

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE MATEMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

ROMEU INOJOSA LUSTOSA PIRES

CONSTRUINDO REDES EFICIENTES E ROBUSTAS MINIMIZANDO DISTÂNCIAS
EM GRAFOS BICONEXOS

RIO DE JANEIRO
2021

ROMEU INOJOSA LUSTOSA PIRES

CONSTRUINDO REDES EFICIENTES E ROBUSTAS MINIMIZANDO DISTÂNCIAS
EM GRAFOS BICONEXOS

Trabalho de conclusão de curso de graduação
apresentado ao Departamento de Ciência da
Computação da Universidade Federal do Rio
de Janeiro como parte dos requisitos para ob-
tenção do grau de Bacharel em Ciência da
Computação.

Orientador: Prof. Daniel Sadoc Menasche

RIO DE JANEIRO

2021

CIP - Catalogação na Publicação

P667c Pires, Romeu Inojosa Lustosa
Construindo redes eficientes e robustas
minimizando distâncias em grafos biconexos / Romeu
Inojosa Lustosa Pires. -- Rio de Janeiro, 2021.
49 f.

Orientador: Daniel Sadoc Menasche.
Trabalho de conclusão de curso (graduação) -
Universidade Federal do Rio de Janeiro, Instituto
de Matemática, Bacharel em Ciência da Computação,
2021.

1. Grafos. 2. Algoritmos. I. Menasche, Daniel
Sadoc, orient. II. Título.

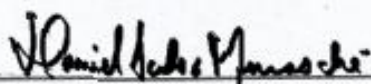
ROMEU INOJOSA LUSTOSA PIRES

CONSTRUINDO REDES EFICIENTES E ROBUSTAS MINIMIZANDO DISTÂNCIAS
EM GRAFOS BICONEXOS

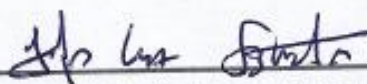
Trabalho de conclusão de curso de graduação
apresentado ao Departamento de Ciência
da Computação da Universidade Federal do
Rio de Janeiro como parte dos requisitos para
obtenção do grau de Bacharel em Ciência da
Computação.

Aprovado em 24 de novembro de 2021

BANCA EXAMINADORA:



Prof. Daniel Sadoc Menasché,
Orientador
Ph.D. (UMass)



Prof. Jayme Luiz Szwarcfiter
Ph.D. (University of Newcastle)



Profa. Márcia Rosana Cerioli
D.Sc. (UFRJ)

Documento assinado digitalmente
gov.br MITRE COSTA DOURADO
Data: 03/12/2021 19:26:10-0300
Verifique em <https://verificador.ri.br>

Prof. Mitre Costa Dourado
D.Sc. (UFRJ)

RESUMO

Motivados pela importância de construir redes eficientes e robustas, apresentamos uma conjectura sobre uma classe de grafos, os theta-grafos generalizados, que minimizam o índice de Wiener (implicando, de uma maneira geral, em mais eficiência) ao mesmo tempo que se mantêm biconexos ("primeiro nível" de robustez). Também é apresentado um algoritmo otimizado para reforçar a conjectura e uma ferramenta lúdica e iterativa para construção de grafos.

Palavras-chave: Grafos biconexos; Robustez; Índice de Wiener; Topologia.

ABSTRACT

Motivated by the importance of building efficient and robust networks, we present a conjecture about a class of graphs, the generalized theta-graphs, which minimize the Wiener index (generally implying more efficiency) while remaining biconnected ("first level" of robustness). A web-based, playful and iterative tool for graph construction is also presented.

Keywords: Biconnected graphs; Robustness; Wiener index; Topology.

LISTA DE ILUSTRAÇÕES

Figura 1 – Dois grafos fundamentais para a construção da classe na qual $m = 2n - 5$. Vértices de grau 2 podem ser inseridos entre quaisquer dois vértices não diretamente conectados, sem abrir mão da propriedade $d = 2$	16
Figura 2 – Distâncias entre vértices internos (x_i) de uma ponte com um vértice central y	19
Figura 3 – Distâncias entre vértices internos de uma mesma ponte. d é a menor distância entre os vértices x_1 e x_n que não passa por dentro da ponte.	19
Figura 4 – Distâncias entre os vértices internos x_i de uma ponte P_a com os vértices internos y_i de uma ponte P_b	22
Figura 5 – Tempo de execução em função do número de vértices em theta-grafos de 5 pontes.	26
Figura 6 – Tempo de execução em theta-grafos gerados aleatoriamente (número de vértices distribuído uniformemente entre 20 e 1000)	27
Figura 7 – Histograma do tempo de execução em função do número de vértices em grafos arbitrariamente gerados.	27
Figura 8 – Histograma do tempo de execução em diferentes grafos arbitrários com 200 vértices.	28
Figura 9 – Classe dos grafos biconexos com $m=n+1$ arestas.	30
Figura 10 – Visualização de solução com somatório das colunas	32
Figura 11 – Visualização de solução com somatório das áreas, agrupadas por linhas iguais	32
Figura 12 – Exemplo de saída do programa geng : grafos biconexos(- C) de 11 vértices e 12 arestas. O formato apresentado é uma codificação da forma matricial comprimida, graph6. (Brendan McKay, 2015)	33
Figura 13 – Topologias possíveis com $m = n + 1$ e sem vértices de grau 1. Note que nem todas são biconexas: grafos biconexos não podem ter vértices centrais em ambas as extremidades de uma ponte. A topologia de theta-grafo é a única topologia biconexa.	35
Figura 14 – Crescimento do índice de Wiener ótimo para theta-grafos com 3 pontes($m = n + 1$) em função do número de vértices, n	36
Figura 15 – Comparação do tempo despendido no processo de exploração e cálculo do índice de Wiener na classe $m = n + 1$, entre os dois métodos empregados.	37

Figura 16 – Topologias biconexas possíveis com $m = n + 2$. Apelidamos a segunda topologia de " <i>quatro33</i> "(quatro-três-três), em decorrência dos graus dos vértices centrais. A terceira topologia nos referimos como a de " <i>tetraedro</i> " e a última de " <i>cilindro</i> ", ambas pelas formas visuais. . . .	38
Figura 17 – Crescimento do índice de Wiener em função do número de vértices, n , para grafos com $m = n + 2$ arestas. Também foi incluído os valores para a topologia de theta-grafo com 3 pontes($m = n + 1$), para comparação.	40
Figura 18 – Comparação do tempo despendido no processo de exploração e cálculo do índice de Wiener na classe $m = n + 2$, entre os dois métodos empregados.	41
Figura 19 – Demonstração da aplicação "about graphs": cada componente conexa é colorida com uma cor. Pode-se optar por mostrar a excentricidade e centralidade de cada vértice. É possível ativar(padrão) ou desativar o comportamento de "mola" nas arestas, para auxiliar na organização espacial.	43

SUMÁRIO

1	MOTIVAÇÃO E CONJECTURA	10
1.1	TERMINOLOGIA	10
1.2	OBJETIVOS	11
1.3	CONTRIBUIÇÕES	12
1.4	ROTEIRO	13
2	RESULTADOS FORMAIS	14
3	O ÍNDICE DE WIENER DE GRAFOS MUITO ESPARSOS	18
3.1	O ÍNDICE DE WIENER POR PARTES.	18
3.1.1	Distâncias entre vértices centrais	19
3.1.2	Distâncias entre vértices internos com vértices centrais	19
3.1.3	Distâncias entre vértices internos numa mesma ponte	20
3.1.4	Distâncias entre vértices internos em pontes diferentes	21
3.2	ALGORITMOS	22
3.3	COMPARATIVO DE PERFORMANCE	23
4	EXPLORAÇÃO DE GRAFOS E O PROBLEMA DO PARTI-	
	CIONAMENTO	29
4.1	MOTIVAÇÃO	29
4.2	O PROBLEMA DO PARTIÇÃOAMENTO	29
5	RESULTADOS NUMÉRICOS	33
5.1	ANALISANDO GRAFOS COM GENG	33
5.2	ANALISANDO GRAFOS COM O ALGORITMO ESPARSO.	34
5.2.1	Topologias $m = n + 1$	34
5.2.2	Topologias $m = n + 2$	35
5.2.3	Topologias $n + 3 \leq m \leq 2n - 6$	41
6	JOGO PROPOSTO	43
6.1	ABOUT GRAPHS	43
6.1.1	Possíveis melhorias	43
7	TRABALHOS RELACIONADOS	45
8	CONCLUSÃO	46
8.1	TRABALHOS FUTUROS	46

REFERÊNCIAS 48

1 MOTIVAÇÃO E CONJECTURA

Robustez e eficiência são duas das características desejáveis em sistemas computacionais. Em redes de computadores, por exemplo, robustez refere-se ao fato de as redes serem tolerantes a falhas, e eficiência refere-se ao fato de os caminhos entre os nós serem curtos. A busca por topologias de redes robustas e eficientes motivou inúmeros trabalhos na interseção entre redes de computadores, teoria dos grafos e inteligência artificial.

Embora robustez e eficiência sejam ambas desejáveis, pode haver um compromisso entre os dois. Considere, por exemplo, que exista um custo para adicionar uma conexão entre dois nós dentro de uma rede. Assuma que o orçamento para uma rede com n vértices só comporte $m = n + 1$ conexões diretas (arestas). Nesse caso, é possível alcançar eficiência máxima, por exemplo, conectando todos os demais vértices por meio de um nó central (semelhante a uma forma de estrela), envolvendo $n - 1$ arestas, e as duas arestas adicionais sendo conectadas arbitrariamente, conferindo uma distância entre nós sempre menor ou igual a 2. Em outras palavras, a distância entre nós é $O(1)$. Entretanto, esta topologia derivada da estrela é muito frágil, tendo em vista que a remoção do nó central desconecta os nós restantes. Alternativamente, pode-se ter uma solução muito robusta, contendo um ciclo envolvendo todos os nós, na qual é impossível desconectar quaisquer dois nós com a remoção de um terceiro. Em contrapartida, nesse caso a distância entre os nós será da ordem de $O(n)$. A topologia derivada a partir do ciclo é muito mais robusta, mas ao mesmo tempo muito menos eficiente em comparação com a topologia derivada a partir da estrela.

1.1 TERMINOLOGIA

Neste trabalho, ao analisarmos diferentes topologias, tentaremos nos restringir a um certo conjunto de medidas, relativas aos conceitos de eficiência e robustez. Serão elas:

- K -vértice-conexo
 - Quando se trata de robustez de uma rede, consideramos importante a capacidade de perder um ou mais nós e ainda assim a rede estar conexa. Nos restringiremos aos casos onde a rede pode admitir a perda de **qualquer** porém no máximo **apenas um** nó. Ou seja, consideramos grafos 2-vértice-conexos, também chamados de biconexos.
- Índice de Wiener
 - Seja $V(G)$ o conjunto de vértices do grafo G e $d_G(u, v)$ a distância mínima entre dois vértices u e v tais que $u, v \subseteq V(G)$. O índice de Wiener $W(G)$ pode

ser calculado da seguinte maneira:

$$W(G) = \sum_{(u,v) \subseteq V(G)} d_G(u,v)$$

- Observe que cada par de vértices (u, v) pode ser contabilizado também como (v, u) . A conversão entre $W(G)$ e o somatório das distâncias considerando a ordem do par irrelevante, $W'(G)$, é simples e direta:

$$W(G) = 2W'(G)$$

- Consideraremos o índice de Wiener a principal medida de eficiência. É válida porém a constatação de que em certos contextos certos vértices devem ser priorizados na minimização de suas distâncias e, nesse caso, a topologia com o menor índice de Wiener pode não ser a mais eficiente. Ainda assim, neste estudo nos restringiremos ao esforço de minimizar o índice de Wiener, deixando como objeto para trabalhos futuros o estudo de outras métricas de eficiência, por exemplo levando em conta critérios de centralidade, para a generalização do problema.
- Relação número de arestas *versus* número de vértices
 - Caso não houvesse necessidade de regular o número de arestas em relação com o número de vértices, sempre poderíamos pedir que o número de arestas m fosse $m = \frac{n(n-1)}{2}$, e neste caso, poderíamos construir um grafo completo, cuja eficiência e robustez são obviamente sempre ótimas.

1.2 OBJETIVOS

Dados os critérios acima, visamos responder a seguinte pergunta:

- restringindo-se a redes robustas, ou seja, biconexas, quais as topologias mais eficientes?

Surpreendentemente, embora esta pergunta seja muito simples, sua resposta ainda não é conhecida.

Quanto à relação entre m (número de arestas) e n (número de vértices), algumas classes às quais este estudo se dedica ou são dignas de nota são:

- $m = \frac{n(n-1)}{2}$
 - Este é o limite superior para os possíveis valores de m . O único grafo possível com esta configuração é o grafo completo, onde o índice de Wiener é o menor possível e o grafo é o mais robusto possível, sendo $(n - 1)$ -vértice-conexo.

- $m = 2n - 5$
 - Em (MURTY, 1968) foi apresentado um método de construção de grafos biconexos com m arestas, n vértices e diâmetro d onde $m = 2n - 5$ e $d = 2$. Foi provado que o número de arestas não pode ser menor sem abrir mão da propriedade $d = 2$
- $m = n, m = n + 1, m = n + 2, \dots, m = 2n - 6$
 - O caso $m = n$ é o ponto de partida para os grafos biconexos. Provaremos que sua topologia é única (ciclo). Os casos em sequência são analisados a partir do anterior. Tentaremos encontrar topologias “extremas” nestes casos, ou seja, quais topologias minimizam o índice de Wiener.

Motivados por redes nas quais seja custoso adicionar arestas, ou seja, nas quais seja necessário trabalhar no regime esparso, propomos então a seguinte conjectura:

Conjectura 1. *Dados o número de arestas (m) e o número de vértices (n), e restringindo-se a grafos biconexos e esparsos, em que $m < 2n - 5$, sempre existe um grafo theta-generalizado que minimiza a soma das distâncias entre os vértices.*

Definimos como grafo theta-generalizado um grafo com todos os vértices sendo de grau 2, com exceção de apenas 2, com grau maior ou igual à 3.

1.3 CONTRIBUIÇÕES

Ao tentarmos provar a conjectura acima, e visando responder a pergunta chave levantada neste trabalho, fazemos as seguintes contribuições:

- **Verificação da conjectura até 17 nós:** por meio de análise exaustiva de todos os grafos biconexos até 17 nós, verificamos que a conjectura sempre vale nesses casos. Usamos o programa geng, do pacote nauty, adaptado para nossos fins, e acabamos verificando que nesses casos o grafo teta-generalizado *balanceado* minimiza a soma das distâncias. Verificamos também que para 20 ou mais nós, o grafo teta-generalizado *balanceado* nem sempre minimiza a soma das distâncias, mas o grafo teta-generalizado *não balanceado* continua sendo um candidato
- **Caracterização dos grafos biconexos muito esparsos:** quando os grafos são muito esparsos ($m = n + k; k \leq 2$), caracterizamos completamente as possíveis configurações dos mesmos, abrindo espaço para provar a conjectura nesses casos
- **Fórmulas eficientes para soma das distâncias e para número de grafos:** derivamos várias fórmulas fechadas para somas das distâncias de vários grafos especiais de interesse, e para o número de grafos em cada classe. Tais fórmulas poderão vir a ser instrumentais para finalmente provar a conjectura em questão

- **Ambiente/jogo interativo:** propusemos um ambiente/jogo para os usuários poderem tentar achar contra-exemplos de forma recreativa

1.4 ROTEIRO

No capítulo 2 mostraremos alguns teoremas relevantes para o trabalho e suas demonstrações. No capítulo 3 apresentaremos um algoritmo para calcular o índice de Wiener de forma mais eficiente. No capítulo 5, à procura de provas que invalidem a proposta conjectura 1, mostraremos experimentos realizados com ferramentas previamente disponíveis e como foi possível melhorá-los com um método de iteração descrito pelo capítulo 4 e o algoritmo do capítulo 3. No capítulo 6 apresentamos uma ferramenta iterativa criada para a plataforma web (HTML5) e, por último, no capítulo 7 apresentamos trabalhos relacionados e no capítulo 8 fazemos uma breve conclusão.

2 RESULTADOS FORMAIS

Teorema 1 (diâmetro e distâncias). *Considerando grafos conexos com n vértices e m arestas (n e m dados), se o diâmetro é igual a 2, a soma das distâncias é mínima, e igual a $2(n(n-1) - 2m) + 2m$.*

Prova: Começamos considerando que $m = n(n-1)/2$, ou seja, que temos um grafo completo. Nesse caso, o diâmetro é igual a 1. Logo, os grafos completos não se enquadram nas premissas do teorema, e podemos assumir, daqui em diante, que $m < n(n-1)/2$, ou seja, $n(n-1) - 2m > 0$.

Se o diâmetro é igual a 2, todos os nós alcançam todos os outros em no máximo 2 passos. Temos $n(n-1) - 2m$ pares de nós que não estão diretamente conectados uns aos outros (observe que este é um número válido pois é maior que zero). A soma das distâncias entre esses nós é igual a $2(n(n-1) - 2m)$. A soma das distâncias entre os pares de nós ligados diretamente é igual a $2m$. Logo, a soma das distâncias é igual a $2(n(n-1) - 2m) + 2m$. Por contradição, não é possível alcançar uma soma de distâncias menor que essa. Caso fosse possível, ou o diâmetro seria menor que 2, ou então mais do que $2m$ pares de nós teriam distância igual a 1. Ambas as possibilidades são contradições. \square

Abaixo, apresentamos uma demonstração alternativa para o teorema acima.

Prova: Demonstraremos que, fixos n e m , é com $d = 2$ que obtemos o menor valor possível para o índice de Wiener.

Seja G_d um grafo biconexo de diâmetro $d \geq 2$, n vértices, $m = 2n - 5$ arestas e índice de Wiener o menor possível. Uma das maneiras de se calcular o índice de Wiener de um grafo G é a partir do somatório entre o número de pares distantes por k arestas, $S_k(G)$, para todo k . Ou seja:

$$W(G_d) = \sum_{i=1}^d iS_i(G_d) \quad (2.1)$$

O somatório $\sum_{i=1}^d S_i(G_d)$ é conhecido e independe de d :

$$\sum_{i=1}^d S_i(G_d) = n(n-1)$$

Logo,

$$W(G_d) = n(n-1) + S_2(G_d) + \sum_{i=3}^d (i-1)S_i(G_d)$$

A fim de minimizar o somatório acima, precisamos determinar como os $n(n-1) - 2m$ pares de nós que não estão diretamente conectados irão contribuir para o somatório.

Claramente, alocar todos estes pares no conjunto correspondente a $S_2(G_d)$ irá minimizar a expressão acima, tendo em vista que os elementos alocados a este conjunto estão associados ao fator multiplicativo 1, enquanto que todos os demais termos estão associados a fatores multiplicativos maiores. Assim sendo, $W(G_d)$ é minimizado quando o diâmetro do grafo é igual a 2,

$$W(G_d) = n(n-1) + n(n-1) - 2m$$

Concluimos que, dados m e n , o índice de Wiener é mínimo quando o diâmetro do grafo é igual a 2. \square

Uma observação interessante sobre (2.1) consiste no fato de que ela motiva o chamado polinômio de Hosoya (CASH, 2002),

$$W(z; G_d) = \sum_{i=1}^d z^i S_i(G_d) \quad (2.2)$$

Claramente, a derivada de $W(z; G_d)$ com relação a z , no ponto $z = 1$, gera o índice de Wiener. Em geral, o polinômio de Hosoya é a função geradora associada ao índice de Wiener.

Corolário 1. *Em grafos biconexos, com n vértices e m arestas, se o diâmetro é igual a 2, a soma das distâncias é mínima e igual a $2(n(n-1) - 2m) + 2m$.*

O corolário acima segue imediatamente a partir do Teorema 1. Claramente, a recíproca do corolário acima não é válida, como indicado pelo seguinte teorema.

Teorema 2. *Se $m = n$, só existe um grafo biconexo, o ciclo, cujo diâmetro é $O(n)$.*

Esta é uma proposição simples porém de suma-importância para o desenvolvimento das classes de grafo onde $m > n$.

Prova: Seja d_v o grau do vértice v pertencente ao grafo G , com n vértices e m arestas. É impossível que exista um vértice v tal que $d_v = 1$, pois isso significa que com a retirada do vértice conectado à v , o vértice v estaria desconexo dos demais e o grafo não seria biconexo. Logo, é certo afirmar que $d_v \geq 2; \forall v \in G$. Suponha que todos os vértices possuem grau $d = 2$, ou seja, $d_v = 2; \forall v \in G$. Dessa maneira, o número total de arestas m pode ser facilmente contado da seguinte maneira:

$$m = \frac{nd}{2} = n$$

Observe que é necessário que nenhum vértice tenha grau maior do que 2 para esta fórmula ser válida. Caso houvesse um vértice de grau maior ou igual a 3, o valor de m ultrapassaria o valor de n . Logo, todos os vértices necessariamente precisam ser de grau 2, o que permite apenas uma configuração: a de ciclo. \square

Teorema 3 (grafos densos). *Em grafos biconexos, se $m \geq 2n - 5$ não existe tradeoff entre robustez e eficiência. Ou seja, se $m \geq 2n - 5$ a soma das distâncias mínima alcançável com n vértices e m arestas, levando em conta todos os grafos, sem restrições, é também alcançável por um grafo biconexo.*

Prova: Em (MURTY, 1968) o autor mostra que, para grafos biconexos, se $m \geq 2n - 5$ o diâmetro igual a 2 é alcançável, e o resultado segue a partir do Teorema 1. \square

Alternativamente, a demonstração do teorema acima segue o seguinte roteiro:

- Seja $G_{d=2}(n)$ o conjunto definido por grafos biconexos de diâmetro $d = 2$ e número de vértices $n \geq 5$.
- Seja $G_{min}(n)$ o subconjunto de $G_{d=2}(n)$ tal que o número de arestas m é mínimo.
- Em (MURTY, 1968) é provado que $m = 2n - 5$ para todo grafo $G \in G_{min}(n)$.

Os grafos da classe $G_{min}(n)$ são bem conhecidos e facilmente construídos a partir de dois modos, conforme explicado em (MURTY, 1968), sendo um deles válido apenas para os casos onde $n \geq 7$.

Em resumo: $m = 2n - 5$ arestas são necessárias e suficientes para atingir o diâmetro 2 dentre os grafos biconexos com n vértices. Qualquer aresta adicionada posteriormente, suponhamos que entre dois vértices genéricos $\{u, v\}$, produzirá o mesmo efeito sobre o índice de Wiener: subtraí-lo em 2, pois apenas um caminho, o menor entre u e v , será reduzido, com a única redução possível: de tamanho 2 para 1. É esta propriedade que define a ausência do *tradeoff* entre robustez e eficiência para $m \geq 2n - 5$.

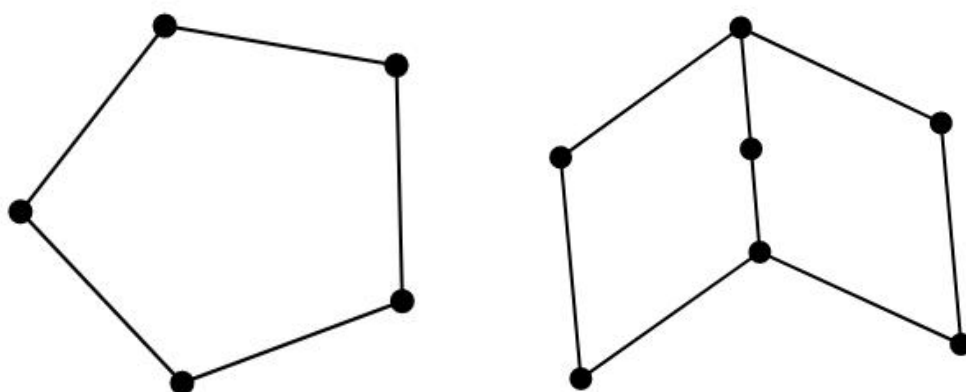


Figura 1 – Dois grafos fundamentais para a construção da classe na qual $m = 2n - 5$. Vértices de grau 2 podem ser inseridos entre quaisquer dois vértices não diretamente conectados, sem abrir mão da propriedade $d = 2$.

Uma particularidade interessante dessa classe $G_{min}(n)$ é a de que, como por definição o diâmetro $d = 2$ é fixo, o *índice de Wiener* depende apenas do número de vértices – o número de arestas é igual a $2n - 5$. Nesse regime, não há necessidade de sacrificar robustez em prol de eficiência. Dado o número de vértices n , com $2n - 5$ arestas consegue-se produzir grafos robustos (biconexos) e eficientes (cuja soma das distâncias é a mínima possível dentre todos os grafos com n vértices e $2n - 5$ arestas).

Teorema 4. *Entre todos os grafos biconexos com $m = 2n - 5$ arestas, existe uma classe de grafos, denotada por $G_{min}(n)$, que possui diâmetro 2. Essa é a classe de grafos que minimiza o índice de Wiener.*

Prova: Já foi demonstrado pelo Teorema 1 que só existe um valor possível para o índice de Wiener em grafos de diâmetro $d = 2$, e pelo Corolário 1 sabemos que, dados m e n , e restringindo-se a grafos biconexos, se o diâmetro for 2 então o índice de Wiener é minimizado. Em (MURTY, 1968) foi demonstrado como construir, para $m = 2n - 5$, grafos com diâmetro 2. Tal classe de grafos é denotada por $G_{min}(n)$. Em (MURTY, 1968) também é demonstrado que $m = 2n - 5$ é o menor número de arestas possível para se alcançar $d = 2$ em grafos biconexos. \square

Temos então o número mínimo de arestas necessárias para se obter eficiência máxima mantendo o grafo biconexo: $m = 2n - 5$. Abaixo de tal cota, ou seja, para grafos esparsos, existe um compromisso (*tradeoff*) entre robustez e eficiência. É exatamente nesse regime ($m < 2n - 5$) que iremos trabalhar daqui em diante.

Teorema 5. *O grafo theta-generalizado balanceado é a solução ótima para $n \leq 17$ e $m < 2n - 5$.*

Prova: Verificamos exaustivamente todos os grafos com até 17 vértices, de acordo com estratégias mostradas no capítulo 5. \square

Teorema 6. *O grafo theta-generalizado balanceado nem sempre é a solução ótima para $m < 2n - 5$.*

Prova: Verificamos numericamente que para $n = 20$ e $m = 25$ existe um grafo theta-generalizado não balanceado que possui soma de distâncias menor que o grafo theta-generalizado balanceado. \square

Combinando as demonstrações dos dois teoremas acima, percebemos que ainda existe uma lacuna no entendimento dos grafos que minimizam as somas das distâncias para $n = 19$ e $n = 18$. Estamos no momento avaliando tais grafos, para identificar se nesses casos o grafo theta-generalizado balanceado é suficiente, ou não, para minimizar as somas das distâncias.

3 O ÍNDICE DE WIENER DE GRAFOS MUITO ESPARSOS

Neste capítulo apresentamos um algoritmo mais performático para o cálculo do índice de Wiener em grafos esparsos, ou seja, considerando o regime $m < 2n + 5$. Nossa expectativa é de que, ao analisar tais casos especiais, abriremos caminho para provar a conjectura apresentada neste trabalho no cenário geral em que $m < 2n - 5$.

O meio mais usual de se calcular o índice de Wiener de um grafo genérico é a partir da matriz de caminhos mínimos, podendo ser obtida à partir do algoritmo de Floyd-Warshall, de complexidade $O(n^3)$.

Desde 1988 existe um esforço para encontrar maneiras de se calcular o índice de Wiener sem a necessidade da matriz de caminho mínimo, que é um método considerado de força-bruta. Químicos utilizam um método para calculá-lo em certas classes de grafos, que representam cadeias de carbono. Tal método se utiliza do fato de que estas cadeias são repetitivas e simétricas. (Paul Manuel, Indra Rajasingh, Bharati Rajan, R. Sundara Rajan, 2013)

Motivados que a restrição de escopo possa nos beneficiar, buscamos um algoritmo para calcular o índice de Wiener na classe de grafos a qual estamos interessados: a de grafos biconexos com $m < 2n - 5$. A principal vantagem de se trabalhar no regime muito esparsos consiste na ocorrência de muitas sequências de vértices de grau 2, o que chamamos de pontes. É possível utilizar essas premissas para sermos mais eficiente no cálculo do índice de Wiener. Em resumo, o algoritmo apresentado tem complexidade $O((n - n_1 - n_2)^3)$, sendo n_1 e n_2 o número de vértices de grau 1 e grau 2, respectivamente.

3.1 O ÍNDICE DE WIENER POR PARTES.

Calcularemos o índice de Wiener por partes, dividindo os vértices em duas categorias: (1) *vértices internos* e (2) *vértices centrais*. Chamamos de *vértices internos* todos os vértices de grau 2 ou menos, e de *vértices centrais* todo o restante, ou seja, de grau 3 ou mais. O índice de Wiener pode ser calculado pelo somatório...

1. ... das distâncias entre os vértices centrais.
2. ... das distâncias entre os vértices internos com os vértices centrais.
3. ... das distâncias entre os vértices internos pertencentes à mesma ponte.
4. ... das distâncias entre os vértices internos pertencentes à pontes diferentes.

3.1.1 Distâncias entre vértices centrais

Esta parcela é a mais simples de ser calculada. Em verdade, aqui valem os métodos usuais de recuperar o caminho mínimo, como o algoritmo de Floyd-Warshall. Uma otimização que faremos será a de que, na tarefa de recuperar a distância entre dois vértices centrais, consideraremos apenas os vértices de grau maior ou igual à 3.

3.1.2 Distâncias entre vértices internos com vértices centrais

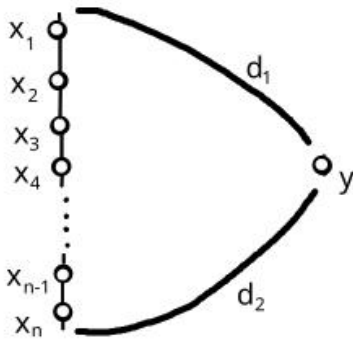


Figura 2 – Distâncias entre vértices internos (x_i) de uma ponte com um vértice central y

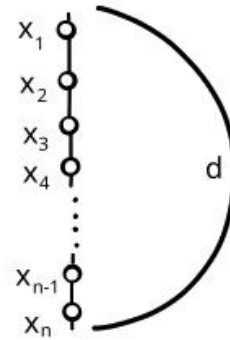


Figura 3 – Distâncias entre vértices internos de uma mesma ponte. d é a menor distância entre os vértices x_1 e x_n que não passa por dentro da ponte.

O somatório $W_{pc}(P, y)$ das distâncias entre o vértice central y e os vértices internos x_i pertencentes à ponte P , como demonstrado na figura 2, é dado por:

$$W_{pc}(P, y) = \sum_{i=1}^n d(x_i, y) \quad (3.1)$$

Sendo $d(x_i, y)$ a distância entre um único vértice interno x_i e y . Note que existem dois caminhos possíveis nesse caso: um que inclui o caminho d_1 e outro que inclui o caminho d_2 . Queremos considerar o menor caminho:

$$d(x_i, y) = \min(i - 1 + d_1, n - i + d_2) \quad (3.2)$$

Ainda que o vértice y possua uma aresta diretamente conectada à x_1 ou x_n ($d_1 = 1$ ou $d_2 = 1$, respectivamente), é possível que o menor caminho entre um vértice interno da mesma ponte e y não passe por tal aresta, mas por outra ponte.

O somatório da equação 3.1 pode ser simplificado em uma fórmula direta. É preciso notar que existe um vértice interno $x_{i'}$ tal que $d(x_{i'}, y) \geq d(x_j, y), \forall j$. i' pode ser encontrado como o valor de i na interseção das retas $i - 1 + d_1$ e $n - 1 + d_2$:

$$i' = \lfloor \frac{n + 1 + d_2 - d_1}{2} \rfloor$$

É preciso que, após aplicar essa fórmula, i' tenha o valor mínimo de 1 ou o valor máximo de n .

Conhecendo o valor de i' , o somatório $W_{pc}(P, y) = \sum_{i=1}^n d_{ic}(x_i, y)$ pode ser dividido em duas partes:

$$W_{pc}(P, y) = \sum_{i=1}^n d(x_i, y) \quad (3.3)$$

$$= \sum_{i=1}^{i'} d(x_i, y) + \sum_{i=i'+1}^n d(x_i, y) \quad (3.4)$$

$$= \sum_{i=1}^{i'} i - 1 + d_1 + \sum_{i=i'+1}^n n - i + d_2 \quad (3.5)$$

$$= (d_1 - 1)i' + \sum_{i=1}^{i'} i + (n + d_2)(n - i') - \sum_{i=i'+1}^n i \quad (3.6)$$

$$= (d_1 - 1)i' + \frac{i'(i' + 1)}{2} + (n + d_2)(n - i') - \frac{(n - i')(n + i' + 1)}{2} \quad (3.7)$$

3.1.3 Distâncias entre vértices internos numa mesma ponte

A figura 3 nos mostra o conjunto de vértices que estamos calculando as distâncias internas. É preciso se atentar que o menor caminho entre dois vértices desse conjunto, por exemplo, x_a e x_b , $b > a$, não necessariamente tem tamanho $(b - a)$. É possível que o caminho menor se passe por fora da ponte, quando $(b - a) \geq \frac{(n-1)+d}{2}$. Neste último caso, o tamanho da distância entre os dois vértices x_a e x_b seria $(n - 1) + d - (b - a)$.

É importante perceber que a distância entre dois vértices desta ponte depende apenas da diferença dos índices ordenados a e b , $b > a$. Seja $k = |b - a|$ a diferença dos índices entre dois vértices internos x_a e x_b . Existem $n - k$ combinações de vértices com índices distantes em k . Seja $d_{ii}(n, k)$ a distância entre dois vértices internos cujos índices distam de k , ambos pertencentes à mesma ponte P com n vértices.

$$d_{ii}(n, k) = \min(k, n - 1 + d - k)$$

O somatório $W_p(P)$ de todas as distâncias entre os vértices da ponte P com n vértices internos, como representado na figura 3, pode ser calculado de acordo com a equação:

$$W_p(P) = \sum_{k=1}^{n-1} (n-k)d_{ii}(n, k) \quad (3.8)$$

Semelhantemente ao caso anterior, é importante notar que existe um valor de k' que maximiza o valor de $d_{ii}(n, k)$, e este é o valor para k presente na interseção das retas k e $n + d - 1 - k$.

$$k' = \lfloor \frac{n-1+d}{2} \rfloor \quad (3.9)$$

Considere então a seguinte partição do somatório apresentado:

$$W_p(P) = \sum_{k=1}^{k'} (n-k)d_{ii}(n, k) + \sum_{k=k'+1}^{n-1} (n-k)d_{ii}(n, k) \quad (3.10)$$

$$= \sum_{k=1}^{k'} (n-k)k + \sum_{k=k'+1}^{n-1} (n-k)(n-1+d-k) \quad (3.11)$$

$$= \sum_{k=1}^{k'} (nk - k^2) + \sum_{k=k'+1}^{n-1} (n^2 - n + dn - 2kn + k - dk + k^2) \quad (3.12)$$

$$= \sum_{k=1}^{k'} nk + \sum_{k=k'+1}^{n-1} (n^2 - n + dn) + \sum_{k=k'+1}^{n-1} (1-d-2n)k - \sum_{k=1}^{k'} k^2 + \sum_{k=k'+1}^{n-1} k^2 \quad (3.13)$$

Então

$$W_p(P) = \quad (3.14)$$

$$= \frac{nk'(k'+1)}{2} + (n^2 - n + dn)(n - k' - 1) + \frac{(1-d-2n)(n+k')(n-k'-1)}{2} - \quad (3.15)$$

$$- \sum_{k=1}^{k'} k^2 + \sum_{k=k'+1}^{n-1} k^2$$

Com a equação acima, podemos calcular $W_p(P)$ em tempo constante. As parcelas com somatório de k^2 podem ser substituídas por uma fórmula conhecida:

$$\sum_{i=a}^b i^2 = \frac{b(b+1)(2b+1)}{6} - \frac{a(a-1)(2a-1)}{6}$$

3.1.4 Distâncias entre vértices internos em pontes diferentes

Nos resta apenas encontrar uma maneira mais performática para o cálculo do somatório $W_{pp}(P_a, P_b)$ das distâncias entre os vértices internos de uma ponte P_a com os de

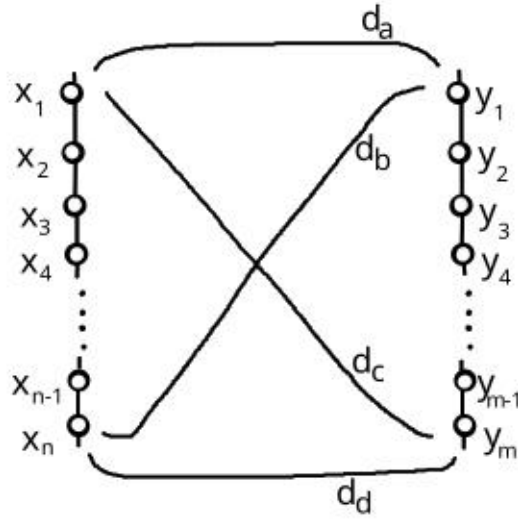


Figura 4 – Distâncias entre os vértices internos x_i de uma ponte P_a com os vértices internos y_i de uma ponte P_b

uma ponte P_b . Nestes casos, como demonstrado na figura 4, iremos aplicar a fórmula 3.7 sucessivamente, considerando que a ponte em questão é $P_a = \{x_1, x_2, \dots, x_n\}$ e, y_i , como se fosse o vértice central, para todo $y_i \in \{y_1, y_2, \dots, y_m\}$.

$$W_{pp}(P_a, P_b) = \sum_{i=1}^m W_{pc}(P_a, y_i) \quad (3.16)$$

Atente-se para o detalhe de que, nesta "transposição" de fórmula, tanto d_1 quanto d_2 podem assumir dois valores diferentes, a depender de d_a, d_b, d_c e d_d :

$$d_1 = \min(d_a + j, d_c + m - 1 - j)$$

$$d_2 = \min(d_d + m - 1 - j, d_b + j)$$

Desta maneira, executaremos uma fórmula fechada em m iterações, onde m não deve ultrapassar a quantidade de vértices do grafo. Todas as fórmulas apresentadas até aqui são necessárias e suficientes para o cálculo do índice de Wiener dos grafos que serão apresentados.

3.2 ALGORITMOS

Nesta seção serão apresentados algoritmos mais para o cálculo mais performático dos menores caminhos e do índice de Wiener em grafos esparsos. É conveniente que a estrutura de dados utilizada para o grafo G seja uma lista/vetor dinâmico de adjacências, de modo

que seja possível retornar o tamanho de cada lista/vetor(ou seja, o grau de determinado vértice) em tempo constante. No algoritmo 3 utilizaremos as fórmulas apresentadas neste capítulo, isto é, W_{pp}, W_p e W_{pc} .

Com as estruturas criadas no algoritmo 1, podemos calcular não apenas a distância entre dois vértices em tempo constante pelo algoritmo 2, mas também o índice de Wiener de G rapidamente com o algoritmo 3. Será necessário a matriz de caminhos mínimos de G_{aux} , grafo auxiliar com os vértices de grau maior ou igual à 3 de G , gerado pelo algoritmo 1. A matriz pode ser calculada seguindo o algoritmo de Floyd-Warshall.

3.3 COMPARATIVO DE PERFORMANCE

Apresentaremos alguns resultados que comparam a performance do cálculo do índice de Wiener pelo usual algoritmo de Floyd-Warshall contra o algoritmo 1 em conjunto com o algoritmo 3. Os experimentos foram executados sequencialmente, ou seja, sem uso de processamento paralelo, todos em um processador *Intel Core i5-3330*.

Como era de se esperar, o tempo de execução nos theta-grafos (que é o nosso foco nesse trabalho) se mostrou quase constante independentemente do número de vértices, mas apenas do número de pontes. As figuras 5 e 6 mostram o comparativo do tempo necessário para a execução dos dois algoritmos em theta-grafos.

Os gráficos apresentados nas figuras 7 e 8 mostram a performance dos algoritmos sendo executados em grafos aleatórios. O modo de geração de cada grafo seguiu um algoritmo simples: inicializamos um grafo com um dado número de vértices e, enquanto este grafo não for conexo, criamos uma aresta entre dois vértices arbitrários. Isto significa que nenhum dos grafos gerados dessa maneira é biconexo, e a razão do número de arestas com o número de vértices se mantém aproximadamente 3.

O histograma da figura 8 mostra o tempo de execução em 300 grafos diferentes, todos com 200 vértices. Nele, fica explícito que, embora o algoritmo que apresentamos possa ser mais útil em determinadas classes de grafos, em alguns casos pode haver um tempo extra necessário, muito provavelmente em grafos com um maior número de arestas e poucos vértices de grau 1 ou 2. Ainda assim podemos esperar um ganho médio modesto.

Para grafos esparsos(regime $m < 2n - 5$), nosso interesse neste trabalho, o tempo praticamente constante neste cálculo é crítico para podermos, no capítulo 5, responder a pergunta: "Entre todas as topologias biconexas possíveis com um dado número de vértices e arestas, qual possui o menor índice de Wiener?".

Algoritmo 1: Algoritmo de criação para estrutura de G esparso.

Data: Grafo G na forma de lista de adjacências.

Result: Grafo auxiliar G_{aux} , mapa de pontes V_{mapa_pontes} e vetor de pontes

```

1   $V_{pontes}$ 
2  struct {
3  |   Vetor internos;
4  |   Vetor extremidades;
5  } Ponte;
6  Inicializar um grafo  $G_{aux}$ , com peso nas arestas, inicialmente vazio;
7  Inicializar um vetor de inteiros,  $V_{mapa\_pontes}$ , com  $|V(G)|$  espaços;
8  Inicializar um vetor de Ponte,  $V_{pontes}$ , inicialmente vazio;
9  Inicializar uma lista  $L_{descobertos}$ , inicialmente vazia;
10 for vértice  $v_i$  in  $V(G)$  do
11 |   if  $v_i \in L_{descobertos}$  then
12 |   |   continue
13 |   end
14 |    $V_{mapa\_pontes}[i] = -1$ ;
15 |   if  $grau(v_i) \geq 3$  then
16 |   |    $v_i$  passa a pertencer também à  $G_{aux}$ ;
17 |   else
18 |   |   Inicializar uma Ponte,  $P$ ;
19 |   |   Inserir  $P$  ao final de  $V_{pontes}$ ;
20 |   |    $p_i$  recebe o índice de  $P$  em  $V_{pontes}$ ;
21 |   |   Inicializar uma fila  $F_{descobrimento}$ ;
22 |   |   Inserir  $v_i$  à  $F_{descobrimento}$ ;
23 |   |   while  $F_{descobrimento}$  não está vazia do
24 |   |   |   vértice  $v_j$  é o próximo vértice da fila  $F_{descobrimento}$ ;
25 |   |   |    $F_{descobrimento}$  retira  $v_j$ ;
26 |   |   |   if  $grau(v_j) \geq 3$  then
27 |   |   |   |   Inserir  $v_j$  em  $P.extremidades$ ;
28 |   |   |   |    $v_j$  passa a pertencer também à  $G_{aux}$ ;
29 |   |   |   else
30 |   |   |   |    $V_{mapa\_pontes}[i] \leftarrow p_i$ ;
31 |   |   |   |   Inserir  $v_j$  em  $P.internos$ ;
32 |   |   |   |   for vértice  $v_k$  adjacente à  $v_j$  do
33 |   |   |   |   |   Inserir  $v_k$  à  $F_{descobrimento}$ ;
34 |   |   |   |   end
35 |   |   |   end
36 |   |   |   Adicionar  $v_j$  à  $L_{descobertos}$ ;
37 |   |   end
38 |   |   Criar uma aresta em  $G_{aux}$  entre os vértices  $P.extremidades[0]$  e
39 |   |    $P.extremidades[1]$ , com peso igual à  $tamanho(P.internos) + 1$ ;
40 |   end
41 |   Adicionar  $v_i$  à  $L_{descobertos}$ ;
42 |   Retorna  $G_{aux}$ ,  $V_{mapa\_pontes}$  e  $V_{pontes}$ .
43 end

```

Algoritmo 2: Algoritmo para recuperar distância entre dois vértices de G em tempo constante à partir das estruturas criadas pelo algoritmo 1

Data: Grafo G_{aux} , V_{mapa_pontes} e V_{pontes} criados pelo algoritmo 1 aplicado ao grafo G , matriz de caminhos mínimos aplicado à G_{aux} , $M_{G_{aux}}$, e os dois vértices, v_i e v_j , com a distância a ser calculada.

Result: Tamanho do caminho mínimo entre v_i e v_j .

```

1 if grau( $v_i, G$ )  $\geq$  3 & grau( $v_j, G$ )  $\geq$  3 then
2   | Retorna  $M_{G_{aux}}[v_i][v_j]$ ;
3 else if grau( $v_i, G$ )  $\leq$  2 & grau( $v_j, G$ )  $\geq$  3 then
4   |  $P_i\_index \leftarrow V_{mapa\_pontes}[v_i]$ ;
5   |  $P_i \leftarrow V_{pontes}[P_i\_index]$ ;
6   |  $P_i\_size \leftarrow tamanho(P_i.internos)$ ;
7   |  $d_0 \leftarrow M_{G_{aux}}[P_i.extremidades[0]][v_j]$ ;
8   |  $d_1 \leftarrow M_{G_{aux}}[P_i.extremidades[1]][v_j]$ ;
9   |  $v_i\_index \leftarrow index(v_i, P_i.internos)$ ;
10  | Retorna  $min(d_0 + v_i\_index, d_1 + P_i\_size - v_i\_index)$ ;
11 else
12  |  $P_i\_index \leftarrow V_{mapa\_pontes}[v_i]$ ;
13  |  $P_j\_index \leftarrow V_{mapa\_pontes}[v_j]$ ;
14  | if  $P_i\_index == P_j\_index$  then
15  |   |  $P \leftarrow V_{pontes}[P_i\_index]$ ;
16  |   |  $v_i\_index \leftarrow index(v_i, P.internos)$ ;
17  |   |  $v_j\_index \leftarrow index(v_j, P.internos)$ ;
18  |   |  $delta\_index \leftarrow abs(v_i\_index - v_j\_index)$ ;
19  |   |  $P\_size \leftarrow tamanho(P.internos)$ ;
20  |   |  $d \leftarrow M_{G_{aux}}[P.extremidades[0]][P.extremidades[1]]$ ;
21  |   | Retorna  $min(delta\_index, d + P\_size - delta\_index + 1)$ ;
22  | else
23  |   |  $P_i \leftarrow V_{pontes}[P_i\_index]$ ;
24  |   |  $P_j \leftarrow V_{pontes}[P_j\_index]$ ;
25  |   |  $v_i\_index \leftarrow index(v_i, P_i.internos)$ ;
26  |   |  $v_j\_index \leftarrow index(v_j, P_j.internos)$ ;
27  |   |  $delta\_index \leftarrow abs(v_i\_index - v_j\_index)$ ;
28  |   |  $P_i\_size \leftarrow tamanho(P_i.internos)$ ;
29  |   |  $P_j\_size \leftarrow tamanho(P_j.internos)$ ;
30  |   |  $d_{00} \leftarrow M_{G_{aux}}[P_i.extremidades[0]][P_j.extremidades[0]]$ ;
31  |   |  $d_{01} \leftarrow M_{G_{aux}}[P_i.extremidades[0]][P_j.extremidades[1]]$ ;
32  |   |  $d_{10} \leftarrow M_{G_{aux}}[P_i.extremidades[1]][P_j.extremidades[0]]$ ;
33  |   |  $d_{11} \leftarrow M_{G_{aux}}[P_i.extremidades[1]][P_j.extremidades[1]]$ ;
34  |   | Retorna  $min($ 
35  |     |  $d_{00} + v_i\_index + v_j\_index,$ 
36  |     |  $d_{01} + v_i\_index + P_j\_size - v_j\_index,$ 
37  |     |  $d_{10} + P_i\_size - v_i\_index + v_j\_index,$ 
38  |     |  $d_{11} + P_i\_size - v_i\_index + P_j\_size - v_j\_index)$  ;

```

Algoritmo 3: Algoritmo para recuperar índice de Wiener de G à partir das estruturas criadas pelo algoritmo 1

Data: Grafo G_{aux} , V_{mapa_pontes} e V_{pontes} criados pelo algoritmo 1 aplicado ao grafo G , matriz de caminhos mínimos aplicado à G_{aux} , $M_{G_{aux}}$.

Result: Índice de Wiener do grafo G .

```

1  $W \leftarrow 0$ ;
2 for  $v_i \in V(G_{aux})$  do
3   for  $v_j \in V(G_{aux})$  do
4      $W \leftarrow W + M_{G_{aux}}[v_i][v_j]$ ;
5   end
6   for  $P \in V_{pontes}$  do
7      $W \leftarrow W + W_{pc}(P, v_i)$ ;
8   end
9 end
10 for  $P_i \in V_{pontes}$  do
11    $W \leftarrow W + W_p(P_i)$ ;
12   for  $P_j \in V_{pontes}$  do
13     if  $P_i \neq P_j$  then
14        $W \leftarrow W + W_{pp}(P_i, P_j)$ ;
15     end
16   end
17 end
18 Retorna  $W$ ;

```

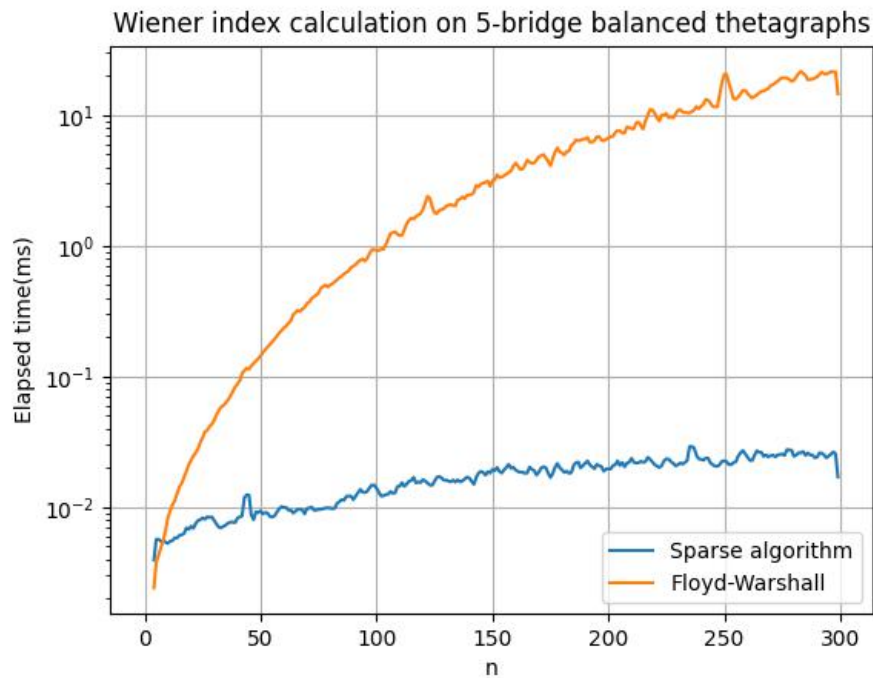


Figura 5 – Tempo de execução em função do número de vértices em theta-grafos de 5 pontes.

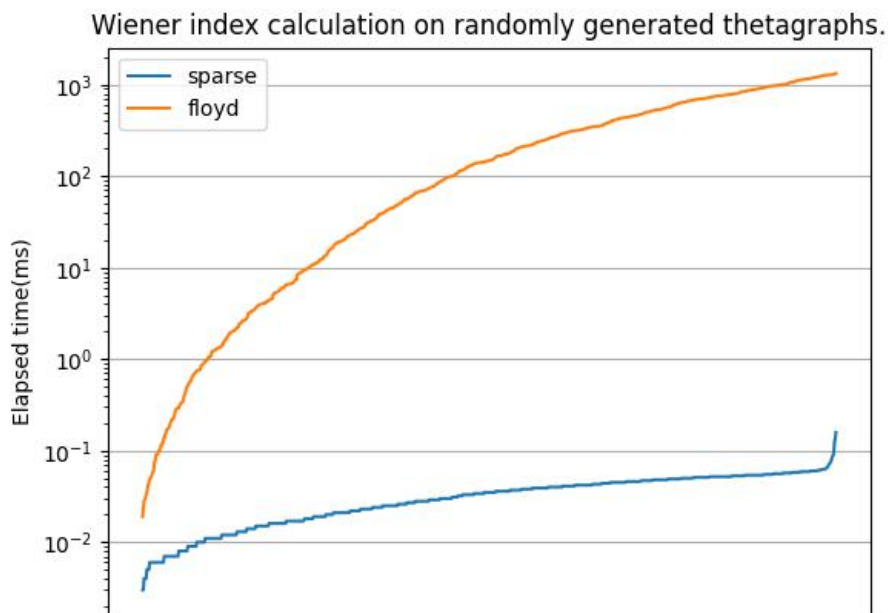


Figura 6 – Tempo de execução em theta-grafos gerados aleatoriamente (número de vértices distribuído uniformemente entre 20 e 1000)

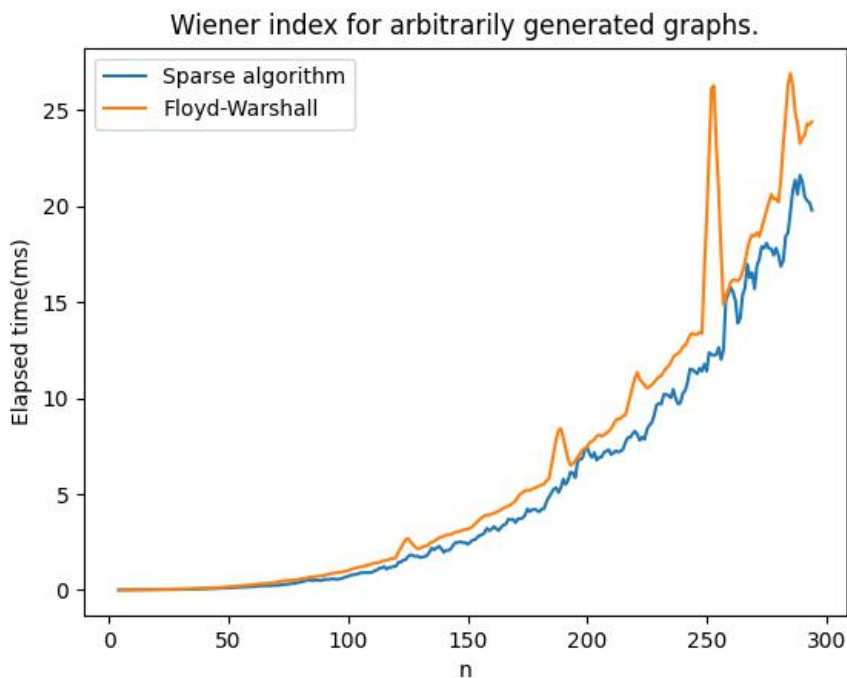


Figura 7 – Histograma do tempo de execução em função do número de vértices em grafos arbitrariamente gerados.

Wiener index calculation on arbitrarily generated graphs with 200 vertices

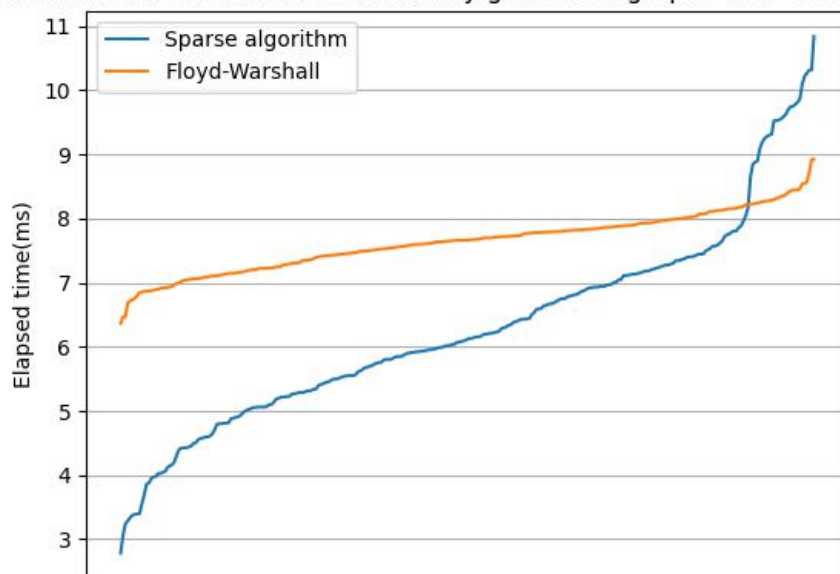


Figura 8 – Histograma do tempo de execução em diferentes grafos arbitrários com 200 vértices.

4 EXPLORAÇÃO DE GRAFOS E O PROBLEMA DO PARTICIONAMENTO

4.1 MOTIVAÇÃO

Nosso objetivo principal com esse trabalho é reforçar a proposta conjectura 1 para um número maior de grafos. Para tanto, o capítulo 5 descreve experimentos que exploram o maior número de grafos possíveis(até onde a complexidade permitir). Um dos desafios nessa tarefa é o de calcular o índice de Wiener para **todos** os grafos com um dado número de vértices(n) e arestas(m).

Uma abordagem ingênua seria iterar sobre todas as combinações de m pares escolhidos entre todos possíveis($\binom{n}{2}$). Essa solução, embora seja correta, não faz proveito do fato de que não queremos iterar sobre dois grafos isomorfos ou grafos que não sejam biconexos, e é, por causa disso, inviável pôr em prática devido ao crescimento combinatorial do número de pares que enfrentaríamos.

Procuramos implementar uma abordagem mais analítica, que nos permite recuperar todos os grafos biconexos com $n \leq 100$ e $m \leq n + 2$ em tempo hábil. O número de grafos ainda cresce de forma combinatorial, mas pudemos reforçar a conjectura proposta em mais alguns casos.

4.2 O PROBLEMA DO PARTICIONAMENTO

Sendo m o número de arestas de um grafo genérico G , d o seu diâmetro e n_i o número de vértices de grau i , podemos afirmar que:

$$n = n_1 + n_2 + n_3 + \dots + n_d \quad (4.1)$$

$$m = \frac{n_1 + 2n_2 + 3n_3 + \dots + dn_d}{2} \quad (4.2)$$

Na classe dos biconexos, vale ressaltar que $n_1 = 0$, logo, nesse caso:

$$m = n_2 + \frac{3n_3}{2} + \frac{4n_4}{2} + \dots + \frac{dn_d}{2} \quad (4.3)$$

Note que ao abordar grafos biconexos com n vértices e $m = n + k$ arestas, no regime onde $k \ll n$ temos a maioria dos vértices sendo de grau 2. Quando $k = 1$, por exemplo, é impossível que $n_i \geq 1$ para qualquer $n_i \in \{n_5, n_5, \dots, n_d\}$. Na verdade, para valores pequenos de k , é fácil contabilizarmos todas as soluções possíveis na equação apresentada. No caso $k = 1$, é fácil ver que as únicas soluções possíveis são $(n_2, n_3, n_4) = (n - 2, 2, 0)$ e $(n_2, n_3, n_4) = (n - 1, 0, 1)$.

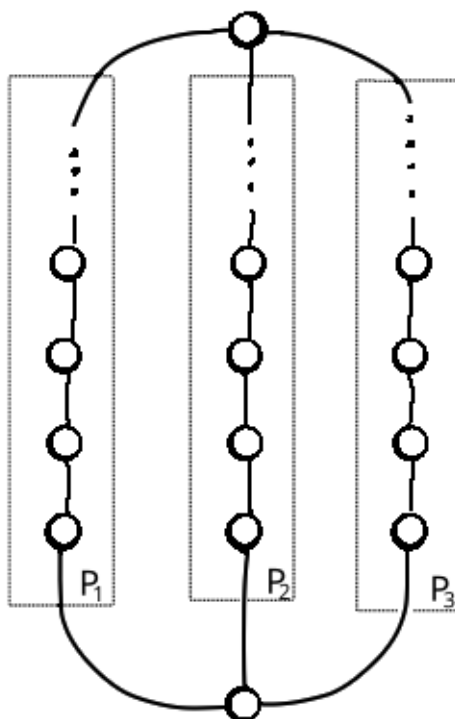


Figura 9 – Classe dos grafos biconexos com $m=n+1$ arestas.

Isto significa que podemos tentar exaurir todas as topologias possíveis para uma dada solução da equação apresentada. A partir de uma dada topologia, como, por exemplo, o theta-grafo de 3 pontes, é do nosso interesse calcular o índice de Wiener com todas as combinações possíveis de tamanhos de pontes, que é um problema conhecido de particionamento.

Tomando como exemplo os theta-grafos de 3 pontes, queremos analisar todos os valores possíveis para o número de vértices presentes nas pontes P_1, P_2, P_3 , com algumas ressalvas: não é válido duas ou mais pontes com 0 vértices internos, pois isso configuraria duas arestas entre dois vértices. Outras regras a serem respeitadas: $|P_1| + |P_2| + |P_3| = n - 2$ (número total de vértices internos) e $|P_1| \leq |P_2| \leq |P_3|$. As últimas inequações são necessárias para não contabilizarmos grafos isomorfos múltiplas vezes. Ou seja, não consideramos um grafo onde $(|P_1|, |P_2|, |P_3|) = (2, 1, 5)$, pois este mesmo grafo já é contemplado pela configuração $(1, 2, 5)$.

Embora comumente seja interessante buscar todas as partições de um certo número, iremos procurar apenas soluções com um certo número de partições como, por exemplo, 3, no caso $m = n + 1$.

Definiremos a função $f(n)$ como o número de maneiras de somar n com 1 ou mais parcelas não negativas e $f(n, m)$ como sendo o número de maneiras de somar n com m parcelas não negativas. Por exemplo, $f(6, 3) = 7$, pois são 7 maneiras de somar 6 com 3 parcelas:

- $6 = 6 + 0 + 0$
- $6 = 5 + 1 + 0$
- $6 = 4 + 2 + 0$
- $6 = 4 + 1 + 1$
- $6 = 3 + 3 + 0$
- $6 = 3 + 2 + 1$
- $6 = 2 + 2 + 2$

Note que $f(6, 3) - 1 = 6$ representa o número de configurações possíveis para os grafos-tetas de 8 vértices (6 vértices internos). É preciso desconsiderar o caso "6 + 0 + 0" pois não é possível ter duas pontes com 0 vértices internos.

Apresentaremos uma maneira de calcular $f(n, m)$ recursivamente (via programação dinâmica), mas também, mais importante ainda, uma maneira de percorrer todas as soluções representadas na contagem.

Teorema 7. *O número de soluções para a equação em x_i :*

$$N = x_1 + x_2 + x_3 + \dots + x_{m-1} + x_m \mid x_1 \geq x_2 \geq x_3 \geq \dots \geq x_m \geq 0, x_k \in \mathbb{N}$$

é o mesmo número de soluções para a equação em y_i :

$$N = y_1 + 2y_2 + 3y_3 + \dots + my_m \mid y_k \in \mathbb{N}$$

Prova: Como uma forma intuitiva de visualizar a primeira equação, atente-se à figura 10, que representa uma das soluções contabilizadas por $f(16, 6)$, $(x_1, x_2, x_3, x_4, x_5, x_6) = (5, 3, 3, 3, 1, 1)$, sendo cada coluna uma das variáveis. Essa configuração pode também ser representada como uma solução para a segunda equação, onde $(y_1, y_2, y_3, y_4, y_5, y_6) = (2, 0, 0, 2, 0, 1)$. A mesma configuração visual pode ser utilizada, como demonstrado na figura 11. Linhas de mesmo tamanho k são agrupadas, formando retângulos de altura y_k , e o número total de quadrados é contabilizado pelo somatório das áreas dos agrupamentos ($k \cdot y_k$). \square

Teorema 8. *Seja $f(N, m)$ o número de soluções para a equação:*

$$N = y_1 + 2y_2 + 3y_3 + \dots + my_m \mid y_k \in \mathbb{N}, \forall k \in [1, m]$$

Então

$$f(N, m) = f(N, m-1) + f(N-m, m-1) + f(N-2m, m-1) + \dots + f(N \bmod m, m-1)$$

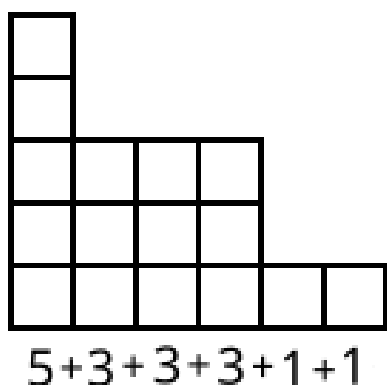


Figura 10 – Visualização de solução com somatório das colunas

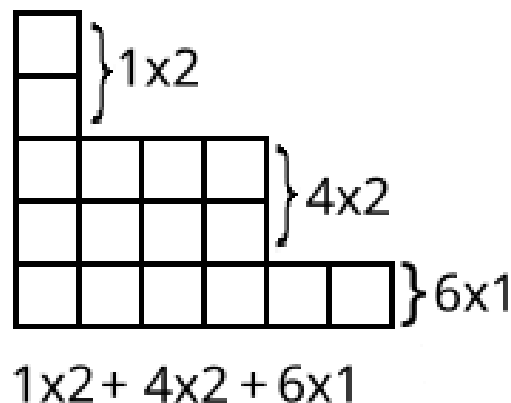


Figura 11 – Visualização de solução com somatório das áreas, agrupadas por linhas iguais

Prova: Como todas as parcelas ky_k são positivas para todo k , é certo afirmar que:

$$N \geq my_m$$

Podemos então numerar todos os valores possíveis para y_m :

$$0 \leq y_m \leq \lfloor \frac{N}{m} \rfloor$$

Se fixarmos um valor para y_m , é certo que isso nos restringiria à $f(N - my_m, m - 1)$ soluções para a equação apresentada. Logo, o somatório $\sum_{i=0}^{\lfloor \frac{N}{m} \rfloor} f(N - mi, m - 1)$, que considera todos os valores possíveis para y_m , representa, assim como $f(N, m)$, o total de soluções para a mesma equação. \square

O caso trivial da função recursiva é $f(N, 1) = 1$, pois obviamente só existe uma maneira de somar qualquer número com apenas uma parcela.

Existe uma vasta literatura sobre como recuperar $f(N)$, com uma nota especial para a recorrência de MacMahon (NOTES... , 2013), que nos permite recuperar este número em $2\sqrt{\frac{2N}{3}}$ passos. Entretanto, não seria possível, como é a fórmula apresentada no teorema 8, "visitarmos" cada uma destas soluções, sendo possível apenas contabilizarmos.

5 RESULTADOS NUMÉRICOS

À fim de reforçar a conjectura 1, realizamos alguns experimentos de força-bruta em grafos relevantes (biconexos). Neste capítulo mostramos duas abordagens que realizamos, a primeira com **geng** (Brendan McKay, Adolfo Piperno, 2020), uma ferramenta já existente, e a segunda com uma proposta menos genérica porém mais eficiente, fazendo uso dos métodos apresentados nos capítulos 3 e 4.

5.1 ANALISANDO GRAFOS COM GENG

geng é um programa para gerar grafos de classes definidas até a sua exaustão.

Com essa metodologia foi possível analisar todos os grafos biconexos com até **17** vértices no regime com $m < 2n - 5$ arestas. Nossa Conjectura 1, de que o theta-grafo é a(uma das) configuração(ões) com menor índice de Wiener, se provou verdadeira nesses casos. Nestes casos, os theta-grafos ótimos são todos balanceados.

Alguns valores encontrados são mostrados na tabela a seguir:

```

rilpires $> geng -C 11 12
>A ./geng -Cd2D4 n=11 e=12
J??ED@OJ?w?
J??ED?WI@W?
J??E@_KiAg?
J?AAD@OM?[?
J?AAD?oU?[?
J?AAD?W[?[?
J?AA@_gs?[?
J?AEB?oECW?
J?AEB?oEAW?
J?AEB?oE@W?
J?AEB?oE?w?
>Z 11 graphs generated in 0.00 sec

```

Figura 12 – Exemplo de saída do programa **geng**: grafos biconexos(-C) de 11 vértices e 12 arestas. O formato apresentado é uma codificação da forma matricial comprimida, graph6. (Brendan McKay, 2015)

n	m	Índice de Wiener ótimo
10	11	204
11	12	272
12	13	348
13	14	440
14	15	550
15	16	670
16	17	810
16	18	688
16	19	612
16	26	432

O tempo de execução pra um dado grupo cresce em função não só do número de vértices mas também do número de arestas. Devido à natureza combinatorial do problema, podemos perceber que mesmo para um valor de n tão pequeno como 18 pode ter o tempo de exploração em ordens de grandeza de segundos ($m = n + 1$) até dias ($m = 2n - 6$).

Ainda que tivéssemos todos os grafos definidos em uma lista, restaria calcular o índice de Wiener em cada um dos grafos para compará-los. Mas como o número de vértices não passou de 25, o tempo de execução do cálculo não chegou a ser relevante. A maior parte do tempo do experimento foi despendida com a exploração realizada pelo *geng*. Daí a motivação de uma exploração mais inteligente, fazendo uma análise mais minuciosa e analítica das topologias possíveis nestas classes.

5.2 ANALISANDO GRAFOS COM O ALGORITMO ESPARSO.

5.2.1 Topologias $m = n + 1$

A classe dos biconexos $m = n + 1$ é a primeira a qual é possível se obter um theta-grafo. A configuração de theta-grafo também é, como veremos a seguir, a única configuração possível para esta classe.

Vimos anteriormente que existem duas soluções para a equação 4.3 onde não existem vértices de grau 1:

1. $n_3 = 0, n_4 = 1$
2. $n_3 = 2$

Conseguimos exaurir todas as configurações de pontes entre estes vértices centrais, como demonstrado na figura 13. As opções biconexas nunca terão nenhum vértice central conectado simultaneamente com as duas extremidades de uma ponte, ou, caso contrário,

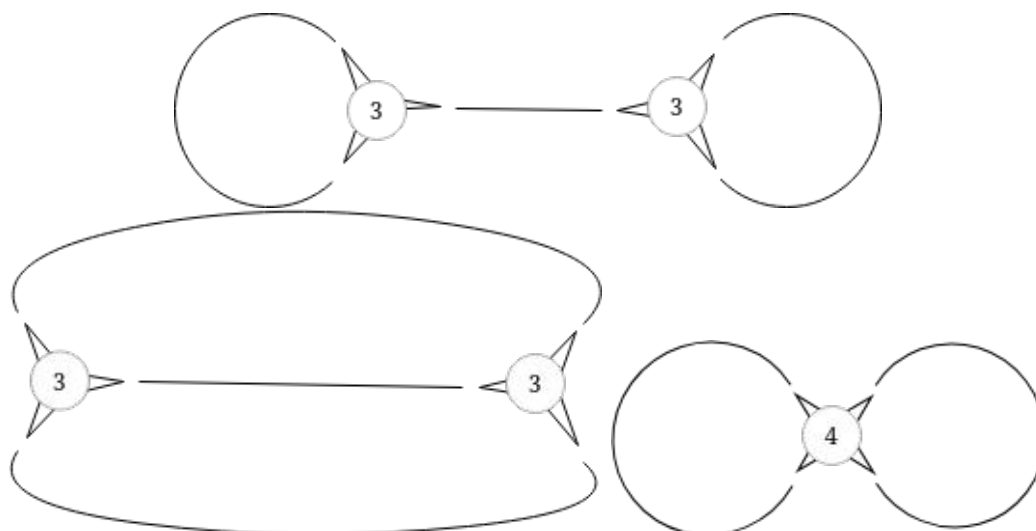


Figura 13 – Topologias possíveis com $m = n + 1$ e sem vértices de grau 1. Note que nem todas são biconexas: grafos biconexos não podem ter vértices centrais em ambas as extremidades de uma ponte. A topologia de theta-grafo é a única topologia biconexa.

a ponte estaria desconexa dos demais vértices quando o vértice central fosse retirado e, portanto, o grafo não seria biconexo. Com isso notamos que apenas a topologia de theta-grafo é biconexa.

Já vimos na figura 9 um modo de generalizar os grafos dessa classe: um simples particionamento onde $P_1 + P_2 + P_3 = n - 2$, não podendo duas ou mais pontes possuírem 0 vértices internos simultaneamente.

Com o algoritmo esparso, calculamos o índice de Wiener de qualquer configuração de (P_1, P_2, P_3) em tempo $O(P_2)$. Dessa maneira, fomos capazes de percorrer **todos os grafos biconexos com $m = n + 1$ arestas** e até 1000 vértices ($n = 1000$). O gráfico da figura 14 mostra o crescimento do índice de Wiener em função de n .

Foi interessante notar que o theta-grafo balanceado foi sempre a configuração com o menor índice de Wiener, empatado, quando possível, com uma configuração em "escada", ($P_3 = P_2 + 1 = P_1 + 2$).

5.2.2 Topologias $m = n + 2$

De acordo com a equação 4.3, para $m = n + 2$, contabilizamos todas as distribuições em vértices centrais possíveis:

1. $n_3 = 0, n_4 = 0, n_5 = 0, n_6 = 1$
2. $n_3 = 1, n_4 = 0, n_5 = 1$
3. $n_3 = 0, n_4 = 2$

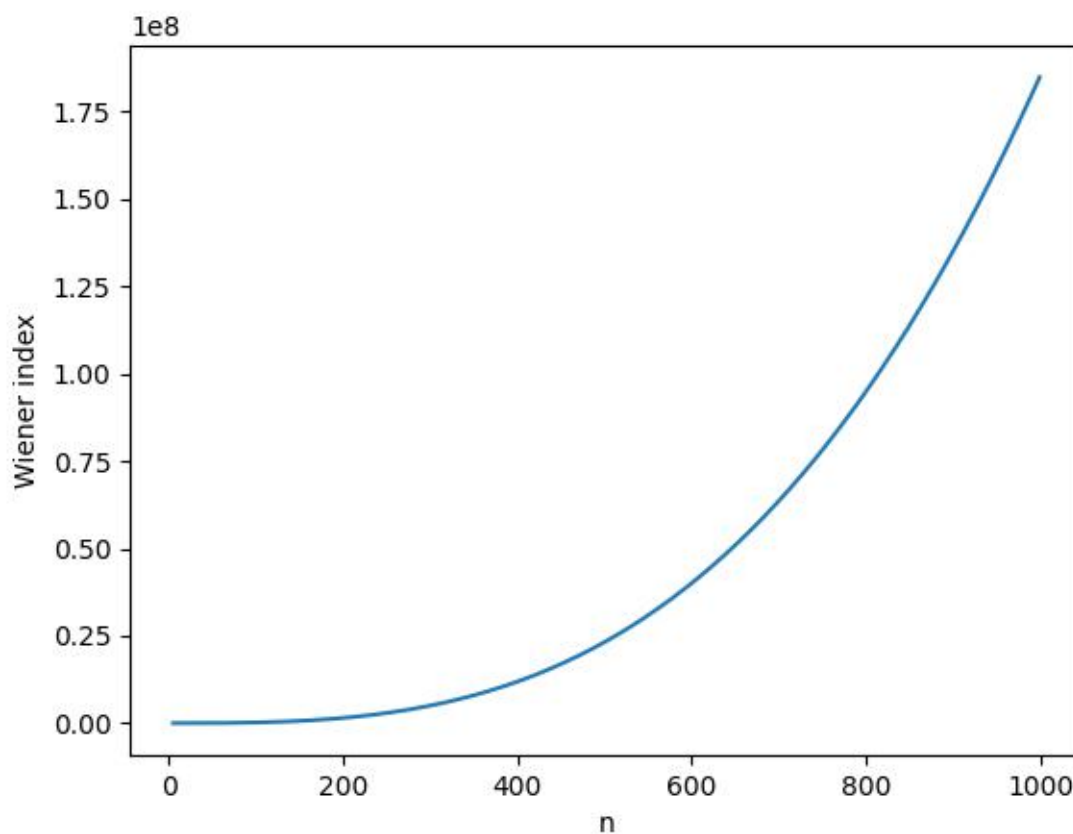


Figura 14 – Crescimento do índice de Wiener ótimo para theta-grafos com 3 pontes ($m = n + 1$) em função do número de vértices, n .

4. $n_3 = 2, n_4 = 1$

5. $n_3 = 4$

De um modo semelhante ao caso $m = n + 1$, isto é, descartando as topologias onde um vértice central possui uma ponte conectada à si mesmo, aqui podemos verificar que existem apenas 4 topologias biconexas, e o theta-grafo de 4 pontes é uma delas, como demonstrado na figura 16. Para um dado valor de n , queremos comparar o índice de Wiener entre todos os grafos possíveis que são derivados de uma das 4 topologias.

1. A derivação dos theta-grafos é semelhante ao caso anterior: verificar todas as partições (P_1, P_2, P_3, P_4) que somam $n - 2$.
2. A derivação dos grafos "quatro33" é feita da seguinte maneira: Para toda solução $X + Y = n - 3$... o valor de Y representa o número de vértices internos na ponte que conecta os dois vértices de grau 3. O valor de X representa o total de vértices internos entre as demais pontes. Particionar X em X_1 e X_2 , cujos valores representam o total de vértices internos dentro de cada dupla de pontes que conectam os vértices

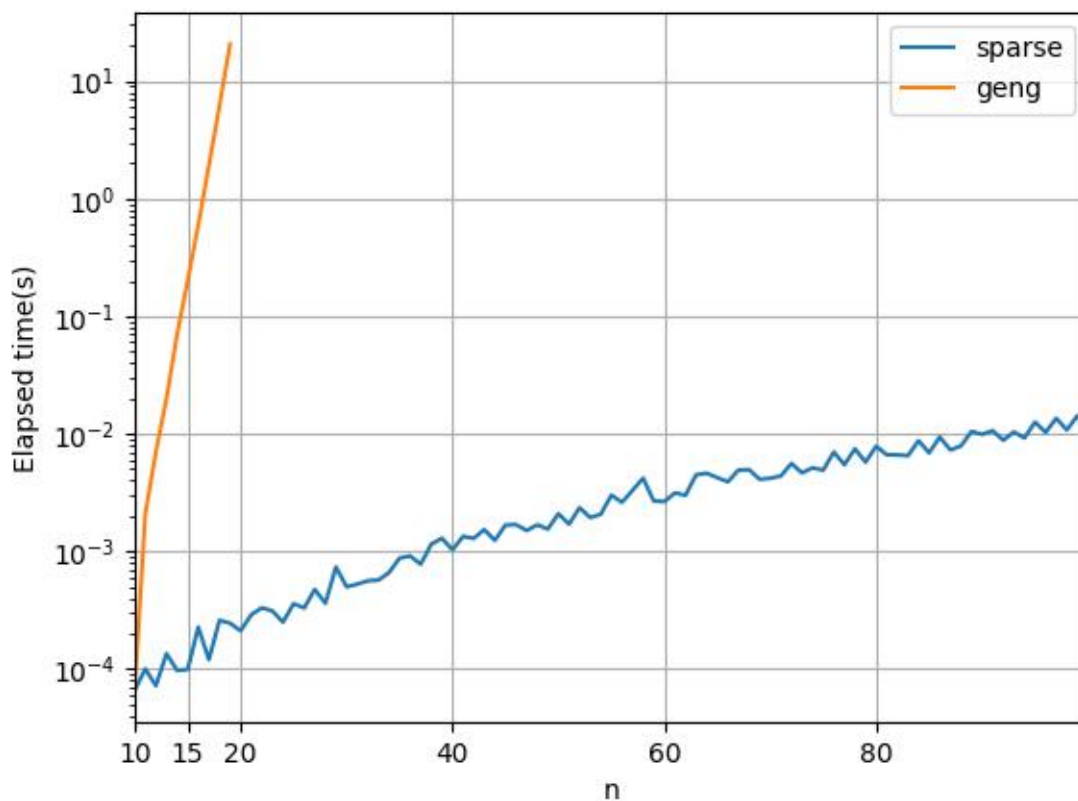


Figura 15 – Comparação do tempo despendido no processo de exploração e cálculo do índice de Wiener na classe $m = n + 1$, entre os dois métodos empregados.

de grau 3 e 4. Para cada valor de X_1 e X_2 , particionar em ainda mais duas subpartições: X_{11} e X_{12} , que somam X_1 e representam o total de vértices internos em cada uma das duas pontes de X_1 . Da mesma maneira, X_2 deve ser sub-particionado em X_{21} e X_{22} .

3. A derivação dos grafos *cilindros* é feita semelhantemente ao caso anterior, com a diferença de que $X + Y = n - 4$ e que Y deve ser sub-particionado em Y_1 e Y_2 , representando o total de vértices internos em cada uma das pontes que conectam sozinhas os vértices de grau 3.
4. A derivação dos grafos *tetraedros* é ligeiramente mais complexa e descrita da seguinte maneira:
 - a) Sub-particionaremos o total de vértices internos, $n - 4$, em dois valores: X e Y .
 - b) Cada vértice central c_i , possui 3 pontes conectadas à si, com um total de vértices internos igual a $T(c_i)$. O valor de Y deve representar $\max_{c_i} T(c_i)$.

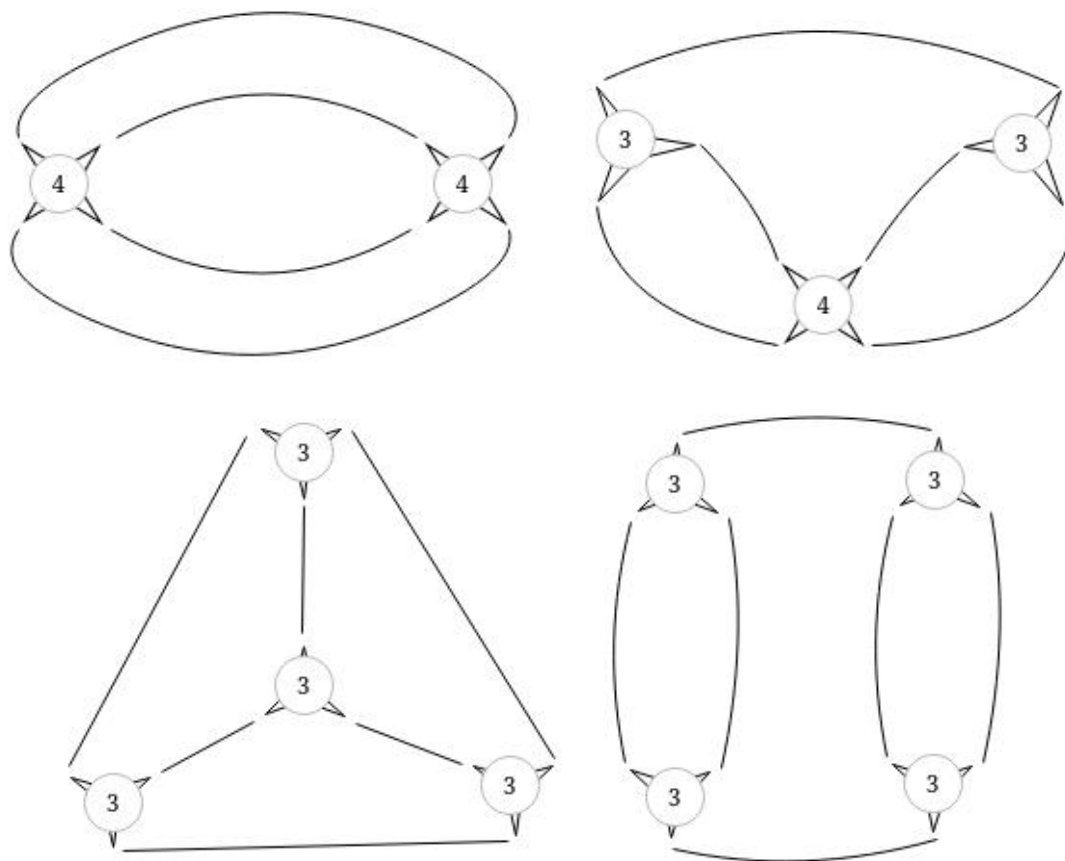


Figura 16 – Topologias biconexas possíveis com $m = n + 2$. Apelidamos a segunda topologia de "quatro33" (quatro-três-três), em decorrência dos graus dos vértices centrais. A terceira topologia nos referimos como a de "tetraedro" e a última de "cilindro", ambas pelas formas visuais.

- c) O valor de X deve representar os demais vértices internos que não são contabilizados em Y . Note que, de fato, $Y \geq X$: **Prova:** Seja c_m o vértice central com o maior valor de $T(c_i)$ e c_a, c_b, c_c os demais. Sejam A, B e C o total de vértices internos nas pontes conectadas à c_m e D, E e F o total de vértices internos das pontes restantes. Então:

$$T(c_m) = A + B + C$$

$$T(c_a) = A + D + E$$

$$T(c_b) = B + E + F$$

$$T(c_c) = C + F + D$$

$$T(c_m) \geq T(c_a)$$

$$T(c_m) \geq T(c_b)$$

$$T(c_m) \geq T(c_c)$$

$$3T(c_m) \geq T(c_a) + T(c_b) + T(c_c)$$

$$3A + 3B + 3C \geq A + B + C + 2D + 2E + 2F$$

$$2(A + B + C) \geq 2(D + E + F)$$

$$A + B + C \geq D + E + F$$

□

- d) O valor de X deverá ser particionado em A, B e C , enquanto o valor de Y deverá ser particionado em D, E e F . Uma dada configuração de tetraedro, (A, B, C, D, E, F) , ainda permitiria até **6 grafos diferentes(ou seja, não isomorfos)**, quando variamos quais arestas dentre (A, B, C) ficam opostas às arestas em (D, E, F) .

Teorema 9. *Para $m = n + 2$ e $n \leq 100$, existe sempre um theta-grafo generalizado que minimiza a soma das distâncias dentre todos os grafos biconexos com um dado número n de vértices e m de arestas.*

Prova:

Com os métodos de iteração apresentados acima, pudemos percorrer **todos os grafos biconexos com $m=n+2$ arestas** e até 100 vértices ($n = 100$). **A nossa conjectura proposta se consolidou verdadeira nestes casos**, isto é, a topologia com o menor índice de Wiener foi a de theta-grafo. □

A figura 17 mostra um gráfico com o crescimento do melhor índice de Wiener, em função de n , para cada uma das quatro topologias possíveis. Os índices de Wiener ótimos de cada topologia são ligeiramente diferente. Foi incluído também os índices de Wiener ótimos para a topologia de theta-grafo com 3 pontes ($m = n + 1$).

É interessante notar que as configurações ótimas dentro de cada topologia **não** são sempre simétricas. De fato, as configurações se mantêm simétricas, na topologia de theta-grafo, por exemplo, apenas para valores de $n < 35$. Cada uma das topologias(exceto a de tetraedro, que não admite nenhuma configuração simétrica ótima), possui um limiar pro valor de n , no qual para valores menores a sua configuração ótima é simétrica/balanceada, enquanto que, para valores maiores, a configuração ótima passa a ser outra, que ainda segue um padrão.

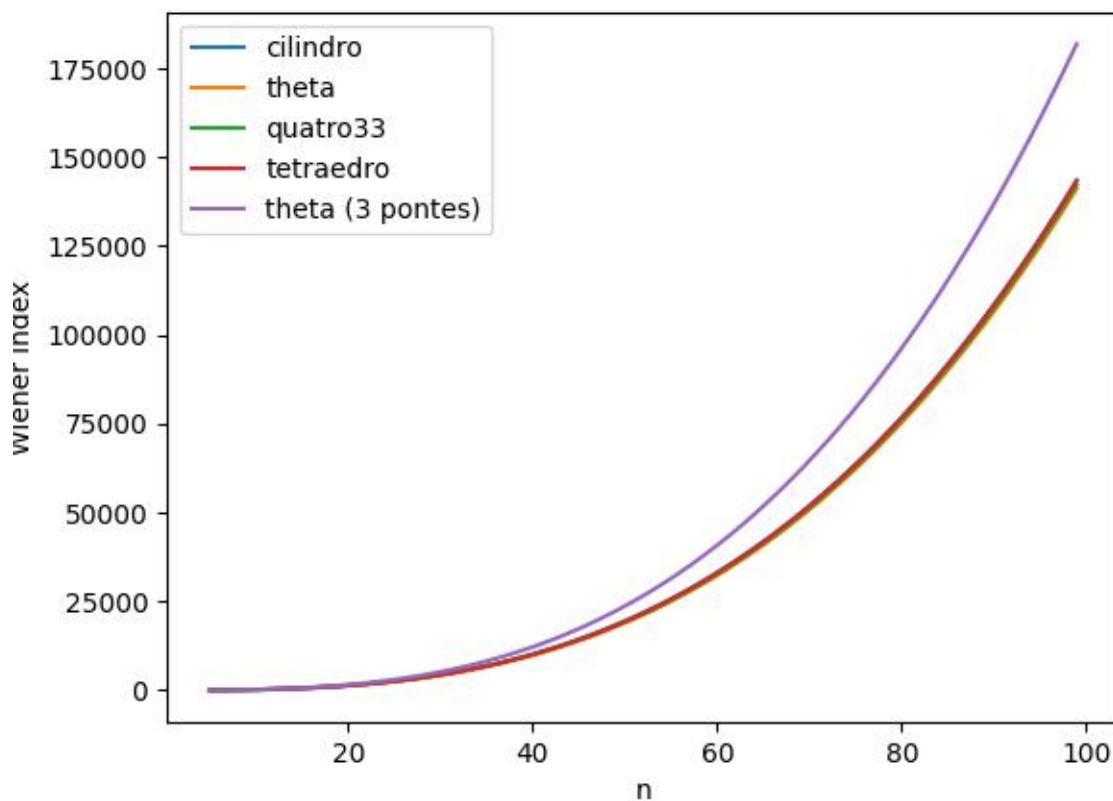


Figura 17 – Crescimento do índice de Wiener em função do número de vértices, n , para grafos com $m = n + 2$ arestas. Também foi incluído os valores para a topologia de theta-grafo com 3 pontes ($m = n + 1$), para comparação.

n	Índice de Wiener ótimo	$[P_1, P_2, P_3, P_4]$
30	4272	[7,7,7,7]
31	4696	[7,7,7,8]
32	5152	[7,7,8,8]
33	5640	[7,8,8,8]
34	6162	[8,8,8,8]
35	6686	[0,11,11,11]
36	7274	[0,11,11,12]
37	7880	[0,11,11,13]
38	8522	[0,11,12,13]
39	9182	[0,11,13,13]
40	9878	[0,12,13,13]
41	10592	[0,13,13,13]

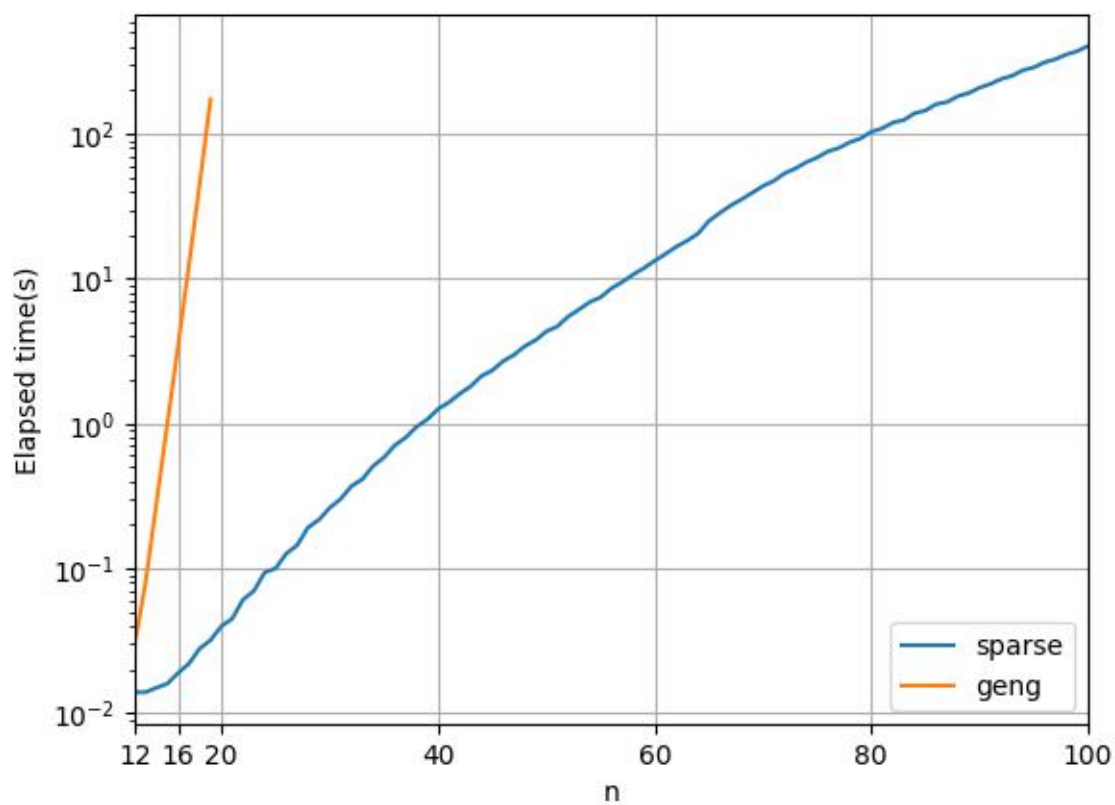


Figura 18 – Comparação do tempo despendido no processo de exploração e cálculo do índice de Wiener na classe $m = n + 2$, entre os dois métodos empregados.

5.2.3 Topologias $n + 3 \leq m \leq 2n - 6$

De acordo com a equação 4.3, para $m = n + 3$ e $n_1 = 0$, todas as distribuições de n_i são apresentadas a seguir:

1. $n_3 = 0, n_4 = 0, n_5 = 0, n_6 = 0, n_7 = 0, n_8 = 1$
2. $n_3 = 1, n_4 = 0, n_5 = 0, n_6 = 0, n_7 = 1$
3. $n_3 = 0, n_4 = 1, n_5 = 0, n_6 = 1$
4. $n_3 = 2, n_4 = 0, n_5 = 0, n_6 = 1$
5. $n_3 = 0, n_4 = 0, n_5 = 2$
6. $n_3 = 1, n_4 = 1, n_5 = 1$
7. $n_3 = 3, n_4 = 0, n_5 = 1$
8. $n_3 = 0, n_4 = 3$

9. $n_3 = 2, n_4 = 2$

10. $n_3 = 4, n_4 = 1$

11. $n_3 = 6$

Cada uma das distribuições ainda permite múltiplas topologias, onde apenas algumas são biconexas. Por exemplo, na distribuição $(n_1 = 0, n_2 = n - 2, n_3 = 0, n_4 = 0, n_5 = 2)$, são 3 topologias possíveis (outras distribuições permitem mais):

1. Com 5 pontes entre os dois vértices de grau 5 (única topologia biconexa)
2. Com 3 pontes entre os dois vértices de grau 5 e cada vértice de grau 5 possuindo uma ponte conectada à si.
3. Com 1 ponte entre os dois vértices de grau 5 e cada vértice de grau 5 possuindo uma ponte conectada à si.

Podemos estimar que a complexidade de analisar, seguindo a nossa abordagem, todas as topologias biconexas para $m = n + k$ com $k \geq 3$, cresce exponencialmente à medida que aumentamos k , semelhante ao problema de particionamento.

6 JOGO PROPOSTO

6.1 ABOUT GRAPHS

“About graphs” é uma ferramenta iterativa que foi criada para auxiliar a análise de todo o trabalho. Foi desenvolvida com o motor de jogos livre e de código aberto *Godot engine* (Juan Linietsky, Ariel Manzur e contribuidores, 2021) e compilada para HTML5, podendo ser facilmente acessível em https://rilpires.github.io/about_graphs/.

Seu objetivo inicial era auxiliar fazer uma rápida validação manual de pequenos experimentos e de forma lúdica e iterativa.

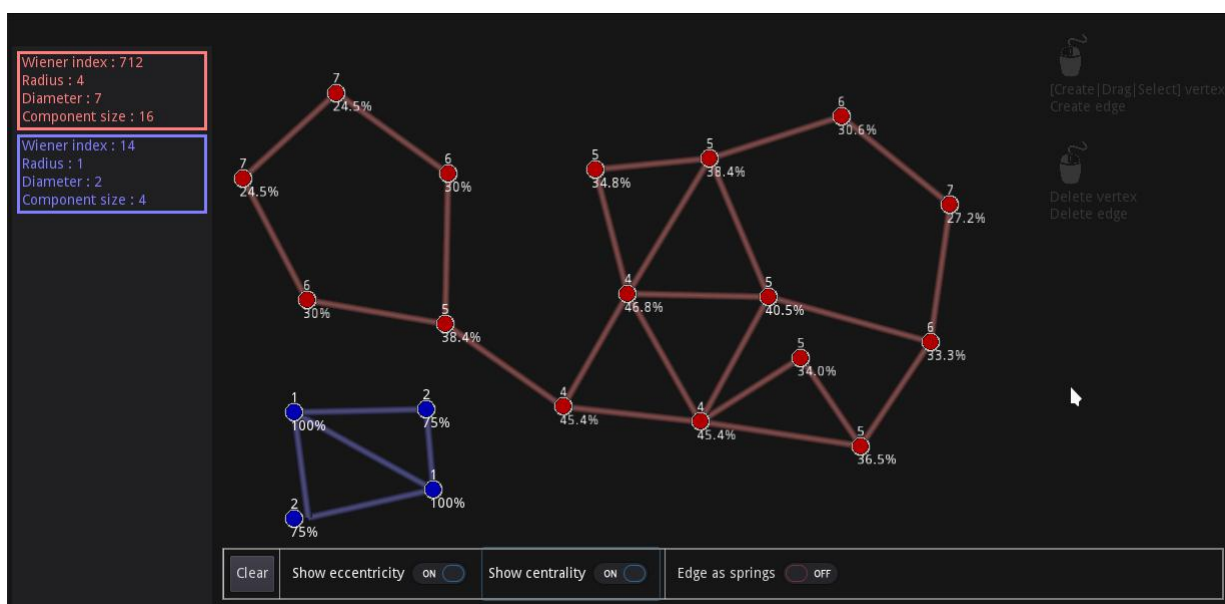


Figura 19 – Demonstração da aplicação "about graphs": cada componente conexa é colorida com uma cor. Pode-se optar por mostrar a excentricidade e centralidade de cada vértice. É possível ativar(padrão) ou desativar o comportamento de "mola" nas arestas, para auxiliar na organização espacial.

A aplicação pode começar a dar sinais de lentidão quando o número de vértices se aproxima de 25, pois é necessário, a cada alteração no grafo, recalcular a matriz de caminhos mínimos para não só mostrar o índice de Wiener mas também indicar outras métricas por vértice, como excentricidade e centralidade.

6.1.1 Possíveis melhorias

A performance certamente poderia ser otimizada fazendo a implementação dos algoritmos em C ou C++ ao invés da linguagem interpretada GDScript, que é voltada para a rápida prototipação. Atualmente a *Godot Engine* permite embutir códigos em C++.

Foi avaliada uma forma de registrar grafos construídos por qualquer usuário em nuvem. Seria uma proposta de jogo interessante para incentivar e apresentar o problema ao público, com rankings online mostrando, juntamente com os respectivos autores, as melhores configurações em diferentes métricas (índice de Wiener, centralidade média, etc.). O desafio seria manusear grafos maiores: pode ser muito demorado para modificar o grafo do jeito que o usuário já tem em mente.

7 TRABALHOS RELACIONADOS

A literatura sobre grafos extremos é vasta (MEYERSON; TAGIKU, 2009; BOLLOBÁS, 2004; DENG; XIAO; TANG, 2010; XU et al., 2014; ZHANG et al., 2021; ZHOU; WANG; LU, 2018). Em particular, árvores extremas com relação ao índice de Wiener foram consideradas em (DENG; XIAO; TANG, 2010). Entretanto, não é de nosso conhecimento nenhum trabalho relacionado que tenha considerado grafos biconexos extremos com relação ao índice de Wiener.

Grafos biconexos extremos com relação ao diâmetro foram considerados em (JARRY; LAUGIER, 2012; ENOMOTO; USAMI, 1999). Conforme indicado neste trabalho de conclusão de curso, a extensão do problema, trocando-se diâmetro por índice de Wiener, não é trivial.

O cálculo eficiente do índice de Wiener para grafos especiais foi considerado em (KLAVZAR; GUTMAN; MOHAR, 1995). Entretanto, não é de nosso conhecimento nenhum resultado especializando o cálculo do índice de Wiener para grafos biconexos. Um algoritmo para o cálculo eficiente do índice de Wiener para grafos biconexos é uma das contribuições deste trabalho de conclusão de curso.

A relação do índice de Wiener com outras métricas de grafos foi extensamente considerada na literatura (KNOR; POTOČNIK; ŠKREKOVSKI, 2014; DANKELMANN et al., 2014; DANKELMANN et al., 2009). Neste trabalho, focamos exclusivamente no índice de Wiener, nos restringindo a grafos biconexos.

8 CONCLUSÃO

Intuitivamente, esperávamos encontrar soluções ótimas que fossem simétricas. Nossa conjectura inicialmente era a de que os theta-grafos com menores índices de Wiener fossem sempre balanceados, mas com este trabalho foi possível perceber de que existe um limiar para o qual os theta-grafos passam a "preferir" grafos desbalanceados.

Também foi uma surpresa o fato de que o mesmo acontece até para a topologia de tetraedro. E que o grafo "*quatro33*" é mais conexo do que o tetraedro, talvez por ser parecer mais com o theta-grafo. Aparentemente, grafos extremos não são tão amigos da simetria quanto pensávamos.

Nosso trabalho propôs uma abordagem naturalmente impossível de provar a conjectura proposta, que é a abordagem de força-bruta em uma infinidade de topologias e número de vértices, sem contar com o cálculo necessário do índice de Wiener que, embora tenha sido especializado para grafos esparsos, ainda cresce em complexidade com o número de vértices.

Dada a ligeira diferença entre os índices de Wiener ótimos para cada topologia com o mesmo número de vértices e arestas, acreditamos que, em uma aplicação real, provavelmente seja mais conveniente aderir à uma topologia por outros fatores, como localização espacial e distância mínima/máxima entre vértices.

Embora não podemos dizer ainda com certeza que os theta-grafos possuem sempre o menor índice de Wiener entre os grafos biconexos com $m < 2n - 5$, temos fortes indícios que isso seja verdade, visto que a classe ótima com $m = 2n - 5$ é um theta-grafo (com $n - 3$ pontes) e nosso trabalho comprovou que, até $n = 100$, os grafos biconexos ótimos com $m = n + 1$ e $m = n + 2$ também são theta-grafos.

8.1 TRABALHOS FUTUROS

Acreditamos que a fórmula 3.16, para o cálculo de $W_{pp}(P_a, P_b)$, ainda possa ser traduzida para uma fórmula de execução em tempo constante, em vez de realizar um somatório com $\min|P_a|, |P_b|$ parcelas. Desta maneira, seria possível analisar a derivada de todo o índice de Wiener em função dos tamanhos das pontes e, com isso, provar se a topologia de theta-grafo sempre vence pelo índice de Wiener, para qualquer valor de n .

Seria interessante a implementação da exploração analítica no pacote *nauty* (ao qual o utilitário *geng* está incluído), restrita apenas aos casos estudados (biconexos $m = n + 1$ e $m = n + 2$). É válido lembrar que o método empregado pelo *nauty*, embora no nosso caso possa não ter sido favorável, foi amplamente utilizado desde sua concepção em 1978 por ter sido o primeiro programa do tipo para grafos genéricos que pudesse suportar algumas centenas de vértices (Brendan McKay, Adolfo Piperno, 2014).

A abordagem analítica para construção de grafos pode ser mais quando exaustiva, como em problemas de enunciado mais simples. Por exemplo: se nos restringíssemos à um grafo ciclo com adição sequencial de arestas, qual a melhor maneira de fazê-la para termos o menor índice de Wiener? (MEYERSON; TAGIKU, 2009; Daniel S. Menasche, 2021) Note que este é um problema que leva em consideração apenas um subconjunto dos grafos da conjectura que tentamos provar neste trabalho.

REFERÊNCIAS

- BOLLOBÁS, B. **Extremal graph theory**. [S.l.]: Courier Corporation, 2004.
- Brendan McKay. **Description of graph6, sparse6 and digraph6 encodings**. 2015. Acessado em 25 de out. de 2021. Disponível em: <<https://users.cecs.anu.edu.au/~bdm/data/formats.txt>>.
- Brendan McKay, Adolfo Piperno. Practical graph isomorphism, ii. 2014. Acessado em 03 de nov. de 2021. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0747717113001193>>.
- Brendan McKay, Adolfo Piperno. **Nauty Traces**. 2020. Acessado em 25 de ago. de 2021. Disponível em: <<https://pallini.di.uniroma1.it/>>.
- CASH, G. G. Relationship between the hosoya polynomial and the hyper-wiener index. **Applied mathematics letters**, Elsevier, v. 15, n. 7, p. 893–895, 2002.
- Daniel S. Menasche. **What is best way to augment a cycle with shortcut edges to minimize the sum of distances?** 2021. Acessado em 8 de nov. de 2021. Disponível em: <<https://math.stackexchange.com/questions/4282701/what-is-best-way-to-augment-a-cycle-with-shortcut-edges-to-minimize-the-sum-of-d>>.
- DANKELMANN, P. et al. The edge-wiener index of a graph. **Discrete Mathematics**, Elsevier, v. 309, n. 10, p. 3452–3457, 2009.
- DANKELMANN, P. et al. On the eccentric connectivity index and wiener index of a graph. **Quaestiones Mathematicae**, Taylor & Francis, v. 37, n. 1, p. 39–47, 2014.
- DENG, H.; XIAO, H.; TANG, F. On the extremal wiener polarity index of trees with a given diameter. **MATCH Commun. Math. Comput. Chem**, v. 63, n. 1, p. 257–264, 2010.
- ENOMOTO, H.; USAMI, Y. Extremal 2-connected graphs with given diameter. **Tokyo Journal of Mathematics**, Publication Committee for the Tokyo Journal of Mathematics, v. 22, n. 1, p. 1–16, 1999.
- JARRY, A.; LAUGIER, A. On the minimum number of edges of two-connected graphs with given diameter. **Discrete Mathematics**, Elsevier, v. 312, n. 4, p. 757–762, 2012.
- Juan Linietsky, Ariel Manzur e contribuidores. **Godot Engine**. 2021. Acessado em 4 de out. de 2021. Disponível em: <<https://godotengine.org/>>.
- KLAVZAR, S.; GUTMAN, I.; MOHAR, B. Labeling of benzenoid systems which reflects the vertex-distance relations. **Journal of chemical information and computer sciences**, ACS Publications, v. 35, n. 3, p. 590–593, 1995.
- KNOR, M.; POTOČNIK, P.; ŠKREKOVSKI, R. Relationship between the edge-wiener index and the gutman index of a graph. **Discrete Applied Mathematics**, Elsevier, v. 167, p. 197–201, 2014.

MEYERSON, A.; TAGIKU, B. Minimizing average shortest path distances via shortcut edge addition. In: **Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques**. [S.l.]: Springer, 2009. p. 272–285.

MURTY, U. On some extremal graphs. **Acta Mathematica Academiae Scientiarum Hungarica**, Springer, v. 19, n. 1-2, p. 69–74, 1968.

NOTES on partitions and their generating functions. 2013. Acessado em 01 de nov. de 2021. Disponível em: <<https://math.berkeley.edu/~mhaiman/math172-spring10/partitions.pdf>>.

Paul Manuel, Indra Rajasingh, Bharati Rajan, R. Sundara Rajan. A new approach to compute wiener index. 2013. Acessado em 24 de out. de 2021. Disponível em: <https://www.researchgate.net/publication/241279743_A_New_Approach_to_Compute_Wiener_Index>.

XU, K. et al. A survey on graphs extremal with respect to distance-based topological indices. **MATCH Commun. Math. Comput. Chem**, v. 71, n. 3, p. 461–508, 2014.

ZHANG, L. et al. Wiener polarity index calculation of square-free graphs and its implementation to certain complex materials. **Mathematical Problems in Engineering**, Hindawi, v. 2021, 2021.

ZHOU, Q.; WANG, L.; LU, Y. Some sufficient conditions on k-connected graphs. **Applied Mathematics and Computation**, Elsevier, v. 325, p. 332–339, 2018.