

UFRRJ
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM
MODELAGEM MATEMÁTICA E COMPUTACIONAL

DISSERTAÇÃO

O Problema do 1-Centro em Árvores:
Variações e Aplicações

Isis Paulo do Nascimento

2023



**UNIVERSIDADE FEDERAL RURAL DO RIO DE JANEIRO
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM
MATEMÁTICA E COMPUTACIONAL**

**O PROBLEMA DO 1-CENTRO EM ÁRVORES:
VARIAÇÕES E APLICAÇÕES**

ISIS PAULO DO NASCIMENTO

Sob orientação de
Carlos Andrés Reyna Vera-Tudela

e co-orientação de
Aquiles Braga de Queiroz

Dissertação submetida como requisito parcial para obtenção do grau de **Mestre** no Programa de Pós-Graduação em Modelagem Matemática e Computacional, Área de Concentração em Modelagem Matemática e Computacional.

Seropédica, RJ, Brasil
Agosto de 2023

Universidade Federal Rural do Rio de Janeiro
Biblioteca Central / Seção de Processamento Técnico

Ficha catalográfica elaborada
com os dados fornecidos pelo(a) autor(a)

N736p Nascimento , Isis , 16/10/1992-
O problema do 1-centro em árvores: variações e
aplicações. / Isis Nascimento . - Mesquita, 2023.
54 f.: il.

Orientador: Carlos Andrés Reyna Vera-Tudela.
Coorientador: Aquiles Braga de Queiroz.
Dissertação(Mestrado). -- Universidade Federal
Rural do Rio de Janeiro, PROGRAMA DE PÓS-GRADUAÇÃO EM
MODELAGEM MATEMÁTICA E COMPUTACIONAL, 2023.

1. árvores. 2. problemas de localização. 3. 1
centro. I. Reyna Vera-Tudela, Carlos Andrés , 1964- ,
orient. II. Braga de Queiroz, Aquiles , 1971- ,
coorient. III Universidade Federal Rural do Rio de
Janeiro. PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM
MATEMÁTICA E COMPUTACIONAL. IV. Título.



**MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL RURAL DO RIO DE JANEIRO
INSTITUTO DE CIÊNCIAS EXATAS**



PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM MATEMÁTICA E COMPUTACIONAL

Seropédica-RJ, 30 de agosto de 2023.

ISIS PAULO DO NASCIMENTO

Dissertação submetida como requisito parcial para a obtenção de grau de Mestre no Programa de Pós-Graduação em Modelagem Matemática e Computacional-PPGMMC, área de concentração em Modelagem Matemática e Computacional

DISSERTAÇÃO APROVADA EM 30/08/2023

Carlos Andres Reyna Vera Tudela Drº UFRRJ (Orientador, Presidente da Banca)

Marcelo Dib Cruz Drº UFRRJ (membro interno)

Robson Mariano da Silva Drº UFRRJ (membro interno)

Paulo Eustáquio Duarte Pinto Drº UERJ (Externo à Instituição)



Emitido em 30/08/2023

ATA Nº ata/2023 - ICE (12.28.01.23)

(Nº do Documento: 3571)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 06/09/2023 13:38)

CARLOS ANDRES REYNA VERA TUDELA

PROFESSOR DO MAGISTERIO SUPERIOR

DeptM (12.28.01.00.00.63)

Matrícula: ###336#3

(Assinado digitalmente em 08/09/2023 11:09)

MARCELO DIB CRUZ

PROFESSOR DO MAGISTERIO SUPERIOR

DCOMP (11.39.97)

Matrícula: ###680#1

(Assinado digitalmente em 11/09/2023 21:39)

ROBSON MARIANO DA SILVA

PROFESSOR DO MAGISTERIO SUPERIOR

ICE (12.28.01.23)

Matrícula: ##82#8

(Assinado digitalmente em 20/09/2023 12:59)

PAULO EUSTAQUIO DUARTE PINTO

ASSINANTE EXTERNO

CPF: ###.###.367-##

Visualize o documento original em <https://sipac.ufrrj.br/documentos/> informando seu número: **3571**, ano: **2023**, tipo: **ATA**, data de emissão: **06/09/2023** e o código de verificação: **1fa9763010**

Agradecimentos

Em primeiro lugar, dedico essa dissertação ao único que é digno de receber toda honra e glória, à Deus meu amado pai que tens me capacitado e suprido todos esses anos. Sou grata pelo fôlego de vida e pelo lindo propósito que ele colocou em meu coração. Agradeço por ter me rodeado de pessoas especiais como meus orientadores Carlos Andrés e Aquiles Braga que foram imprescindíveis nesse processo, sem a orientação, conselhos e suporte de vocês nada disso seria possível, louvo a Deus pelas suas vidas.

A minha doce e amada família por todo carinho e dedicação demonstrada de inúmeras formas, em especial meus pais Sócrates e Rosane, minha tia Rosangela, ao meu irmão Roney, cunhada Suellen e ao meu pequeno Zion por todas as conversas, apoio, desabafos, abraços e palavras de conforto quando as coisas não iam bem, obrigada por serem minha rede de apoio.

Aos meus professores e amigos, dedico minha gratidão por me motivarem, auxiliarem e aconselharem, ter amigos que se preocupam e se dedicam é como ter um tesouro na terra, e sou muito abençoada por ter vocês na minha vida.

A minha psicóloga Renata Marinho por me desafiar a romper meus medos e me encorajar a iniciar o mestrado e desenvolver meu projeto, obrigada por me fazer enxergar todo potencial e capacidade que eu tenho e me auxiliar a desenvolver cada etapa com carinho e assertividade, você foi e é instrumento de Deus na minha vida, nunca se esqueça disso.

Agradeço também ao IMPA pela dedicação em fornecer materiais que foram determinantes para a realização deste trabalho, juntamente agradeço também a UFRRJ por me conceder a oportunidade de cursar a pós-graduação em um programa de referência que é o PPGMMC. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, ao qual agradeço pelo apoio financeiro que foi essencial para minha permanência no mestrado.

*"A Matemática, quando a
compreendemos bem, possui não
somente a verdade, mas também a
suprema beleza."
(Bertrand Russel)*

DO NASCIMENTO, Isis Paulo. **O Problema do 1-Centro em Árvores: Variações e Aplicações**. 2023. 44f. Dissertação (Mestrado em Modelagem Matemática e Computacional). Instituto de Ciências Exatas, Universidade Federal Rural do Rio de Janeiro, Seropédica, RJ, 2023.

Problemas de localização possuem aplicações em diversas áreas, incluindo o estudo do planejamento de redes de distribuição de energia. No presente trabalho, apresentamos o problema do 1-centro modificado em árvores com aplicações ao estudo do redimensionamento de redes de energia, bem como um algoritmo para a resolução do problema em tempo $O(n)$, onde consideramos pesos e distâncias positivas.

A pesquisa também inclui a apresentação de resultados computacionais para alguns dos métodos apresentados, como os métodos de resolução em tempo $O(n^2)$, $O(n \log n)$ e $O(n)$, assim como novas estratégias para a aplicação de problemas de localização ao projeto de redes de distribuição de energia.

Palavras-chave: Árvores, problemas de localização, 1-centro.

GENERAL ABSTRACT

DO NASCIMENTO, Isis Paulo. **The Problem of 1-Center in Trees: Variations and Applications**. 2023. 44p. Dissertation (Master in Mathematical and Computational Modeling). Instituto de Ciências Exatas, Universidade Federal Rural do Rio de Janeiro, Seropédica, RJ, 2023.

Location problems have applications in many areas, including the study of the planning of power distribution systems. In the present work, we present the modified 1-center problem in trees, with applications to the study of resizing of power distribution networks, as well as an algorithm to solve the problem in $O(n)$ time, considering positive weights and distances.

The research also includes the presentation of computational results for some of the methods presented, such as the resolution methods in $O(n^2)$, $O(n \log n)$ and $O(n)$ time, and new strategies for the application of location problems to the design of power distribution networks.

Keywords: Trees, location problems, 1-center.

1	INTRODUÇÃO	1
1.1	Motivação do Tema	1
1.2	Escopo da Dissertação	2
1.3	Objetivo Geral	2
1.4	Objetivos Específicos	2
2	CONCEITOS PRELIMINARES	3
2.1	Definições Principais	3
2.2	NP-Compleitude	6
2.3	O Problema do 1-Centro Modificado	8
3	PROBLEMAS DE LOCALIZAÇÃO	11
3.1	Definições de Problemas	11
3.2	NP-Compleitude do Problema de Determinar os p-Centros	14
3.3	Método de Kariv e Hakimi	16
3.4	Método de Megiddo	18
4	ALGORITMOS POLINOMIAIS E RESULTADOS PARA O PROBLEMA 1-CENTRO MODIFICADO	24
4.1	Algoritmo de Tempo Quadrático	24
4.2	Algoritmo de Tempo $O(n \log n)$	25
4.3	Algoritmo de Tempo Linear	26
4.4	Resultados	31
4.5	Observações	33
5	CONCLUSÃO	39
5.1	Considerações Finais	39
5.2	Trabalhos Futuros	39
6	REFERÊNCIAS BIBLIOGRÁFICAS.....	41

Lista de Figuras

Figura 2.1 – Grafo $G = (V, E)$; Árvore $G' = (V', E')$	4
Figura 2.2 – Árvore $T = (V, E)$; Representação de T	4
Figura 2.3 – Cobertura de vértices de $G = (V, E)$	6
Figura 2.4 – Conjunto dominante de $G = (V, E)$	6
Figura 2.5 – Mapa de estudo.....	8
Figura 2.6 – Representação da rede de distribuição.	9
Figura 2.7 – Definição do cálculo de queda de tensão. (a) Distância entre os vértices v_j e v_k . (b) Demanda acumulada em uma subárvore $T(v_j, v_k)$. (c) Constante do material utilizado no trecho (v_j, v_k)	10
Figura 3.1 – Problema da 1-mediana em grafos.	11
Figura 3.2 – Problema da p -mediana em grafos.	12
Figura 3.3 – Problema do 1-centro em grafos.	13
Figura 3.4 – Problema do p -centro em grafos.	13
Figura 3.5 – Redução do problema da cobertura de vértices para o problema do conjunto dominante.....	15
Figura 3.6 – Distância $d(u, x)$	20
Figura 3.7 – Vértices a e b	20
Figura 3.8 – Definição de t_{ab}	21
Figura 4.1 – Exemplo do problema do 1-centro modificado.	24
Figura 4.2 – Árvore $T = T_0$ e os valores de perdas máximas $R(v_0, v_{0,1}), \dots, R(v_0, v_{0,d})$ a partir de v_0 nas subárvores $T_{v_0, v_{0,1}}^+, \dots, T_{v_0, v_{0,d}}^+$	27
Figura 4.3 – Perda máxima em $T = T_0$ a partir de v_0	27
Figura 4.4 – Perda máxima em $T_0 \setminus T_{v_0, k}$ a partir de v_0	28
Figura 4.5 – Árvore T'_0	28
Figura 4.6 – Peso do vértice u	29
Figura 4.7 – Distância $d(v_0, u)$	29
Figura 4.8 – Performance dos métodos $O(n^2)$, $O(n \log n)$ e $O(n)$ para o problema do 1-centro modificado em instâncias com $n = 7, 10, \dots, 37$	32
Figura 4.9 – Árvores com 3 e 4 vértices.	33

Lista de Tabelas

Tabela 4.1 – Número de operações elementares.....	32
Tabela 4.2 – Recorrências dos métodos.....	34

1.1 Motivação do Tema

A distribuição de energia elétrica no Brasil possui regulamentações e normas técnicas que devem ser seguidas, objetivando-se excelência na prestação do serviço [ANEEL, 2022]. De acordo com [GARCIA et al., 2003], o sistema de distribuição de energia elétrica é a parte do sistema de potência que abrange desde as subestações rebaixadoras até os transformadores (sistema de distribuição primário) e destes até a entrada elétrica dos consumidores (sistema de distribuição secundário).

Considerando fatores impeditivos ao controle do Estado em planejar ou remanejar novas redes de distribuição que atue em regiões menos favorecidas, em muitas vezes, uma porcentagem da população toma a iniciativa na aquisição de serviços públicos, ainda que de maneira temporária. No caso da eletricidade, por vezes, seu fornecimento se dá por meio de conexões não regulares, o chamado popularmente de "gato", com o redirecionamento de energia do sistema elétrico principal a casas e/ou empresas inicialmente não registradas.

Com o objetivo de atender de forma padrão a população em tais regiões e, por outro lado, representando um benefício a concessionárias de energia elétrica, o estudo e a elaboração de uma nova rede de energia tornaram-se imprescindíveis, assim como o estudo do problema do redimensionamento de energia elétrica. Determinadas aplicações de algoritmos de particionamento e de localização em grafos ao planejamento e redimensionamento de redes de energia são apresentadas em [ASSIS; FRANCA; USBERTI, 2014], [GARCIA et al., 2003] e [SILVA; FRANCA; SILVEIRA, 1996].

O problema de alocação de equipamentos envolve a decisão sobre como alocar de forma otimizada recursos limitados de maneira a atender determinadas demandas, podendo incluir a minimização do custo total, a maximização da eficiência ou o equilíbrio de ambos. As técnicas utilizadas para a resolução deste tipo de problema podem incluir programação linear, algoritmos genéticos e heurísticas [GARCIA et al., 2003].

Neste trabalho, apresentamos o problema do 1-centro modificado, aplicado ao redimensionamento de uma rede de energia em árvores, assim como dois algoritmos, de tempos $O(n \log n)$ e $O(n)$, para sua solução, nos quais consideramos pesos e distâncias positivas. O trabalho inclui a apresentação de resultados computacionais obtidos na implementação dos métodos.

1.2 Escopo da Dissertação

Na presente dissertação, apresentamos um estudo de métodos existentes para a resolução de problemas de localização, dentre os quais, os métodos de [KARIV; HAKIMI, 1979a] e [MEGIDDO, 1983] para o problema do 1-centro.

O Capítulo 2 apresenta os conceitos, as definições principais em teoria dos grafos e as notações utilizadas no trabalho. São também apresentadas as classes P, NP e NP-Completo, de problemas de decisão e, por fim, é apresentado o problema do 1-centro modificado em árvores e suas generalizações.

O Capítulo 3 apresenta definições de problemas de localização em grafos, como os problemas da 1-mediana, da p -mediana, do 1-centro e do p -centro. No capítulo são também apresentados resultados de NP-Completeness referentes aos problemas estudados, e os algoritmos de Kariv e Hakimi, e de Megiddo, para a obtenção do 1-centro em árvores.

No Capítulo 4, apresentamos algoritmos para a resolução do problema do 1-centro em árvores. O primeiro algoritmo, de tempo $O(n^2)$, é apresentado em [NASCIMENTO, 2018]. Um segundo método, de tempo $O(n \log n)$, é apresentado, e segue a estratégia do centróide para a obtenção do 1-centro modificado. Os resultados que fundamentam tal método são igualmente apresentados. O terceiro método requer tempo $O(n)$ e utiliza propriedades da função a ser minimizada para a redução do problema, de uma instância de tamanho n a uma instância de tamanho $\frac{n}{2} + 2$. O capítulo também inclui a apresentação de resultados computacionais referentes à implementação dos métodos e um estudo das recorrências em linguagem C++, indicando um número menor de operações do método $O(n)$, se comparado a algoritmos de localização de grafos conhecidos. Por fim, são apresentados no capítulo variações de problemas de localização em grafos relacionadas ao problema do 1-centro modificado em árvores e aos problemas de localização abordados no estudo do redimensionamento de redes.

O Capítulo 5 é dedicado à apresentação das considerações finais e indicações de trabalhos futuros.

1.3 Objetivo Geral

Este trabalho possui como objetivo reduzir a complexidade de tempo do algoritmo para obtenção do 1-centro modificado em árvores.

1.4 Objetivos Específicos

Como objetivos específicos, temos:

- A pesquisa de técnicas existentes para a resolução de problemas de localização em árvores, tais como a aplicação de centróides (Método de Kariv e Hakimi) e procedimentos de redução (Método de Megiddo).
- Implementar algoritmos existentes e os desenvolvidos, objetivando uma primeira análise de performance dos métodos desenvolvidos.
- Apresentar problemas de localização abertos e outros correlatos ao problema do redimensionamento de redes.

Conceitos Preliminares

Neste capítulo, são apresentados conceitos, definições principais e notações utilizadas em teoria dos grafos, com o objetivo de mostrar a aplicação da teoria a um problema de alocação de equipamentos, no contexto do problema do redimensionamento de redes. Além de tais definições, são apresentadas de modo não formal as classes P, NP e NP-Completo, de problemas de decisão e exemplos de problemas dessas classes, assim como a definição do problema do 1-centro modificado em árvores e suas generalizações.

2.1 Definições Principais

Um *grafo* é um par ordenado $G = (V, E)$, onde V é um conjunto finito, não vazio, de elementos denominados *vértices* e E é um conjunto de pares não ordenados distintos de elementos de V denominados *arestas*. Se $e = (u, v)$ é uma aresta de G , dizemos que u e v são *adjacentes* ou *vizinhos* em G . Tais vértices são as *extremidades* da aresta e a aresta é *incidente* aos vértices u e v . A *ordem* do grafo G é o número de vértices de V , denotado por $|V|$. O número de arestas de E é denotado por $|E|$. Dado um grafo G qualquer, utilizamos $|V| = n$ e $|E| = m$. A *vizinhança* de um vértice v , denotada por $Adj(v)$, é o conjunto de seus vizinhos. O *grau* $d(v)$ de um vértice v é o número de vizinhos que v contém, ou o número de arestas incidentes a ele ([COGIS; ROBERT, 2003], [CORMEN et al., 2009]).

Um grafo $G' = (V', E')$ é um *subgrafo* de $G = (V, E)$ quando V' é um subconjunto de V e E' é um subconjunto de E . Um *subgrafo induzido* de um grafo $G = (V, E)$ é um subgrafo $G' = (V', E')$ cujas arestas são todas as arestas de G cujas extremidades estão em V' . Neste caso, dizemos que $G' = G[V']$ é o subgrafo de G induzido por V' . Um *caminho* de um vértice s para um vértice t em um grafo G é uma sequência $\langle v_1, \dots, v_k \rangle$ de vértices, tal que $s = v_1$, $t = v_k$ e $(v_j, v_{j+1}) \in E(G)$, para $1 \leq j < k$. O caminho é dito *simples* se todos os vértices forem distintos. O *comprimento* do caminho é dado pelo seu número de arestas. Um *ciclo* é um caminho no qual $v_1 = v_k$.

Um grafo $G = (V, E)$ é dito ser uma *árvore* se for acíclico (*i.e.* não possuir ciclos) e *conexo* (*i.e.* existir um caminho entre qualquer par de vértices). Como um resultado clássico, em qualquer grafo $G = (V, E)$, a soma dos graus de todos os seus vértices é igual a duas vezes o seu número de arestas, isto é, $\sum_{v \in V(G)} d(v) = 2|E|$. Adicionalmente, se $T = (V, E)$ é uma árvore, então $m = n - 1$.

Dizemos que um grafo $G = (V, E)$ é *ponderado* em vértices quando cada vértice possui um

peso associado a ele. Denotaremos por $w(v)$ o peso do vértice $v \in V(G)$. Para um conjunto de vértices $V' \subseteq V$, definimos o peso de V' por $w(V') = \sum_{v \in V'} w(v)$. Sendo $G' = (V', E')$ um subgrafo de $G = (V, E)$, definimos o peso de G' por $w(G') = w(V') = \sum_{v' \in V'} w(v')$. Dizemos que G é *ponderado* em arestas quando cada aresta possui um peso associado a ela, e denotaremos por $w(e)$ o peso da aresta $e \in E(G)$, conforme a Figura 2.1 abaixo:

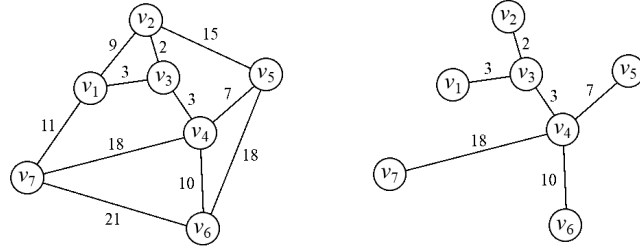


Figura 2.1 – Grafo $G = (V, E)$; Árvore $G' = (V', E')$

Dados dois vértices u e v em um grafo $G(V, E)$, denominamos *distância* entre u e v o comprimento do menor caminho entre esses dois vértices. No caso da não existência de tal caminho, consideramos a distância como infinita. Denotaremos por $d(u, v)$ a distância entre os vértices u e v (ver [CAMICIA; VICENTE, 2008]). Em um grafo conexo, distância é uma métrica, isto é, para todo vértice u, v e x de $G(V, E)$, temos:

- i. $d(u, v) \geq 0$ com $d(u, v) = 0$ se e somente se $u = v$;
- ii. $d(u, v) = d(v, u)$ ocorre apenas quando o grafo é não orientado;
- iii. $d(u, v) + d(v, x) \geq d(u, x)$.

Uma árvore T na qual determinamos um de seus vértices como vértice *raiz* é dita árvore *enraizada*. Como exemplo, na Figura 2.2 temos a representação de uma árvore T em uma implementação. Sendo v uma representação de um vértice $v \in V(T)$ na implementação, os campos para cada vértice v incluem $v.rotulo$, $v.peso$ e $v.grau$. O conjunto $v.adj[1 \dots v.grau]$ representa os vértices adjacentes a v na árvore T . Sendo $k = grau(v)$ e $T_{(v,v_1)}, T_{(v,v_2)}, \dots, T_{(v,v_k)}$ as subárvores de T obtidas pela remoção de v em T , o conjunto $v.peso_subarvore[1 \dots v.grau]$ representa os pesos $w(T_{(v,v_1)}), w(T_{(v,v_2)}), \dots, w(T_{(v,v_k)})$ das subárvores $T_{(v,v_1)}, T_{(v,v_2)}, \dots, T_{(v,v_k)}$. Observamos que tal representação requer espaço $O(n)$.

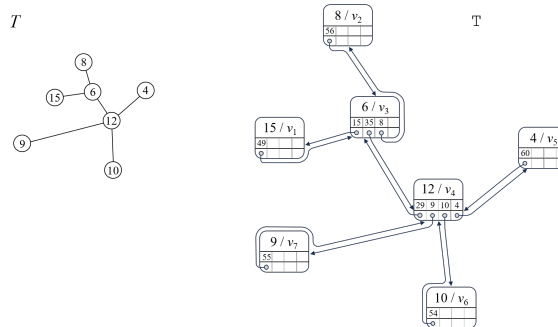


Figura 2.2 – Árvore $T = (V, E)$; Representação de T .

Para um dado vértice $v \in V(G)$, a obtenção dos pesos das subárvores pode ser realizada como segue. Se o vértice v é uma folha de T , então, para sua única subárvore $T_{(v,v_1)}$, temos

$$w(T_{(v,v_1)}) = w(T - v) = w(T) - w(v). \quad (2.1)$$

Se v não é uma folha de T , então, para cada subárvore $T_{(v,v_i)}$ de T enraizada em v_i , temos

$$w(T_{(v,v_i)}) = w(T) - w(v) - \sum_{\substack{j=1, \dots, \text{grau}(v) \\ j \neq i}} w(T_{(v,v_j)}). \quad (2.2)$$

Com as Equações 2.1 e 2.2, dado um vértice v de T , temos que o procedimento para o cálculo de $v.\text{peso_subarvore}[1..v.\text{grau}]$ inicia-se pelas folhas de T . Para cada folha v de T , o cálculo de $v.\text{peso_subarvore}[1]$ é imediato. As folhas v de T podem ser então suprimidas de T , dando origem a novas folhas em T . Para cada nova folha v de T , podemos aplicar a Equação 2.2 para o cálculo de $v.\text{peso_subarvore}[i]$. O procedimento segue sucessivamente até que não se tenha mais vértices em T . O tempo total requerido pelo procedimento é $O(n)$. Observamos que os pesos de todas as subárvores de todos os vértices em T podem igualmente ser obtidos em tempo linear pela aplicação de um algoritmo de busca em profundidade em T .

Diversos conceitos em teoria dos grafos são apresentados por [COGIS; ROBERT, 2003] e [CORMEN et al., 2009]. Em [CORMEN et al., 2009], temos as definições das classes de problemas P, NP e NP-Completo, assim como diversos resultados de NP-Completo. Em algumas demonstrações de NP-Completo de problemas de localização em grafos, temos a utilização de reduções em tempo polinomial de problemas como o problema da cobertura de vértices e o problema do conjunto dominante (ver [KARIV; HAKIMI, 1979a], [KARIV; HAKIMI, 1979b] e [GAREY; JOHNSON, 1979]).

Para a apresentação dos problemas, considerando-se um grafo $G = (V, E)$, uma *cobertura de vértices* de G é um subconjunto $V' \subseteq V(G)$ tal que se $(u, v) \in E(G)$, então ou $u \in V'$, ou $v \in V'$, ou $u, v \in V'$. Ou seja, cada vértice cobre suas arestas incidentes, e uma cobertura de vértices para G é um conjunto de vértices que cobre todas as arestas em $E(G)$. O *tamanho* de uma cobertura de vértices é dado pelo número de vértices nela. No *problema da cobertura de vértices*, devemos encontrar uma cobertura de vértices de tamanho mínimo para um dado grafo. Podemos enunciar este problema de otimização como um problema de decisão: dado um grafo $G = (V, E)$ e um tamanho k , o grafo G possui uma cobertura de vértices de tamanho k . Como um exemplo, o grafo G da Figura 2.3 possui o conjunto $\{v_2, v_3, v_4, v_5\}$ como uma de suas coberturas de vértices, a qual não é a de menor tamanho.

Dado um grafo $G = (V, E)$, um *conjunto dominante* de G é um subconjunto $V' \subseteq V(G)$ tal que, para todo $u \in V(G)$, ou $u \in V'$, ou um vizinho $v \in V'$. Ou seja, cada vértice está em V' ou possui um vizinho em V' . O *tamanho* de um conjunto dominante é dado pelo seu número de vértices. O *número de dominação* $\gamma(G)$ de G é o número de vértices no menor conjunto dominante para G . Na versão de decisão do *problema do conjunto dominante*, para um dado grafo $G = (V, E)$ e um tamanho k , devemos decidir se o grafo possui um conjunto dominante de tamanho no máximo k . Como problema de otimização, dado um grafo $G = (V, E)$, devemos

G

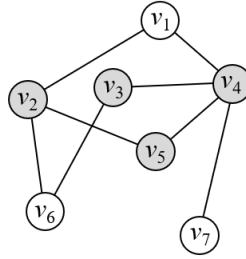


Figura 2.3 – Cobertura de vértices de $G = (V, E)$.

determinar $\gamma(G)$. Como um exemplo, o grafo G da Figura 2.4 possui o conjunto dominante $\{v_1, v_2\}$, o qual não é uma cobertura de vértices.

G

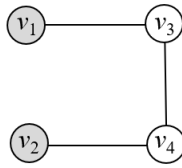


Figura 2.4 – Conjunto dominante de $G = (V, E)$.

2.2 NP-Compleitude

Segundo [CORMEN et al., 2009], problemas podem ser classificados como *tratáveis* se puderem ser solucionados em tempo polinomial, e como *intratáveis* se requererem tempo super-polinomial para a obtenção de uma solução. O *status* da classe dos problemas NP-Completo é desconhecido. Não são conhecidos algoritmos de tempo polinomiais para a solução de um problema NP-Completo. Por outro lado, não existe uma prova de limite inferior de tempo super-polinomial para qualquer problema desta classe.

A classe P, informalmente, consiste dos problemas que podem ser resolvidos em tempo polinomial. Especificamente, por [CORMEN et al., 2009], são problemas que podem ser solucionados em tempo $O(n^k)$, para alguma constante k , onde n é o tamanho da entrada para o problema. A classe NP consiste dos problemas que são “verificáveis” em tempo polinomial. Isto é, se possuímos um “certificado” de uma solução, então podemos verificar que o certificado está correto em tempo polinomial no tamanho da entrada do problema. Um problema em P está também em

NP, pelo fato de que se o problema está em P então ele pode ser resolvido em tempo polinomial sem ser fornecido um certificado. Isto é, temos $P \subseteq NP$. Contudo, a questão $P = NP$ permanece em aberto. De modo não formal, um problema é NP-Completo se está em NP e é ao menos tão “difícil” quanto qualquer problema em NP. Por [CORMEN et al., 2009], se algum problema NP-Completo puder ser solucionado em tempo polinomial, então todo problema em NP possui um algoritmo de tempo polinomial. Uma demonstração de que um problema é NP-Completo pode representar uma boa evidência de sua intratabilidade. Neste caso, em lugar de buscarmos um algoritmo eficiente para a solução exata de um problema, podemos buscar desenvolver um algoritmo de aproximação ou ainda resolver um caso especial tratável. (Determinados problemas NP-Difíceis para a classe geral de grafos podem ser resolvidos eficientemente quando restritos a árvores. Como exemplo, pode ser mostrado que se T é uma árvore com $n \geq 2$ vértices, então seu número cromático é dado por $\chi(T) = 2$. Contudo, para a classe geral de grafos, a determinação de $\chi(G)$ é um problema NP-Difícil [NASCIMENTO, 2018].)

Em um *problema de otimização*, cada solução viável possui um valor associado, e buscamos encontrar uma solução viável com o melhor valor. A NP-Compleitude aplica-se diretamente a *problemas de decisão*, para os quais a resposta é “sim” ou “não”. Apesar deste fato, podemos fazer uso da relação entre problemas de decisão e problemas de otimização. Como um exemplo de problema de decisão e seu problema de otimização correlato, temos o problema de decidirmos se existe uma coloração válida dos vértices de um dado grafo $G = (V, E)$ que utilize no máximo k cores (sendo k um número de cores dado), e o problema de encontrarmos $\chi(G)$. Podemos resolver este problema de decisão encontrando o valor de $\chi(G)$, e então comparando $\chi(G)$ com o parâmetro k do problema de decisão. Por [CORMEN et al., 2009], o problema de decisão é em um sentido *mais fácil*, ou pelo menos *não mais difícil* do que o problema de otimização. Expressando este fato de forma relevante à NP-Compleitude, se podemos fornecer evidência de que um problema de decisão é difícil, também fornecemos evidência de que seu problema de otimização relacionado é difícil [CORMEN et al., 2009].

A noção de que um problema não é mais fácil ou mais difícil do que outro pode ser aplicada quando ambos os problemas são problemas de decisão, e pode ser utilizada em quase todas as provas de NP-Compleitude. Seja A um problema de decisão, que desejamos solucionar em tempo polinomial. Seja B um problema de decisão que já saibamos como resolver em tempo polinomial. Um *algoritmo de redução* é um procedimento de tempo polinomial que transforma qualquer instância α do problema de decisão A em uma instância β do problema de decisão B , de forma que suas respostas sejam iguais. Isto é, a resposta para α é “sim” se e somente se a resposta para β for “sim”. Desta forma, para resolvermos o problema A em tempo polinomial, dada uma instância α do problema A , utilizamos o algoritmo de redução em tempo polinomial para transformá-la em uma instância β do problema B . Em seguida, executamos o algoritmo de decisão em tempo polinomial para B , tendo como entrada a instância β . Por fim, utilizamos a resposta para β como resposta para α . Por outro lado, poderíamos utilizar a técnica para mostrarmos a inexistência de algoritmo de tempo polinomial para um problema B . Primeiramente, podemos supor um problema de decisão A para o qual não possa existir nenhum algoritmo de tempo polinomial. Supomos, além disso, que temos uma redução em tempo polinomial que transforma instâncias de A em instâncias de B . Podemos utilizar, então, uma prova por contradição para mostrar que nenhum algoritmo de tempo polinomial pode existir para B . (Se houvesse um algoritmo de tempo polinomial para B , poderíamos utilizar a redução em tempo polinomial para transformar uma instância α de A em uma instância β de B , obtendo uma resposta para α também em tempo polinomial.) Como apresentado por [CORMEN et al., 2009], em uma demonstração de NP-Compleitude, não assumimos a inexistência de um algo-

ritmo de tempo polinomial para o problema A . Temos a demonstração de que o problema B é NP-Completo com a suposição de que o problema A também seja NP-Completo.

2.3 O Problema do 1-Centro Modificado

Conforme [RABUSKE, 1992], a teoria dos grafos proporciona ferramentas simples e poderosas para a construção de modelos e resolução de problemas em matemática discreta. Baseada na simples idéia de pontos interligados por linhas, a teoria dos grafos combina estes elementos em um rico sortimento de formas e dota estas propriedades com características flexíveis, tornando-se uma ferramenta útil no estudo de diversos tipos de sistemas.

No estudo do problema do redimensionamento de redes, temos a aplicação de conceitos e algoritmos de localização em grafos em etapas que incluem, por exemplo, a obtenção de soluções iniciais para a posterior aplicação de outras técnicas (ver [GARCIA et al., 2003], [NASCIMENTO, 2018]). Para a definição do problema do 1-centro modificado, apresentado por [NASCIMENTO, 2018], consideramos primeiramente mapas de localização de área das comunidades ou o mapa de rotas. Após a definição da área de estudo, temos sua representação por um mapa de estudo, ver Figura 2.5, em desenho técnico (incluindo cabeamento, postes, transformadores, distância entre os postes, quantidade de consumo (kVA) em cada poste pertencente às vias da comunidade estudada, etc.).

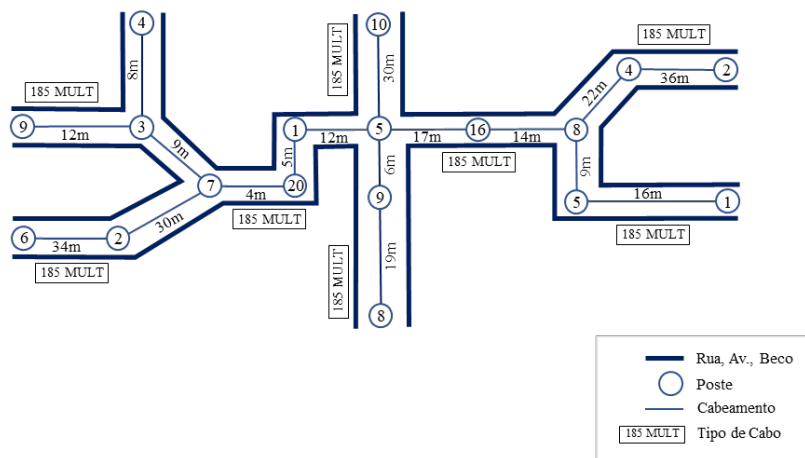


Figura 2.5 – Mapa de estudo.

Conforme a Figura 2.6 podemos observar uma rede de distribuição em árvore com n vértices e $n - 1$ arestas, os vértices da árvore representam os postes e as arestas o cabeamento.

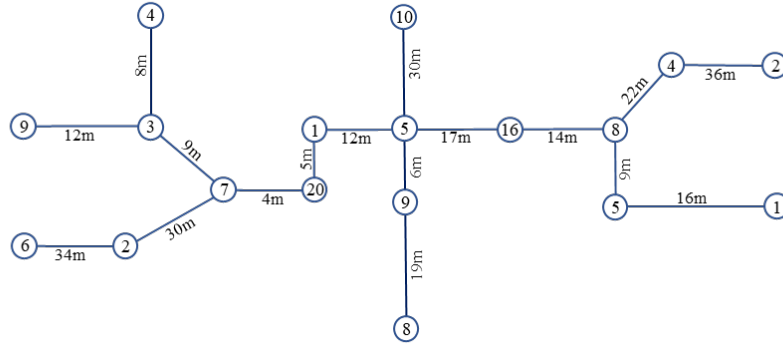


Figura 2.6 – Representação da rede de distribuição.

Seja $T = (V, E)$ uma árvore ponderada, com uma função peso $k(v_i)$ associando a cada vértice $v_i \in V$ um real positivo, e uma função distância $d(e_k)$ associando a cada aresta $e_k \in E$ um real positivo. Sendo $e_k = (v_i, v_j)$, denotaremos por $d(e_k) = d(v_i, v_j)$ a distância entre v_i e v_j .

Seja $V' \subseteq V$. A função peso k pode ser estendida ao conjunto de vértices V' como a soma dos pesos dos vértices em V' . Isto é, temos

$$k(V') = \sum_{v_i \in V'} k(v_i). \quad (2.3)$$

Seja $T' = (V', E')$ uma subárvore de $T = (V, E)$. Denotamos por $k(T') = k(V')$ o peso da subárvore T' . Para uma aresta (v_i, v_j) de $T = (V, E)$, seja $T(v_i)$ a árvore T enraizada em v_i e seja $T(v_i, v_j)$ a subárvore de $T(v_i)$ enraizada em v_j . Para tal subárvore, a notação $T_{(v_i, v_j)} = T(v_i, v_j)$ será igualmente utilizada.

No estudo do problema do 1-centro modificado, consideramos uma rede secundária de energia, dada por uma árvore ponderada $T = (V, E)$. Por [NASCIMENTO, 2018], consideramos uma fonte fixa de energia t a ser implantada em um vértice da árvore, denominada *transformador*. No problema, devemos definir a localização de um transformador, que minimize a queda de tensão máxima na rede. Por [NASCIMENTO, 2018], quanto maior for o comprimento do condutor de energia, maior será a queda de tensão, consequente do aumento de resistência elétrica.

Para a árvore $T = (V, E)$, caso o transformador seja instalado em um vértice $v_i \in V(T)$, devemos calcular a *queda de tensão total* para cada vértice $v_j \in T$, com $v_j \neq v_i$. A *queda de tensão total* de v_i para um vértice v_j é definida como sendo a soma das quedas de tensão ocorridas em cada uma das arestas do único caminho de v_i a v_j em T .

Para a definição da *queda de tensão* em uma aresta $(v_j, v_k) \in E(T)$, consideraremos o peso $k(v_i)$ como a demanda de energia no vértice $v_i \in T$. Assim, definimos a *queda de tensão* em uma aresta $(v_j, v_k) \in E(T)$, denotada por $q(v_j, v_k)$, por

$$q(v_j, v_k) = \frac{d(v_j, v_k) * k(T(v_j, v_k)) * \mu(v_j, v_k)}{100}. \quad (2.4)$$

onde

- $d(v_j, v_k)$ é a distância de v_j a v_k ,
- $k(T(v_j, v_k))$ é o peso da subárvore $T(v_j, v_k)$, ou *demanda acumulada* de $T(v_j, v_k)$, e
- $\mu(v_j, v_k)$ é a constante do material condutor utilizado no trecho entre v_j e v_k .

Conforme apresentado na Figura 2.7 abaixo:

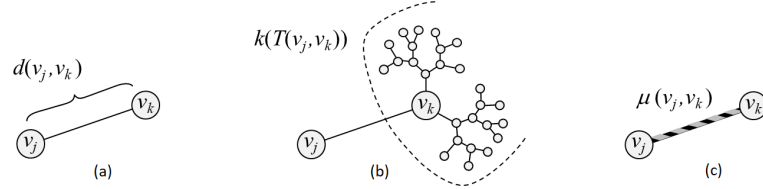


Figura 2.7 – Definição do cálculo de queda de tensão. (a) Distância entre os vértices v_j e v_k . (b) Demanda acumulada em uma subárvore $T(v_j, v_k)$. (c) Constante do material utilizado no trecho (v_j, v_k) .

Observamos que, para uma aresta $(v_j, v_k) \in E(T)$, temos $d(v_j, v_k) = d(v_k, v_j)$ e $\mu(v_j, v_k) = \mu(v_k, v_j)$. Contudo, não necessariamente temos $k(T(v_j, v_k)) = k(T(v_k, v_j))$. Seja $P(v_i, v_j) = \langle v_i, u_1, u_2, \dots, u_p, v_j \rangle$ o caminho em T do vértice v_i em que se localiza o transformador a um vértice qualquer de T . Definimos a *queda de tensão total* entre v_i e v_j por

$$Q(v_i, v_j) = q(v_i, u_1) + q(u_1, u_2) + \dots + q(u_p, v_j) = \sum_{(v_k, v_l) \in P(v_i, v_j)} q(v_k, v_l). \quad (2.5)$$

Isto é, $Q(v_i, v_j)$ é a soma das quedas de tensão ocorridas nos trechos ao longo do caminho $P(v_i, v_j)$. Se $Q(v_i, v_j)$ representa uma queda máxima em alguma subárvore de $T(v_i)$, o caminho $\langle v_i, u_1, u_2, \dots, u_p, v_j \rangle$, é dito *caminho de queda máxima*. Para $v_i \in V(T)$, seja

$$\Phi(v_i) = \max_{v_k \in V(G)} \{Q(v_i, v_k)\} \quad (2.6)$$

a *queda de tensão total máxima* a partir de v_i , isto é, a queda de tensão total máxima considerando-se um transformador em v_i . O *1-centro modificado* de T é um vértice $v^* \in V(T)$ que minimize esta queda de tensão total máxima, isto é, um vértice tal que

$$\Phi(v^*) = \min_{v \in V(T)} \{\Phi(v)\}. \quad (2.7)$$

Como exemplos de algoritmos de localização em árvores, estão o algoritmo de Kariv e Hakimi, e o algoritmo de Megiddo, para a obtenção do 1-centro (clássico) em árvores. No Capítulo 3, temos a descrição dos métodos, e de suas complexidades, como um estudo das técnicas existentes em árvores.

No Capítulo 4, apresentamos métodos para a obtenção do 1-centro modificado em árvores. Especificamente, apresentamos um algoritmo de tempo $O(n)$ com o menor número de operações, se comparado a algoritmos clássicos para o problema do 1-centro em árvores.

Problemas de Localização

Em [NASCIMENTO, 2018], foram estudados problemas de localização correlatos ao problema do redimensionamento de redes, abordado no presente trabalho. Tais problemas, pesquisados por [KARIV; HAKIMI, 1979a], [HAKIMI; SCHMEICHEL; PIERCE, 1978] e [KARIV; HAKIMI, 1979b], contextualizaram o presente trabalho, e representaram abordagens iniciais para a obtenção de novos métodos para problemas de localização. Neste capítulo, apresentamos as definições dos problemas de localização clássicos, os principais resultados de NP-Completeness em problemas de localização, assim como os métodos de [KARIV; HAKIMI, 1979b], e de [MEGIDDO, 1983], para a resolução do problema do 1-centro em árvores.

3.1 Definições de Problemas

As definições de problemas como o da 1-mediana, p -mediana, 1-centro, p -centro e conjuntos dominantes de raio r em redes, nos permitem avaliar suas possíveis aplicações ao problema do redimensionamento de energia, podendo incluir grafos ponderados e não ponderados, e problemas em vértices ou absolutos.

No *problema da 1-mediana* (ou *1-median problem*) em um grafo, devemos encontrar um vértice que minimiza a soma das distâncias de caminhos mínimos ponderadas do próprio vértice a todos os outros vértices, cada um associado a um peso positivo, como representado na Figura 3.1 abaixo:

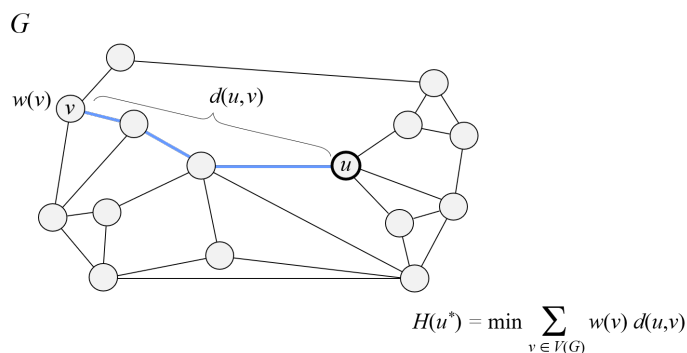


Figura 3.1 – Problema da 1-mediana em grafos.

Por [BURKARD; KRARUP, 1998], para problemas NP-Difíceis definidos sobre um grafo, casos especiais bem resolvidos podem ocorrer se, por exemplo, o grafo possui certas propriedades. Como um dos casos especiais bem resolvidos, o problema da 1-mediana em uma árvore foi considerado primeiramente por [HUA, 1961]. Em [GOLDMAN, 1962] um algoritmo simples foi desenvolvido para este problema. Como uma observação, se uma aresta (u, v) é removida na árvore, obtemos duas árvores T_1 e T_2 . Seja W_i o peso de todos os vértices na árvore T_i , para $i = 1, 2$. Então a localização ótima se situa na árvore com o maior peso acumulado. Esta observação baseia o algoritmo apresentado em [GOLDMAN, 1962] e faz uso do fato de que todos os pesos são não negativos, mas não é válida se pesos de vértices negativos são permitidos.

Conforme a Figura 3.2, o *problema da p -mediana* (ou *p -median problem*) é o problema de identificarmos um subconjunto X_p de p vértices que minimizam a soma das distancias de caminhos mínimos ponderadas de cada outro vértice no grafo ao vértice mais próximo em X_p .

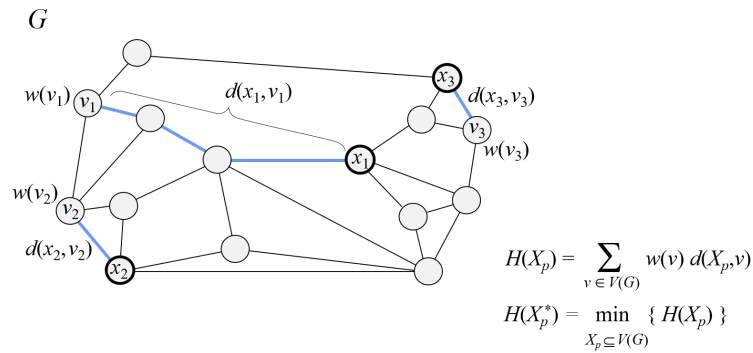


Figura 3.2 – Problema da p -mediana em grafos.

Por [KARIV; HAKIMI, 1979b], o problema da p -mediana em um grafo é NP-Difícil. O problema permanece NP-Difícil mesmo quando a rede possui uma estrutura simples, como por exemplo um grafo planar com o grau de vértice máximo igual a 3 [KARIV; HAKIMI, 1979b]. Contudo, os autores apresentaram resultados que conduziram a algoritmos eficientes quando a rede é uma árvore. Em particular, [KARIV; HAKIMI, 1979b] mostraram que a 1-mediana de uma árvore é idêntica a seu w -centróide. O trabalho também apresentou um algoritmo que encontra a p -mediana de uma árvore (para $p > 1$) em tempo $O(n^2 p^2)$. Em [TAMIR, 1996] foi apresentado um algoritmo em tempo $O(pn^2)$ para o problema. Em [DASKIN; MAASS, 2015] são apresentados alguns resultados existentes na literatura para o problema clássico.

O *problema do 1-centro* (ou *1-center problem*), representado na Figura 3.3, pode ser definido como segue: dado um conjunto de n pontos de demanda, um espaço de localizações viáveis para um serviço (ou uma instalação) e uma função para calcular o custo de transporte entre um serviço (ou instalação) e qualquer ponto de demanda, encontrar um posicionamento ótimo do serviço que minimize o custo máximo de transporte do serviço ao ponto de demanda.

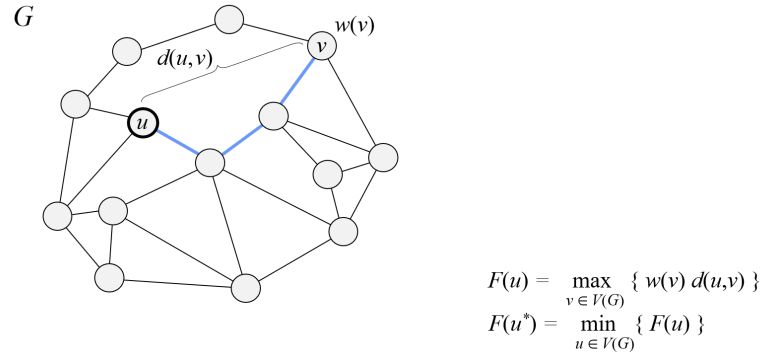


Figura 3.3 – Problema do 1-centro em grafos.

Na Figura 3.4 podemos observar que dado um conjunto $X_p = x_1, \dots, x_p$ de p pontos em um grafo $G = (V, E)$, a distância $d(X_p, v_j)$ entre X_p e um nó v_j é computada como $\min_{i=1, \dots, p} d(x_i, v_j)$.

Seja

$$F(X_p) = \max_{v \in V(G)} \{ w(v) d(X_p, v) \}. \quad (3.1)$$

Seja X_p^* tal que

$$F(X_p^*) = \min_{X_p \subseteq V(G)} \{ F(X_p) \}. \quad (3.2)$$

Então X_p é chamado p -centro de G e $F(X_p^*)$ é chamado p -raio de G , sendo denotado por $r_p(G)$. Isto é, no problema do p -centro (ou p -center problem), devemos encontrar um conjunto X_p de p pontos em G tal que $\max_{j=1, \dots, n} w(v_j) \cdot d(v_j, X_p)$ seja minimizado.

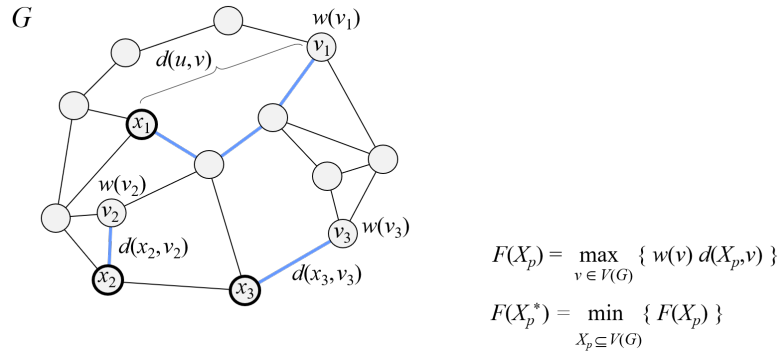


Figura 3.4 – Problema do p -centro em grafos.

O problema do 1-centro absoluto ponderado foi definido e resolvido por [HAKIMI, 1965] (ver [TANSEL; FRANCIS; LOWE, 1983a]). Para o problema do 1-centro *absoluto*, é viável que

sua solução esteja em um vértice do grafo ou em um ponto no interior de uma aresta (que não seja a extremidade de tal aresta). Em [HAKIMI; SCHMEICHEL; PIERCE, 1978] mostraram que o método de Hakimi pode ser implementado em tempo $O(|E|n^2 \log n)$. Refinamentos posteriores do procedimento foram obtidos por [KARIV; HAKIMI, 1979b], resultando em um algoritmo $O(|E|n \log n)$ para o caso ponderado e $O(|E|n)$ para o caso não ponderado.

O problema do p -centro foi formulado por [HAKIMI, 1965]. Em [KARIV; HAKIMI, 1979a] mostraram que o problema em uma rede geral é NP-Difícil. Em [HANDLER; MIRCHANDANI, 1979] foram considerados os problemas do p -centro absoluto e em vértice em uma árvore para o caso especial de $p = 2$, e foram obtidos dois algoritmos $O(n)$ similares. Em [KARIV; HAKIMI, 1979a] descreveram um algoritmo de tempo $O(n^2 \log n)$ para a obtenção do p -centro absoluto para uma árvore vértice ponderada. Em [TANSEL; FRANCIS; LOWE, 1983b] apresentaram um *survey* sobre os problemas da p -mediana e do p -centro.

3.2 NP-Completeness do Problema de Determinar os p -Centros

Em [KARIV; HAKIMI, 1979a], é mostrado que a NP-Completeness do problema do conjunto dominante implica que os problemas de encontrarmos p -centros (absolutos ou em vértices) são NP-Difíceis, mesmo quando a rede é um grafo planar com grau máximo de vértices igual a 3. Isto implica que é altamente improvável que existam algoritmos de tempo polinomial para os problemas de p -centros.

Um problema pode ser NP-Completo apenas se ele pertence ao conjunto NP. Isto é, apenas se ele é um problema de decisão para o qual um certificado pode ser verificado em tempo polinomial. Desta forma, o problema do p -centro, como um problema de otimização, não pode ser classificado como um problema NP-Completo. Contudo, Kariv e Hakimi mostram que o problema do p -centro é NP-Difícil, isto é, mostram que o problema do conjunto dominante, como um problema NP-Completo, pode ser reduzido em tempo polinomial ao problema do p -centro. Desta forma, existe um algoritmo de tempo polinomial para o problema do p -centro se e somente se $P = NP$. O seguinte lema é apresentado por Kariv e Hakimi.

Lema 1 ([GAREY; JOHNSON, 1979]). *Seja $G = (V, E)$ um grafo e seja p um inteiro, $1 \leq p \leq n$. O problema de definirmos se existe em G um conjunto dominante de cardinalidade $\leq p$ (i.e. de definirmos se o número de dominação de G é $\leq p$) é NP-Completo, mesmo no caso em que G é um grafo planar de grau de vértice máximo igual a 3.*

A prova do Lema 1 foi apresentada por [GAREY; JOHNSON, 1979], sendo baseada na NP-Completeness do problema da cobertura de vértices. Garey e Johnson provaram que o problema da cobertura de vértices geral é NP-Completo mesmo quando G é um grafo planar com grau de vértices máximo igual a 3. Na redução do problema da cobertura de vértices para o problema do conjunto dominante, substituímos cada aresta (u, v) do grafo G pela estrutura apresentada na Figura 3.5, e consideramos $p = |E| + k$.

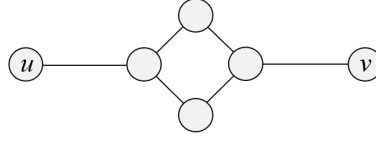


Figura 3.5 – Redução do problema da cobertura de vértices para o problema do conjunto dominante.

Os seguintes teoremas são apresentados por Kariv e Hakimi.

Teorema 1 ([KARIV; HAKIMI, 1979a]). *Os problemas do p -centro em vértices e do p -centro absoluto são NP-Difíceis, mesmo no caso onde a rede é um grafo planar não ponderado em vértices, de grau máximo igual a 3, e com as arestas com comprimento 1.*

Teorema 2 ([KARIV; HAKIMI, 1979a]). *Os problemas do p -centro em vértices e do p -centro absoluto são NP-Equivalentes.*

Em [KARIV; HAKIMI, 1979b], é mostrado que a NP-Compleitude do problema do conjunto dominante implica igualmente que o problema da p -mediana é NP-Completo, mesmo quando a rede é um grafo planar com grau máximo de vértices igual a 3. O seguinte teorema é apresentado por Kariv e Hakimi.

Teorema 3 ([KARIV; HAKIMI, 1979b]). *O problema de encontrar uma p -mediana é NP-Difícil, mesmo no caso onde a rede é um grafo planar com grau máximo igual a 3, com as arestas com comprimento 1 e todos os vértices com peso 1.*

A demonstração do Teorema 3 apresentada por Kariv e Hakimi, igualmente utiliza uma redução em tempo polinomial do problema do conjunto dominante de cardinalidade p . Seja o problema de decisão derivado do problema da p -mediana, no qual, dado um grafo $G = (V, E)$, um inteiro p ($1 < p < n$), e um valor real positivo h , devemos decidir se existe um subconjunto V_p^* com p vértices de V , tal que $H(V_p^*) = \sum_{v \in V} w(v)d(v, V_p^*) \leq h$. Como uma observação apresentada por Kariv e Hakimi, este problema pertence a NP, e pela demonstração do Teorema 3, o problema é NP-Difícil. Portanto, o problema de decisão é NP-Completo.

Como apresentado no capítulo anterior, o estudo do redimensionamento é concluído com êxito no momento em que encontramos um ponto ótimo na rede atendendo a todas as restrições. Apresentaremos a seguir os métodos devidos a Kariv e Hakimi, e a Megiddo, para a resolução do problema do 1-centro clássico, em árvores. Observamos que, como uma primeira abordagem, consideramos a rede particionada para realizar em seguida a busca pelo melhor ponto de localização.

3.3 Método de Kariv e Hakimi

Centróide de uma Árvore

Em [KARIV; HAKIMI, 1979a] é apresentado um método para a obtenção do 1-centro absoluto (ou em vértice) de uma árvore T , com a utilização do método do centróide. Seja $v \in V(T)$, seja d_v o grau de v em T , e sejam $T_{v,1}, T_{v,2}, \dots, T_{v,d_v}$ as subárvores de $T - v$. Denote por $T_{v,i}^+$ a subárvore que consiste de $T_{v,i}$, o vértice v , e a aresta que conecta v a $T_{v,i}$. O lema e o corolário a seguir são apresentados por Kariv e Hakimi, e definem, respectivamente, a subárvore contendo o 1-centro de T , e uma condição para a definição do 1-centro de T .

Lema 2 ([KARIV; HAKIMI, 1979a]). *Seja $v \in V$ um vértice fixo e seja \tilde{v} um vértice tal que $w(\tilde{v}).d(\tilde{v}, v) = \max_{v' \in V} \{w(v').d(v', v)\}$. Seja $T_{v,l}$ a subárvore de $T - v$ à qual \tilde{v} pertence. Então, o 1-centro de T está em $T_{v,l}^+$.*

Corolário 1 ([KARIV; HAKIMI, 1979a]). *Sejam \tilde{v} e \hat{v} dois vértices tais que $\tilde{v} \in T_{v,l}$, $\hat{v} \in T_{v,k}$, $k \neq l$, e $w(\tilde{v}).d(\tilde{v}, v) = w(\hat{v}).d(\hat{v}, v) = \max_{v' \in V} \{w(v').d(v', v)\}$. Então v é o 1-centro de T .*

Por uma observação apresentada por Kariv e Hakimi, se o vértice v no Lema 1 não é uma folha de T , então $T_{v,l}^+$ é uma subárvore própria de T . Desta forma, em um algoritmo baseado no lema para a obtenção do 1-centro de uma árvore ponderada, a árvore T_0 é considerada como sendo a árvore original. Escolhemos um vértice v_0 de T_0 (onde v_0 não é uma folha de T_0) e encontramos a subárvore T_{v_0,l_0}^+ na qual (de acordo com o Lema 1) o 1-centro está. Seja $T_1 = T_0 \cap T_{v_0,l_0}^+$. Escolhemos um vértice v_1 de T_1 (onde v_1 não é uma folha de T_1) e encontramos a subárvore T_{v_1,l_1}^+ de T na qual o 1-centro de T está. Desta forma, por Kariv e Hakimi, o 1-centro precisa estar na subárvore $T_2 = T_1 \cap T_{v_1,l_1}^+$. Pelo algoritmo apresentado, o processo é repetido iteradamente até obtermos uma subárvore T_k que consiste em uma única aresta. O centro local nesta aresta é o 1-centro da árvore T , e um (ou ambos) extremo(s) desta aresta é o 1-centro em vértice da árvore T .

Por uma análise apresentada por Kariv e Hakimi, exceto pelo último passo de encontrarmos o centro local (ou encontrarmos qual extremo é o 1-centro em vértice), cada etapa (*i.e.*, a obtenção das subárvores T_{v_i,l_i}^+ e T_{i+1}) requer tempo $O(n)$. A complexidade total do algoritmo é $O(nk + n \lg n)$ para o 1-centro absoluto, e de $O(nk)$ para o 1-centro em vértice. (O número de etapas é dado por k , e o custo da última etapa é dado por $n \lg n$, no caso do 1-centro absoluto. No caso do 1-centro em vértice, a última etapa requer tempo $O(n)$). Por Kariv e Hakimi, o valor de k depende do modo no qual escolhemos o vértice v_i a cada etapa i do algoritmo. Para uma boa escolha de v_i , é apresentada a noção de centróide de uma árvore.

Para um vértice v de T , considerando-se $T_{v,1}, T_{v,2}, \dots, T_{v,d_v}$ as subárvores de $T - v$, seja $|T|$ o número de vértices em T e defina $N(v)$ por

$$N(v) = \max_{1 \leq i \leq d_v} \{|T_{v,i}|\}. \quad (3.3)$$

Por [KARIV; HAKIMI, 1979a], e [HARARY, 1969], um *centróide* da árvore T é um vértice v_c para o qual $N(v)$ é mínimo, isto é,

$$N(v_c) = \min_{v \in V} \{N(v)\}. \quad (3.4)$$

Observamos que uma árvore pode ter um ou dois centróides. No último caso, os dois centróides são conectados por uma aresta [HARARY, 1969]. Adicionalmente, observamos que $N(v_c) \leq \lfloor n/2 \rfloor$. Mais exatamente, o número de vértices em cada uma das subárvores $T_{v_c,1}, T_{v_c,2}, \dots, T_{v_c,d_{v_c}}$ não é maior do que $\lfloor n/2 \rfloor + 1$. No algoritmo de Kariv e Hakimi, se a cada etapa escolhemos um centróide de T_i como sendo o vértice v_i , então $|T_{i+1}| \leq \lfloor |T_i|/2 \rfloor + 1$, e o número k de etapas será $O(\log n)$. Desta forma, objetivando a apresentação de um algoritmo $O(n \log n)$ para a obtenção do 1-centro de uma árvore, Kariv e Hakimi forneceram um algoritmo $O(n)$ para a obtenção do centróide de uma árvore.

Por [KARIV; HAKIMI, 1979a], a desigualdade $N(v_c) \leq \lfloor n/2 \rfloor$ é uma condição necessária e suficiente para que um vértice v_c seja um centróide da árvore. Com base nesta propriedade, uma versão do algoritmo de [GOLDMAN, 1971] pode ser utilizada para encontrarmos um centróide de uma árvore em $O(n)$ passos. Na execução do algoritmo, utilizamos uma cópia T' da árvore original T como uma árvore auxiliar sobre a qual o algoritmo funciona. A cada vértice v da árvore, temos a definição de uma variável $n(v)$. Durante o algoritmo, se v é uma folha de T' , então $T' - v$ está contida em uma das subárvores de $T - v$, e $n(v)$ fornece então o número de vértices desta subárvore.

```

CENTRÓIDE( $T$ )
   $T' = T$  // Inicialização
  para cada vértice  $v \in T'$  faça // Inicialização
     $n(v) = n - 1$ ;
  enquanto a árvore auxiliar  $T'$  não consistir de um único vértice  $v_0$  faça
    seja  $v$  uma folha da árvore auxiliar  $T'$ ;
    se  $n(v) \leq \lfloor n/2 \rfloor$  então
      PARE //  $v$  é um centróide da árvore original  $T$ ;
    senão
      seja  $u$  o vértice adjacente a  $v$  em  $T'$ ;
       $n(u) = n(u) - (n - n(v))$ ;
      Remova o vértice  $v$  (e a aresta  $(u, v)$ ) de  $T'$ ;
  RETORNE o vértice  $v_0$  como um centróide de  $T$ ;

```

Em [KARIV; HAKIMI, 1979a], a prova detalhada da validade do algoritmo CENTRÓIDE, ou a prova de que ele requer tempo $O(n)$ para a obtenção do centróide de uma árvore T , é deixada ao leitor. Com o algoritmo CENTRÓIDE, Kariv e Hakimi apresentam o algoritmo 1-CENTRO-ARVORE para a obtenção do 1-centro de uma árvore. No algoritmo, as variáveis T' , T'' , e T''' representam a cada etapa as subárvores T_i , T_{v_i, l_i}^+ , e T_{v_i, l_i} , respectivamente. O algoritmo faz uso de uma sub-rotina para a obtenção do centro-local sobre uma aresta e de T .

Um *centro-local* de um grafo G sobre uma aresta $e \in E(G)$ é um ponto $x^*(e)$ sobre e , tal que $F(x^*(e)) = \min_{x(e) \text{ sobre } e} \{F(x(e))\}$, e $F(x^*(e))$ é chamado o *raio-local* de G sobre e . Kariv e Hakimi apresentam um algoritmo de tempo $O(n \log n)$ para a obtenção do centro-local de um grafo G sobre uma aresta $e \in E(G)$.

1-CENTRO-ARVORE(T)

1. $T' = T$; // Inicialização
2. **se** T' possui uma única aresta e **então** // Para 1-centro absoluto
 Encontre o centro-local x^* de T sobre e e PARE ; // x^* é o 1-centro de T
 se T' possui uma única aresta $e = (v_r, v_s)$ **então** // Para 1-centro em vértice
 Sejam:
 $d_r = \max_{v' \in V} \{w(v').d(v', v_r)\}$;
 $d_s = \max_{v' \in V} \{w(v').d(v', v_s)\}$;
 se $d_r < d_s$ **então** v_r é o 1-centro em vértice de T ;
 se $d_r > d_s$ **então** v_s é o 1-centro em vértice de T ;
 se $d_r = d_s$ **então** cada um de v_r e v_s é o 1-centro em vértice de T ;
 PARE ;
3. $v_c = \text{CENTRÓIDE}(T')$;
4. Seja \tilde{v} um vértice de T tal que $w(\tilde{v}).d(\tilde{v}, v_c) = \max_{v' \in V} \{w(v').d(v', v_c)\}$;
 Seja T''' a componente de $T - v_c$ que contém \tilde{v} ;
 Seja $T'' = T''' +$ o vértice $v_c +$ a aresta que conecta v_c a T''' ;
5. **se** existe um vértice \bar{v} tal que $\bar{v} \notin T''$ e $w(\bar{v}).d(\bar{v}, v_c) = w(\tilde{v}).d(\tilde{v}, v_c)$ **então**
 v_c é o 1-centro em vértice de T ;
 PARE ;
6. $T' = T' \cap T''$;
 Retorne a 2.

Por uma observação em [KARIV; HAKIMI, 1979a], na condicional do algoritmo, se $n(v)$ é o número de vértices da subarvore de $T - v$ à qual u pertence, então $n - n(v)$ é o número de vértices da subarvore de $T - u$ que contém v . Portanto, $n(u)$ é corretamente atualizado, *i.e.* quando u se torna uma folha, $n(u)$ fornece o número de vértices da subarvore de $T - u$ que contém $T' - u$. Adicionalmente, se a relação $n(v) \leq \lfloor n/2 \rfloor$ não é válida, então v não pode ser um centróide e sua remoção de T' ainda preserva o(s) centróide(s) de T em T' . Ver [KARIV; HAKIMI, 1979a].

3.4 Método de Megiddo

Sendo $T = (V, E)$ uma árvore com n vértices, com um comprimento não negativo $d(v_i, v_j)$ associado a cada aresta (v_i, v_j) e um peso não negativo $w(v_i)$ associado a cada vértice v_i , em [MEGIDDO, 1983] temos a apresentação de uma abordagem ao problema do 1-centro análoga à abordagem por Kariv e Hakimi. Uma aresta (v_i, v_j) é identificada com um segmento de reta de comprimento $d(v_i, v_j)$, de forma que podemos considerar qualquer ponto sobre a aresta (v_i, v_j) . Por Megiddo, um ponto $x = (v_i, v_j; t)$ é caracterizado como estando localizado a uma distância t de v_i , e a uma distância $d(v_i, v_j) - t$ de v_j . Portanto a distância $d(x, y)$ entre quaisquer dois pontos sobre a árvore está bem definida, a saber, é o comprimento do único caminho de x a y na árvore. O *centro ponderado* de T é um ponto x que minimiza a função

$$r(x) = \max \{w(v_i)d(x, v_i) : v_i \in V\}. \quad (3.5)$$

O centro absoluto é único a menos que todos os pesos sejam iguais a zero. O problema relacionado de encontrarmos um vértice v_j que minimize a função $r(x)$, *i.e.*, o ponto x estando restrito a ser um vértice da árvore, é também apresentado por [MEGIDDO, 1983], assim como um algoritmo de tempo linear para o caso ponderado geral.

Observações Iniciais

Por [MEGIDDO, 1983], a função $r(x)$ é convexa sobre qualquer caminho simples da árvore. Isto é, se P é um caminho simples e v_i é qualquer vértice, então considere o vértice v_j que esteja sobre o caminho P e seja mais próximo de v_i . O vértice v_j particiona o caminho em duas partes sobre cada uma das quais a função

$$gv_i(x) = d(x, v_i) \quad (3.6)$$

é linear e crescente à medida em que nos distanciamos de v_j . Desta forma, $gv_i(x)$ é linear por partes sobre P (com no máximo duas partes) e convexa. A função $r(x) = \max\{w(v_i)d(x, v_i) : v_i \in V\}$ é portanto linear por partes e convexa, sendo o máximo de funções convexas.

Como uma observação apresentada por Megiddo, dado qualquer ponto x , podemos determinar em tempo $O(n)$ em qual das subárvores o centro se situa, e o algoritmo de Kariv e Hakimi está fundamentado nesta observação. Dado que o centro se situa em uma subárvore T' , o procedimento verifica o centróide x de T' objetivando determinar em qual das subárvores de T' , enraizada em x , está o centro.

Visto que o centróide define subárvores cujos tamanhos são no máximo metade do tamanho de T' , o procedimento termina em $O(\log n)$ testes e portanto requer tempo $O(n \log n)$. A melhoria sugerida por [MEGIDDO, 1983] está na redução do custo de um teste. No algoritmo apresentado por [MEGIDDO, 1983], também são efetuados $O(\log n)$ testes. Contudo, o custo de cada teste não é maior do que três quartos do custo do teste precedente.

O Algoritmo de Tempo Linear

Seja v_c denotando o centróide de T , isto é, v_c é um vértice tal que para todo vértice adjacente v_j , temos $|T_{v_c, v_j}| \leq n/2$ (onde n é o número de vértices de T). Seja

$$r_{v_j}(v_c) = \max\{w(v_i)d(v_c, v_i) : v_i \in V(T_{v_c, v_j})\}. \quad (3.7)$$

Por [MEGIDDO, 1983] e [KARIV; HAKIMI, 1979a], o centróide v_c pode ser encontrado em tempo $O(n)$. Então, supondo-se que v_c seja conhecido, primeiramente, calculamos $r_{v_j}(v_c)$, para cada vértice v_j adjacente a v_c , o que pode ser feito em tempo $O(n)$. Se existem dois vértices v_{j_1}, v_{j_2} ($v_{j_1} \neq v_{j_2}$) adjacentes a v_c , tais que $r_{v_{j_1}}(v_c) = r_{v_{j_2}}(v_c) = r(v_c)$, então o próprio v_c é o

centro. Caso contrário, sendo v_j adjacente a v_c e $r_{v_j}(v_c) > r_{v_k}(v_c)$ para todo outro vértice v_k adjacente a v_c , sabemos que o centro situa-se na subárvore T_{v_c, v_j} .

Seja u um vértice não pertencente à subárvore T_{v_c, v_j} . Se o centro x está em T_{v_c, v_j}^+ a uma distância t a partir de v_c , então $d(u, x) = d(u, v_c) + t$. A seguir na Figura 3.6, temos o centróide v_c da árvore T , com um vértice que não está em T_{v_c, v_j} , com um vértice x em T_{v_c, v_j} a uma distância t a partir de v_c .

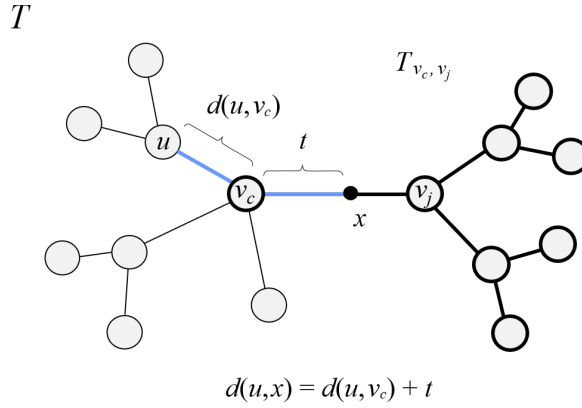


Figura 3.6 – Distância $d(u, x)$.

Conforme a Figura 3.7, sejam a e b vértices não pertencentes a T_{v_c, v_j} . Iremos assumir, sem perda de generalidade, que

$$w(a)d(a, v_c) \geq w(b)d(b, v_c). \quad (3.8)$$

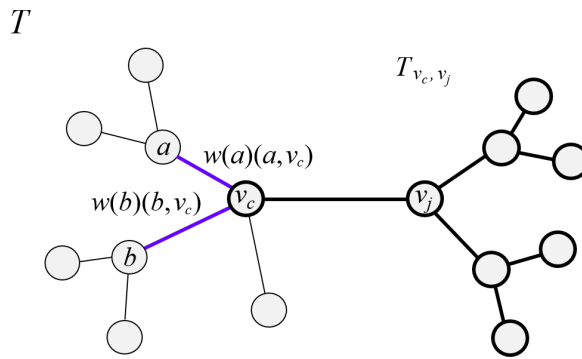


Figura 3.7 – Vértices a e b .

Resolvendo (para t) a equação

$$w(a)(d(a, v_c) + t) = w(b)(d(b, v_c) + t), \quad (3.9)$$

podemos afirmar que existe um valor t_{ab} (com $0 \leq t_{ab} \leq \infty$) tal que, para todo x em T_{v_c, v_j}^+ a uma distância t de v_c , temos

$$w(a)(d(a, x)) \geq w(b)(d(b, x)) \quad \Leftrightarrow \quad 0 \leq t \leq t_{ab}. \quad (3.10)$$

Como apresentado na Figura 3.8 a seguir,

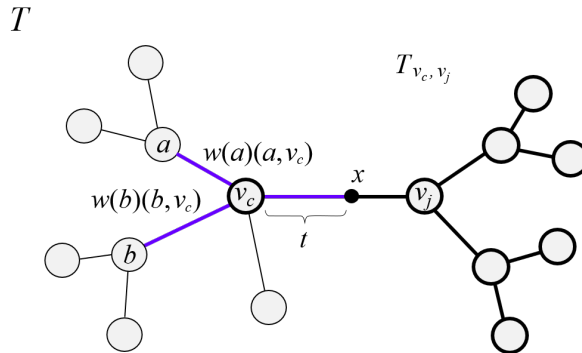


Figura 3.8 – Definição de t_{ab} .

Substituindo $d(a, x) = d(a, v_c) + t$ e $d(b, x) = d(b, v_c) + t$ na equação $w(a)(d(a, x)) = w(b)(d(b, x))$, temos

$$w(a)d(a, v_c) + w(a)t = w(b)d(b, v_c) + w(b)t \quad (3.11)$$

e

$$[w(a) - w(b)]t = w(b)d(b, v_c) - w(a)d(a, v_c). \quad (3.12)$$

Definimos, então,

$$t_{ab} = \frac{w(b)d(b, v_c) - w(a)d(a, v_c)}{w(a) - w(b)}. \quad (3.13)$$

Desta forma, se sabemos que o centro se situa a uma distância menor do que t_{ab} de v_c , então poderíamos desconsiderar o vértice b no processo de encontrar o centro. Similarmente, o vértice a poderia ser eliminado se sabemos que o centro se situa a uma distância maior do que t_{ab} a partir de v_c .

Megiddo mostra como explorar eficientemente esta observação, e como reconhecer se o centro se situa dentro de uma distância t de x ou não, onde x é qualquer vértice folha e t é qualquer número real positivo. Adicionalmente, por Megiddo, podemos decidir se o centro está dentro de uma distância de t a partir de x em tempo $O(n)$.

Considerando-se T a árvore original, seu centróide v_c e a subárvore T_{v_c, v_j}^+ a qual é conhecida conter o centro, o algoritmo de Megiddo arranja os vértices que não estão na subárvore T_{v_c, v_j}^+ em pares disjuntos $(a_1, b_1), (a_2, b_2), \dots, (a_s, b_s)$ (desconsiderando um vértice, se o número de vértices for ímpar). Observamos que existirão ao menos $(n/4) - 1$ de tais pares visto que no máximo $n/2$ vértices pertencem a T_{v_c, v_j} . Para cada par (a, b) , assumindo, sem perda de generalidade, que

$$w(a)(d(a, v_c) + t) \geq w(b)(d(b, v_c) + t), \quad (3.14)$$

se $w(a) \geq w(b)$, então o vértice b é “descartado”. Caso contrário, seja

$$t_{ab} = \frac{w(b)d(b, v_c) - w(a)d(a, v_c)}{w(a) - w(b)}. \quad (3.15)$$

O algoritmo obtém a mediana t_m de todos os valores $t_{a_i b_i}$ (para todos os pares dos quais nenhum vértice foi descartado) em tempo $O(n)$ (ver [AHO; HOPCROFT; ULLMAN, 1974] e [MEGIDDO, 1983]). Em seguida, o algoritmo verifica em tempo $O(n)$ se o centro se encontra (em T_{v_c, v_j}) dentro de uma distância de t_m de v_c . Se o centro de fato se situa dentro de uma distância de t_m de v_c , considerando um par (a, b) tal que $t_{ab} \geq t_m$, então, onde quer que o centro x^* se situe (dado que ele está em $T_j(c)$ a uma distância de não mais que t_m de c), temos $w(a)d(a, x^*) \geq w(b)d(b, x^*)$. Desta forma, por Megiddo, podemos seguramente descartar o vértice b neste caso.

Similarmente, se x^* está a uma distância maior do que t_m de v_c , então dos pares (a, b) , tais que $t_{ab} \leq t_m$, o vértice a pode ser descartado. Pela argumentação apresentada [MEGIDDO, 1983], um vértice é descartado de aproximadamente metade dos pares. Isto é, são descartados aproximadamente $1/8$ dos vértices da árvore. O problema é então reduzido ao problema do centro ponderado em uma nova árvore T' . Para cada vértice a não pertencente a T_{v_c, v_j} , e não descartado, formamos uma aresta (u, v_c) , com comprimento $d(a, v_c)$, com $w(a)$ permanecendo o mesmo como em T . Todas estas arestas são adicionadas à árvore T_{v_c, v_j} , formando a nova árvore T' .

Tendo em vista que $n/8$ vértices foram descartados, o tempo requerido $time(n)$ para uma árvore com n vértices satisfaz

$$time(n) \leq time(7n/8) + Cn. \quad (3.16)$$

Desta forma, temos $time(n) = O(n)$. Por [MEGIDDO, 1983], pela convexidade da função $r(x)$, o vértice que minimiza $r(x)$ é ou idêntico ou adjacente ao ponto no qual $r(x)$ possui seu

mínimo global. Portanto, com a obtenção do mínimo global, temos no máximo dois vértices (extremos de uma aresta) candidatos a mínimo global de $r(x)$ em vértice.

No método apresentado por Megiddo, para cada par (a, b) de vértices não pertencentes à árvore T_{v_c, v_j}^+ , são verificadas as condições para a remoção do vértice a ou do vértice b do processo de obtenção do centro de T . Esta estratégia utilizada permitiu a remoção de $n/8$ vértices da árvore T a cada etapa do algoritmo, dando origem à nova árvore T' , de menor ordem.

Para o problema do 1-centro modificado, consideremos T a árvore original, v_c o centróide de T , T_{v_c, v_j}^+ a árvore que contém o 1-centro modificado de T , e a e b dois vértices não pertencentes a T_{v_c, v_j}^+ . Se a corresponde ao vértice de perda máxima a partir de v_c , dentre todos os vértices não pertencentes a T_{v_c, v_j}^+ , então podemos seguramente remover o vértice b da árvore T , independentemente da distância do 1-centro modificado x^* ao centróide v_c ou a qualquer folha de T_{v_c, v_j}^+ . Consequentemente, todos os vértices não pertencentes a T_{v_c, v_j}^+ podem ser removidos de T , com exceção do vértice a , o qual deverá ter seu peso e sua distância $d(a, v_c)$ ao centróide v_c modificados para a definição da árvore T' , de menor ordem. Observamos que, como será apresentado no Capítulo 4, pela natureza do problema e da função a ser minimizada, e pela estratégia utilizada, o tempo requerido pelo algoritmo para o problema do 1-centro modificado será dado por

$$S(n) \leq S\left(\frac{n}{2} + 2\right) + Cn. \quad (3.17)$$

Algoritmos Polinomiais e Resultados para o Problema 1-Centro Modificado

Nesta seção apresentamos métodos para a resolução do problema do 1-centro em árvores, abordado no trabalho de [NASCIMENTO, 2018]. São apresentadas as melhorias em complexidade obtidas, algumas das quais, baseadas nos métodos estudados no Capítulo 3. Em seguida, são apresentados resultados de avaliação de performance dos métodos, referentes às implementações desenvolvidas e aos experimentos computacionais realizados. Por fim, são apresentadas observações referentes ao critério de parada utilizado no algoritmo linear, à eficiência do método, quando comparado ao método para o problema do 1-centro clássico em árvores, e referentes às variações do problema do p -centro clássico estudadas.

4.1 Algoritmo de Tempo Quadrático

Objetivando a adoção de terminologia correlata à apresentada por [KARIV; HAKIMI, 1979a], definimos v^* como o *1-centro modificado* de T e $\Phi(v^*)$ como o *1-raio modificado* de T , denotado por r_1 . Observamos que, no problema do 1-centro modificado são considerados pesos de subárvores e, com isso, temos a assimetria nos valores utilizados para as arestas. Desta forma, no exemplo da Figura 4.1, a árvore T possui o vértice v_1 como 1-centro e o vértice v_2 como seu 1-centro modificado, evidenciando a distinção entre os dois problemas.

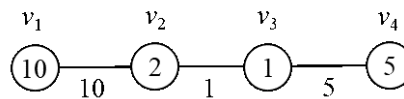


Figura 4.1 – Exemplo do problema do 1-centro modificado.

Como uma primeira abordagem para a determinação de polinomialidade do problema, um primeiro método para a obtenção do 1-centro modificado pode consistir na aplicação de n buscas em profundidade na árvore T , cada uma possuindo um vértice $v_i \in V(T)$ como raiz. Para cada vértice $v_i \in V(T)$, a busca em profundidade enraizada em v_i calcula as quedas de tensão ocorridas nas arestas de cada caminho raiz-folha da árvore de profundidade. Então, calcula a

queda de tensão total em cada folha da árvore de profundidade enraizada em v_i e define a queda de tensão total máxima ocorrida, considerando-se o transformador em v_i . Tendo em vista que o cálculo das demandas acumuladas de cada subárvore requer tempo linear, o método utiliza um pré-processamento em tempo $O(n)$ para o cálculo de todas as demandas acumuladas das subárvores de T . O algoritmo de pré-processamento procede de forma similar ao algoritmo de [HUA, 1961] para o cálculo da 1-mediana em árvores, iniciando a definição dos pesos das subárvores a partir das folhas de T . Ao término, o 1-centro modificado de T será o vértice v_i com a menor queda de tensão total máxima. Tendo em vista a aplicação de n buscas em profundidade em T , esta primeira abordagem requer tempo $O(n^2)$.

4.2 Algoritmo de Tempo $O(n \log n)$

Seja T uma árvore de entrada para o problema do redimensionamento de redes, com pesos em vértices e distâncias em arestas estritamente positivos. Seja v um vértice de T com grau d_v . Seja $T - v$ o grafo obtido de T pela remoção do vértice v . Objetivando uma adequação aos resultados apresentados por [KARIV; HAKIMI, 1979a], $T - v$ consiste de d_v subárvores $T_{v,1}, T_{v,2}, \dots, T_{v,d_v}$. Denotamos por $T_{v,i}^+$ a subárvore que consiste de $T_{v,i}$, o vértice v , e a aresta que conecta v a $T_{v,i}$. Com base nos resultados apresentados por Kariv e Hakimi, temos a seguinte extensão.

Lema 3. *Seja $v \in V(G)$ um vértice fixo e seja \tilde{v} um vértice tal que a queda de tensão total seja máxima a partir de v , i.e. $Q(v, \tilde{v}) = \max_{v' \in V} Q(v, v')$. Seja $T_{v,l}$ a subárvore de $T - v$ á qual \tilde{v} pertence. Então, o 1-centro modificado de T está em $T_{v,l}^+$.*

Prova. Assuma que o 1-centro modificado v^* de T não esteja em $T_{v,l}^+$. Então $Q(v^*, \tilde{v}) > Q(v, \tilde{v})$, considerando-se que as demandas e distâncias sejam todas positivas em T . Portanto, se r_1 é o 1-raio modificado de T , então $r_1 \geq Q(v^*, \tilde{v}) > Q(v, \tilde{v}) = \max_{v' \in V} Q(v, v')$. Desta forma, por um argumento análogo ao apresentado por Kariv e Hakimi, a escolha de v como o 1-centro modificado de T é melhor do que a escolha de v^* , o que é uma contradição. □

O lema conduz ao resultado a seguir.

Corolário 2. *Sejam \tilde{v} e \hat{v} dois vértices tais que $\tilde{v} \in T_{v,l}$, $\hat{v} \in T_{v,k}$, $k \neq l$, e $Q(v, \tilde{v}) = Q(v, \hat{v}) = \max_{v' \in V} Q(v, v')$. Então v é o 1-centro modificado de T .*

Observamos que, se o vértice v no Lema 3 não é uma folha de T , então $T_{v,l}^+$ é uma subárvore própria de T .

Aplicação da Abordagem por Centróides

Analogamente à abordagem por [KARIV; HAKIMI, 1979a], com a utilização o algoritmo CENTRÓIDE, podemos definir o método para a obtenção do 1-centro modificado de uma árvore. No algoritmo a seguir, as variáveis T' , T'' , e T''' representam a cada passo as subárvores T_i , T_{v_i,l_i}^+ , e T_{v_i,l_i} , respectivamente.

Por argumentações análogas às apresentadas por Kariv e Hakimi, o algoritmo obtém o 1-centro modificado de uma árvore em $O(n \log n)$ passos. A cada passo, escolhemos um centróide de T_i como sendo o vértice v_i . Temos $|T_{i+1}| \leq \lfloor T_i/2 \rfloor + 1$, e o número k de passos é $O(\log n)$. Considerando-se n uma potência de 2, isto é, sendo $n = 2^k$, para algum k , se a cada passo diminuirmos o número de vértices da árvore corrente à metade, temos a seguinte sequência de cardinalidades para $V(T)$:

$$\frac{n}{2^0} \rightarrow \frac{n}{2^1} \rightarrow \frac{n}{2^2} \rightarrow \dots \frac{n}{2^k}.$$

Se $n = 2^k$, então $\log_2 n = \log_2 2^k = k$. Ou seja, o número de passos é dado por $k = O(\log n)$.

1-CENTRO MODIFICADO(T)

$T' = T$

enquanto T' possuir mais de uma aresta **faça**

$v_c = \text{CENTRÓIDE}(T')$;

seja $\tilde{v} \in T$ tal que $Q(v_c, \tilde{v}) = \max_{v' \in V(T)} Q(v_c, v')$;

seja T''' a componente de $T - v_c$ que contém \tilde{v} ;

seja T'' a subárvore que consiste de T''' , o vértice v_c ,

e a aresta que conecta v_c a T''' ;

se existe um vértice \bar{v} tal que $\bar{v} \notin T''$ e $Q(v_c, \bar{v}) = Q(v_c, \tilde{v})$ **então**

RETORNE o vértice v_c ;

// Pelo Corolário 2, o vértice v_c é o 1-centro modificado de T ;

$T' = T' \cap T''$;

se T' possui uma única aresta $e = (v_r, v_s)$ **então**

seja $d_r = \max_{v' \in V} Q(v_r, v')$;

seja $d_s = \max_{v' \in V} Q(v_s, v')$;

se $d_r < d_s$ **então** RETORNE o vértice v_r ;

senão RETORNE o vértice v_s ;

4.3 Algoritmo de Tempo Linear

Para a resolução do problema do 1-centro em árvores, a estratégia apresentada por [MEGIDDO, 1983] inclui a remoção de alguns vértices da árvore a cada iteração. Com tal estratégia, é apresentada a seguinte recorrência.

$$time(n) \leq time(7n/8) + Cn. \quad (4.1)$$

Esta recorrência segue do fato de que $n/8$ vértices são descartados a cada iteração no método apresentado por Megiddo. Desta forma, o tempo requerido $time(n)$ para uma árvore com n vértices é linear, i.e. $time(n) = O(n)$. Para o problema do 1-centro modificado em árvores, uma estratégia de remoção de vértices a cada iteração também pode ser utilizada para a obtenção de um algoritmo de tempo linear. Contudo, para o problema do 1-centro modificado em árvores,

tal estratégia conduz a um método com melhor convergência, comparativamente ao método apresentado em [MEGIDDO, 1983].

Seja $T = T_0$ uma árvore instância do problema do 1-centro modificado com n vértices. Seja v_0 o centróide de T_0 . A partir de v_0 , na busca pela subárvore que possui o 1-centro modificado, verificamos qual das subárvores corresponde à perda máxima. Isto é, verificamos qual das subárvores corresponde ao valor máximo de $Q(v_0, v')$.

Sejam $v_{0,1}, \dots, v_{0,d}$ os vizinhos de v_0 em T_0 , e sejam $R(v_0, v_{0,1}), \dots, R(v_0, v_{0,d})$ os valores de perdas máximas em $T_{v_0, v_{0,1}}^+, \dots, T_{v_0, v_{0,d}}^+$. Isto é, os valores de perdas máximas a partir de v_0 nas direcções de $v_{0,1}, \dots, v_{0,d}$, respectivamente (ver Fig. 4.2). Seja $R(v_0, v_{0,k}) = \max_{1 \leq i \leq d} \{R(v_0, v_{0,i})\}$. Como exemplo, para a árvore $T = T_0$ apresentada na Figura 4.3, temos $k = 5$.

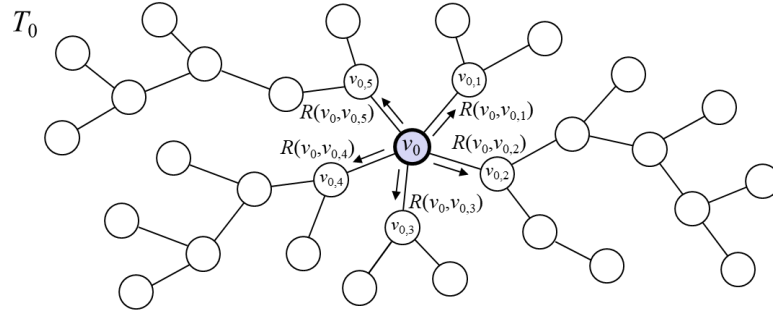


Figura 4.2 – Árvore $T = T_0$ e os valores de perdas máximas $R(v_0, v_{0,1}), \dots, R(v_0, v_{0,d})$ a partir de v_0 nas subárvores $T_{v_0, v_{0,1}}^+, \dots, T_{v_0, v_{0,d}}^+$.

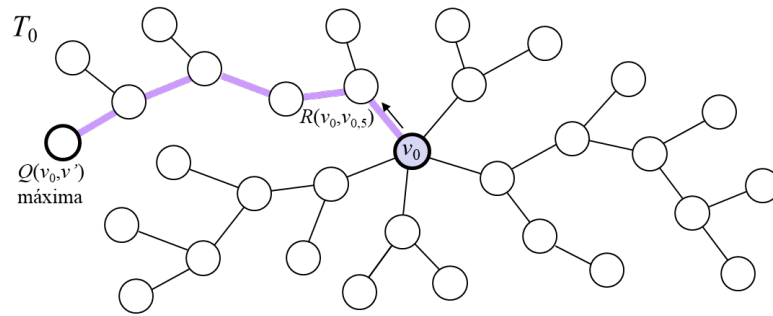


Figura 4.3 – Perda máxima em $T = T_0$ a partir de v_0 .

Seja $R(v_0, v_{0,q}) = \max_{\substack{1 \leq i \leq d \\ i \neq k}} \{R(v_0, v_{0,i})\}$. Isto é, $R(v_0, v_{0,q})$ corresponde à perda máxima a partir de v_0 na subárvore $T_0 \setminus T_{v_0, k}$ (ver Figura 4.4).

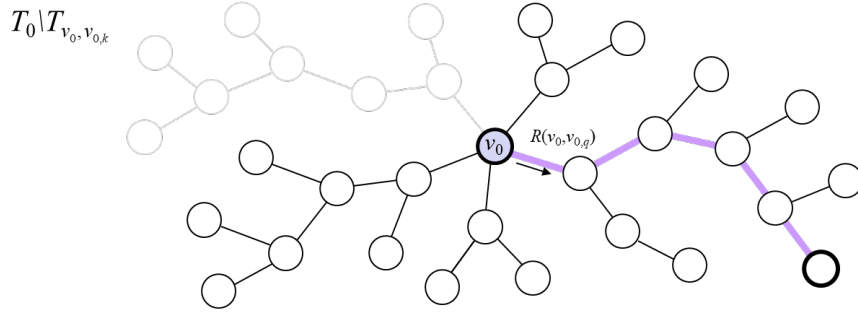


Figura 4.4 – Perda máxima em $T_0 \setminus T_{v_0, k}$ a partir de v_0 .

Se $R(v_0, v_{0,k}) = R(v_0, v_{0,q})$ então, pelo Corolário 2, sabemos que v_0 é o 1-centro modificado de T_0 . Caso contrário, seja T'_0 a árvore obtida de $T_{v_0, v_{0,k}}^+$ pela inclusão de um *vértice falso* u , adjacente a v_0 (ver Figura 4.5).

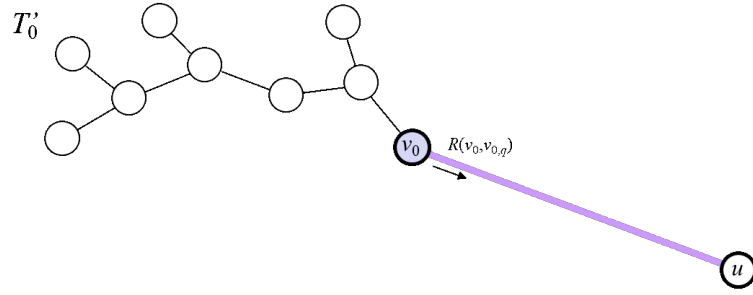


Figura 4.5 – Árvore T'_0 .

Em T'_0 , sejam

- $w(u) = \sum_{v \in T_0 \setminus T_{v_0, v_{0,k}}^+} w(v)$, o somatório dos pesos dos vértices removidos de T_0 na criação de T'_0 (ver Figura 4.6),
- $d(v_0, u) = \frac{100 * R(v_0, v_{0,q})}{w(u)}$ (ver Figura 4.7), e
- $\mu(v_0, u) = 1$.

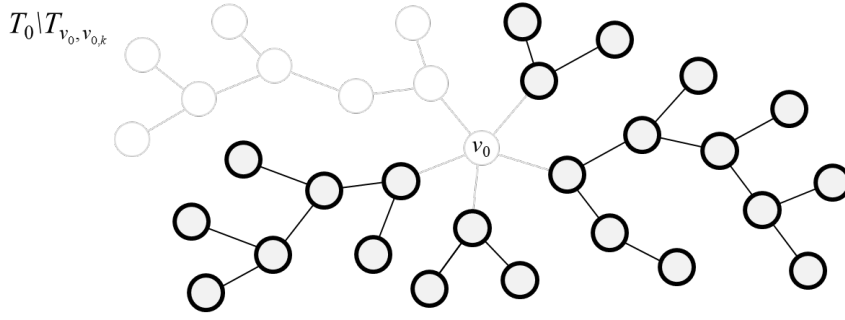


Figura 4.6 – Peso do vértice u .

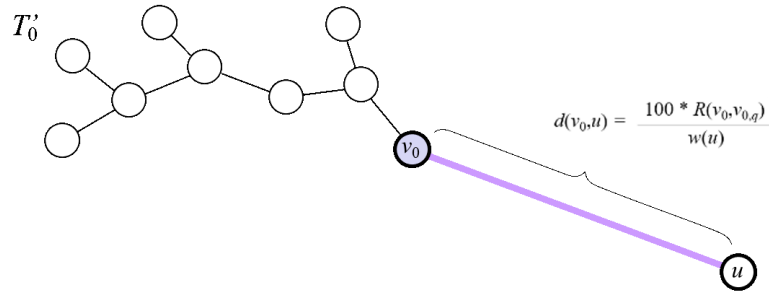


Figura 4.7 – Distância $d(v_0, u)$.

Sendo $|V(T'_0)| \geq 4$, os lemas a seguir estabelecem uma redução para o problema do redimensionamento de redes em árvores.

Lema 4. Em T'_0 , considerando-se pesos, distâncias e constantes estritamente positivos, a perda máxima total a partir de u é maior do que a perda máxima total a partir de v_0 .

Prova. Em T'_0 , a perda máxima a partir de v_0 é dada por $R(v_0, v_{0,k})$, e a perda máxima a partir de u é dada por $R(v_0, v_{0,k}) + [100 * R(v_0, v_{0,k})/w(u)] * w(T'_0 - u)/100$. \square

Como consequência, o vértice u não pode ser o 1-centro modificado da árvore T'_0 . Além disso, cada vértice falso u em T'_i , para $i = 0, 1, \dots$, é uma folha de T'_i e, por construção, foi originado pela remoção de uma subárvore que não continha um vértice que minimizasse a perda máxima.

Se v_i é um centróide de T_i , com $|V(T_i)| > 4$, então v_i não pode ser vizinho de dois vértices falsos em T'_i . Se $|V(T_i)| = 3$, então $|V(T'_i)| = 3$. Se $|V(T_i)| = 4$, então $|V(T'_i)| = 4$. Desta forma, um possível critério de parada em um algoritmo para a definição do 1-centro modificado de uma árvore $T = T_0$ pode ser dado por $|V(T_i)| \leq 4$.

Lema 5. Um vértice v em T_0 é um 1-centro modificado de T_0 se e somente se v for um 1-centro modificado de T'_0 .

Prova. Em T_0 e em T'_0 , as perdas máximas a partir de qualquer vértice v diferente de u são iguais, pelos valores definidos para $w(u)$, $d(v_0, u)$ e $\mu(v_0, u)$. \square

Lema 6. Sendo T_0 uma árvore de ordem n , a árvore T'_0 possui ordem no máximo $\lfloor n/2 \rfloor + 2$.

Prova. Este lema segue da construção da árvore T'_0 a partir de T_0 , e da definição de centróide. \square

O método 1-CENTRO MODIFICADO LINEAR(T) a seguir determina o 1-centro modificado da árvore T e é baseado nos Lemas 3, 4, 5, e 6.

1-CENTRO MODIFICADO LINEAR(T)

$T' = T$

enquanto $|V(T')| > 4$ **faça**

$v_c = \text{CENTRÓIDE}(T')$;

seja $\tilde{v} \in T'$ tal que $Q(v_c, \tilde{v}) = \max_{v' \in V(T')} Q(v_c, v')$;

seja T''' a componente de $T' - v_c$ que contém \tilde{v} ;

seja T'' a subárvore que consiste de T''' , o vértice v_c , o vértice u
e a aresta que conecta v_c a T''' , e a aresta (v_c, u) ;

defina $w(u)$, $d(v_c, u)$ e $\mu(v_c, u)$;

se existe um vértice \bar{v} tal que $\bar{v} \notin T''$ e $Q(v_c, \bar{v}) = Q(v_c, \tilde{v})$ **então**

RETORNE o vértice v_c ;

$T' = T''$;

RETORNE o vértice v_s com menor perda máxima, dentre os vértices em $V(T')$,
com $|V(T')| \leq 4$;

Pelo Lema 6, temos uma redução do problema, de uma instância de tamanho n a uma instância de tamanho menor ou igual a $\lfloor n/2 \rfloor + 2$. Isto é, temos a seguinte recorrência para o tempo requerido pelo algoritmo:

$$\text{tempo}(n) \leq \text{tempo}(\lfloor n/2 \rfloor + 2) + Cn. \quad (4.2)$$

Lema 7. Seja T uma árvore ponderada com n vértices, com distâncias, pesos e constantes positivas, como instância do problema do redimensionamento de redes. O algoritmo 1-CENTRO MODIFICADO LINEAR(T) requer tempo $O(n)$.

Prova. Seja

$$S(n) = S\left(\frac{n}{2} + 2\right) + Cn. \quad (4.3)$$

Aplicando o método da substituição, temos

$$\begin{aligned}
S(n) &= S\left(\frac{n}{2^2} + 1 + 2\right) + C\left(\frac{n}{2} + 2\right) + Cn \\
&= S\left(\frac{n}{2^3} + \frac{1}{2} + 1 + 2\right) + C\left(\frac{n}{2^2} + 1 + 2\right) + C\left(\frac{n}{2} + 2\right) + Cn \\
&\vdots
\end{aligned} \tag{4.4}$$

Supondo $n = 2^k$, para um inteiro positivo k , temos

$$\begin{aligned}
S(n) &= S\left(\frac{n}{2^k} + 2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots\right)\right) + Cn\left(1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-2}}\right) + \\
&\quad + C\left(2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-2}}\right)\right) + C\left(2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-3}}\right)\right) + \\
&\quad + \dots + C\left(2\left(1 + \frac{1}{2} + \frac{1}{4}\right)\right) + C\left(2\left(1 + \frac{1}{2}\right)\right) + C(2(1)).
\end{aligned} \tag{4.5}$$

Substituindo, temos

$$\begin{aligned}
S(n) &= S\left(\overbrace{\frac{n}{2^k}}^{\leq 1} + \overbrace{2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots\right)}^{\leq 4}\right) + Cn\left(\overbrace{1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-2}}}^{\leq 2}\right) + \\
&\quad + C\left(\overbrace{2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-2}}\right)}^{\leq 4}\right) + C\left(\overbrace{2\left(1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^{k-3}}\right)}^{\leq 4}\right) + \\
&\quad + \dots + C\left(\overbrace{2\left(1 + \frac{1}{2} + \frac{1}{4}\right)}^{\leq 4}\right) + C\left(\overbrace{2\left(1 + \frac{1}{2}\right)}^{\leq 4}\right) + C\left(\overbrace{2(1)}^{\leq 4}\right).
\end{aligned} \tag{4.6}$$

Isto é,

$$S(n) \leq O(1) + 2Cn + 4C \log n = O(n). \tag{4.7}$$

□

4.4 Resultados

Problemas de localização possuem aplicações que incluem a cobertura de uma determinada região com a localização de p hospitais, torres de telefonia móvel, instalação de sirenes de alerta e de depósitos de baterias em uma rede de *delivery* por drones, dentre outras ([LIU, 2019]). Nesta seção, apresentamos uma primeira avaliação da performance computacional dos métodos desenvolvidos para a resolução do problema do 1-centro modificado. Os experimentos foram realizados com a utilização de um Intel(R) Core(TM) i7 - 9750H CPU @ 2.60GHz, 16.0 GB RAM - Windows 11.

O desenvolvimento em C++ dos softwares utilizados incluiu a implementação de classes para o armazenamento de instâncias do problema em espaço linear, e para a execução dos métodos CENTRÓIDE em tempo $O(n)$ e 1-CENTRO MODIFICADO em tempo $O(n^2)$, $O(n \log n)$ e $O(n)$. A abordagem incluiu a criação de instâncias de tamanhos de 7, 10, 13, 16, ..., e 37 vértices, assim como uma comparação de desempenho com o primeiro método de cálculo do 1-centro modificado em tempo quadrático, apresentado em [NASCIMENTO, 2018]. Para o conjunto de instâncias geradas, com a representação de pesos de vértices e arestas, demandas acumuladas, quedas de tensão e constantes de condutividade elétrica em precisão simples, o método 1-CENTRO MODIFICADO em tempo linear representou uma melhoria de 18,92% em média, no tempo de execução (em operações elementares) comparativamente aos métodos $O(n^2)$ e $O(n \log n)$.

Dentre as principais classes desenvolvidas, estão `class TFile`, `class TVertice`, `class TArvore`, `class TArvoreLinear`. Dentre os principais métodos desenvolvidos, estão `void alg_1_Centro_Modif_Quad(void)`, `void alg_1_Centro_Modif_nlogn(void)`, e `void alg_1_Centro_Modif_n(void)`, referentes aos métodos de tempo $O(n^2)$, $O(n \log n)$ e $O(n)$, respectivamente. Para o conjunto de instâncias geradas, com a representação de pesos de vértices e arestas, demandas acumuladas, quedas de tensão e constantes de condutividade elétrica em precisão simples, a seguir a Tabela 4.1 apresenta de forma aproximada os números de operações elementares em cada um dos métodos e a Figura 4.8 nos mostra a performance dos métodos.

Métodos	$n = 7$	$n = 10$	$n = 13$	$n = 16$	$n = 19$	$n = 22$
$O(n^2)$	647	1378	2341	3552	5125	6809
$O(n \log n)$	641	987	1816	1735	2979	3638
$O(n)$	478	848	1031	1113	1394	1804

Métodos	$n = 25$	$n = 28$	$n = 31$	$n = 34$	$n = 37$
$O(n^2)$	8723	11906	13147	16169	19291
$O(n \log n)$	4294	5065	4214	4881	7712
$O(n)$	1941	2082	1799	2143	2742

Tabela 4.1 – Número de operações elementares.

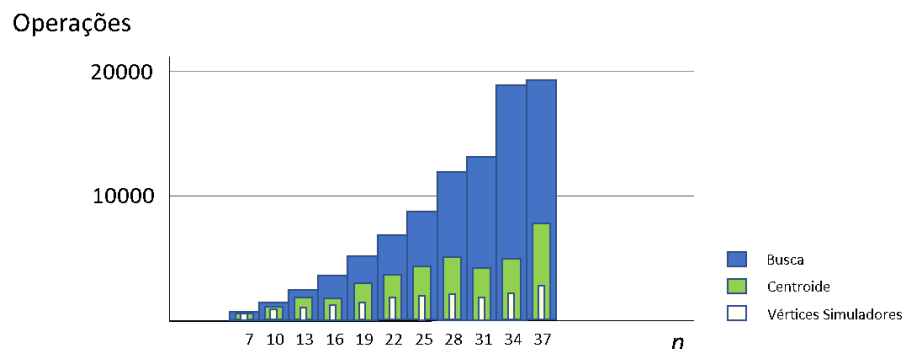


Figura 4.8 – Performance dos métodos $O(n^2)$, $O(n \log n)$ e $O(n)$ para o problema do 1-centro modificado em instâncias com $n = 7, 10, \dots, 37$.

Observamos que, para as instâncias geradas, o método de tempo $O(n)$ mostrou-se sensivelmente mais rápido do que a estratégia de tempo $O(n \log n)$, representando uma melhoria de 64% em operações elementares, para determinadas instâncias.

Como novas abordagens para a resolução do problema do redimensionamento de redes de energia, podemos citar a aplicação de uma variação do problema do 1-centro inverso, objetivando uma adequação de setores de uma rede em projeto à capacidade de equipamentos utilizados na distribuição de energia.

4.5 Observações

Critérios de Parada

Seja T' uma árvore com 3 vértices, e T'' e T''' árvores com 4 vértices, como apresentado na Figura 4.9 a seguir. Pela localização do centróide v_c de T' , a remoção de qualquer subárvore de v_c para a inclusão de um vértice simulador não acarretará em uma diminuição na ordem de T' . Considerando-se T'' , pela localização do centróide v_c , a remoção de duas subárvores quaisquer de v_c para a inclusão de um vértice simulador acarretará em uma diminuição na ordem de T'' , dando origem a uma árvore isomorfa a T' . Em T''' , a remoção de uma subárvore de v_c para a inclusão de um vértice simulador pode dar origem a uma nova árvore isomorfa a T' ou isomorfa à própria T''' .

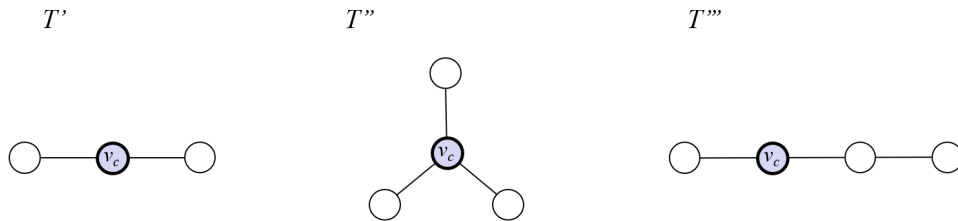


Figura 4.9 – Árvores com 3 e 4 vértices.

Para qualquer árvore com ao menos 5 vértices, a remoção de subárvores do centróide v_c dá origem a uma nova árvore de ordem necessariamente menor. Desta forma, o critério de parada utilizado assegura o tempo $O(n)$ do algoritmo.

Eficiência do Método

Seja T uma árvore com n vértices, seja v_0 o centróide de T , seja $R(v_0, v_{0,k}) = \max_{1 \leq i \leq d} \{R(v_0, v_{0,i})\}$ o valor de perda máxima a partir do centróide v_0 , seja v' o vértice para o qual ocorre a perda máxima a partir de v_0 em T , i.e. seja $R(v_0, v_{0,k}) = Q(v_0, v')$. Seja T_{v_0, v_k} a subárvore de $T - v_0$ que contém v' e seja $T_{v_0, v_k}^+ = T_{v_0, v_k} + v_0$.

Seja $R(v_0, v_{0,q}) = \max_{1 \leq i \leq d, i \neq k} \{R(v_0, v_{0,i})\}$. Isto é, seja $R(v_0, v_{0,q})$ o valor de perda máxima a partir de v_0 na subárvore $T \setminus T_{v_0, v_k}^+$. Seja v'' o vértice para o qual ocorre a perda máxima a

partir de v_0 em $T \setminus T_{v_0, v_k}^+$, i.e. seja $R(v_0, v_0, q) = Q(v_0, v'')$.

Seja x um vértice em T_{v_0, v_k}^+ . Se $R(x, p)$ é o valor de perda máxima a partir do vértice x em T , e se o caminho de perda máxima $\langle x, \dots, v_0, \dots, v_p \rangle$ contém o centróide v_0 , então $R(x, p) = Q(x, v_p) = Q(x, v'')$. Isto é, a perda máxima a partir de x na árvore T ocorre no caminho para o vértice v'' . Consequentemente, todos os vértices em $T \setminus T_{v_0, v_k}^+$ podem ser removidos da árvore T , com exceção do vértice v'' , sendo transformado em um vértice simulador. Na definição de $u = v''$ como um novo vértice simulador, temos a redefinição do peso $w(u) = \sum_{v \in T \setminus T_{v_0, v_k}^+} w(v)$, e a inclusão da aresta (v_0, u) , com $d(v_0, u) = Q(v_0, v'')$ e $\mu(v_0, u) = 1$.

Desta forma, a nova árvore T' é definida com a devida redefinição do peso $w(v'')$, da distância $d(v_0, v'')$ e de $\mu(v_0, v'')$, de forma a considerar todas as demandas acumuladas dos vértices em $T \setminus T_{v_0, v_k}^+$, para qualquer caminho de x a um vértice de $T \setminus T_{v_0, v_k}^+$.

Observamos que, diferentemente da estratégia apresentada por Megiddo, tais remoções podem ser realizadas independentemente da localização do 1-centro modificado em T_{v_0, v_k}^+ . Esta possibilidade se deve ao fato de que, um caminho de perda máxima deve ser composto de subcaminhos de perda máxima. Especificamente, para o caminho de perda máxima $x \rightsquigarrow v_0 \rightsquigarrow v_p$, composto pelos subcaminhos $x \rightsquigarrow v_0$ e $v_0 \rightsquigarrow v_p$ em T , devemos ter $v_0 \rightsquigarrow v_p$ como um subcaminho de perda máxima. Caso contrário, poderíamos substituir tal subcaminho por um subcaminho de maior perda a partir de v_c , contradizendo o valor máximo do caminho original $x \rightsquigarrow v_0 \rightsquigarrow v_p$.

Portanto, em uma etapa do método 1-CENTRO MODIFICADO LINEAR, sendo T' a árvore corrente com n' vértices, temos a remoção de $\frac{n'}{2} - 2$ vértices para a obtenção de uma nova árvore de menor ordem. O conjunto de remoções no método dá origem à recorrência $S(n) \leq S((n/2) + 2) + Cn$. O conjunto de remoções no algoritmo de Megiddo resulta na recorrência $S(n) \leq S(7n/8) + Cn$. Em um estudo comparativo entre as duas recorrências, considerando-se $C = 30$ e $S(n) = 118$ (para $n \leq 4$) obtemos os valores apresentados na tabela 4.2 a seguir. Os valores podem indicar uma eficiência do método apresentado, com respeito aos métodos clássicos para a resolução do problema do 1-centro em árvores.

Recorrências	$n = 7$	$n = 10$	$n = 13$	$n = 16$	$n = 19$	$n = 22$
$S(n) = S(7n/8) + Cn$	658	1198	1648	2458	3028	3688
$S(n) = S((n/2) + 2) + Cn$	478	778	1078	1258	1378	1738

Recorrências	$n = 25$	$n = 28$	$n = 31$	$n = 34$	$n = 37$
$S(n) = S(7n/8) + Cn$	4018	4828	4214	5518	6898
$S(n) = S((n/2) + 2) + Cn$	1888	2098	2218	2398	2758

Tabela 4.2 – Recorrências dos métodos.

Generalizações do Problema

Seja $G = (V, E)$ um grafo conexo, ponderado em vértices e arestas. Objetivando a generalização do problema do 1-centro modificado em árvores a grafos conexos, observamos que as

funções de distância $d(v_j, v_k)$ e de peso $k(G')$, estendem-se de forma imediata, assim como a constante $\mu(v_j, v_k)$ do material condutor utilizado no trecho entre v_j e v_k . Contudo, o conceito de *demandas acumuladas* pode ser revisto, de acordo com a aplicação da definição de 1-centro ou de p -centro a ser utilizada no estudo do problema do redimensionamento de redes.

Seja $e_i = (v_j, v_k) \in E(G)$. Seja $G(v_j, v_k)$ o subgrafo de G induzido pelos vértices alcançáveis a partir de v_k , sem a utilização da aresta (v_j, v_k) . Se G é um grafo sem pontes, então $G(v_j, v_k) = G$, e portanto

$$k(G(v_j, v_k)) = k(G). \quad (4.8)$$

Como apresentado no Capítulo 2, na definição da *queda de tensão* em uma aresta $(v_j, v_k) \in E(G)$, consideraremos o peso $k(v_i)$ como a demanda de energia no vértice $v_i \in T$. Assim, definimos a *queda de tensão* em uma aresta $(v_j, v_k) \in E(G)$, denotada por $q(v_j, v_k)$, por

$$q(v_j, v_k) = \frac{d(v_j, v_k) * k(G(v_j, v_k)) * \mu(v_j, v_k)}{100} = \frac{d(v_j, v_k) * k(G) * \mu(v_j, v_k)}{100}. \quad (4.9)$$

Tendo em vista o valor fixo $k(G)$, consideramos $\mu(v_j, v_k) = 100$ em uma redução em tempo polinomial imediata do problema do p -centro em grafos com pesos uniformes em vértices para o problema do p -centro modificado com tal formulação, indicando a NP-Completeness do problema.

Seja $T = (V, E)$ uma árvore, ponderada em vértices e arestas. Para a generalização do problema do 1-centro modificado em árvores a um p -centro modificado, observamos que novamente as funções de distância $d(v_j, v_k)$ e de peso $k(T(v_j, v_k))$ estendem-se de forma imediata, assim como a constante $\mu(v_j, v_k)$ do material condutor utilizado no trecho entre v_j e v_k . Contudo, o conceito de *demandas acumuladas* pode ser revisto, de acordo com as definições de p -centro e de suprimento de energia, utilizadas no estudo do problema do redimensionamento de redes.

Variações de Problemas de Localização Correlatos

Na definição e na abordagem ao problema do 1-centro modificado em árvores, foram estudadas inúmeras variações do problema do 1-centro clássico. As variações estudadas são apresentadas a seguir, muitas das quais não incluindo a assimetria observada nas demandas acumuladas, definidas no problema do 1-centro modificado. Em [BEN-MOSHE et al., 2007], apresentam algoritmos eficientes para resolver os problemas de centros em redes como cactos ponderados. Em particular, para redes ponderadas em cactos de tamanho n , são propostos um algoritmo de tempo $O(n \log n)$ para a resolução do problema do 1 centro e um algoritmo de tempo $O(n \log^3 n)$ para a resolução do problema de 2-centro contínuo ponderado. Também são fornecidas soluções aprimoradas para os problemas gerais de p -centro em redes cactos, e as ideias desenvolvidas são então aplicadas para a resolução do problema do 1-centro “indesejável” em redes cactos ponderadas.

Em [BHATTACHARYA; DAS; DEV, 2019] apresentam um algoritmo de tempo linear para o problema do k -centro ponderado em árvores para um k fixo, resolvendo parcialmente a questão

de longa data sobre o limite inferior da complexidade de tempo do problema. A complexidade de tempo atual do algoritmo conhecido para o problema com k como parte da entrada é $O(n \log n)$ de Wang *et al.* (2018). Se existe um algoritmo de tempo $O(n)$ para k 's arbitrários ainda permanece em aberto.

Segundo [BHATTACHARYA *et al.*, 2020] um estudo é apresentado sobre alguns problemas fundamentais de localização de instalações do ponto de vista da eficiência de espaço. Em [BHATTACHARYA; SHI, 2014], é mostrado que o problema do 1-centro no espaço Euclidiano e em redes de árvores pode ser resolvido eficientemente no modelo de espaço constante. Os autores apresentam um algoritmo que requer espaço $O(1)$ para o problema do 1-centro em árvores.

Por [BHATTACHARYA; SHI, 2014], em um modelo de problema de localização de instalações, pesos de vértices incertos são representados por intervalos de pesos possíveis, e procura-se uma solução “robusta” que minimize o máximo “arrependimento” (Kouvelis *et al.*, 1993). Primeiro, Bhattacharya *et al.* fornecem um algoritmo de tempo $O(n)$ para as redes de caminho e, em seguida, apresentam um algoritmo de tempo $O(n \log n)$ para as redes em árvore, que melhora o algoritmo anteriormente melhor para esse problema (Yu *et al.*, 2008), com complexidade de tempo $O(n \log^2 n)$. Os autores apresentam também um algoritmo de tempo $O(n \log n)$ para redes uniciclo, possuindo apenas um ciclo. O algoritmo apresentado para cactos requer tempo $O(n \log^2 n)$. Por [BHATTACHARYA; SHI, 2014], não existem resultados publicados anteriormente, adaptados especificamente para redes de cactos. Ao resolver esses problemas, são apresentadas várias estruturas de dados, que podem ser úteis para outras aplicações.

Conforme [BHATTACHARYA; SHI, 2014] é apresentado um problema do p -centro ($p \geq 2$) em redes gerais pode ser transformado no conhecido problema de medida de Klee (Overmars e Yap, 1991), resultando em um algoritmo significativamente melhorado para o caso contínuo, de tempo $O(m^p n^{p/2} 2^{\log^* n} \log n)$ para $p \geq 3$, onde n é o número de vértices, m é o número de arestas, e $\log^* n$ denota o logaritmo iterado de n (ver [CORMEN *et al.*, 2009]). Para $p = 2$, o tempo de execução do algoritmo melhorado é $O(m^2 n \log^2 n)$.

Visto [BHATTACHARYA *et al.*, 2006] é considerado o problema de localizar uma instalação em forma de caminho ou em forma de árvore (extensiva) em árvores, sob a condição de que as instalações existentes já estejam localizadas. Os autores apresentam um método de poda paramétrica para resolver os problemas de localização de 1-centro ponderado extensivo discreto/contínuo condicional em árvores em tempo linear. Os resultados apresentados melhoram os resultados de tempo de $O(n \log n)$, devidos a Tamir *et al.* (J. Algebra 56:50–75, 2005).

De acordo com [CABELLO; ROTE, 2010] consideramos o problema de encontrar centros “indesejáveis” em grafos. Para grafos arbitrários com n vértices e m arestas, os autores apresentam um algoritmo randômico com tempo esperado $O(n \log^2 n + m \log n)$. Para grafos planares, os autores apresentam algoritmos com tempo esperado $O(n \log n)$ e tempo de pior caso $O(n \log^3 n)$. Para grafos com largura de árvore limitada, é fornecido um algoritmo que requer tempo $O(n \log n)$ no pior caso. Os algoritmos fazem uso de busca paramétrica e diversos resultados para o cálculo de distâncias em grafos com largura de árvore limitada e grafos planares.

[GØRTZ; WIRTH, 2006] abordam o problema do k -centro, algumas de suas variações e assimetria. Variações do k -centro podem modelar problemas da vida real com mais precisão do que a formulação original. Por Gørtz e Wirth, a assimetria é um impedimento para a aproximação em muitos problemas em grafos, como k -centro, localização de instalações, k -mediana e o TSP. Os autores fornecem um algoritmo de $O(\log^* n)$ -aproximação para o problema do

k -centro ponderado assimétrico. No trabalho, os vértices são ponderados, com um custo total para a abertura de centros. Na variação de p -vizinho cada vértice deve ter p centros (não ponderados) próximos. Por fim, são apresentadas versões não aproximáveis do problema do k -centro assimétrico.

[HARTMANN; LENDL; WOEGINGER, 2020] apresentam um estudo sobre um problema de localização contínuo em grafos não direcionados onde todas as arestas possuem comprimento unitário. O objetivo é cobrir todo o grafo com um número mínimo de centros com alcance de cobertura $\delta > 0$. Equivalentemente, no problema abordado devemos posicionar o menor número possível de centros com a condição de que cada ponto em cada aresta esteja a uma distância no máximo δ de um desses centros. Os autores investigam este problema de cobertura em termos do parâmetro racional δ , e apresentam condições em que o problema pode ser resolvido em tempo polinomial, e em que o problema é NP-difícil. No trabalho, também é apresentada uma análise de complexidade parametrizada, com o tamanho da solução como parâmetro: o problema resultante é tratável por parâmetro fixo, para $\delta < 3/2$, e é W[2]-difícil para $\delta \geq 3/2$.

[NGUYEN; ANH, 2015] consideram uma generalização do problema da 1-mediana inverso, o problema do k -centro inverso, em árvores com pesos de vértices variáveis. Em contraste com a possibilidade de resolução em tempo linear do problema da 1-mediana inverso em árvores, os autores provam que o problema do k -centro inverso em árvores é NP-difícil. Adicionalmente, o problema do 1-centro inverso, um caso especial do problema apresentado com $k = 1$, em uma árvore com n vértices pode ser resolvido em tempo $O(n^2)$.

No problema do 1 centro inverso em uma rede devemos modificar os comprimentos das arestas ou os pesos dos vértices dentro de certos limites, de modo que o vértice pré-especificado se torne um 1-centro (absoluto) da rede perturbada e o custo de modificação seja minimizado. Em [NGUYEN; NGUYEN-THU; HUNG, 2018], apresentam um estudo sobre o problema do 1-centro inverso em uma árvore ponderada com custo uniforme de modificação do comprimento da aresta, uma generalização para o problema análogo em uma árvore não ponderada.

[PUERTO; RODRÍGUEZ-CHÍA; TAMIR, 2010] apresentam um modelo minimax unificando várias classes de problemas de localização de centro planar de instalação única. Os autores consideram o problema do p -centro em árvores onde os clientes são modelados como subárvores contínuas. São abordados modelos não ponderados e ponderados. Os autores mostram que uma modificação relativamente simples dos algoritmos de tempo linear clássicos de Handler para problemas não ponderados de 1- e 2-centros em relação a clientes pontuais resolve linearmente os problemas não ponderados de 1- e 2-centros com adendos do modelo de cliente de subárvore acima. Também são desenvolvidos algoritmos de tempo polinomial para os problemas de p -centro baseados na resolução de problemas de cobertura e busca sobre domínios especiais.

[WANG; ZHANG, 2017] apresentam um algoritmo de tempo $O(n \log n)$ para o problema do k -centro em árvores.

Em [WANG; ZHANG, 2016], consideram o problema do 1-centro para dados incertos em redes de árvores. Neste problema, é dada uma árvore T e n pontos incertos (ponderados), cada um com m possíveis localizações em T associadas a probabilidades. O objetivo é encontrar um ponto x^* em T tal que a distância máxima (ponderada) esperada de x^* a todos os pontos incertos seja minimizada. Segundo Wang e Zhang, esse problema não foi investigado anteriormente. Os autores propõem uma técnica refinada de poda e busca que resolve o problema em tempo linear.

[YU; LI; LEE, 2018] propõem um novo tipo de problema de localização de rede para a definição de instalações múltiplas, mas distintas, chamado de problema dos centros de p -serviços. Neste

problema, devemos localizar instalações no grafo, cada uma das quais fornecendo um serviço distinto requerido por todos os vértices. Para cada vértice, sua distância de p -serviço é a soma de suas distâncias ponderadas aos p centros. O objetivo é minimizar o valor máximo entre as distâncias de p -serviços de todos os vértices. Yu *et al.* mostram que o problema dos centros de p -serviços em um grafo geral é NP-difícil e propõem um algoritmo simples de aproximação com fator p/c para qualquer constante inteira c . Além disso, apresentam um estudo do caso básico $p = 2$ em caminhos e árvores. Quando a rede subjacente é um caminho, o problema dos centros de 2-serviços é resolvido em tempo $O(n)$, onde n é o número de vértices. Quando a rede subjacente é uma árvore, é apresentado um algoritmo de tempo $O(n^3)$ para o caso de pesos não negativos, um algoritmo de tempo $O(n \log n)$ para o caso de pesos positivos e um algoritmo de tempo $O(n)$ para o caso de pesos uniformes.

[YU; LIN; WANG, 2008] apresentam algoritmos eficientes para os problemas do 1-centro e da 1-mediana *minmax-regret* em um grafo geral e uma árvore com pesos de vértices incertos. Para o problema do 1-centro *minmax-regret* em um grafo geral, é apresentada uma melhoria no limite superior anterior de $O(mn^2 \log n)$ para $O(mn \log n)$. Para o problema em uma árvore, o limite superior é melhorado de $O(n^2)$ para $O(n \log^2 n)$. Para o problema da 1-mediana *minmax-regret* em um grafo geral, o limite superior é melhorado de $O(mn^2 \log n)$ para $O(mn^2 + n^3 \log n)$. Para o problema em uma árvore, o limite superior é melhorado de $O(n \log^2 n)$ para $O(n \log n)$.

[BANIK et al., 2016] apresentam dois algoritmos melhorados para o problema do p -centro discreto ponderado para redes em árvore com n vértices. Um dos algoritmos propostos roda em tempo $O(n \log n + p \log^2 n \log(n/p))$. É observado que, para todos os valores de p , o algoritmo apresentado mostrou-se tão rápido quanto ou mais rápido que o algoritmo de tempo $O(n \log^2 n)$ mais eficiente, obtido pela aplicação da técnica de aceleração de Cole ao algoritmo devido a Megiddo e Tamir.

[CALIK; LABBÉ; YAMAN, 2015] apresentam diferentes variantes do problema do p -centro. Revisam casos polinomiais especiais, determinam a complexidade dos problemas, apresentam formulações de programação linear inteira mista, algoritmos exatos e heurísticas, e apresentam revisões de várias extensões do problema.

Generalizando um resultado de [PLESNÍK, 1987] apresentam um algoritmo polinomial com uma razão de erro de pior caso de 2 para o problema do p -centro em grafos conexos com comprimentos em arestas e pesos em vértices. Uma pequena modificação deste algoritmo fornece razão 2 também para o problema do p -centro absoluto. Ambas as heurísticas são melhores possíveis no sentido de que qualquer razão menor implicaria $P = NP$.

[HANDLER; MIRCHANDANI, 1979] apresentam algoritmos para um conjunto de problemas de otimização baseados em redes, considerando aplicações em redes de comunicação, redes de distribuição de energia e redes de transportes.

Em [JAEGER; GOLDBERG, 1994] estendem a polinomialidade do problema do p -centro não capacitado ao caso em que cada centro pode servir a um número limitado de clientes e mostram que o problema do p -centro capacitado em árvores pode ser solucionado em tempo polinomial quando as capacidades são idênticas.

5.1 Considerações Finais

Nesta dissertação, apresentamos um estudo sobre o problema do 1-centro modificado em árvores, como aplicação ao problema do redimensionamento de redes de energia. Em tal estudo, foram apresentados resultados fundamentando a técnica de utilização de centróides para a obtenção do 1-centro modificado de uma árvore. Os resultados estendem os apresentados por [KARIV; HAKIMI, 1979a] em algoritmos para a obtenção do 1-centro. Com a utilização da técnica, foi desenvolvido um primeiro método de tempo $O(n \log n)$ para a obtenção do 1-centro modificado em uma árvore. O estudo igualmente incluiu a apresentação de condições para a utilização da técnica de *vértices simuladores*. Em tal estratégia, diversas subárvores são substituídas por um único vértice simulador, possibilitando a eliminação de inúmeros vértices a cada etapa, com a sensível diminuição do tempo requerido por um algoritmo de tempo linear, quando comparado aos algoritmos clássicos existentes para a obtenção de 1-centro.

A implementação dos métodos desenvolvidos possibilitou a verificação da superioridade do algoritmo de tempo $O(n)$, quando comparado a outros métodos. Igualmente, foram resolvidos casos especiais da recorrência do método de tempo linear apresentado para o problema do 1-centro modificado em árvores e da recorrência do método para o problema do 1-centro clássico em árvores apresentado por [MEGIDDO, 1983], indicando um menor tempo requerido pelo método desenvolvido no presente trabalho, embora ambos requeiram tempo $O(n)$.

Aplicações de conceitos de teoria da localização em problemas de redimensionamento de energia são apresentadas em [GARCIA et al., 2003] e [SILVA; FRANCA; SILVEIRA, 1996]. Além disso, com os estudos de NP-Completeness sobre problemas de localização existentes, foi possível a verificação de NP-Completeness para uma determinada formulação de generalização do problema do 1-centro modificado em árvores para o p -centro, considerando-se a classe geral de grafos.

5.2 Trabalhos Futuros

Como apresentado no Capítulo 4, [BEN-MOSHE et al., 2007] apresentam algoritmos eficientes para a resolução de problemas de centros em redes de cactos ponderados. Em particular, para as redes ponderadas de cactos de tamanho n , é proposto um algoritmo de tempo $O(n \log n)$ para a resolução do problema do 1-centro e um algoritmo de tempo $O(n \log^3 n)$ para a resolução do

problema do 2-centro contínuo ponderado.

[BHATTACHARYA; DAS; DEV, 2019] apresentam um algoritmo de tempo linear para o problema do k -centro ponderado em árvores para um k fixo, resolvendo parcialmente uma questão sobre o limite inferior da complexidade de tempo do problema.

No *problema do 1-centro inverso* em uma rede devemos modificar os comprimentos das arestas ou os pesos dos vértices dentro de certos limites, de modo que um vértice pré-especificado se torne um 1-centro 1 (absoluto) da rede perturbada e o custo de modificação seja minimizado. Em [NGUYEN; NGUYEN-THU; HUNG, 2018], é apresentado um estudo sobre o problema do 1-centro inverso em uma árvore ponderada com custo uniforme de modificação do comprimento da aresta.

[YE; LI; WANG, 2018] apresentam um estudo sobre o problema de encontrar um centro de caminho em uma árvore na qual os pesos dos vértices são incertos e a incerteza é descrita por intervalos dados. Devemos encontrar uma solução de arrependimento *minmax*, que minimize a perda de pior caso na função objetivo. [YE; LI; WANG, 2018] apresentam um algoritmo de tempo $O(n \log n)$ para o problema.

Como atividades futuras, podem ser citados estudos de aplicações de algoritmos de 1-centro *modificado* para redes cactos, no contexto do problema do redimensionamento de redes. Além disso, a complexidade de tempo atual do algoritmo conhecido para o problema do k -centro ponderado em árvores para um k fixo, com k sendo parte da entrada, é $O(n \log n)$. Como um problema em aberto citado, está a verificação da existência de um algoritmo de tempo $O(n)$ para k 's arbitrários. Por fim, pode ser investigada a aplicação do problema do 1-centro inverso e do problema do centro de caminho em uma árvore, ao problema do redimensionamento de redes de energia.

Referências Bibliográficas

AHO, A.; HOPCROFT, J.; ULLMAN, J. *The Design and Analysis of Computer Algorithms*. [S.l.]: Addison-Wesley — Reading, MA, 1974.

ANEEL. *Agência Nacional de Energia Elétrica – Regulação dos Serviços de Distribuição*. 2022. Disponível em: <https://antigo.aneel.gov.br/web/guest/regulacao-dos-servicos-de-distribuicao>.

ASSIS, L. de; FRANCA, P.; USBERTI, F. A redistricting problem applied to meter reading in power distribution networks. *Comput. Oper. Res.*, Elsevier Science Ltd., GBR, v. 41, p. 65–75, jan 2014. ISSN 0305-0548. Disponível em: <https://doi.org/10.1016/j.cor.2013.08.002>.

BANIK, A. et al. The $\$p\$$ -center problem in tree networks revisited. *CoRR*, abs/1604.07535, 2016. Disponível em: <http://arxiv.org/abs/1604.07535>.

BEN-MOSHE, B. et al. Efficient algorithms for center problems in cactus networks. *Theoretical Computer Science*, v. 378, n. 3, p. 237–252, 2007. ISSN 0304-3975. Algorithms and Computation. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0304397507001247>.

BHATTACHARYA, B.; DAS, S.; DEV, S. R. The Weighted k-Center Problem in Trees for Fixed k. In: LU, P.; ZHANG, G. (Ed.). *30th International Symposium on Algorithms and Computation (ISAAC 2019)*. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019. (Leibniz International Proceedings in Informatics (LIPIcs), v. 149), p. 27:1–27:11. ISBN 978-3-95977-130-6. ISSN 1868-8969. Disponível em: <https://drops.dagstuhl.de/opus/volltexte/2019/11523>.

BHATTACHARYA, B. et al. Optimal algorithms for the path/tree-shaped facility location problems in trees. In: *Proceedings of the 17th International Conference on Algorithms and Computation*. Berlin, Heidelberg: Springer-Verlag, 2006. (ISAAC'06), p. 379–388. ISBN 3540496947. Disponível em: https://doi.org/10.1007/11940128_39.

BHATTACHARYA, B.; SHI, Q. Improved algorithms to network p-center location problems. *Computational Geometry*, v. 47, n. 2, Part C, p. 307–315, 2014. ISSN 0925-7721. The 20th Canadian Conference on Computational Geometry (CCCG 2008). Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925772113000278>.

- BHATTACHARYA, B. K. et al. Constant work-space algorithms for facility location problems. *Discrete Applied Mathematics*, v. 283, p. 456–472, 2020. ISSN 0166-218X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0166218X20300573>.
- BURKARD, R.; KRARUP, J. A linear algorithm for the pos/neg-weighted 1-median problem on a cactus. *Computing*, v. 60, p. 193–215, 1998.
- CABELLO, S.; ROTE, G. Obnoxious centers in graphs. *SIAM Journal on Discrete Mathematics*, v. 24, n. 4, p. 1713–1730, 2010. Disponível em: <https://doi.org/10.1137/09077638X>.
- CALIK, H.; LABBÉ, M.; YAMAN, H. p -center problems, g. laporte, s. nickel, f. saldanha da gama (eds.). In: SPRINGER (Ed.). *Location Science*. [S.l.]: Springer International Publishing Switzerland, 2015.
- CAMICIA, R.; VICENTE, A. Grafos e madiana. In: *Anais da XXII Semana Acadêmica de Matemática da Unioeste*. Cascavel, PR - Brasil: [s.n.], 2008.
- COGIS, O.; ROBERT, C. *Théorie des Graphes: Au-delà des Ponts de Königsberg, Problèmes, Théorèmes, Algorithmes*. [S.l.]: Vuibert, 2003.
- CORMEN, T. H. et al. *Introduction to Algorithms, Third Edition*. 3rd. ed. [S.l.]: The MIT Press, 2009. ISBN 0262033844.
- DASKIN, M.; MAASS, K. The p -median problem. In: LAPORTE, G.; NICKEL, S.; GAMA, F. S. da (Ed.). *Location Science*. Switzerland: Springer International Publishing, 2015.
- GARCIA, V. et al. Grasp para o problema de planejamento de redes ssecundárias de distribuição de energia elétrica. In: *Anais do XXXV SBPO*. Natal, RN - Brasil: [s.n.], 2003. p. 1427–1437.
- GAREY, M.; JOHNSON, D. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. [S.l.]: W. H. Freeman, 1979. ISBN 0-7167-1044-7.
- GOLDMAN, A. Optimal center location in simple networks. *Transportations Sci.*, v. 2, p. 77–91, 1962.
- GOLDMAN, A. Optimal center location in simple networks. *Transportations Sci.*, v. 5, p. 212–221, 1971.
- GØRTZ, I. L.; WIRTH, A. Asymmetry in k -center variants. *Theoret. Comput. Sci*, v. 361, p. 188–199, 2006.
- HAKIMI, S. Optimum distribution of switching centers in a communication network and some related graph theoretic problems. *Operations Research*, v. 13, p. 462–475, 1965.
- HAKIMI, S.; SCHMEICHEL, E.; PIERCE, J. On p -centers in networks. *Transportation Sci.*, v. 12, p. 1–15, 1978.
- HANDLER, G.; MIRCHANDANI, P. *Location on Networks: Theory and Algorithms*. [S.l.]: The MIT Press, 1979.
- HARARY, F. *Graph Theory*. [S.l.]: Addison-Wesley — Reading, MA, 1969.
- HARTMANN, T. A.; LENDL, S.; WOEGINGER, G. J. Continuous facility location on graphs. In: BIENSTOCK, D.; ZAMBELLI, G. (Ed.). *Integer Programming and Combinatorial Optimization*. [S.l.]: Springer, 2020. p. 171–181.

- HUA, L. Applications of mathematical models to wheat harvesting (english translation in *chin. math.* 2 (1962), 77-91). *Acta Mathematica Sinica*, v. 11, p. 63–75, 1961.
- JAEGER, M.; GOLDBERG, J. A polynomial algorithm for the equal capacity p -center problem on trees. *Transportation Sci.*, v. 28, p. 167–175, 1994.
- KARIV, O.; HAKIMI, S. An algorithmic approach to network location problems. part 1: The p -centers. *SIAM J. Appl. Math.*, v. 37, p. 513–538, 1979.
- KARIV, O.; HAKIMI, S. An algorithmic approach to network location problems. part 2: The p -medians. *SIAM J. Appl. Math.*, v. 37, p. 539–560, 1979.
- LIU, Y. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones. *Computers & Operations Research*, v. 111, p. 1–20, 2019.
- MEGIDDO, N. Linear-time algorithms for linear programming in \mathbb{R}^3 and related problems. *SIAM J. Comput.*, v. 12, p. 759–776, 1983.
- NASCIMENTO, I. *Algoritmos de Particionamento e de Localização em Árvores — Monografia de Graduação. Bacharelado em Matemática Aplicada e Computacional*. Dissertação (B.S. Thesis) — Instituto Multidisciplinar - Universidade Federal Rural do Rio de Janeiro, 2018.
- NGUYEN, K.; ANH, L. Inverse k -centrum problem on trees with variable vertex weights. *Mathematical Methods of Operational Research*, v. 82, 05 2015.
- NGUYEN, K.; NGUYEN-THU, H.; HUNG, N. Combinatorial algorithms for the uniform-cost inverse 1-center problem on weighted trees. *Acta Mathematica Vietnamica*, v. 44, 08 2018.
- PLESNÍK, J. A heuristic for the p -center problems in graphs. *Discrete Applied Mathematics*, v. 17, n. 3, p. 263–268, 1987. ISSN 0166-218X. Disponível em: <https://www.sciencedirect.com/science/article/pii/0166218X87900291>.
- PUERTO, J.; RODRÍGUEZ-CHÍA, A.; TAMIR, A. On the planar piecewise quadratic 1-center problem. *Algorithmica (New York)*, v. 57, p. 252–283, 02 2010.
- RABUSKE, M. A. *Introdução à Teoria dos Grafos*. [S.l.]: Editora da UFSC, 1992.
- SILVA, M.; FRANCA, P.; SILVEIRA, P. Long-range planning of power distribution systems: Secondary networks. *Computers & Electrical Engineering*, v. 22, p. 179–191, 1996.
- TAMIR, A. An $o(pn^2)$ algorithm for the p -median and related problems on tree graphs. *Oper. Res. Lett.*, v. 19, p. 59–64, 1996.
- TANSEL, B.; FRANCIS, R.; LOWE, T. Location on networks: A survey. part i: The p -center and p -median problems. *Management Science*, v. 29, p. 482–497, 1983.
- TANSEL, B.; FRANCIS, R.; LOWE, T. Location on networks: A survey. part ii: Exploiting tree network structure. *Management Science*, v. 29, p. 498–511, 1983.
- WANG, H.; ZHANG, J. Computing the center of uncertain points on tree networks. *Algorithmica*, v. 78, 05 2016.
- WANG, H.; ZHANG, J. An $o(n \log n)$ -time algorithm for the k -center problem in trees. *CoRR*, abs/1705.02752, 2017. Disponível em: <http://arxiv.org/abs/1705.02752>.
- YE, J.-H.; LI, C.-Y.; WANG, B.-F. An improved algorithm for the minmax regret path centdian problem on trees. *Journal of Computer and System Sciences*, v. 97, 06 2018.

YU, H.; LI, C.; LEE, D. T. The multi-service center problem. *Theor. Comput. Sci.*, v. 705, p. 58–74, 2018. Disponível em: <https://doi.org/10.1016/j.tcs.2017.09.034>.

YU, H.; LIN, T.; WANG, B. Improved algorithms for the minmax-regret 1-center and 1-median problems. *ACM Trans. Algorithms*, v. 4, n. 3, p. 36:1–36:27, 2008. Disponível em: <https://doi.org/10.1145/1367064.1367076>.