

UNIVERSIDADE FEDERAL FLUMINENSE  
INSTITUTO DE CIÊNCIA E TECNOLOGIA  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Alonso Leonardo Souza de Oliveira

Um Algoritmo Computacional para o  
Estudo de Convexidade Geodésica em Grafos

Rio das Ostras-RJ

2016

Ficha catalográfica automática - SDC/BRO  
Gerada com informações fornecidas pelo autor

O48a Oliveira, Alonso Leonardo Souza de  
Um Algoritmo Computacional para o Estudo de Convexidade  
Geodésica em Grafos / Alonso Leonardo Souza de Oliveira ;  
Danilo Artigas, orientador. Niterói, 2016.  
35 f. : il.

Trabalho de Conclusão de Curso (Graduação em Ciência da  
Computação)-Universidade Federal Fluminense, Instituto de  
Ciência e Tecnologia, Rio das Ostras, 2016.

1. Convexidade em grafos. 2. Conjunto geodésico. 3.  
Algoritmos em grafos. 4. Produção intelectual. I. Artigas,  
Danilo, orientador. II. Universidade Federal Fluminense.  
Instituto de Ciência e Tecnologia. III. Título.

CDD -

ALONSO LEONARDO SOUZA DE OLIVEIRA

UM ALGORITMO COMPUTACIONAL PARA O  
ESTUDO DE CONVEXIDADE GEODÉSICA EM GRAFOS

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Instituto de Ciência e Tecnologia da Universidade Federal Fluminense, como parte dos requisitos necessários à obtenção do Grau de Bacharel em Ciências. Área de Concentração: Algoritmos e Combinatória.

Orientador: Prof. Dr. DANILO ARTIGAS DA ROCHA

Rio das Ostras-RJ

2016

ALONSO LEONARDO SOUZA DE OLIVEIRA

UM ALGORITMO COMPUTACIONAL PARA O  
ESTUDO DE CONVEXIDADE GEODÉSICA EM GRAFOS

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Instituto de Ciência e Tecnologia da Universidade Federal Fluminense, como parte dos requisitos necessários à obtenção do Grau de Bacharel em Ciências. Área de Concentração: Algoritmos e Combinatória.

Aprovada em JULHO de 2016.

BANCA EXAMINADORA

---

Prof. Dr. DANILO ARTIGAS DA ROCHA - Orientador  
UFF

---

Profa. Dra. MAISE DANTAS DA SILVA  
UFF

---

Prof. Dr. ANDRÉ RENATO VILLELA DA SILVA  
UFF

---

Me. RODRIGO LAMBLET MAFORT  
UFF

Rio das Ostras-RJ  
2016

À José Francisco, que viverá eternamente em memória dentro de mim. Dedico essa conquista ao meu falecido pai, que me deu sabedoria para contestar realidades e propor sempre um novo universo de possibilidades.

# Agradecimentos

Em primeiro lugar, eu agradeço a Deus que iluminou o meu caminho durante essa longa jornada, pois sem Ele eu não teria forças para concluir essa etapa da minha vida; aos meus pais, José e Aparecida, por sempre terem sido a minha referência de tantas maneiras; ao meu irmão, Alan, por me apresentar e sempre me incentivar a crescer no caminho da ciência e tecnologia; aos meus familiares, em especial, aos meus tios, Rosalina, Jair e Iêda, por me darem todo o apoio necessário para conclusão desse curso.

Ao meu professor e orientador deste trabalho, Danilo, pela orientação, pelo apoio e pela amizade, que foram tão importantes na minha vida acadêmica e no desenvolvimento deste trabalho.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

"If the prospect of living in a world where trying to respect the basic rights of those around you and valuing each other simply because we exist are such daunting, impossible tasks then what sort of world are we left with? And what sort of world do you want to live in?"

Wonder Woman, Wonder Woman #170

# Lista de Figuras

1.1	Ilustração das Sete Pontes de Königsberg. . . . .	1
1.2	Modelagem com grafos das Sete Pontes de Königsberg. . . . .	2
2.1	Grafo $G$ cordal e as excentricidades de seus vértices. . . . .	5
2.2	Grafos $G_i$ tal que $Ct(G_i) = \{a, b, c\}$ e $I[Ct(G_i)] = V - d$ . . . . .	6
2.3	Grafos <i>buraco</i> , <i>casa</i> , <i>dominó</i> e <i>gem</i> . . . . .	7
2.4	Grafo bipartido $G$ , com diâmetro 8, cujo contorno não é geodésico. Os números na figura correspondem às excentricidades de cada vértice. Os vértices marcado com retângulo são vértices de contorno de $G$ , mas $v \notin I[Ct(G)]$ . . . . .	7
3.1	Grafo $G$ . . . . .	9
3.2	Grafo <i>circulante</i> $C_{10}(1, 2, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0)$ . . . . .	10
3.3	Grafo $G$ é uma generalização do grafo $G_1$ da Figura 2.2. $G$ possui $i - k - 2$ vértices gêmeos de $a$ , $j - 1$ vértices gêmeos de $d$ , $k - 1$ vértices gêmeos de $b$ e $l - 1$ vértices gêmeos do vértice abaixo de $c$ . . . . .	11
3.4	Grafo com $Ct(G) = \{a, b, c\}$ e os quatro vértices brancos não pertencem ao $I[Ct(G)]$ . . . . .	11
4.1	(a) Grafo com 11 vértices tal que o contorno não é geodésico e o número de arestas é mínimo; (b) Grafo com 11 vértices tal que o contorno não é geodésico e o número de arestas é máximo. . . . .	15
4.2	Grafo bipartido $G$ , com diâmetro 8, cujo contorno não é geodésico. Este foi encontrado através da implementação do Algoritmo 2. Os números na figura correspondem às excentricidades de cada vértice. Os vértices marcado com retângulo são vértices de contorno de $G$ , mas $v \notin I[Ct(G)]$ . . . . .	16



# Sumário

<b>Agradecimentos</b>	<b>v</b>
<b>Lista de Figuras</b>	<b>vii</b>
<b>Resumo</b>	<b>ix</b>
<b>Abstract</b>	<b>x</b>
<b>1 Introdução</b>	<b>1</b>
1.1 O problema da convexidade em grafos . . . . .	1
1.2 Organização da monografia . . . . .	3
<b>2 Definições e resultados iniciais</b>	<b>4</b>
2.1 Preliminares . . . . .	4
2.2 Definições sobre convexidade . . . . .	5
2.3 Histórico do problema . . . . .	6
<b>3 Resultados Estruturais</b>	<b>9</b>
<b>4 Resultados Computacionais</b>	<b>12</b>
4.1 O programa <i>geng</i> . . . . .	12
4.2 Busca em Largura . . . . .	12
4.3 O Algoritmo . . . . .	13
4.4 Resultados obtidos . . . . .	14
<b>5 Conclusão</b>	<b>17</b>
5.1 Considerações finais . . . . .	17
5.2 Trabalhos futuros . . . . .	17
<b>Referências Bibliográficas</b>	<b>18</b>
<b>A Ferramenta Computacional</b>	<b>20</b>
A.1 Implementações . . . . .	20
A.2 Complexidade de Tempo . . . . .	25

# Resumo

Seja  $G = (V, E)$  um grafo simples, finito e conexo. O contorno de um grafo  $G$  é o conjunto dos vértices de  $G$  cuja excentricidade é maior ou igual a de seus vizinhos. O intervalo fechado de  $S \subseteq V$  é o conjunto de todos os vértices que se encontram em algum caminho mínimo entre pares de vértices de  $S$ . Dizemos que  $S$  é um conjunto geodésico de  $G$  se o intervalo fechado de  $S$  for igual ao conjunto de vértices de  $G$ . Neste trabalho, desenvolvemos uma ferramenta computacional para encontrar resultados para dois problemas propostos por Cáceres et al. em 2005: (i) determinar se o contorno de um grafo  $G$  é geodésico; (ii) determinar se existe um grafo  $G$  tal que o intervalo fechado do seu contorno não é geodésico. Com isso, estabelecemos novas propriedades estruturais e resultados computacionais. Apresentamos um resultado estrutural que auxilia a decidir se um conjunto é geodésico. Também estabelecemos um teorema de existência de grafos cujo contorno apresenta algumas características pré-determinadas. Finalmente, aplicando nossa abordagem computacional, determinamos os menores grafos  $G$  e os menores grafos *bipartidos*, em número de vértices, cujo contorno não é geodésico. Por fim, consideramos o problema (ii) proposto em Cáceres et al., e obtivemos um limite inferior no número de vértices onde uma resposta positiva para este problema pode ser encontrada.

Palavras-chave: Convexidade em grafos. Conjunto geodésico. Algoritmos em grafos.

# Abstract

Let  $G = (V, E)$  be a finite, simple and connected graph. The contour of a graph  $G$  is the set of all vertices of  $G$  whose eccentricity is greater than or equal to its neighbours. The closed interval of  $S \subseteq V$  is the set of all vertices lying on shortest paths between any pair of vertices of  $S$ . We say that  $S$  is a geodetic set of  $G$  if the closed interval of  $S$  is equal to all vertices of  $G$ . In this work, we develop a computational tool to find results for two problems proposed by Cáceres et al. in 2005: (i) to determine whether the contour of a graph  $G$  is geodetic. (ii) to determine if there exists a graph  $G$  such that the closed interval of its contour is not geodetic. Thus, we establish new structural and computational results. We show a structural result that helps decide if a set is geodetic. We also establish a theorem of existence of graphs whose contour has some predetermined characteristics. Finally, using our computational approach, we establish the minimum graphs  $G$  and the minimum *bipartite* graphs, in number of vertices, for which the contour is not geodetic. Lastly, we consider the problem (ii) proposed by Cáceres et al., and we obtain a lower bounds to the number of vertices where a positive answer to this problem can be found.

Keywords: Convexity. Geodetic set. Graph algorithm.

# Capítulo 1

## Introdução

### 1.1 O problema da convexidade em grafos

O campo da teoria dos grafos surgiu em 1736, quando Leonhard Euler afirmou que a resposta era negativa para o problema das Sete Pontes de Königsberg. O problema é baseado na cidade de Königsberg (atual Kaliningrado). Na época, sete pontes ligavam a cidade, que é cortada pelo Rio Prególia, como na Figura 1.1. Era uma lenda popular a possibilidade de atravessar todas as pontes sem repetir nenhuma. Euler modelou o problema, transformando os caminhos em arestas e suas intersecções em vértices, criando, possivelmente, o primeiro grafo da história, como o grafo da Figura 1.2. A modelagem usando grafos passou a ser aplicada em diversos ramos da ciência, como redes de computadores, estruturas moleculares, inteligência artificial, entre outras. Além disso, diversos conceitos foram estendidos para teoria dos grafos.

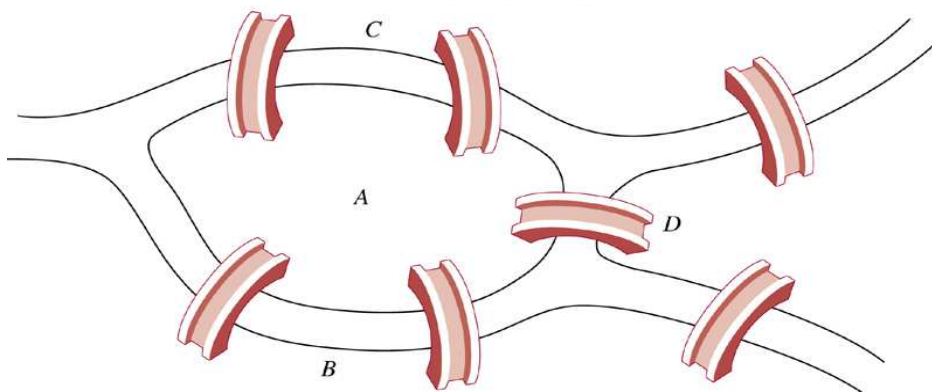


Figura 1.1: Ilustração das Sete Pontes de Königsberg.

O conceito de convexidade em estruturas discretas foi estendido a partir do conceito para matemática contínua. Nos últimos anos, houve um grande número de publicações estendendo conceitos de convexidade para teoria dos grafos. Estes conceitos em teoria dos grafos possuem interessantes aplicações, como propagação de doenças e propagação de vírus em uma rede [1]. Neste trabalho, nós consideramos problemas relacionados à convexidade em grafos.

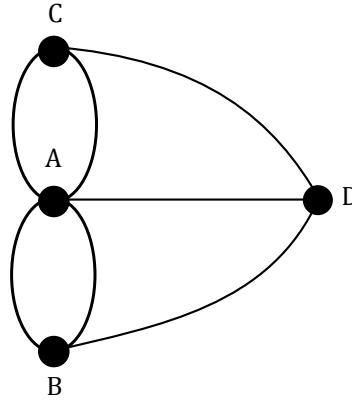


Figura 1.2: Modelagem com grafos das Sete Pontes de Königsberg.

Um conjunto de vértices  $S$  é denominado *geodésico* se todo vértice do grafo se encontra em algum caminho mínimo entre um par de vértices de  $S$ . O problema NÚMERO GEODÉSICO consiste em determinar o número de vértices de um conjunto geodésico mínimo de um grafo  $G$ . A versão de decisão de NÚMERO GEODÉSICO é um problema  $NP$ -completo para um grafo qualquer.

O problema de determinar se um conjunto  $S$  é geodésico tem aplicações diretas em combinatória e em geometria. A partir de um conjunto geodésico  $S$  é possível recuperar todos os vértices do grafo utilizando um operador que determina todos os caminhos mínimos. Dessa forma, este problema tem aplicações com computação, pois se trata de uma forma de compressão de dados.

Em Cáceres et al. [2], os autores definiram o *contorno* de um grafo como um conjunto de vértices cuja excentricidade é maior ou igual à excentricidade de seus vizinhos. O conjunto de contorno é um candidato natural a conjunto geodésico de um grafo. Ainda em [2], foram propostos dois problemas: (i) determinar se o contorno de um grafo é geodésico; (ii) sendo  $S$  um conjunto de vértices que se encontram em algum caminho mínimo entre vértices do contorno, determinar se existe um grafo  $G$  para o qual  $S$  não é geodésico.

Até o presente momento, a literatura utilizou uma abordagem analítica para solucionar os dois problemas propostos em [2]. Tal abordagem é custosa; primeiramente, porque tais grafos não são comuns, mas, principalmente, porque a remoção ou inserção de uma aresta ou vértice pode mudar as excentricidades de todos os vértices de  $G$ . Com isso, há um grande trabalho necessário para calcular manualmente as excentricidades dos vértices de  $G$ , determinar o contorno e verificar se ele é geodésico. A falta de êxito destas abordagens e a dificuldade de se verificar manualmente estas propriedades tornam a abordagem computacional promissora para a análise dos problemas, e este é o foco deste trabalho.

Contribuímos com os problemas propostos em [2] apresentando um resultado estrutural que auxilia a decidir se um conjunto é geodésico; um teorema de existência de grafos cujo contorno apresenta algumas características predeterminadas; e resultados computacionais que, dentre outros resultados, permitiram estabelecer quais os menores grafos cujo contorno não é geodésico.

## 1.2 Organização da monografia

Neste primeiro capítulo, apresentamos as definições essenciais para esta monografia, o problema da convexidade geodésica, bem como uma preliminar dos nossos resultados.

No Capítulo 2, introduzimos todas as definições e notações necessárias para esta monografia e definimos os conceitos básicos de Teoria dos Grafos e os conceitos avançados de intervalo fechado, conjuntos geodésicos e conjunto de contorno. Por fim, apresentamos os resultados iniciais da literatura.

No Capítulo 3, consideramos o problema de decidir se o contorno de um grafo é geodésico. Apresentamos os nossos resultados estruturais, e estabelecemos um teorema que relaciona a cardinalidade do contorno de um grafo com o problema de determinar se o contorno é geodésico. Estabelecemos também um teorema que diz como construir famílias infinitas de grafos cujo contorno não é geodésico.

No Capítulo 4, introduzimos nossa abordagem computacional e apresentamos nossos principais resultados obtidos através dessa abordagem.

No Capítulo 5, apresentamos nossas conclusões e alguns problemas e abordagens que são de interesse para trabalhos futuros.

Por último, no Apêndice A, apresentamos e discutimos nossa ferramenta computacional, utilizada para obtenção dos nossos principais resultados.

Os resultados desta monografia foram apresentados no CNMAC 2014, com resumo publicado no "Proceeding Series of the Brazilian Society of Computational and Applied Mathematics" [3], e no LawCliques 2014, com resumo publicado no "Proceedings of the 6th Latin American Workshop on Cliques in Graphs" [4] e artigo publicado na revista Matemática Contemporânea [5]. Além disso, os resultados parciais foram também apresentados no Seminário de Iniciação Científica de 2014 e 2015 e publicados em seus respectivos livros de resumos [6, 7]. Por fim, todos os resultados obtidos se tornaram um artigo completo publicado na "International Transactions in Operational Research" [8].

## Capítulo 2

# Definições e resultados iniciais

Este capítulo introduz as definições e notações necessárias para este trabalho. Ao final do capítulo, apresentamos os primeiros resultados da literatura.

### 2.1 Preliminares

Um grafo  $G$  *finito e simples* é uma dupla ordenada  $(V(G), E(G))$ , onde  $V(G)$  é um conjunto finito, cujos elementos são denominados *vértices*, e  $E(G)$  é um conjunto de subconjuntos de  $V(G)$  com cardinalidade dois; os elementos de  $E(G)$  são denominados *arestas*. Uma aresta entre os vértices  $v, w$  de  $V(G)$  será representada por  $\{v, w\}$ . Na falta de ambiguidade, escreveremos  $V$  e  $E$  e representaremos a cardinalidade de  $V$  por  $|V|$ .

Um *subgrafo*  $H$  de  $G$  é um grafo onde  $V(H) \subseteq V(G)$  e  $E(H) \subseteq E(G)$ . Um *subgrafo induzido*  $H$  de  $G$  é um subgrafo de  $G$  tal que  $E(H) = \{(v, w) \in E(G) \mid v, w \in V(H)\}$ .

Um *caminho*  $\mathcal{P}$ , com comprimento  $k$ , entre  $v_0$  e  $v_k$  em  $G$  é uma sequência finita de vértices distintos  $v_0, v_1, \dots, v_k$ , onde  $v_{i-1}, v_i$  são adjacentes, para  $1 \leq i \leq k$ . Representamos o comprimento de  $\mathcal{P}$  por  $|\mathcal{P}|$ . Observe que  $|\mathcal{P}|$  é exatamente o número de arestas em  $\mathcal{P}$ .

Um *ciclo* é uma sequência de vértices  $v_0, v_1, \dots, v_k$ , para  $k \geq 3$ , tal que  $v_0, \dots, v_k$  é um caminho e  $v_k = v_0$ .

Um grafo  $G$  é *conexo* quando existe um caminho entre cada par de vértices de  $V(G)$ , do contrário, ele é dito *desconexo*. Os grafos  $G$  adotados neste trabalho são *finitos, simples e conexos*.

A *distância*,  $d(v, w)$ , entre dois vértices  $v, w \in V(G)$  é o número de arestas em um caminho mínimo entre  $v$  e  $w$ . A *excentricidade*,  $ecc(v)$ , de um vértice  $v$  é o máximo de  $d(v, w)$  para todo vértice  $w \in V(G)$ , ou seja,  $ecc(v) = \max\{d(v, w) \mid w \in V(G)\}$ . O *diâmetro* de  $G$ ,  $diam(G)$ , é o máximo  $ecc(v)$  para todo vértice  $v \in V(G)$ . O *raio* de  $G$ ,  $r(G)$ , é o mínimo  $ecc(v)$  para todo vértice  $v \in V(G)$ . A *vizinhança* de  $v$  em  $G$ ,  $N(v)$ , é o conjunto  $\{w \in V(G) \mid d(v, w) = 1\}$ . Dois vértices  $v, w$  são *gêmeos falsos* se  $N(v) = N(w)$ . Por simplicidade, usaremos apenas o termo *gêmeos* para fazer referência aos gêmeos falsos.

Um grafo  $G$  é *bipartido* quando seu conjunto de vértices  $V(G)$  puder ser particionado em dois

subconjuntos  $V'(G)$  e  $V''(G)$ , tais que toda aresta  $\{v, w\} \in E(G)$  une um vértice  $v \in V'(G)$  a outro vértice  $w \in V''(G)$ . Os grafos bipartidos são adequados para modelagem de problemas de acoplamento. Mais conceitos básicos de teoria dos grafos podem ser encontrados em [9].

## 2.2 Definições sobre convexidade

Nesta seção, definiremos alguns conceitos sobre convexidade geodésica em grafos e algumas classes de grafos.

**Definição 1.** [2] *Seja  $G$  um grafo. Uma geodésica entre dois vértices  $v, w$  é um caminho entre  $v$  e  $w$  com comprimento  $d(v, w)$ .*

Observe que uma geodésica entre dois vértices é exatamente um caminho mínimo entre os dois vértices.

**Definição 2.** [2] *O intervalo fechado,  $I[S]$ , de um conjunto  $S \subseteq V(G)$  é o conjunto de todos os vértices que se encontram em alguma geodésica entre pares de vértices de  $S$ , incluindo os vértices em  $S$ .*

**Definição 3.** [2] *Um conjunto  $S \subseteq V(G)$  é denominado conjunto geodésico de  $G$  se  $I[S] = V(G)$ .*

Como exemplo, utilizaremos a Figura 2.1. Seja  $S_1 = \{a, c\}$  então  $I[S_1] = V(G)$ , logo  $S_1$  é um conjunto geodésico de  $G$ . Seja  $S_2 = \{a, b, d\}$  então  $I[S_2] = S_2$ , logo  $S_2$  não é um conjunto geodésico de  $G$ .

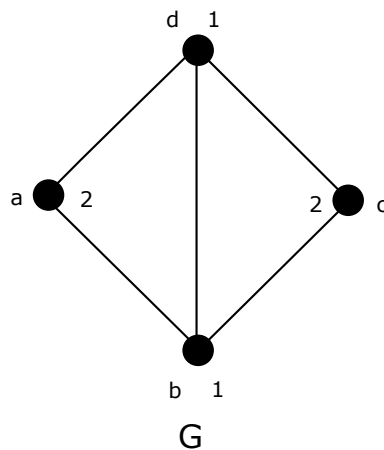


Figura 2.1: Grafo  $G$  cordal e as excentricidades de seus vértices.

**Definição 4.** [2] *Um vértice  $v$  é denominado vértice de contorno de  $G$  se  $ecc(v) \geq ecc(w)$  para todo vértice  $w$  adjacente a  $v$ . O contorno,  $Ct(G)$ , de  $G$  é o conjunto de todos os vértices de contorno de  $G$ .*

Os números na Figura 2.1 representam a excentricidade de cada vértice, logo o  $Ct(G) = \{a, c\}$ . Observe que, na Figura 2.2, o grafo  $G_1$  possui  $Ct(G_1) = \{a, b, c\}$  e  $I[Ct(G_1)] = V(G_1) \setminus \{d\}$ . Logo,  $Ct(G_1)$  não é um conjunto geodésico.

**Definição 5.** [2] *Um vértice  $v$  é denominado vértice de excentricidade de  $w$ ,  $e(w)$ , se  $ecc(w) = d(w, v)$ .*



Como exemplo, na Figura 2.1, os vértices  $a, c, d$  são vértices de excentricidade de  $b$ , pois estão à distância 1 de  $b$  e  $\text{ecc}(b) = 1$ . Já o grafo  $G_1$ , da Figura 2.2, possui  $\text{ecc}(a) = d(a, c) = 5$ . Logo,  $c$  é um *vértice de excentricidade* de  $a$ .

Com as definições apresentadas podemos apresentar os dois problemas que serão abordados neste trabalho.

(I) CONTORNO GEODÉSICO

**Instância:** Classe de grafos  $\mathcal{C}$ .

**Pergunta:** Existe um grafo  $G \in \mathcal{C}$  tal que  $\text{Ct}(G)$  é geodésico?

(II) INTERVALO FECHADO DO CONTORNO GEODÉSICO

**Instância:** Conjunto de todos os grafos  $\mathcal{C}$ .

**Pergunta:** Existe  $G$  tal que  $I[\text{Ct}(G)]$  é geodésico?

## 2.3 Histórico do problema

Os problemas (I) e (II) foram propostos por Cáceres et al. [2] em 2005. Neste artigo, os autores definiram o conceito de contorno de um grafo, apresentaram o grafo  $G_2$  da Figura 2.2, cujo contorno não é geodésico, e iniciaram o estudo do problema (I) para algumas classes de grafos. A partir desse ponto, diversos artigos foram publicados sobre este tema [10, 11, 12, 13].

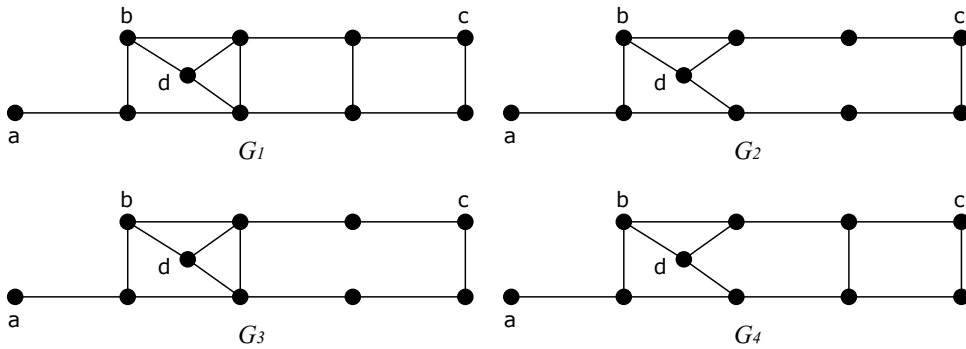


Figura 2.2: Grafos  $G_i$  tal que  $\text{Ct}(G_i) = \{a, b, c\}$  e  $I[\text{Ct}(G_i)] = V - d$ .

Dados dois grafos  $G$  e  $H$ , dizemos que  $G$  contém  $H$  se  $H$  é um subgrafo induzido de  $G$ . Denotamos por  $P_k$  um caminho induzido com  $k$  vértices e  $C_k$ ,  $k \geq 3$ , um ciclo induzido com  $k$  vértices. Um grafo *buraco* é um  $C_k$  para  $k \geq 5$  (Figura 2.3a). Um grafo *casa* é o complemento de um  $P_5$  (Figura 2.3b). Um grafo *dominó* é um grafo bipartido obtido através da adição de exatamente uma corda a um  $C_6$  (Figura 2.3c). Um grafo *gem* é um grafo de cinco vértices obtido através da adição de um vértice universal a um  $P_4$  (Figura 2.3d).

Em 2005, Cáceres et al. [2] provou que o contorno de grafos *distância hereditária*, que são grafos que não contém *casa*, *buraco*, *dominó* ou *gem*, é geodésico, e apresentou um grafo -  $G_2$  da Figura 2.2,

cujo contorno não é geodésico. Em 2008, Eroh et al. [13] provou que o contorno de grafos HHD-free, que não contém *casa*, *buraco* ou *dominó*, é geodésico.

Em 2008, Cáceres et al. [11] apresentou o grafo  $G_1$  da Figura 2.2 que é um grafo de permutação cujo contorno não é geodésico. Além disso, provou que o contorno de um grafo *cordal* (grafo que não contém  $C_k$  induzido para  $k \geq 4$ ) é geodésico e propôs como problema em aberto decidir se o contorno é geodésico para grafos bipartidos, grafos cocordais (complementos de cordais) e grafos de paridade.

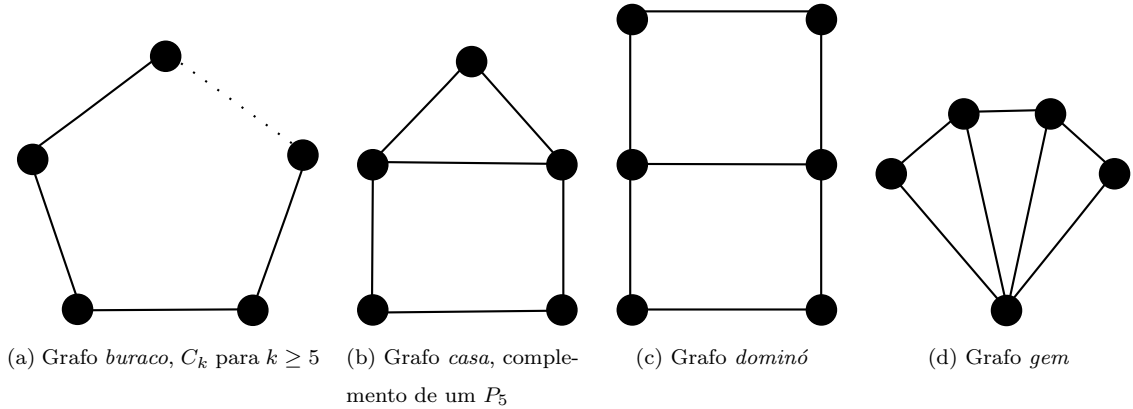


Figura 2.3: Grafos *buraco*, *casa*, *dominó* e *gem*

Em 2013, os problemas propostos em [11] foram resolvidos por Artigas et al. [10]. Neste artigo, os autores provaram que o contorno de grafos cocordais é geodésico e apresentaram o grafo da Figura 2.4, que é um grafo bipartido e de paridade tal que o contorno não é geodésico. Os autores também apresentaram os Teoremas 3 e 4, apresentados no Capítulo 4, que relacionam o problema de decidir se o contorno é geodésico com o diâmetro do grafo.

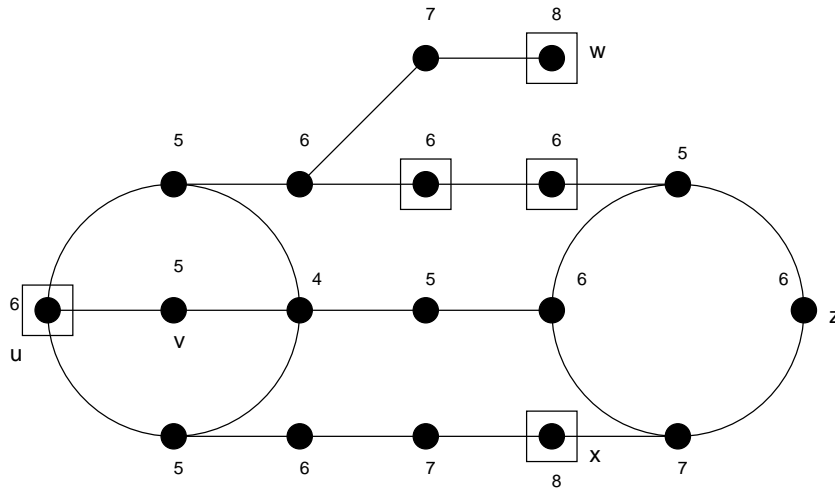


Figura 2.4: Grafo bipartido  $G$ , com diâmetro 8, cujo contorno não é geodésico. Os números na figura correspondem às excentricidades de cada vértice. Os vértices marcado com retângulo são vértices de contorno de  $G$ , mas  $v \notin I[Co(G)]$ .

Em 2015, Mezzini et al. [14] propôs uma técnica para auxiliar a reconhecer se o contorno é geodé-

sico analisando os blocos (componentes biconexos) do grafo. Em outro artigo dos mesmos autores [15], foi provado que o contorno de grafos *bridged* é geodésico. Um grafo  $G$  é *bridged* se cada ciclo  $C$  de tamanho, pelo menos, quatro contém dois vértices cuja distância para cada vértice em  $G$  é estritamente menor do que em  $C$ . Neste mesmo ano, Artigas et al. [16] provaram que o contorno de um grafo *bipartido cordal* (bipartido e sem *buraco*) é geodésico.

Por fim, em 2016, Mezzini [17] resolveu o problema (II), apresentando um grafo  $G$  tal que  $I^2[Ct(G)] \neq V(G)$ . Denotamos  $I^2[Ct(G)] = I[I[Ct(G)]]$ .

## Capítulo 3

# Resultados Estruturais

Neste capítulo, investigamos aspectos estruturais do problema de determinar se o contorno de um grafo é geodésico. Os resultados apresentados foram obtidos analiticamente.

Primeiramente, introduziremos algumas ferramentas básicas necessárias para o desenvolvimento dos nossos resultados.

**Observação 1.** Se  $v, u \in V$ , então  $|ecc(v) - ecc(u)| \leq d(v, u)$ . Em particular, se  $\{v, w\} \in E$  então  $|ecc(v) - ecc(w)| \leq 1$ .

**Lema 1.** [11] Se  $u_0 \in V$  e  $\mathcal{P} = u_0, u_1, \dots, u_t$  é um caminho em  $G$  tal que  $ecc(u_{i+1}) = ecc(u_i) + 1$ , para cada  $i \in \{0, 1, \dots, t-1\}$ , então, para cada vértice  $u_t$  de excentricidade  $e(u_t)$ , existe uma geodésica entre  $e(u_t)$  e  $u_t$  que contém  $\mathcal{P}$ . Além disso,  $e(u_t)$  é um vértice de excentricidade de cada vértice em  $\mathcal{P}$ .

Na Figura 3.1, o vértice  $c$  é um vértice de excentricidade de  $u_2$ , e o caminho  $\mathcal{P} = u_0, u_1, u_2$  atende a propriedade mencionada no Lema 1. Portanto, o Lema 1 garante a existência de uma geodésica entre  $u_2$  e  $c$  passando por  $u_1$  e  $u_0$ .

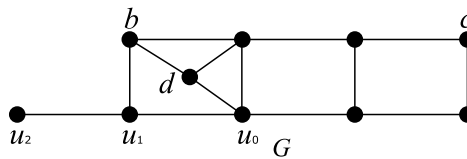


Figura 3.1: Grafo  $G$ .

O primeiro resultado estrutural encontrado, Teorema 1, estabelece uma relação entre a cardinalidade do contorno de um grafo e o problema de determinar se o contorno é geodésico.

**Teorema 1.** Se  $Ct(G) \subseteq S \subseteq V$  e  $|V \setminus S| \leq r(G)$ , então  $S$  é geodésico.

*Demonstração.* Suponha que um vértice  $v_0 \in V \setminus S$ , tal que  $v_0 \notin I[S]$ . Uma vez que  $v_0 \notin I[S]$ , pela Observação 1, existe um caminho  $P = v_0, v_1, \dots, v_k$  onde  $ecc(v_i) + 1 = ecc(v_{i+1})$  e  $v_i \in V \setminus S$ , para  $0 \leq i \leq k-1$ , e  $v_k \in S$ .

Seja  $e(v_k)$  um vértice de excentricidade de  $v_k$ . Pelo Lema 1, existe uma geodésica  $\mathcal{P}$  entre  $v_k$  e  $e(v_k)$  passando por  $P$ . Se  $\mathcal{P}$  tem um vértice  $x \neq v_k$  tal que  $x \in S$ , então a geodésica entre  $x$  e  $v_k$ , formada

pelos vértices de  $\mathcal{P}$ , é uma geodésica entre dois vértices de  $S$  contendo  $v_0$ , contradição. Daí, nenhum vértice de  $\mathcal{P}$ , exceto  $v_k$ , pertence ao conjunto  $S$ . Já que  $\text{ecc}(v_0) < \text{ecc}(v_k)$ , temos que  $\text{ecc}(v_k) \geq r(G) + 1$ . Portanto,  $|\mathcal{P}| \geq r(G) + 2$ . Desta forma, uma vez que o único vértice de  $\mathcal{P}$  em  $S$  é  $v_k$ , concluímos que  $|V \setminus S| \geq r(G) + 1$ , uma contradição à hipótese.  $\square$

A partir deste teorema, é possível provar que o contorno dos grafos *circulantes*, (Figura 3.2), é geodésico. Sejam  $n, m$  e  $a_1, \dots, a_m$  inteiros positivos. Um grafo  $G$  tal que  $V(G) = \{0, \dots, n-1\}$  e  $E(G) = \{\{i, (i+a_j) \pmod{n}\} \mid 0 \leq i \leq n-1, 1 \leq j \leq m\}$  é chamado de grafo *circulante*, e é denotado por  $C_n(a_1, \dots, a_m)$ .

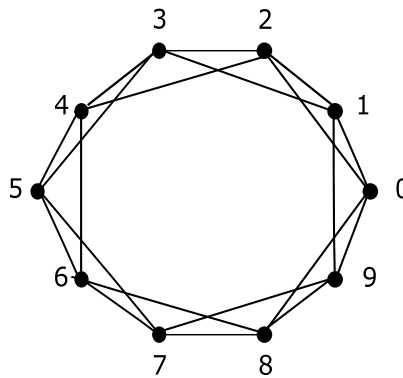


Figura 3.2: Grafo *circulante*  $C_{10}(1, 2, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0)$ .

**Observação 2.** Sejam  $v, v'$  vértices gêmeos de  $G = (V, E)$  tal que  $\text{ecc}(v) > 2$  e  $\text{ecc}(v') > 2$ . Assim, para todo  $x \in V \setminus \{v, v'\}$ ,  $d(v, x) = d(v', x)$ . Portanto,  $\text{ecc}(v) = \text{ecc}(v')$ .

**Observação 3.** Se  $v, v'$  são vértices gêmeos de  $G = (V, E)$ , então, para todo vértice  $x \in V \setminus \{v, v'\}$  temos que  $I[v, x] \setminus \{v\} = I[v', x] \setminus \{v'\}$ .

**Observação 4.** Sejam  $v, v'$  vértices gêmeos de  $G = (V, E)$  e  $x, y \in V \setminus \{v, v'\}$ . Um caminho  $\mathcal{P} = x, \dots, w, v, z, \dots, y$  é uma geodésica entre  $x$  e  $y$  contendo  $v$  se, e somente se, o caminho  $\mathcal{P}' = x, \dots, w, v', z, \dots, y$  é uma geodésica entre  $x$  e  $y$  contendo  $v'$ .

As Observações 2, 3 e 4 estabelecem que vértices gêmeos preservam as relações de distância em um grafo. O próximo teorema é uma consequência direta dessas observações. Ele garante que é possível construir grafos com uma quantidade de vértices de contorno e uma quantidade de vértices que não estão no  $I[Ct(G)]$ .

**Teorema 2.** Sejam inteiros  $(i, j, k, l)$  tais que  $j, k, l \geq 1$  e  $i \geq k$ . É possível construir um grafo  $G$  com  $i$  vértices de contorno,  $j$  vértices que não pertencem ao  $I[Ct(G)]$  e  $k$  vértices de contorno com  $l$  vértices de excentricidade que não são vértices de contorno.

*Demonstração.* Decorre diretamente das Observações 2, 3 e 4.  $\square$

Na Figura 3.3, apresentamos um grafo  $G$  que satisfaz as condições do Teorema 2. Aplicando o Teorema 2, podemos gerar o grafo  $G$  da Figura 3.4, que é um exemplo de um grafo tal que a volta do

Teorema 1 não é válido, ou seja,  $Ct(G) \subseteq I[Ct(G)] \subseteq V$  e  $|V \setminus I[Ct(G)]| = 4 > 3 = r(G)$ , mas  $I[Ct(G)]$  é geodésico.

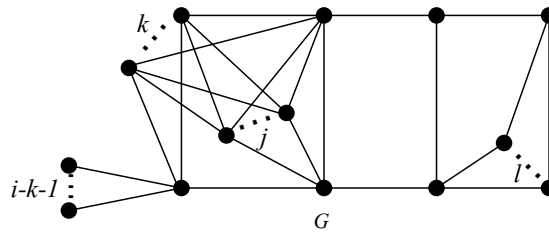


Figura 3.3: Grafo  $G$  é uma generalização do grafo  $G_1$  da Figura 2.2.  $G$  possui  $i - k - 2$  vértices gêmeos de  $a$ ,  $j - 1$  vértices gêmeos de  $d$ ,  $k - 1$  vértices gêmeos de  $b$  e  $l - 1$  vértices gêmeos do vértice abaixo de  $c$ .

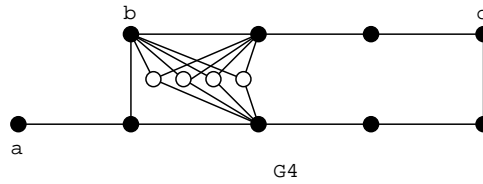


Figura 3.4: Grafo com  $Ct(G) = \{a, b, c\}$  e os quatro vértices brancos não pertencem ao  $I[Ct(G)]$ .

## Capítulo 4

# Resultados Computacionais

Neste capítulo, mostraremos os resultados que obtivemos computacionalmente através de uma implementação do Algoritmo 2 para os problemas (I) e (II). A implementação do Algoritmo 2 se encontra no Apêndice A.

### 4.1 O programa *geng*

Iniciamos nossa abordagem computacional procurando por grafos cujo contorno não é geodésico. O primeiro passo foi gerar todos os grafos não isomorfos com um número fixo de vértices. Para isso, utilizamos o programa *geng* desenvolvido por McKay & Piperno [18]. O *geng* pode gerar grafos não isomorfos rapidamente, além de gerar instâncias de algumas classes de grafos, como bipartidos, que são de grande interesse neste trabalho. A utilização do *geng* é viável, pois o mesmo pode ser utilizado gratuitamente para fins não lucrativos.

### 4.2 Busca em Largura

Nesta seção, inicialmente, vamos definir o algoritmo de *Busca em Largura*. Ele foi utilizado para os cálculos de distâncias entre cada par de vértices, da excentricidade de cada vértice, do diâmetro e do raio do grafo  $G$ .

A *Busca em Largura* é um algoritmo para percurso em grafos. Este procedimento permite calcular a distância entre um vértice inicial  $s$  e todos os vértices de um grafo  $G$ . Consequentemente, também é possível obter a excentricidade de  $s$  e, após  $|V|$  execuções da *Busca em Largura* (uma para cada vértice), também calculamos o raio e o diâmetro de  $G$ .

A *Busca em Largura* possui como entrada um grafo  $G$  e um vértice inicial  $s$ . Ela é capaz de calcular a distância de  $s$  aos demais vértices de  $G$  usando a seguinte ideia: inicialmente a busca começa pelo vértice  $s$  que recebe distância 0; depois, a busca visita os vizinhos de  $s$ , que recebem distância 1; na iteração seguinte a busca visita os vizinhos dos vizinhos de  $s$  que ainda não foram visitados (estes vértices correspondem àqueles com distância 2 do vértice  $s$ ); a busca procede sucessivamente desta forma até que todos os vértices do grafo sejam visitados.

A seguir, apresentamos o algoritmo de *Busca em Largura* adaptado do livro *Algorithms* [19]. O algoritmo 1 utiliza uma fila  $Q$  para controlar a ordem em que os vértices serão visitados, um vetor  $dist[v]$  que armazena a distância de  $s$  a cada vértice  $v$ , e uma variável  $ecc(s)$  que, ao término do procedimento, terá o valor da excentricidade de  $s$ .

---

**Algoritmo 1** *Busca em Largura*.

---

**Entrada:** Grafo  $G$  e vértice de origem  $s \in V(G)$ .

**Saída:** Distância de  $s$  para todo vértice  $u \in V(G)$  e excentricidade de  $s$ .

1. **para** todo  $u \in V(G)$  **faça**
  2.      $dist(u) \leftarrow \infty$
  3. **fim para**
  4.  $dist(s) \leftarrow 0$
  5.  $ecc(s) \leftarrow 0$
  6. enfileirar( $Q, s$ )
  7. **enquanto**  $Q$  não está vazia **faça**
  8.      $u \leftarrow$  desenfileirar( $Q$ )
  9.     **para** todas as arestas  $(u, v) \in E$  **faça**
  10.         **se**  $dist(v) = \infty$  **então**
  11.             enfileirar( $Q, v$ )
  12.              $dist(v) \leftarrow dist(u) + 1$
  13.              $ecc(s) \leftarrow \max(ecc(s), dist(v))$
  14.         **fim se**
  15.     **fim para**
  16. **fim enquanto**
- 

Por fim, para calcular o raio e o diâmetro do grafo, basta calcular a excentricidade de todos os vértices de  $G$  e descobrir quais são os valores mínimo e máximo que correspondem, respectivamente, a  $r(G)$  e  $diam(G)$ .

A complexidade da *Busca em Largura* é  $O(|V| + |E|)$ . Devido a esta complexidade, é preferível utilizar  $|V|$  execuções deste algoritmo que qualquer um dos algoritmos presentes na literatura para cálculo de distância entre todos os pares de vértices de um grafo.

### 4.3 O Algoritmo

Nesta seção, apresentamos o Algoritmo 2, que desenvolvemos para que, a partir dos grafos gerados pelo *geng*, seja determinado se o contorno deste grafo é um conjunto geodésico.



---

**Algoritmo 2** Determinar se  $I[Ct(G)] = V$ .

---

**Entrada:** Grafo  $G$ .

**Saída:**  $Ct(G)$  é geodésico ou não é geodésico.

1. Calcular a excentricidade de cada vértice de  $G$ ;
  2. **se** ( $diam(G) \leq 4$ ) **ou** ( $G$  é bipartido e  $diam(G) \leq 7$ ) **então**
  3.     **retorna**  $Ct(G)$  é geodésico.
  4. **fim se**
  5. Determinar para cada vértice  $v$  se  $v \in Ct(G)$ ;
  6. **se**  $|V \setminus Ct(G)| \leq r(G)$  **então**
  7.     **retorna**  $Ct(G)$  é geodésico.
  8. **fim se**
  9. Verificar se  $I[Ct(G)] = V$ .
- 

O algoritmo tem complexidade  $O(|V|^3)$ . Uma análise detalhada dessa complexidade pode ser vista na Seção A.2. No entanto, verificar se existe um grafo  $G$  com tal propriedade, para um número específico de vértices é um trabalho custoso. Por exemplo, o número total de grafos  $G$  não isomorfos de 10 vértices é 11.716.571. Com isso, é necessário que o algoritmo seja executado 11.716.571 vezes somente para  $|V| = 10$ . Portanto, para melhorar o desempenho computacional, incluímos os testes apresentados na literatura [10], descritos nos Teoremas 3 e 4.

**Teorema 3.** [10] *Se  $G$  é um grafo com  $diam(G) \leq 4$ , então  $Ct(G)$  é geodésico.*

**Teorema 4.** [10] *Se  $G$  é um grafo bipartido com  $diam(G) \leq 7$ , então  $Ct(G)$  é geodésico.*

No passo 1 do Algoritmo 2, para calcular a excentricidade de cada vértice, aplicamos  $|V|$  vezes o algoritmo de *Busca em Largura*. Os passos 2 a 4 são a inclusão dos Teoremas 3 e 4. No passo 5, para determinar se  $v \in Ct(G)$  para cada vértice  $v \in V(G)$ , comparamos a excentricidade de  $v$  com todo vértice  $w \in N(v)$ . Além do teste do passo 2, para melhorar o desempenho computacional, incluímos nos passos 6 a 8 o nosso primeiro resultado estrutural descrito no Teorema 1. Por fim, no passo 9, o algoritmo verifica para cada vértice  $v \in V(G)$  se existem  $x, y \in Ct(G)$  tal que  $v$  se encontra em um caminho mínimo entre  $x$  e  $y$ .

## 4.4 Resultados obtidos

Usando a implementação do Algoritmo 2 do Apêndice A, desenvolvemos testes computacionais para investigar os problemas (I) e (II). Desse modo, estabelecemos computacionalmente os seguintes teoremas.

**Teorema 5.** *Se  $G$  é um grafo com no máximo 9 vértices, então  $Ct(G)$  é geodésico.*

Com isso, a contra-positiva deste teorema nos garante que, se  $I[Ct(G)] \neq V(G)$ , então  $G$  possui pelo menos 10 vértices.

**Teorema 6.** *Existem somente quatro grafos  $G$  com 10 vértices tais que  $Ct(G)$  não é geodésico.*

Os quatro grafos exibidos na Figura 2.2 não possuem o contorno geodésico. Nossos resultados asseguram que não existem outros grafos com 10 vértices tal que o contorno não seja geodésico. Logo, estes são os menores grafos, em número de vértices, com esta propriedade. Sabíamos previamente da existência de tais grafos, porém era desconhecido que, dentre os milhões de grafos de 10 vértices, estes são os únicos grafos cujo contorno não é geodésico.

**Teorema 7.** *Existem 307 grafos  $G$  com 11 vértices tais que  $Ct(G)$  não é geodésico.*

Apresentamos, na Figura 4.1, dois exemplos de grafos com 11 vértices tais que  $I[Ct(G)] \neq V(G)$  com número mínimo e o máximo de arestas, respectivamente. Observe que ambos satisfazem as condições do Teorema 2. O grafo exibido em (a) na Figura 4.1 é uma generalização do grafo  $G_2$  da Figura 2.2, pois possui  $l = j = k = 1$  e  $i = 4$ , ou seja, um vértice gêmeo de  $a$ . Por outro lado, o grafo exibido em (b) na Figura 4.1 é uma generalização do grafo  $G_1$  da Figura 2.2, pois temos  $l = j = k = 1$ ,  $i = 3$  e um vértice gêmeo do vértice abaixo do vértice  $d$ .

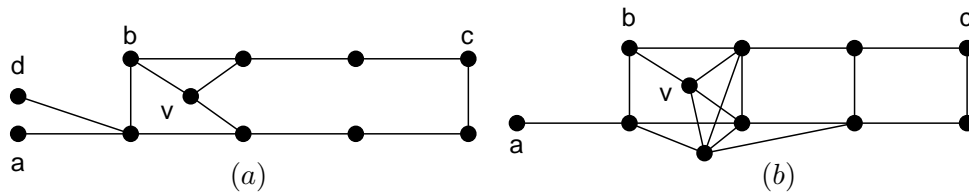


Figura 4.1: (a) Grafo com 11 vértices tal que o contorno não é geodésico e o número de arestas é mínimo; (b) Grafo com 11 vértices tal que o contorno não é geodésico e o número de arestas é máximo.

**Teorema 8.** *Existem 23710 grafos  $G$  com 12 vértices tais que  $Ct(G)$  não é geodésico.*

Foi apresentado em [10] um grafo bipartido com 18 vértices cujo contorno não é geodésico, (grafo  $G$  da Figura 2.4). Entretanto, no Teorema 9, mostramos que este é o menor grafo bipartido, em número de vértices, que satisfaz esta propriedade. Além disso, encontramos outro grafo bipartido com 18 vértices cujo contorno não é geodésico, (grafo  $G$  da Figura 4.2).

**Teorema 9.** *Se  $G$  é um grafo bipartido com no máximo 17 vértices, então  $Ct(G)$  é geodésico.*

E por fim, para o problema (II) proposto por [2], que consiste em descobrir se existe um grafo  $G$  tal que  $I[Ct(G)]$  não é geodésico, apresentamos o Teorema 10. Além disso, investigamos este problema para grafos bipartidos e estabelecemos o Teorema 11.

**Teorema 10.** *Se  $G$  é um grafo com no máximo 12 vértices, então  $I[Ct(G)]$  é geodésico.*

**Teorema 11.** *Se  $G$  é um grafo bipartido com no máximo 17 vértices, então  $I[Ct(G)]$  é geodésico.*

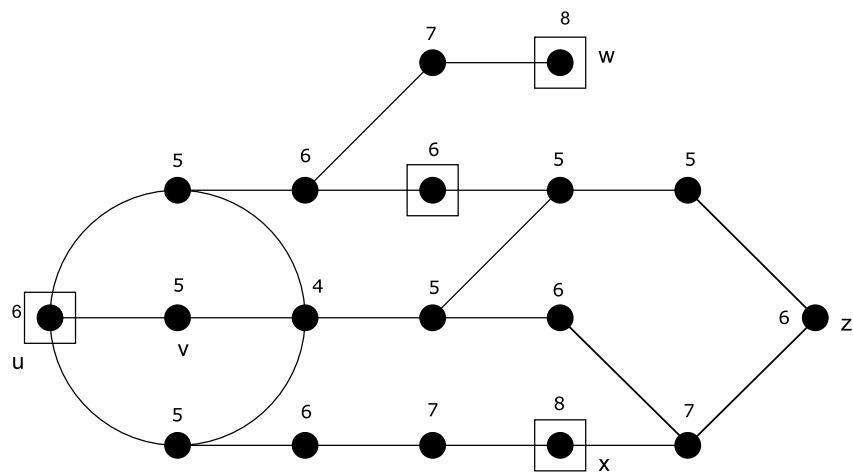


Figura 4.2: Grafo bipartido  $G$ , com diâmetro 8, cujo contorno não é geodésico. Este foi encontrado através da implementação do Algoritmo 2. Os números na figura correspondem às excentricidades de cada vértice. Os vértices marcado com retângulo são vértices de contorno de  $G$ , mas  $v \notin I[Ct(G)]$ .

# Capítulo 5

## Conclusão

Neste capítulo, apresentamos nossas considerações finais e possíveis trabalhos futuros.

### 5.1 Considerações finais

Esse texto é uma exposição consolidada dos nossos trabalhos [3, 4, 5, 6, 7, 8].

Consideramos neste trabalho o problema de determinar se o contorno de um grafo é geodésico. Observamos que, em 8 anos, poucos grafos com tais propriedades foram apresentados na literatura, pois estes grafos são raros e os autores utilizaram de uma abordagem analítica para resolver o problema. A partir dessas circunstâncias, introduzimos uma nova abordagem para encontrar resultados para este problema. Com os resultados encontrados, foi possível desenvolver teoremas inéditos na literatura.

Usando nossa abordagem computacional, encontramos 24.020 grafos novos com tais propriedades, e também estabelecemos quais são os menores grafos e os menores grafos bipartidos, em número de vértices, com esta propriedade. Além disso, encontramos resultados estruturais para construir famílias infinitas de grafos tais que o contorno não é geodésico. Por fim, estabelecemos um novo limite inferior para o número de vértices tal que  $I[Ct(G)]$  não é geodésico.

### 5.2 Trabalhos futuros

Como trabalhos futuros, temos o interesse em desenvolver novos testes e analisar o problema para algumas classes de grafos. E, por fim, utilizar programação paralela para melhorar o desempenho computacional e, com os resultados obtidos, se possível, caracterizar os grafos que não possuem o contorno geodésico.

# Referências Bibliográficas

- [1] DOURADO, M. C., RAUTENBACH, D., SANTOS, V. F., SCHÄFER, P. M., SZWARCFITER, J. L., TOMAN, A. “On the radon number for p3-convexity”, *Lecture Notes in Computer Science*, v. 7256, pp. 267–278, 2012.
- [2] CÁCERES, J., MÁRQUEZ, A., OELLERMANN, O. R., PUERTAS, M. “Rebuilding convex sets in graphs”, *Discrete Mathematics*, v. 297, pp. 26–37, 2005.
- [3] OLIVEIRA, A. L. S., ARTIGAS, D. “Uma abordagem computacional do contorno de grafos”, *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, v. 3, n. 1, 2015a. doi: 10.5540/03.2015.003.01.0227. Disponível em: <<http://dx.doi.org/10.5540/03.2015.003.01.0227>>.
- [4] ARTIGAS, D., DANTAS, S., OLIVEIRA, A. L. S., SILVA, T. M. D. “New results on the geodeticity of the contour of a graph”. In: *Proceedings of the 6th Latin American Workshop on Cliques in Graphs*, pp. 57–57, 2014.
- [5] ARTIGAS, D., DANTAS, S., OLIVEIRA, A. L. S., SILVA, T. M. D. “New Results of the Geodeticity of the Contour of a Graph”, *Matemática Contemporânea*, v. 44, 2016a. Disponível em: <<http://mc.sbm.org.br/wp-content/uploads/sites/15/2016/02/44-7.pdf>>.
- [6] OLIVEIRA, A. L. S., ARTIGAS, D. “Uma Abordagem Computacional do Contorno de Grafos”. In: *Revista PIBIC - Ano 2014*, Niterói, 2014.
- [7] OLIVEIRA, A. L. S., ARTIGAS, D. “Uma Abordagem Computacional do Contorno de Grafos”. In: *Revista PIBIC - Ano 2015*, Niterói, 2015b.
- [8] ARTIGAS, D., DANTAS, S., OLIVEIRA, A. L. S., SILVA, T. M. D. “Computational and structural analysis of the contour of graphs”, *International Transactions in Operational Research*, 2016b. ISSN: 1475-3995. doi: 10.1111/itor.12290. Disponível em: <<http://dx.doi.org/10.1111/itor.12290>>.
- [9] BONDY, J. A., MURTY, U. S. R. *Graph Theory*. Springer, 2008.
- [10] ARTIGAS, D., DANTAS, S., DOURADO, M., SZWARCFITER, J., YAMAGUCHI, S. “On the contour of graphs”, *Discrete Applied Mathematics*, v. 161, pp. 1356–1362, 2013.

- [11] CÁCERES, J., HERNANDO, C., MORA, M., PELAYO, I. M., PUERTAS, M., C. SEARA. “Geodeticity of the contour of chordal graphs”, *Discrete Applied Mathematics*, v. 156, pp. 1132–1142, 2008.
- [12] CÁCERES, J., HERNANDO, M. C., MORA, M., PELAYO, I. M., PUERTAS, M. L., SEARA, C. “On geodetic sets formed by boundary vertices”, *Discrete Mathematics*, v. 306, n. 2, pp. 188–198, 2006.
- [13] EROH, L., OELLERMANN, O. R. “Geodetic and Steiner geodetic sets in 3-Steiner distance hereditary graphs”, *Discrete Mathematics*, v. 308, pp. 4212–4220, 2008.
- [14] MEZZINI, M., MOSCARINI, M. “On the geodeticity of the contour of a graph”, *Discrete Applied Mathematics*, v. 181, pp. 209–220, 2015a.
- [15] MEZZINI, M., MOSCARINI, M. “The contour of a bridged graph is geodetic”, *Discrete Applied Mathematics*, v. 204, pp. 213–215, 2015b. ISSN: 0166-218X. doi: <http://dx.doi.org/10.1016/j.dam.2015.10.007>. Disponible em: <http://www.sciencedirect.com/science/article/pii/S0166218X15004941>.
- [16] ARTIGAS, D., SRITHARAN, R. “Geodeticity of the contour of chordal bipartite graphs”, *Electronic Notes in Discrete Mathematics*, v. 50, pp. 237 – 242, 2015. ISSN: 1571-0653. doi: <http://dx.doi.org/10.1016/j.endm.2015.07.040>. Disponible em: <http://www.sciencedirect.com/science/article/pii/S157106531500195X>. LAGOS’15 - VIII Latin-American Algorithms, Graphs and Optimization Symposium.
- [17] MEZZINI, M. “On the geodetic iteration number of the contour of a graph”, *Discrete Applied Mathematics*, v. 206, pp. 211–214, 2016. ISSN: 0166-218X. doi: <http://dx.doi.org/10.1016/j.dam.2016.02.012>. Disponible em: <http://www.sciencedirect.com/science/article/pii/S0166218X16300634>.
- [18] MCKAY, B. D., PIPERNO, A. “Practical graph isomorphism, ii”, *J. Symbolic Computation*, v. 60, pp. 94–112, 2013.
- [19] PAPADIMITRIOU, C., DASGUPTA, S., VAZIRANI, U. *Algorithms*. McGraw-Hill Education, 2006.

# Apêndice A

## Ferramenta Computacional

### A.1 Implementações

O algoritmo 2 foi implementado usando a linguagem C. Inicialmente, a implementação define todas as estruturas de dados necessárias, tais como vetores, matriz de adjacência e uma fila para o algoritmo de busca em largura garantir a ordem de chegada dos vértices. Na função *isGeodetic()*, dado um conjunto *S* de entrada, ela responde se *S* é ou não é *geodésico*. As principais rotinas do algoritmo se encontram na função principal, *main()*.

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<string.h>
4
5 #define oo 400
6 #define MAXN 20
7 #define WHITE -1
8 #define BLACK 1
9 #define max(a,b) ((a) > (b) ? a : b)
10 #define min(a,b) ((a) < (b) ? a : b)
11
12 int start;
13 int end;
14 int n;
15 int q[MAXN];
16 int AdjMat[MAXN][MAXN];
17 int dist[MAXN][MAXN];
18 int ecc[MAXN];
19 int diam, raio, per;
20 int bipartite, first;
21
22 void construct() {
```

```

23     start = 0;
24     end = 0;
25     n = 0;
26     memset(q, WHITE, sizeof(q));
27 }
28
29 int full() { return (n == MAXN); }
30
31 int empty() { return (n == 0); }
32
33 void push(int valor) {
34     if(!full()) {
35         q[end] = valor;
36         n++;
37         end = ((start + n) % MAXN);
38     }
39 }
40
41 void pop(int* x) {
42     if(empty())
43         *x = WHITE;
44     else {
45         *x = q[start];
46         q[start] = WHITE;
47         n--;
48         start = ((start+1) % MAXN);
49     }
50 }
51
52 int isGeodesic(int S[], int size, int N) {
53     int I[N], n = 0, i, j, v;
54
55     memset(I, WHITE, sizeof(I));
56
57     for(v = 0; v < N; v++) {
58         if(ecc[v] >= diam-1) {
59             I[v] = BLACK;
60             n++;
61             if(n == N) return 1;
62         }
63         else {
64             for(i = 0; i < size; i++) {

```



```

65     for(j = i+1; j < size; j++) {
66         if(I[v] == WHITE) {
67             // Triangle Identities
68             if(dist[S[i]][S[j]] == dist[S[i]][v]+dist[v][S[j]]) {
69                 I[v] = BLACK;
70                 n++;
71                 if(n == N) return 1;
72             }
73         }
74     }
75 }
76 }
77 }
78
79 return 0;
80 }
81
82 //Breadth-First Search Algorithm
83 void bfs(int N, int s) {
84     int u, x, j;
85     dist[s][s] = 0;
86     construct();
87     push(s);
88     while(!empty()) {
89         pop(&u);
90         for(j=0;j<N;j++) {
91             if(AdjMat[u][j]) {
92                 if(dist[s][j] == WHITE) {
93                     dist[s][j] = dist[s][u]+1;
94                     ecc[s] = max(ecc[s], dist[s][j]);
95                     push(j);
96                 }
97                 else if(first && dist[s][j] == dist[s][u]) bipartite = 0;
98             }
99         }
100     }
101     first = 0;
102 }
103
104 int main(void) {
105     int V, GCTN = 0;
106     char head, bin;

```

```

107 scanf("%c", &head);
108 V = head - 63; // V = number of vertices
109
110 do {
111     int f, i, j, k = 0, size;
112     char v, aux[7], s[7], tail[oo], binary[oo];
113     int Ct[V];
114
115     binary[0] = '\0';
116     memset(AdjMat, 0, sizeof(AdjMat));
117     memset(dist, WHITE, sizeof(dist));
118     memset(ecc, 0, sizeof(ecc));
119     diam = per = 0;
120     raio = oo;
121     bipartite = first = 1;
122
123     scanf("%s", tail);
124
125     // Convert from geng format (graph6 format) to adjacency matrix
126     for(size = 0; size < strlen(tail); size++) {
127         memset(s, '0', sizeof(s));
128
129         itoA(tail[size]-63, aux, 2);
130
131         for(i=(6-strlen(aux)),j=0;i<6;i++,j++)
132             s[i] = aux[j];
133         s[6] = '\0';
134         strcat(binary, s);
135     }
136
137     for(i=0;i<V;i++) {
138         for(j=0;j<i;j++) {
139             if(binary[k] == '1')
140                 AdjMat[i][j] = AdjMat[j][i] = 1;
141             k++;
142         }
143     }
144
145     // Calculate the eccentricity of every vertex of G
146     for(f=0;f<V;f++) {
147         bfs(V, f);
148         diam = max(diam, ecc[f]);

```

```

149     raio = min(raio, ecc[f]);
150 }
151
152 // Tests by Artigas et al. (2013)
153 if((diam > 4) && (!bipartite || diam > 7) && (raio < diam - 1)) {
154     int isContour, n = 0;
155     for(i = 0; i < V; i++) {
156         if(ecc[i] == diam) {
157             per++;
158             isContour = 1;
159         }
160         else {
161             isContour = 0;
162             for(j = 0; j < V; j++) {
163                 if(AdjMat[i][j]) {
164                     if(ecc[j] > ecc[i]) {
165                         isContour = 0;
166                         break;
167                     }
168                     else {
169                         isContour = 1;
170                     }
171                 }
172             }
173         }
174
175         // Contour vertex
176         if(isContour) {
177             Ct[n] = i;
178             n++;
179         }
180     }
181
182     /* Theorem: If  $Ct(G) \subset S \subset V$  and  $|V \setminus S| \leq \text{rad}(G)$ ,
183     then  $S$  is geodetic. */
184     if((n < V-raio) && (n != per)) {
185         if(!isGeodesic(Ct, n, V)) {
186             printf("%c%s", head, tail);
187             printf("\n");
188             GCTN++;
189         }
190     }

```

```

191     }
192     fgetc(stdin);
193 } while (scanf("%c", &head) != EOF);
194
195 return 0;
196 }

```

## A.2 Complexidade de Tempo

Observe que, para calcular a excentricidade de um vértice, o algoritmo aplica a *Busca em Largura*. A complexidade de tempo do algoritmo de *Busca em Largura* no pior caso, aquele em que todos os vértices e arestas são explorados pelo algoritmo, é dada por  $O(|V| + |E|)$ . Porém, a busca é executada  $|V|$  vezes. Logo, para o cálculo das excentricidades, a complexidade de tempo é  $O(|V|^2 + |V||E|)$ .

O desempenho computacional foi otimizado com a inserção dos testes extraídos em [10]. Caso a entrada não obedeça à condição do teste, o algoritmo já tem uma conclusão sobre a entrada e termina a sua execução para esta entrada. Caso contrário, o algoritmo gera o *contorno* do grafo de entrada. A complexidade de tempo para geração do *contorno* é  $O(|V|^2)$ . Em posse do *contorno*, o desempenho computacional foi novamente melhorado com a inserção do teste obtido neste trabalho.

Se a entrada satisfaz este teste, é necessário então que o algoritmo continue para determinar se o *contorno* é *geodésico*. Em posse do *contorno*, o último passo é a geração do *intervalo fechado*. Essa geração é realizada dentro da função *isGeodetic()*. A complexidade desta geração é dada por  $O(|V|(\frac{|Ct(G)|^2 - |Ct(G)|}{2}))$ . Como  $Ct(G) \subseteq V(G)$ , então, no pior caso, a geração do *intervalo fechado* é dada por  $O(\frac{|V|^3 - |V|^2}{2})$ . Por fim, o retorno desta função conclui sobre a *convexidade* do *contorno* do grafo de entrada. Em conclusão, a complexidade de tempo do algoritmo pode ser expressa como  $O(|V|^3)$ .