



Aumento máximo da conectividade algébrica em árvores de diâmetro maior ou igual a 4

Anderson Chaves da Silva

Instituto Militar de Engenharia
Praça General Tibúrcio 80, Praia Vermelha, Rio de Janeiro
andersonchaves.net@gmail.com

Claudia M. Justel

Instituto Militar de Engenharia
Praça General Tibúrcio 80, Praia Vermelha, Rio de Janeiro
cjustel@ime.eb.br

RESUMO

Neste trabalho apresentamos os resultados de experimentos comparativos entre duas heurísticas para o problema Aumento Máximo da Conectividade Algébrica, aplicados em classes de árvores de diâmetro maior que 4. As classes de árvores consideradas são denominadas *broom* e *double broom*. Também foram analisadas árvores *broom* de diâmetro aproximadamente metade do número de vértices da mesma. Nestas últimas árvores os resultados indicaram uma relação entre uma aresta que aumenta o valor da conectividade algébrica (quando inserida) e propriedades estruturais da árvore (grau máximo e vértices característicos).

PALAVRAS CHAVE. Grafos, Conectividade algébrica, Matriz laplaciana, Vetor de Fiedler.

Tópicos (Teoria e Algoritmos em Grafos)

ABSTRACT

This work presents experimental results comparing two heuristic procedures for the Maximum Algebraic Connectivity Augmentation Problem in families of trees with diameter greater than 4. The classes of trees considered were the broom and double broom trees. Moreover, broom trees with diameter fixed as half of the vertices on it were analysed. The results for this particular family of broom trees suggest a relation between the edge that more increases the algebraic connectivity with structural properties, such as maximum degree and characteristic vertices.

KEYWORDS. Graphs, Algebraic connectivity, Laplacian Matrix, Fiedler Vector

Paper topics (Graph Theory and Algorithms)



1. Introdução

A teoria espectral de grafos é o campo de pesquisa que se dedica ao estudo dos grafos com relação aos espectros correspondentes às diferentes matrizes associadas a estes.

Neste contexto, uma importante propriedade espectral a ser analisada é a conectividade algébrica, definida como o segundo menor autovalor da matriz laplaciana de um grafo. Esta propriedade, apresentada por Fiedler [1973], é amplamente analisada pelo fato de ser um bom parâmetro para a determinação de quão conexo é um grafo, e é por vezes mais descritiva do que os parâmetros de conectividade de vértices e conectividade de arestas. Esta medida está relacionada com diversas outras propriedades e problemas da teoria dos grafos, tais como o diâmetro, o número isoperimétrico e o problema do corte máximo (Mohar et al. [1991]).

Em Mosk-Aoyama [2008], é analisado o problema definido como o "Problema do Aumento Máximo da Conectividade Algébrica". Este problema consiste em, dadas algumas restrições, identificar um conjunto de arestas a serem incluídas em um grafo de modo a maximizar a conectividade algébrica resultante. No trabalho citado, foi provado que este problema pertence a classe dos problemas NP-Completo, fato que motiva a criação de algoritmos heurísticos para obter soluções aproximadas.

Ghosh e Boyd [2006] propuseram uma heurística para este fim, a Heurística de Perturbação (HP), a qual consiste em um procedimento iterativo no qual as arestas são adicionadas com base nos componentes do vetor de Fiedler do grafo em cada iteração. Além disso, Wang e Mieghem [2008] também propuseram um outro procedimento, no qual as arestas são escolhidas de forma aleatória.

Em Oliveira [2012] é proposto um algoritmo heurístico alternativo, a Heurística de Excentricidade (HE), o qual baseia-se nos valores de excentricidade de cada um dos vértices que constituem o grafo para a escolha das arestas a serem incluídas. O algoritmo foi proposto com base em resultados obtidos através de experimentos em uma classe específica de grafos, as árvores *double broom*, de diâmetros 3 e 4. Estas árvores foram apresentadas no trabalho de Fallat e Kirkland [1998], e são definidas mais detalhadamente na Seção 2.

Neste trabalho, apresentamos os resultados obtidos através de uma análise comparativa do comportamento da conectividade algébrica com relação aos algoritmos de HE e HP, através de experimentos em árvores *double broom*, de diâmetro superior a 4. Em cada experimento, verificamos o aumento proporcionado pela adição de uma única aresta ao grafo, escolhida por cada algoritmo. Os resultados obtidos demonstraram aumentos mais significativos por parte do algoritmo de HE. Em todos os casos, a aresta escolhida proporcionava um aumento da conectividade algébrica maior ou igual ao da aresta indicada pelo algoritmo de HP. Além disso, através de um algoritmo de força bruta, verificou-se que o aumento proporcionado pela aresta indicada pela HE foi em todos os casos próximo ao aumento máximo possível.

O restante deste trabalho está organizado da seguinte forma: na Seção 2, apresentamos algumas definições de grafos e teoria espectral de grafos utilizadas no trabalho, incluindo a definição de árvores *double broom*. Na Seção 3 incluímos a definição formal do problema proposto por Aoyama, e uma descrição dos algoritmos de HE e HP. A Seção 4 apresenta a metodologia utilizada e os resultados comparativos dos experimentos efetuados. Por último, na Seção 5, são discutidos os resultados obtidos.

2. Conceitos iniciais

As definições que utilizamos nesse artigo são baseadas nos trabalhos de Szwarcfiter [1986] e Abreu et al. [2014]. As definições de árvores *double broom* são baseadas nas apresentadas por Fallat e Kirkland [1998].

Neste artigo, nos referiremos a um **grafo** $G = (V, E)$, onde V é um conjunto finito não vazio de n elementos $v_i \in V, i \in \{1, 2, \dots, n\}$, denominados **vértices**, e E um conjunto de pares não ordenados de elementos distintos, do tipo $(v_i, v_j) \in E$, para $i \neq j, i, j \in \{1, 2, \dots, n\}$, denominados **arestas**. Nesse caso, os vértices v_i e v_j são os **vértices incidentes**, ou extremidades da



aresta (v_i, v_j) . Dizemos que duas arestas são adjacentes caso possuam uma extremidade em comum. Denotaremos por $\text{grau}(v)$ a cardinalidade do conjunto de arestas incidentes em v .

Um **caminho** é definido como uma sequência de vértices v_1, \dots, v_k , tal que $(v_i, v_{i+1}) \in E, 1 \leq i \leq k - 1$. O valor $k - 1$ corresponde ao **comprimento do caminho**. Denominamos **caminho simples** ao caminho que possui todos os vértices distintos. Um **ciclo** é um caminho $v_1, v_2, \dots, v_k, k \geq 3$, no qual v_1, \dots, v_{k-1} é um caminho simples, e $v_1 = v_k$.

A **distância** entre dois vértices $u, v \in V$ corresponde ao comprimento do menor caminho entre u e v , e a representamos por $d(u, v)$. A **excentricidade** de um vértice $v \in V$, denotada por $e(v)$, é a maior distância entre o vértice v e qualquer outro vértice do grafo. O **diâmetro** de um grafo G , denotado por $\text{diam}(G)$, corresponde ao comprimento do caminho que representa a maior distância entre pares de vértices do grafo.

Um **grafo conexo** é definido como um grafo para o qual existe um caminho entre qualquer par de vértices. O grafo que não possui ciclos é chamado de **grafo acíclico**.

Seja $G = (V, E)$ um grafo. O grafo $G' = (V', E')$ que satisfaz $V' \subseteq V$ e $E' \subseteq E$ é dito um **subgrafo** de G . Além disso, se G' é tal que dois vértices são adjacentes em G' se e somente se são adjacentes em G , dizemos que G' é um **subgrafo induzido** de G .

A **conectividade de arestas** de um grafo é definida como a cardinalidade do menor conjunto de arestas que, ao serem retiradas, tornam o grafo desconexo. Analogamente, a **conectividade de vértices** é definida como a cardinalidade do menor conjunto de vértices os quais ao serem removidos (juntamente com as respectivas arestas incidentes) tornam o grafo desconexo.

Definimos como a **matriz Laplaciana** de $G, L = L(G)$, a matriz $n \times n$ tal que:

$$L_{ij} = \begin{cases} \text{grau}(v_i), & \text{se } i = j \\ -1, & \text{se } i \neq j, (i, j) \in E \\ 0, & \text{se } i \neq j, (i, j) \notin E \end{cases}$$

O polinômio característico de L é dado por $p_L(\lambda) = \det(L - \lambda I)$, onde $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ são os **autovalores** de G , e I representa a matriz identidade. Um vetor não nulo $x \in R^n$ que satisfaça a equação $Lx = \lambda x$ é dito um **autovetor** de L associado a λ . O segundo menor autovalor de $L(G)$, λ_2 , é definido como a **conectividade algébrica** de G . Um autovetor correspondente a este autovalor é denominado **vetor de Fiedler**.

Seja $y = (y_1, y_2, \dots, y_n)$ um vetor de Fiedler correspondente ao grafo G . Então, pode ser definida uma função bijetora $f : V \rightarrow R^n$, de forma que $f(v_i) = y_i, i \in \{1, 2, \dots, n\}$. Os componentes do vetor y são chamados **valores característicos** de G .

Uma **árvore** é um grafo conexo e acíclico. Um **grafo caminho** é uma árvore na qual dois de seus n vértices possuem grau 1, e os $n - 2$ vértices restantes possuem grau 2. Denominamos **folha** a um vértice cujo grau é igual a 1.

As **árvores double broom** formam uma classe de estrutura particular. Uma árvore *double broom* de n vértices é uma árvore construída da seguinte forma: seja P_{d-1} um grafo caminho com $d - 1$ vértices, e sejam os vértices de P_{d-1} rotulados de s_1, \dots, s_{d-1} na ordem do caminho. Então, para $0 \leq k \leq \lfloor \frac{n-d+1}{2} \rfloor$, adicionemos k folhas adjacentes ao vértice s_1 , e $l = n - k - d + 1$ folhas ao vértice s_{d-1} , conforme ilustrado na Figura 1. Utilizamos a notação $T(n, d, k)$ para indicar a árvore *double broom* de n vértices, diâmetro d e parâmetro k . As árvores pertencentes a subclasse das *double brooms* para as quais $k = 1$ são chamadas **árvores broom**.

Em Fallat e Kirkland [1998] e Grone e Merris [1990], podem ser encontrados estudos relacionados a esta classe específica de árvores, e em Richter e Rocha [2016] as mesmas são classificadas e ordenadas de acordo com sua conectividade algébrica.

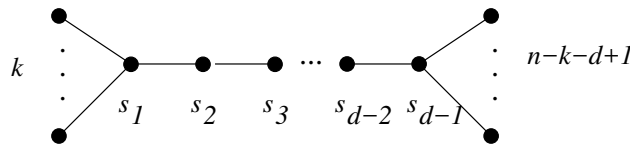


Figura 1: Estrutura de uma árvore *double broom*

Em Fiedler [1975] são definidos os vértices característicos de uma árvore da seguinte forma. Seja T uma árvore e para $v_i \in V$, $f(v_i) = y_i$, $1 \leq i \leq n$, seus valores característicos. Então, dois casos podem ocorrer:

Caso I. Todos os valores de y_i são diferentes de zero. Então T contém exatamente uma aresta (p, q) tal que $y_p > 0$ e $y_q < 0$. Os valores característicos dos vértices ao longo de qualquer caminho em T que inicie em p e que não contenha q serão crescentes; os valores característicos dos vértices ao longo de qualquer caminho que inicie em q e não contenha p serão decrescentes. Os vértices p e q são chamados **vértices característicos**.

Caso II. O conjunto $N_0 = \{i \in \{1, \dots, n\} : y_i = 0\}$ é não vazio. Então o grafo T_0 induzido por N_0 em T é conexo e há exatamente um vértice $j \in N_0$ possuindo ao menos um vizinho que não pertence a N_0 . Os valores característicos ao longo de qualquer caminho em T partindo de j podem crescer, decrescer ou não variar. O vértice j é chamado de **vértice característico**.

A Figura 2 apresenta duas árvores, T_1 e T_2 , correspondentes aos casos II e I, respectivamente. O vértice v é o vértice característico em T_1 , e os vértices i e j são os vértices característicos em T_2 .

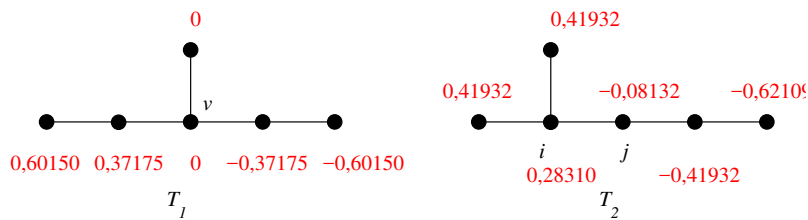


Figura 2: Duas árvores e seus respectivos valores característicos

3. Caracterização do Problema e Heurísticas

Formalmente, o problema do aumento máximo da conectividade algébrica pode ser definido da seguinte maneira: dado um grafo G e um inteiro $k \in \mathbb{N}$, determinar o menor conjunto de arestas $E_{cand} \subseteq E^c$, $|E_{cand}| = k$ a ser inserido em G , de modo a maximizar a conectividade algébrica do grafo resultante. Foi demonstrado por Mosk-Aoyama [2008] que este problema pertence á classe dos NP-Completo (através da sua redutibilidade ao problema da 3-Coloração de Grafos).

A menos que $P = NP$, torna-se impossível a construção de um algoritmo de complexidade polinomial para resolver esse problema. Por esse motivo foram desenvolvidas pesquisas procurando resolver o problema em classes particulares de grafos, assim como criando heurísticas para resolver o problema de forma aproximada para grafos em geral.

A Heurística de perturbação, HP, é o algoritmo heurístico apresentado por Ghosh e Boyd [2006], o qual utiliza o vetor de Fiedler a fim de determinar o conjunto de k arestas a ser incluído no grafo de entrada, para aumentar o valor da conectividade algébrica. A cada uma das k iterações do algoritmo, uma aresta $(u, v) \in E^c$ é escolhida, com base nos componentes de menor e maior valor do vetor de Fiedler do grafo.



A Heurística de excentricidade, HE, conforme proposta por Oliveira [2012] leva em consideração os valores de excentricidade dos vértices que constituem o grafo. A cada uma das k iterações do algoritmo, uma aresta $(u, v) \in E^c$ é escolhida com o objetivo de aumentar a conectividade algébrica do grafo atual. A aresta escolhida deve possuir em suas extremidades um vértice u de excentricidade máxima (i.e., um valor de excentricidade igual ao diâmetro do grafo), enquanto a outra extremidade deve incidir em um vértice v de excentricidade igual a $diam(G) - 1$. A distância entre os vértices u e v também deve ser igual a $diam(G) - 1$. Além disso, u ou v não devem ser vértices folha no grafo original. Como critério de desempate, o algoritmo seleciona a aresta (u, v) com maior valor de $e(u)$ e $grau(v)$. Conforme sugerido no trabalho de Rocha e Chaves [2013], como critério adicional, consideramos também o $grau(u)$ correspondente a aresta selecionada (u, v) .

4. Experimentos e Resultados

Neste trabalho, efetuamos diferentes experimentos comparativos em árvores do tipo *double broom*. Estas árvores foram previamente estudadas no trabalho de Oliveira [2012], para valores de diâmetro 3 e 4. Nesse trabalho, foi identificado qual tipo de aresta proporcionava o maior aumento da conectividade algébrica quando incluída em cada grafo.

Nossos experimentos foram realizados em um computador de processador Intel(R) Core i5, de 2.8 GHz, memória ram de 8GB. Utilizamos a biblioteca Networkx (versão 1.11) para a geração dos grafos, em conjunto com a biblioteca Numpy 1.11.2 para o cálculo dos autovalores e autovetores.

Inicialmente, fixado o número de vértices n e diâmetro d , avaliamos todas as possíveis árvores da classe *double broom*, e confirmamos os resultados citados para as árvores de diâmetro 3 e 4 (arestas de tipo 4 e tipo 8, respectivamente, conforme definidas no trabalho citado). Entretanto, não foi possível identificar qual tipo de aresta proporciona o maior aumento do parâmetro estudado para as árvores de diâmetro $d \geq 5$.

Em seguida, efetuamos diferentes testes comparativos em relação ao desempenho da Heurística de Excentricidade e a Heurística de Perturbação. Em cada teste, foram identificadas duas arestas, e_{HE} e e_{HP} , cada uma correspondente a seleção efetuada por parte da heurística de Excentricidade ou de Perturbação, respectivamente. Cada aresta selecionada foi incluída na árvore avaliada e então calculamos a conectividade algébrica do grafo resultante. Foi identificada também a aresta que proporcionava o maior aumento na conectividade algébrica possível utilizando um algoritmo de força bruta, a qual denotamos por e_{FB} .

Para diâmetros maiores ou iguais a 5, em quase todos os casos a aresta e_{HE} também apresentou melhores resultados. Alguns destes resultados, correspondentes as árvores de diâmetro $d = 20$, são demonstrados nas Tabelas 1 e 2.

Em seguida, restringimos nosso estudo às árvores *broom*. Cada um dos vértices foi rotulado de maneira consecutiva, começando pelos vértices no caminho P_{d-1} , sequencialmente, e continuando pelas $n - k - d + 1$ folhas, conforme ilustra a Figura 3. Efetuamos diferentes testes para grafos com número de folhas aproximadamente igual á metade dos vértices, $d = \lceil \frac{n}{2} \rceil$.

Em todos os casos analisados, as arestas do tipo e_{HE} proporcionaram um aumento maior da conectividade algébrica ou igual ao proporcionado pelas arestas do tipo e_{HP} . As Tabelas 3 e 4 demonstram os resultados obtidos pela inclusão destas arestas para árvores *broom* com número de vértices entre 5 e 40, e diâmetro $d = \lceil \frac{n}{2} \rceil$. Além disso, pode ser observado que, com exceção dos grafos $T(61, 31, 1)$ e $T(76, 38, 1)$, a aresta e_{HE} tem um vértice em comum com a aresta e_{FB} : o vértice d , de grau máximo, um dos critérios considerados pelo algoritmo.

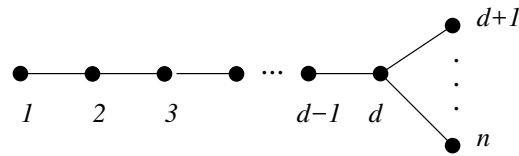


Figura 3: Árvore *broom* com vértices rotulados de 1 a n

d	n	k	l	G	$\lambda_2(G)$	e_{HP}	$\lambda_2(G + e_{HP})$	e_{EH}	$\lambda_2(G + e_{HE})$	e_{FB}	$\lambda_2(G + e_{FB})$
20	22	1	2	T(22, 20, 1)	0,0205	(1, 21)	0,0804	(1, 20)	0,0801	(3, 20)	0,0825
20	23	1	3	T(23, 20, 1)	0,0189	(1, 21)	0,0735	(1, 20)	0,0737	(3, 23)	0,0781
20	23	2	2	T(23, 20, 2)	0,0188	(1, 22)	0,0767	(1, 21)	0,0765	(3, 21)	0,0767
20	24	1	4	T(24, 20, 1)	0,0177	(1, 21)	0,0678	(1, 20)	0,0686	(4, 20)	0,0749
20	24	2	3	T(24, 20, 2)	0,0174	(1, 22)	0,0716	(1, 21)	0,0713	(3, 23)	0,0725
20	25	1	5	T(25, 20, 1)	0,0166	(1, 21)	0,0632	(1, 20)	0,0643	(4, 25)	0,0726
20	25	2	4	T(25, 20, 2)	0,0163	(1, 22)	0,0667	(1, 21)	0,0668	(4, 21)	0,0696
20	25	3	3	T(25, 20, 3)	0,0162	(1, 23)	0,0686	(1, 22)	0,0685	(3, 25)	0,0686
20	26	1	6	T(26, 20, 1)	0,0158	(1, 21)	0,0593	(1, 20)	0,0607	(4, 22)	0,0703
20	26	2	5	T(26, 20, 2)	0,0154	(1, 22)	0,0625	(1, 21)	0,0631	(4, 23)	0,0672
20	26	3	4	T(26, 20, 3)	0,0151	(1, 23)	0,0651	(1, 22)	0,0649	(4, 23)	0,0656
20	27	1	7	T(27, 20, 1)	0,015	(1, 21)	0,0561	(1, 20)	0,0578	(5, 20)	0,0692
20	27	2	6	T(27, 20, 2)	0,0145	(1, 22)	0,0589	(1, 21)	0,0599	(5, 21)	0,0655
20	27	3	5	T(27, 20, 3)	0,0143	(1, 23)	0,0616	(1, 22)	0,0617	(5, 22)	0,0635
20	27	4	4	T(27, 20, 4)	0,0142	(1, 24)	0,0629	(1, 23)	0,0628	(1, 26)	0,0629
20	28	1	8	T(28, 20, 1)	0,0144	(1, 21)	0,0534	(1, 20)	0,0552	(5, 20)	0,0682
20	28	2	7	T(28, 20, 2)	0,0139	(1, 22)	0,0559	(1, 21)	0,0571	(5, 21)	0,0643
20	28	3	6	T(28, 20, 3)	0,0135	(1, 23)	0,0584	(1, 22)	0,0589	(5, 24)	0,0619
20	28	4	5	T(28, 20, 4)	0,0134	(1, 24)	0,0603	(1, 23)	0,0602	(5, 26)	0,0607
20	29	1	9	T(29, 20, 1)	0,0138	(1, 21)	0,051	(1, 20)	0,053	(5, 27)	0,0673
20	29	2	8	T(29, 20, 2)	0,0133	(1, 22)	0,0532	(1, 21)	0,0547	(5, 24)	0,0631
20	29	3	7	T(29, 20, 3)	0,0129	(1, 23)	0,0556	(1, 22)	0,0564	(5, 24)	0,0603
20	29	4	6	T(29, 20, 4)	0,0127	(1, 24)	0,0577	(1, 23)	0,0578	(6, 23)	0,059
20	29	5	5	T(29, 20, 5)	0,0126	(1, 25)	0,0587	(1, 24)	0,0586	(4, 27)	0,0587
20	30	1	10	T(30, 20, 1)	0,0133	(1, 21)	0,049	(1, 20)	0,0511	(5, 20)	0,0662
20	30	2	9	T(30, 20, 2)	0,0128	(1, 22)	0,051	(1, 21)	0,0527	(5, 21)	0,0618
20	30	3	8	T(30, 20, 3)	0,0124	(1, 23)	0,0531	(1, 22)	0,0542	(6, 22)	0,0592
20	30	4	7	T(30, 20, 4)	0,0121	(1, 24)	0,0552	(1, 23)	0,0556	(6, 23)	0,0578
20	30	5	6	T(30, 20, 5)	0,012	(1, 25)	0,0567	(1, 24)	0,0566	(6, 26)	0,057
20	31	1	11	T(31, 20, 1)	0,0129	(1, 21)	0,0473	(1, 20)	0,0495	(5, 20)	0,0652
20	31	2	10	T(31, 20, 2)	0,0123	(1, 22)	0,049	(1, 21)	0,0509	(6, 21)	0,0609
20	31	3	9	T(31, 20, 3)	0,0119	(1, 23)	0,0509	(1, 22)	0,0523	(6, 22)	0,0585
20	31	4	8	T(31, 20, 4)	0,0116	(1, 24)	0,0528	(1, 23)	0,0536	(6, 24)	0,0567
20	31	5	7	T(31, 20, 5)	0,0114	(1, 25)	0,0546	(1, 24)	0,0547	(7, 24)	0,0554
20	31	6	6	T(31, 20, 6)	0,0114	(1, 26)	0,0554	(1, 25)	0,0553	(4, 31)	0,0554
20	32	1	12	T(32, 20, 1)	0,0125	(1, 21)	0,0457	(1, 20)	0,048	(5, 20)	0,0642
20	32	2	11	T(32, 20, 2)	0,0119	(1, 22)	0,0473	(1, 21)	0,0493	(6, 21)	0,0605
20	32	3	10	T(32, 20, 3)	0,0115	(1, 23)	0,0489	(1, 22)	0,0506	(6, 22)	0,0578
20	32	4	9	T(32, 20, 4)	0,0111	(1, 24)	0,0508	(1, 23)	0,0518	(6, 30)	0,0556
20	32	5	8	T(32, 20, 5)	0,0109	(1, 25)	0,0525	(1, 24)	0,053	(7, 24)	0,0546
20	32	6	7	T(32, 20, 6)	0,0108	(1, 26)	0,0538	(1, 25)	0,0537	(7, 29)	0,054
20	33	1	13	T(33, 20, 1)	0,0122	(1, 21)	0,0444	(1, 20)	0,0467	(6, 20)	0,0638
20	33	2	12	T(33, 20, 2)	0,0116	(1, 22)	0,0457	(1, 21)	0,0478	(6, 21)	0,0602
20	33	3	11	T(33, 20, 3)	0,0111	(1, 23)	0,0472	(1, 22)	0,049	(6, 31)	0,0572
20	33	4	10	T(33, 20, 4)	0,0108	(1, 24)	0,0489	(1, 23)	0,0502	(7, 23)	0,0546
20	33	5	9	T(33, 20, 5)	0,0105	(1, 25)	0,0506	(1, 24)	0,0513	(7, 24)	0,0538
20	33	6	8	T(33, 20, 6)	0,0104	(1, 26)	0,0521	(1, 25)	0,0522	(7, 31)	0,0527
20	33	7	7	T(33, 20, 7)	0,0103	(1, 27)	0,0528	(1, 26)	0,0527	(7, 30)	0,0528
20	34	1	14	T(34, 20, 1)	0,0119	(1, 21)	0,0431	(1, 20)	0,0455	(6, 20)	0,0637
20	34	2	13	T(34, 20, 2)	0,0113	(1, 22)	0,0444	(1, 21)	0,0466	(6, 21)	0,0598
20	34	3	12	T(34, 20, 3)	0,0108	(1, 23)	0,0457	(1, 22)	0,0477	(6, 22)	0,0564
20	34	4	11	T(34, 20, 4)	0,0104	(1, 24)	0,0472	(1, 23)	0,0488	(7, 23)	0,0542
20	34	5	10	T(34, 20, 5)	0,0101	(1, 25)	0,0488	(1, 24)	0,0499	(7, 25)	0,053
20	34	6	9	T(34, 20, 6)	0,01	(1, 26)	0,0504	(1, 25)	0,0508	(8, 25)	0,0519
20	34	7	8	T(34, 20, 7)	0,0099	(1, 27)	0,0515	(1, 26)	0,0515	(8, 28)	0,0517

Tabela 1: Resultados para árvores *double broom* onde $d = 20$ e $22 \leq n \leq 34$



d	n	k	l	G	$\lambda_2(G)$	e_{HP}	$\lambda_2(G + e_{HP})$	e_{HE}	$\lambda_2(G + e_{HE})$	e_{FB}	$\lambda_2(G + e_{FB})$
20	35	1	15	T(35, 20, 1)	0,0116	(1, 21)	0,042	(1, 20)	0,0444	(6, 20)	0,0635
20	35	2	14	T(35, 20, 2)	0,011	(1, 22)	0,0431	(1, 21)	0,0454	(6, 21)	0,0594
20	35	3	13	T(35, 20, 3)	0,0105	(1, 23)	0,0444	(1, 22)	0,0464	(6, 22)	0,0558
20	35	4	12	T(35, 20, 4)	0,0101	(1, 24)	0,0457	(1, 23)	0,0475	(7, 23)	0,0538
20	35	5	11	T(35, 20, 5)	0,0098	(1, 25)	0,0472	(1, 24)	0,0485	(7, 28)	0,0522
20	35	6	10	T(35, 20, 6)	0,0096	(1, 26)	0,0487	(1, 25)	0,0495	(8, 25)	0,0514
20	35	7	9	T(35, 20, 7)	0,0095	(1, 27)	0,0501	(1, 26)	0,0502	(8, 34)	0,0506
20	35	8	8	T(35, 20, 8)	0,0095	(1, 28)	0,0507	(1, 27)	0,0506	(8, 31)	0,0507
20	36	1	16	T(36, 20, 1)	0,0114	(1, 21)	0,0411	(1, 20)	0,0435	(6, 20)	0,0634
20	36	2	15	T(36, 20, 2)	0,0107	(1, 22)	0,042	(1, 21)	0,0444	(6, 21)	0,059
20	36	3	14	T(36, 20, 3)	0,0102	(1, 23)	0,0431	(1, 22)	0,0453	(6, 22)	0,0551
20	36	4	13	T(36, 20, 4)	0,0098	(1, 24)	0,0444	(1, 23)	0,0463	(7, 23)	0,0535
20	36	5	12	T(36, 20, 5)	0,0095	(1, 25)	0,0457	(1, 24)	0,0473	(7, 24)	0,0515
20	36	6	11	T(36, 20, 6)	0,0093	(1, 26)	0,0471	(1, 25)	0,0482	(8, 25)	0,0508
20	36	7	10	T(36, 20, 7)	0,0091	(1, 27)	0,0486	(1, 26)	0,049	(9, 26)	0,0497
20	36	8	9	T(36, 20, 8)	0,0091	(1, 28)	0,0496	(1, 27)	0,0496	(9, 31)	0,0498
20	37	1	17	T(37, 20, 1)	0,0112	(1, 21)	0,0402	(1, 20)	0,0426	(6, 20)	0,0632
20	37	2	16	T(37, 20, 2)	0,0105	(1, 22)	0,0411	(1, 21)	0,0434	(6, 21)	0,0586
20	37	3	15	T(37, 20, 3)	0,01	(1, 23)	0,042	(1, 22)	0,0443	(6, 22)	0,0544
20	37	4	14	T(37, 20, 4)	0,0096	(1, 24)	0,0431	(1, 23)	0,0452	(7, 23)	0,0531
20	37	5	13	T(37, 20, 5)	0,0092	(1, 25)	0,0444	(1, 24)	0,0462	(7, 24)	0,0508
20	37	6	12	T(37, 20, 6)	0,009	(1, 26)	0,0457	(1, 25)	0,0471	(8, 28)	0,0502
20	37	7	11	T(37, 20, 7)	0,0088	(1, 27)	0,0471	(1, 26)	0,0479	(9, 26)	0,0493
20	37	8	10	T(37, 20, 8)	0,0087	(1, 28)	0,0483	(1, 27)	0,0485	(9, 29)	0,0489
20	37	9	9	T(20, 37, 9)	0,0087	(1, 29)	0,0489	(1, 28)	0,0489	(4, 37)	0,0489
20	38	1	18	T(38, 20, 1)	0,011	(1, 21)	0,0394	(1, 20)	0,0418	(6, 20)	0,063
20	38	2	17	T(38, 20, 2)	0,0103	(1, 22)	0,0402	(1, 21)	0,0426	(6, 21)	0,0581
20	38	3	16	T(38, 20, 3)	0,0098	(1, 23)	0,041	(1, 22)	0,0434	(7, 22)	0,0543
20	38	4	15	T(38, 20, 4)	0,0093	(1, 24)	0,042	(1, 23)	0,0443	(7, 29)	0,0527
20	38	5	14	T(38, 20, 5)	0,009	(1, 25)	0,0431	(1, 24)	0,0451	(8, 24)	0,0505
20	38	6	13	T(38, 20, 6)	0,0088	(1, 26)	0,0444	(1, 25)	0,046	(8, 31)	0,0497
20	38	7	12	T(38, 20, 7)	0,0086	(1, 27)	0,0457	(1, 26)	0,0468	(9, 26)	0,0489
20	38	8	11	T(38, 20, 8)	0,0085	(1, 28)	0,047	(1, 27)	0,0475	(9, 27)	0,048
20	38	9	10	T(38, 20, 9)	0,0084	(1, 29)	0,048	(1, 28)	0,048	(10, 36)	0,0482
20	39	1	19	T(39, 20, 1)	0,0108	(1, 21)	0,0386	(1, 20)	0,0411	(6, 20)	0,0628
20	39	2	18	T(39, 20, 2)	0,0101	(1, 22)	0,0393	(1, 21)	0,0418	(6, 21)	0,0576
20	39	3	17	T(39, 20, 3)	0,0096	(1, 23)	0,0401	(1, 22)	0,0426	(7, 22)	0,0542
20	39	4	16	T(39, 20, 4)	0,0091	(1, 24)	0,041	(1, 23)	0,0434	(7, 23)	0,0523
20	39	5	15	T(39, 20, 5)	0,0088	(1, 25)	0,042	(1, 24)	0,0442	(8, 24)	0,0503
20	39	6	14	T(39, 20, 6)	0,0085	(1, 26)	0,0431	(1, 25)	0,045	(8, 25)	0,0491
20	39	7	13	T(39, 20, 7)	0,0083	(1, 27)	0,0444	(1, 26)	0,0458	(9, 26)	0,0485
20	39	8	12	T(39, 20, 8)	0,0082	(1, 28)	0,0456	(1, 27)	0,0465	(10, 27)	0,0475
20	39	9	11	T(39, 20, 9)	0,0081	(1, 29)	0,0469	(1, 28)	0,0471	(10, 38)	0,0474
20	39	10	10	T(39, 20, 10)	0,0081	(1, 30)	0,0474	(1, 29)	0,0474	(3, 39)	0,0474
20	40	1	20	T(40, 20, 1)	0,0106	(1, 21)	0,0379	(1, 20)	0,0404	(6, 20)	0,0626
20	40	2	19	T(40, 20, 2)	0,0099	(1, 22)	0,0386	(1, 21)	0,0411	(6, 21)	0,0572
20	40	3	18	T(40, 20, 3)	0,0094	(1, 23)	0,0393	(1, 22)	0,0418	(7, 22)	0,0541
20	40	4	17	T(40, 20, 4)	0,009	(1, 24)	0,0401	(1, 23)	0,0425	(7, 23)	0,0519
20	40	5	16	T(40, 20, 5)	0,0086	(1, 25)	0,041	(1, 24)	0,0433	(8, 24)	0,0502
20	40	6	15	T(40, 20, 6)	0,0083	(1, 26)	0,042	(1, 25)	0,0441	(8, 25)	0,0486
20	40	7	14	T(40, 20, 7)	0,0081	(1, 27)	0,0431	(1, 26)	0,0449	(9, 39)	0,0481
20	40	8	13	T(40, 20, 8)	0,008	(1, 28)	0,0444	(1, 27)	0,0456	(10, 27)	0,0472
20	40	9	12	T(40, 20, 9)	0,0079	(1, 29)	0,0456	(1, 28)	0,0462	(10, 28)	0,0467
20	40	10	11	T(40, 20, 10)	0,0078	(1, 30)	0,0466	(1, 29)	0,0467	(11, 36)	0,0468

Tabela 2: Resultados para árvores *double broom* onde $d = 20$ e $35 \leq n \leq 40$



5. Considerações Finais

Neste trabalho apresentamos os resultados de experimentos comparativos entre a Heurística de Excentricidade e Heurística de Perturbação, em relação ao aumento da conectividade algébrica, aplicados em algumas árvores do tipo *broom* e *double broom*. Os experimentos demonstram um desempenho superior para esta classe de árvores com diâmetro maior que 4, obtidas com a HE, comparado com o resultado obtido pela HP. Os resultados obtidos mostraram um bom desempenho da HE para árvores *broom* de diâmetro aproximadamente metade do número de vértices da mesma, sugerindo uma relação da aresta escolhida por esta heurística e propriedades estruturais da árvore, como o grau máximo e os vértices característicos.

Como trabalhos futuros, sugerimos uma análise da relação entre a aresta que produz o maior aumento de conectividade ao ser inserida em uma *broom* com os vértices característicos da mesma. Com esse estudo mais aprofundado nas árvores *broom* pretendemos obter informações que permitam aprimorar a HE. Além disso, destacamos que os experimentos aqui apresentados não consideraram a inclusão de subconjuntos de arestas de tamanho maior ou igual a 2 nos grafos analisados, e que também serão abordados nos trabalhos futuros.

6. Agradecimentos

Os autores agradecem às agências de fomento, Programa CAPES Demanda Social e CNPq (305677/2013-6).

Referências

- Abreu, N., Del-Vecchio, R., Trevisan, V., e Vinagre, C. (2014). Teoria espectral de grafos - uma introdução. *III Colóquio de Matemática da Região Sul*.
- Fallat, S. e Kirkland, S. (1998). Extremizing algebraic connectivity subject to graph theoretic constraints. *The Electronic Journal of Linear Algebra*, 3(1):7.
- Fiedler, M. (1973). Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2): 298–305.
- Fiedler, M. (1975). A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czechoslovak mathematical journal*, 25(4):619–633.
- Ghosh, A. e Boyd, S. (2006). Growing well-connected graphs. In *IEEE Conference on Decision and Control*, number 45 in IEEE Conference, p. 6605–6611, USA. IEE.
- Grone, R. e Merris, R. (1990). Ordering trees by algebraic connectivity. *Graphs and Combinatorics*, 6(3):229–237.
- Mohar, B., Alavi, Y., Chartrand, G., e Oellermann, O. (1991). The laplacian spectrum of graphs. *Graph Theory, Combinatorics, and Applications*, 2(871-898):12.
- Mosk-Aoyama, D. (2008). Maximum algebraic connectivity augmentation is NP-hard. *Operations Research Letters*, 36(6):677–679.
- Oliveira, C. C. G. F. (2012). Sobre o problema aumento máximo da conectividade algébrica. In *XLIV Simpósio Brasileiro de Pesquisa Operacional, SBPO*, p. 1–11, Rio de Janeiro. SOBRAPO.
- Richter, S. e Rocha, I. (2016). Absolute algebraic connectivity of double brooms and trees. *Discrete Applied Mathematics*, 201(1):213–221.
- Rocha, C. e Chaves, E. (2013). Algoritmos para problemas de perturbações da matriz laplaciana de um grafo. Relatório de iniciação à pesquisa, Instituto Militar de Engenharia.
- Szwarcfiter, J. L. (1986). *Grafos e Algoritmos Computacionais*. Campus, Rio de Janeiro.



Wang, H. e Miegheem, P. V. (2008). Algebraic connectivity optimization via link addition. In *International Conference on Bio-Inspired Models of Network, Information and Computing Systems*, number 3 in ICST, p. 22–30, Huijuan. ICST.