um grafo G , digamos u e v , denotada por $d(u, v)$ é o comprimento do menor caminho entre u e v .

O diâmetro de um grafo conexo G , denotado por $diam(G)$, é a distância máxima entre dois vértices. A excentricidade de um vértice é a distância máxima a partir dele para qualquer outro vértice. O raio de G , denotado por $rad(G)$, é a excentricidade mínima entre os vértices de G .

A distância média de um grafo de ordem n , denotada por $\mu(G)$, é a distância esperada entre um par de vértices distintos escolhidos aleatoriamente. O estudo da distância média começou com o químico Wiener [42], que observou que o ponto de fusão de determinados hidrocarbonetos é proporcional à soma de todas as distâncias entre pares não ordenados de vértices correspondentes ao grafo. Esta

soma, representada por $W(G)$, é chamada índice de Wiener de G . Claramente,

$$W(G) = \binom{n}{2} \mu(G).$$

O índice de Wiener e suas aplicações à química têm recebido muita atenção (ver, por exemplo, [1, 2, 3, 32]).

A matriz de distância Θ de um grafo G conexo é a matriz cujas linhas e colunas são indexados por seus vértices tais que a sua entrada (u, v) é igual a $d(u, v)$. Formalmente temos a seguinte definição.

Definição 7.1.1. *Seja $G = (V, E)$ um grafo conexo com n vértices tal que para todo v_i e $v_j \in V$, $d(v_i, v_j) = d_{ij}$ é o comprimento do menor caminho entre os vértices v_i e v_j . A matriz distância Θ de G é uma matriz quadrada de ordem n e cuja entrada correspondente a (v_i, v_j) é dada pelos valores d_{ij} .*

Por exemplo, a Figura 7.1, ilustra a matriz distância Θ em relação ao grafo G .

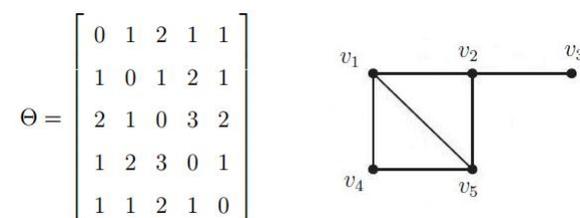


Figura 7.1: Matriz distância do grafo G .

Seja $\mathbf{1}$ o vetor coluna com todas entradas iguais a um, então o índice de Wiener pode ser escrito na forma

$$W = \frac{\mathbf{1}^T \Theta \mathbf{1}}{2}.$$

Os autovalores de Θ são chamados de autovalores da distância de G , formam assim o espectro da distância, e têm várias aplicações do mundo real. Os autovalores da matriz distância foram estudadas pela primeira vez por Graham e Pollack em 1971 para resolver um problema de comunicação de dados [20].

A matriz de distância contém informação em vários passeios em grafos químicos. É útil no cálculo de índices topológicos e propriedades termodinâmicas, tais como pressão e temperatura de coeficientes. Ela contém informações mais estruturais do que a matriz de adjacência [26, 37]. Na literatura química, o maior autovalor de $\Theta(G)$ ajuda a modelar o ponto de ebulição de alcanos [1]. Além de química, matrizes de distância encontram aplicações em teoria musical, ornitologia, biologia molecular, psicologia, arqueologia, etc [4, 8, 31, 37].

No intuito de colaborarmos com novos resultados espectrais em relação à matriz distância, nesse capítulo apresentamos uma extensão de alguns resultados apresentados nos capítulos anteriores para a matriz distância.

7.2 Diagonalizando $\Theta + xI$

A construção da matriz distância Θ de um grafo *threshold* G é obtida através da sequência binária associada a G , onde a enumeração dos vértices é a mesma da sequência binária. Por exemplo, a Figura 7.2, apresenta a matriz distância do grafo *threshold* representado por $(0, 0, 0, 1, 1, 1, 0, 0, 1)$.

$$\begin{bmatrix} 0 & 2 & 2 & 1 & 1 & 1 & 2 & 2 & 1 \\ 2 & 0 & 2 & 1 & 1 & 1 & 2 & 2 & 1 \\ 2 & 2 & 0 & 1 & 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 2 & 2 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 0 & 2 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Figura 7.2: Matriz distância de um *threshold*.

Seja G um grafo *threshold* com sequência binária (b_1, \dots, b_n) , e seja $\Theta = [d_{ij}]$ sua matriz distância. É fácil ver que se $b_i = 1$, então $d_{ij} = d_{ji} = 1$, para $i < j$. E se $b_i = 0$, $d_{ij} = d_{ji} = 2$, para $i < j$.

A técnica de diagonalização da matriz $\Theta + xI$, é análoga ao caso da matriz de adjacência. Existem 4 casos principais:

Caso 1: $b_{m-1} = b_m = 1$ é análoga a matriz de adjacência.

Caso 2: $b_{m-1} = 0$ e $b_m = 1$. Então a submatriz tem a seguinte forma:

$$\begin{bmatrix} & & & 2 & 1 \\ & & & \vdots & \vdots \\ & & & 2 & 1 \\ 2 & \dots & 2 & x & 1 \\ 1 & \dots & 1 & 1 & \alpha \end{bmatrix}.$$

Fazendo as seguintes operações

$$l_m \leftarrow l_m - \frac{1}{2}l_{m-1}$$

$$c_m \leftarrow c_m - \frac{1}{2}c_{m-1}$$

obtemos

$$\begin{bmatrix} & & & 2 & 0 \\ & & & \vdots & \vdots \\ & & & 2 & 0 \\ 2 & \dots & 2 & x & 1 - \frac{x}{2} \\ 0 & \dots & 0 & 1 - \frac{x}{2} & \alpha + \frac{x}{4} - 1 \end{bmatrix}.$$

A maioria dos elementos da linha e coluna m foram anulados. Assim devemos eliminar as duas entradas $1 - \frac{x}{2}$. Dessa forma, existem três subcasos, dependendo se $\alpha + \frac{x}{4} - 1 \neq 0$ e se $x = 2$.

Subcaso 2a: Se $\alpha + \frac{x}{4} - 1 \neq 0$. Então façamos as seguintes operações:

$$l_{m-1} \leftarrow l_{m-1} - \frac{1 - \frac{x}{2}}{\alpha + \frac{x}{4} - 1} l_m$$

$$c_{m-1} \leftarrow c_{m-1} - \frac{1 - \frac{x}{2}}{\alpha + \frac{x}{4} - 1} c_m$$

obtendo

$$\begin{bmatrix} & & & 2 & 0 \\ & & & \vdots & \vdots \\ & & & 2 & 0 \\ 2 & \dots & 2 & \gamma & 0 \\ 0 & \dots & 0 & 0 & \alpha + \frac{x}{4} - 1 \end{bmatrix},$$

onde $\gamma = x - \frac{(1 - \frac{x}{2})^2}{\alpha + \frac{x}{4} - 1} = \frac{\alpha x - 1}{\alpha + \frac{x}{4} - 1}$.

Subcaso 2b: Se $\alpha + \frac{x}{4} = 1$ e $x = 2$, então a matriz é da forma

$$\begin{bmatrix} & & & 2 & 0 \\ & & & \vdots & \vdots \\ & & & 2 & 0 \\ 2 & \dots & 2 & 2 & 0 \\ 0 & \dots & 0 & 0 & 0 \end{bmatrix},$$

ou seja, já está da forma diagonal desejada.

Subcaso 2c: Se $\alpha + \frac{x}{4} = 1$ e $x \neq 2$. Então a matriz é da forma:

$$\begin{bmatrix} & & & 2 & 0 \\ & & & \vdots & \vdots \\ & & & 2 & 0 \\ 2 & \dots & 2 & x & 1 - \frac{x}{2} \\ 0 & \dots & 0 & 1 - \frac{x}{2} & 0 \end{bmatrix}.$$

Como $1 - \frac{x}{2} \neq 0$, para cada $i, 1 \leq i \leq m - 2$, procedemos

$$l_i \leftarrow l_i - \frac{2}{1 - \frac{x}{2}} l_m$$

$$c_i \leftarrow c_i - \frac{2}{1 - \frac{x}{2}} c_m$$

Isso anula todos os uns na linha e coluna $m - 1$:

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & x & 1 - \frac{x}{2} \\ 0 & \dots & 0 & 1 - \frac{x}{2} & 0 \end{bmatrix}.$$

As operações

$$l_{m-1} \leftarrow l_{m-1} + l_m$$

$$c_{m-1} \leftarrow c_{m-1} + c_m$$

substituem a diagonal x por 2, enquanto que as operações

$$l_m \leftarrow l_m - \frac{1 - \frac{x}{2}}{2} l_{m-1}$$

$$c_m \leftarrow c_m - \frac{1 - \frac{x}{2}}{2} c_{m-1}$$

eliminam os elementos fora da diagonal:

$$\begin{bmatrix} & & & 0 & 0 \\ & & & \vdots & \vdots \\ & & & 0 & 0 \\ 0 & \dots & 0 & 2 & 0 \\ 0 & \dots & 0 & 0 & -\frac{1}{2}(1 - \frac{x}{2})^2 \end{bmatrix}.$$

Como a linha e coluna $m - 1$ estão da forma desejada, fazemos a seguinte atribuição

$$m \leftarrow m - 1.$$

Caso 3: $b_{m-1} = b_m = 0$. Então a submatriz tem a seguinte forma:

$$\begin{bmatrix} & & & 2 & 2 \\ & & & \vdots & \vdots \\ & & & 2 & 2 \\ 2 & \dots & 2 & x & 2 \\ 2 & \dots & 2 & 2 & \alpha \end{bmatrix}.$$

É possível decompor este caso em três subcasos, mas de fato, usaremos uma simplificação para reduzi-lo. Consideremos as seguintes operações

$$l_m \leftarrow \frac{1}{2}l_m$$

$$c_m \leftarrow \frac{1}{2}c_m$$

obtemos a matriz:

$$\begin{bmatrix} & & & 2 & 1 \\ & & & \vdots & \vdots \\ & & & 2 & 1 \\ 2 & \dots & 2 & x & 1 \\ 1 & \dots & 1 & 1 & \frac{\alpha}{4} \end{bmatrix}.$$

Dessa forma podemos aplicar as operações do caso 2, já que a matriz está da forma apropriada para esse caso.

Caso 4: $b_{m-1} = 1$ e $b_m = 0$. Então a submatriz tem a seguinte forma:

$$\begin{bmatrix} & & & 1 & 2 \\ & & & \vdots & \vdots \\ & & & 1 & 2 \\ 1 & \dots & 1 & x & 2 \\ 2 & \dots & 2 & 2 & \alpha \end{bmatrix}.$$

Consideremos as seguintes operações

$$l_m \leftarrow \frac{1}{2}l_m$$


```

Algorithm Diagonalize( $G, x$ )
  initialize  $d(i) \leftarrow x$ , for all  $i$ 
  for  $m = n$  to 2
     $\alpha \leftarrow d(m)$ 
    if  $b_m = 0$  //Case 3 or Case 4
       $\alpha \leftarrow \frac{\alpha}{4}$ 
    if  $b_{m-1} = 1$ 
      if  $\alpha + x \neq 2$  //subcases 1a,4a
         $d(m-1) \leftarrow \frac{\alpha x - 1}{\alpha + x - 2}$ 
         $d(m) \leftarrow \alpha + x - 2$ 
      else if  $x = 1$  //subcases 1b,4b
         $d(m-1) \leftarrow 1$ 
         $d(m) \leftarrow 0$ 
      else //subcases 1c,4c
         $d(m-1) \leftarrow 1$ 
         $d(m) \leftarrow -(1-x)^2$ 
         $m \leftarrow m - 1$ 
    else if  $b_{m-1} = 0$ 
      if  $\alpha + \frac{x}{4} \neq 1$  //subcases 2a,3a
         $d(m-1) \leftarrow \frac{\alpha x - 1}{\alpha + \frac{x}{4} - 1}$ 
         $d(m) \leftarrow \alpha + \frac{x}{4} - 1$ 
      else if  $x = 2$  //subcases 2b,3b
         $d(m-1) \leftarrow 2$ 
         $d(m) \leftarrow 0$ 
      else //subcases 2c,3c
         $d(m-1) \leftarrow 2$ 
         $d(m) \leftarrow -\frac{1}{2}(1 - \frac{x}{2})^2$ 
         $m \leftarrow m - 1$ 
  end loop

```

Figura 7.3: Diagonalizando $\Theta + xI$

Para $m = 3$, o algoritmo executa o subcaso 1a, já que $b_3 = b_2 = 1, x = 0$, e $\alpha = \frac{1}{2}$. Logo as atribuições são:

$$d_2 \leftarrow \frac{\alpha x - 1}{\alpha + x - 2} = \frac{2}{3}$$

$$d_3 \leftarrow \alpha + x - 2 = -\frac{3}{2}.$$

Para $m = 2$, o algoritmo executa o subcaso 2a, já que $b_1 = 0, b_2 = 1$ e $x = 0$. Logo as atribuições são:

$$d_1 \leftarrow \frac{\alpha x - 1}{\alpha + \frac{x}{4} - 1} = 3$$

$$d_2 \leftarrow \alpha + \frac{x}{4} - 1 = -\frac{1}{3}.$$

A seguinte tabela fornece os passos do algoritmo em cada estágio:

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	0	0	0	0	0	0	3
1	0	1	0	1	$\frac{2}{3}$	1	$-\frac{1}{3}$
1	0	1	$\frac{1}{2}$	1	$-\frac{3}{2}$	1	$-\frac{3}{2}$
1	0	1	-2	1	-2	1	-2
inicial		após 1a		após 1a		após 2a	

Nosso segundo exemplo, assumimos que G é representado por $(0, 1, 0, 1)$ e $x = 1$. Após a inicialização, temos $m = 4, \alpha = x = 1, b_4 = 1$ e $b_3 = 0$. Então no primeiro passo, o algoritmo executa o subcaso 2a e as atribuições são:

$$d_3 \leftarrow \frac{\alpha x - 1}{\alpha + \frac{x}{4} - 1} = 0$$

$$d_4 \leftarrow \alpha + \frac{x}{4} - 1 = \frac{1}{4}.$$

Para $m = 3$, temos $\alpha = 0, x = 1, b_2 = 1$ e $b_3 = 0$. Então o subcaso 4a é executado e as atribuições são:

$$\alpha \leftarrow \frac{\alpha}{4} = 0$$

$$d_2 \leftarrow \frac{\alpha x - 1}{\alpha + x - 2} = 1$$

$$d_3 \leftarrow \alpha + x - 2 = -1.$$

Para $m = 2$, temos $\alpha = 1, x = 1, b_1 = 0$ e $b_2 = 1$. Então o subcaso 2a é executado e as atribuições são:

$$d_1 \leftarrow \frac{\alpha x - 1}{\alpha + \frac{x}{4} - 1} = 0$$

$$d_2 \leftarrow \alpha + \frac{x}{4} - 1 = \frac{1}{4}.$$

A seguinte tabela fornece os passos do algoritmo em cada estágio:

b_i	d_i	b_i	d_i	b_i	d_i	b_i	d_i
0	1	0	1	0	1	0	0
1	1	1	1	1	1	1	$\frac{1}{4}$
0	1	0	0	0	-1	0	-1
1	1	1	$\frac{1}{4}$	1	$\frac{1}{4}$	1	$\frac{1}{4}$
inicial		após 2a		após 4a		após 2a	

De forma análoga à matriz de adjacência, desejamos localizar os autovalores de um grafo *threshold* G , em relação à matriz distância Θ . Sejam G um grafo *threshold*, Θ sua matriz distância e $(a, b]$ um intervalo real. Primeiramente vamos determinar o número de autovalores de G em $(a, b]$. A prova do seguinte resultado é análoga ao Teorema 3.2.3.

Teorema 7.2.2. *Seja D a matriz diagonal congruente a $\Theta - aI$, onde Θ é real simétrica.*

- i. O número de autovalores de Θ maiores do que a é o número de entradas positivas em D .*
- ii. O número de autovalores de Θ menores do que a é o número de entradas negativas em D .*
- iii. A multiplicidade de um autovalor a é o número de entradas nulas na diagonal de D .*

Corolário 7.2.1. *O número de autovalores de Θ em $(a, b]$, incluindo a multiplicidade é o número de entradas positivas na diagonalização de $\Theta - aI$, menos o número de entradas positivas na diagonalização de $\Theta - bI$.*

Portanto de forma análoga à matriz de adjacência podemos determinar o número de autovalores em um intervalo, chamando o algoritmo de diagonalização duas vezes.

Se soubermos que o intervalo $(a, b]$ contém ao menos um autovalor então podemos localizar o autovalor à direita ou à esquerda, através de um algoritmo de bissecção.

Exemplo 7.2.1. *Consideremos o grafo threshold $G = (0, 1, 0, 1)$. Para $x = 1$, o algoritmo de diagonalização fornece os seguintes valores $d = (0, \frac{1}{4}, -1, \frac{1}{4})$. De acordo com o Teorema 7.2.2, significa que $x = -1$ é um autovalor de G com multiplicidade 1, existem dois autovalores maiores do que -1 e um autovalor menor do que -1 . Quando aplicarmos o algoritmo ao mesmo grafo para $x = 0$, obtemos a diagonal com os valores $d = (\frac{7}{3}, -\frac{3}{7}, -\frac{7}{4}, -1)$. Isso implica que 3 autovalores são negativos e 1 é positivo. Finalmente podemos concluir que existe um único autovalor no intervalo $(-1, 0]$.*

De forma análoga à matriz de adjacência, temos um resultado para a simplicidade dos autovalores de G em relação à matriz distância Θ . A prova é análoga ao Teorema 3.3.1.

Teorema 7.2.3. *Seja Θ a matriz distância de um grafo threshold G . Então todo autovalor $\lambda \neq -1, -2$ de G é simples.*

8 CONCLUSÃO

Nesta tese de doutorado trabalhamos com temas de teoria espectral de grafos. Mais especificamente, estudamos a classe de grafos denominada grafos *threshold*. O estudo dos grafos *threshold* foi intensamente motivador tanto pelas características estruturais desses grafos como por suas propriedades espectrais.

Para desenvolvermos este trabalho, no capítulo 2 apresentamos algumas caracterizações equivalentes dos grafos *threshold*, que consideramos importantes e que são conhecidas. Assim definimos os grafos *threshold* através de uma sequência binária, mostramos que esses grafos são livres de $\{2K_2, C_4, P_4\}$ e mostramos também que eles são grafos *split nested*.

Propriedades espectrais dos grafos *threshold* conhecidas da literatura, também foram apresentadas. Através da estrutura dos grafos *split nested* e da partição equilibrada mínima, mostramos como obter a multiplicidade dos autovalores não principais desses grafos. Além disso, usamos o grafo quociente com objetivo de obter o espectro principal de um *threshold*. Dessa forma apresentamos uma redução para o cálculo do polinômio característico de um grafo *threshold* de acordo com [39].

No intuito de obter um método eficiente para determinar o espectro de grafos *threshold*, a partir do capítulo 3 iniciamos com as nossas contribuições. Mesmo sabendo que os grafos *threshold* têm matrizes de adjacência que podem ser densas, desenvolvemos um algoritmo de tempo linear que constrói uma matriz diagonal D congruente a $A + xI$, onde A é matriz de adjacência de um grafo *threshold* G , x é um número real e I é a matriz de identidade. Embora nosso algoritmo seja baseado em operações elementares de matrizes, na prática sua implementação só depende da sequência binária do grafo *threshold* de ordem n , e pode assim ser implementado usando dois vetores de comprimento n , tal que sua complexidade é de ordem $O(n)$.

A partir desse algoritmo de diagonalização, obtivemos várias aplicações. Como principal aplicação, usamos a lei da Inércia de Sylvester para determinar exatamente quantos autovalores de um *threshold* G pertencem a um intervalo real $(a, b]$. Dessa forma, sabendo que esse intervalo contém ao menos um autovalor então podemos aproximá-lo, com uma certa precisão, usando um algoritmo de bissecção. Ainda no capítulo 3, usamos o algoritmo de diagonalização para mostrar que todo autovalor principal de G é simples.

Percebemos que o algoritmo de diagonalização, além de ser útil para determinar o número de autovalores de um grafo *threshold* em um intervalo qualquer, é uma ferramenta importantíssima para resolver outros problemas espectrais.

Vimos que é possível utilizá-lo para determinar quais são os grafos *threshold* que contêm o menor autovalor mínimo, dentre todos os grafos *threshold* de ordem n . Assim no capítulo 4, primeiramente determinamos o espectro de uma subclasse $G(k, j)$, representada por $0^k 1^j$. Então utilizamos o algoritmo de diagonalização para mostrar que de fato os grafos $G(k, j)$ contêm o menor autovalor mínimo, ou seja, provamos que $G(n - \lfloor \frac{n}{3} \rfloor, \lfloor \frac{n}{3} \rfloor)$ tem o menor autovalor mínimo entre todos os grafos *threshold* de ordem n . Quando $n \equiv 2 \pmod{3}$, o grafo $G(n - \lceil \frac{n}{3} \rceil, \lceil \frac{n}{3} \rceil)$ também possui esse autovalor. Como conhecíamos o espectro dos $G(k, j)$, obtemos uma fórmula para esse autovalor mínimo.

No capítulo 5, focamos nosso estudo para a outra extremidade do espectro de um grafo *threshold*, ou seja, estávamos interessados no índice de um grafo *threshold*. Usamos novamente o algoritmo de diagonalização para mostrar que existe uma ordem parcial sobre os grafos *threshold* G , de ordem n , de modo que seus índices estão relacionados. Além disso, identificamos uma família infinita de grafos *threshold* com espectro integral.

A nossa motivação para o capítulo 6, foi o trabalho de Jacobs e Trevisan [29], que apresenta um algoritmo de ordem $o(n^2)$ para o cálculo do polinômio característico de matrizes de adjacência de árvores. Dessa forma, adaptamos o nos-

so algoritmo de diagonalização para desenvolver um novo algoritmo para o cálculo do polinômio característico de um grafo *threshold* G . Até o presente momento, o método conhecido para resolver tal problema era o da redução apresentada por Sciriha e Farrugia [39].

A execução do nosso algoritmo para o cálculo do polinômio característico depende apenas da sequência binária de um *threshold*. Além disso, tal algoritmo é prático e mais simples do que a redução para o cálculo do polinômio característico apresentada no capítulo 2. Em relação à complexidade deste algoritmo, um possível trabalho futuro, é desenvolvermos um algoritmo com complexidade menor, usando-se a técnica de interpolação.

Dedicamos o capítulo final desta tese a uma extensão de nossos resultados para a matriz distância. Iniciamos apresentando um algoritmo de diagonalização para a matriz distância Θ de um grafo *threshold* G . De forma análoga, como aplicação, determinamos quantos autovalores de Θ pertencem a um intervalo real qualquer. Mostramos também que todo autovalor $\lambda \neq -1, -2$ da matriz distância Θ de um grafo *threshold* G , é simples.

Futuramente pretendemos estudar a energia de grafos *threshold*. A energia de um grafo G , definida por Gutman em [21] é a soma dos valores absolutos dos autovalores de G . Mais precisamente, se $\lambda_1, \lambda_2, \dots, \lambda_n$ são os autovalores da matriz de adjacência associada ao grafo G então a energia de G é dada por

$$E(G) = \sum_{i=1}^n |\lambda_i|.$$

Naturalmente esse invariante foi estendido para a matriz laplaciana. A energia laplaciana de um grafo G é dada por

$$E_L(G) = \sum_{i=1}^n |\mu_i - \bar{d}|,$$

onde (μ_i) são os autovalores laplacianos de G , e \bar{d} é a média dos graus dos vértices de G .

Em relação à energia laplaciana, existem vários problemas em aberto. Um deles é encontrar árvores de n vértices não coespectrais com a mesma energia laplaciana. Um outro problema interessante é sobre a energia laplaciana de grafos *threshold*. No trabalho de Del-Vecchio, Justo, Trevisan e Vinagre [17], foi conjecturado que certo grafo *threshold*, conhecido como abacaxi, possui a maior energia laplaciana dentre todos os grafos conexos de ordem n .

Um problema que estamos trabalhando em paralelo a esta tese, embora não esteja relacionado ao algoritmo de diagonalização, é a obtenção de uma fórmula para o determinante da matriz de adjacência de grafos *threshold*. Mais especificamente, desejamos obter tal fórmula para o determinante a partir da sequência binária de um grafo *threshold* G . A inspiração para obter tal fórmula, se deve a um resultado muito interessante sobre o determinante da matriz distância de uma árvore T , que não depende de sua estrutura mas somente de sua ordem [20]: o determinante da matriz distância de uma árvore T de ordem n , é obtido através da seguinte expressão $\det(\Theta(T)) = (-1)^{n-1}(n-1)2^{n-2}$.

De modo geral existe um considerável número de problemas em aberto relacionados aos grafos *threshold*. Assim, a partir deste trabalho acreditamos que seja possível resolver alguns desses problemas. Também vislumbramos que o nosso algoritmo de diagonalização (matriz de adjacência e distância) possa ser útil e até mesmo ser adaptado à outras classes de grafos, permitindo obtermos novos resultados espectrais.

Bibliografia

- [1] A. BALABAN, D. C., AND MEDELEANU, M. Topological indices and real number vertex invariants based on graph eigenvalues and eigenvectors. *Journal of Chemical Information and Modeling* 31 (1991), 517–523.
- [2] A.A. DOBRYNIN, R. E., AND GUTMAN, I. Wiener index of trees: theory and applications. *Acta Applicandae Mathematicae: An International Survey Journal on Applying Mathematics and Mathematical Applications* 66(3) (2001), 211–249.
- [3] BALABAN, A., AND DIUDEA, M. Real number vertex invariants: regressive distance sums and related topological indices. *Journal of Chemical Information and Modeling* 33 (1993), 421–428.
- [4] BALASUBRAMANIAN, K. Computer generation of distance polynomials of graphs. *Journal of Computational Chemistry* 11 (1990), 829–836.
- [5] BELL, F. K., CVETKOVIĆ, D. M., ROWLINSON, P., AND SIMIĆ, S. K. Graphs for which the least eigenvalue is minimal *i*. *Linear Algebra and its Applications* 429 (2008), 234–241.
- [6] BELL, F. K., CVETKOVIĆ, D. M., ROWLINSON, P., AND SIMIĆ, S. K. Graphs for which the least eigenvalue is minimal *ii*. *Linear Algebra and its Applications* 429 (2008), 2168–2179.
- [7] BIYIKOGLU, T., SIMIĆ, S., AND STANIĆ, Z. Some notes on spectra of cographs. *Ars Combinatoria* 100 (2011), 421–434.
- [8] BOGOMOLNY, E., BOHIGAS, O., AND SCHMIT, C. Spectral properties of distance matrices. *J. Phys. A; Math. Gen* 36 (2003), 3595–3616.
- [9] BRADLEY, G. L. *A Primer of Linear Algebra*. Prentice Hall, Englewood Cliffs, 1975.

- [10] BROUWER, A. E., AND HAEMERS, W. H. *Spectra of Graphs*. Springer, 2012.
- [11] CHVÁTAL, V., AND HAMMER, P. Aggregation of inequalities in integer programming. *Ann. Discrete Mathematics 1* (1977), 145–162.
- [12] CORNEIL, D., PERL, Y., AND STEWART, L. K. A linear recognition algorithm for cographs. *SIAM Journal on Computing 14(4)* (1985), 926–934.
- [13] CORNEIL, D. G., LERCHS, H., AND BURLINGHAM, L. Complement reducible graphs. *Discrete Appl. Math. 3* (1981), 163–175.
- [14] CVETKOVIĆ, D. M. The main part of spectrum, divisors and switching of graphs. *Publ. Inst. Math. (Beograd) 23(37)* (1978), 31–38.
- [15] CVETKOVIĆ, D. M., DOOB, M., AND SACHS, H. *Spectra of Graphs: Theory and Application*. Academic Press Inc, New York, 1980.
- [16] DE ABREU, N. M., DEL-VECCHIO, R. R., VINAGRE, C. T., AND STEVANOVIĆ, D. *Introdução a Teoria Espectral de Grafos com Aplicações*. São Carlos, SP: CNMAC, 2007.
- [17] DEL-VECCHIO, R., JUSTO, D., TREVISAN, V., AND VINAGRE, C. Maximum laplacian energy among threshold graphs. *To appear Linear Algebra and its Applications , 2013*.
- [18] GODSIL, C., AND ROYLE, G. *Algebraic Graph Theory. Graduate Texts in Mathematics, 207*. Springer-Verlag, New York, 2001.
- [19] GOLUBIC, M. C. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, New York, 1980.
- [20] GRAHAM, R., AND POLLAK, H. On the addressing problem for loop switching. *Bell System Tech. J. 50* (1971), 2495–2519.

- [21] GUTMAN, I. The energy of graph. *Ber. Math. Statist. Sect. Forschungszentrum Graz. 103* (1978), 1–22.
- [22] HAGOS, E. M. Some results on graph spectra. *Linear Algebra and its Applications 356* (2002), 103–111.
- [23] HARARY, F., AND SCHWENK, A. J. Which graphs have integral spectra? *Graphs and Combinatorics, Springer-Verlag, Lecture Notes in Mathematics 406* (1974), 55–51.
- [24] HÜCKEL, E. *Quantentheoretische Beiträge Zum Benzolproblem Z. Phys. 70* (1931), 204–286.
- [25] HENDERSON, P., AND Y.ZALCSTEIN. A graph-theoretic characterization of the pv class of synchronizing primitives. *SIAM Journal on Computing 6* (1977), 88–108.
- [26] INDULAL, G. Distance spectrum of graph compositions. *Ars Mathematica Contemporanea 2* (2009), 93–100.
- [27] JACOBS, D., TREVISAN, V., AND TURA, F. Distance eigenvalue location in threshold graphs. *Proceedings of Distance Geometry and Its Applications, Manaus, Amazonas, Brazil* (2013).
- [28] JACOBS, D., TREVISAN, V., AND TURA, F. Eigenvalue location in threshold graphs. *submitted to Linear Algebra and its Applications* (2013).
- [29] JACOBS, D. P., MACHADO, C. M., AND TREVISAN, V. An $o(n^2)$ algorithm for characteristic polynomial of a tree. *Journal of Combinatorial Mathematics and Combinatorial Computing 54* (2005), 213–221.
- [30] JACOBS, D. P., AND TREVISAN, V. Locating the eigenvalues of a tree. *Linear Algebra and its Applications 434* (2011), 81–88.
- [31] LIU, Z. On spectral radius of the distance matrix. *Appl. Anal. Discrete Math 4* (2010), 269–277.

- [32] M. RANDIC, A. K., AND DEALBA, L. Distance matrices. *Journal of Chemical Information and Modeling* 34 (1994), 277–286.
- [33] MAHADEV, N. V. R., AND PELED, U. N. Threshold graphs and related topics. *Ann. Disc. Maths.* 56 (1995).
- [34] MATHPAGES SPECTRAL GRAPH THEORY.
http://www.sgt.pep.ufrj.br/publications.php.
- [35] M.D. KONIG, S. BATTISTON, M. N., AND SCHWEITZER, F. The efficiency and stability of r&d networks. *Games and Economic Behavior* 75 (2012), 694–713.
- [36] MERRIS, R. Degree maximal graphs are laplacian integral. *Linear Algebra and its Applications* 199 (1994), 381–389.
- [37] RAMANE, H., REVANKAR, D. S., GUTMAN, I., AND WALIKAR, H. B. Distance spectra and distance energies of iterated line graphs of regular graphs. *Publications de L’Institut Mathématique* 85 (2009), 39–46.
- [38] ROYLE, G. F. The rank of cographs. *The Electronic Journal of Combinatorics* 10 (2003), N 11.
- [39] SCIRIHA, I., AND FARRUGIA, S. On the spectrum of threshold graphs. *ISRN Discrete Mathematics* (2011).
- [40] SIMIĆ, S. K., BELARDO, F., MARZI, E. M. L., AND TOSIĆ, D. V. Connected graphs of fixed order and size with maximal index: some spectral bounds. *Linear Algebra and its Applications* 432 (2010), 2361–2372.
- [41] STANIĆ, Z. On nested split graphs whose second largest eigenvalue is less than 1. *Linear Algebra and its Applications* 430 (2009), 2200–2211.
- [42] WIENER, H. Structural determination of paraffin boiling points. *Journal American Chemistic Society* 69(1) (1947), 17–20.

Índice

- α -sequência, 51
- algoritmo de diagonalização, 39
- algoritmo do polinômio característico, 69
- algoritmo KS, 21
- autovalor mínimo, 50
- autovalor principal, 44
- caterpillar, 20
- clique e estável, 10, 20
- coárvores, 12
- cografos, 11
- conjunto estável, 9
- definição de grafo *threshold*, 15
- distância de Hamming, 61
- extensão do algoritmo de diagonalização, 82
- grafo interseção, 16
- grafo livre de P_4 , 11
- grafo livre de $2k_2, C_4$ e P_4 , 16
- grafo quociente, 28
- grafo *split nested*, 10, 23
- grafo *split*, 10, 20
- lei da Inércia de Sylvester, 41
- matriz de adjacência, 2
- matriz de adjacência de um *threshold*, 27, 32
- matriz de adjacência do grafo quociente, 29
- matriz distância, 75
- matriz distância de um *threshold*, 76
- matrizes congruentes, 32
- nulidade de um grafo *threshold*, 27
- partição equilibrada, 25
- polinômio característico, 2
- raio espectral ou índice, 57
- redução para o cálculo do polinômio característico, 30
- subgrafo induzido, 12
- união e junção de um grafo, 11
- vértice coduplicado, 26
- vértice dominante, 15
- vértice duplicado, 26
- vértice interior, 12
- vértice isolado, 15
- vetor característico, 9