O Problema do Corte Mínimo de Arestas como Medida de Desempenho de uma Rede de Computadores

Ygor Santos Vieira
CENTRO FEDERAL DE EDUCAÇÃO TECNOLOGICA DE MINAS GERAIS
Campus V
Divinópolis, Brasil
ygorvieira111@gmail.com

***Abstract*—Este trabalho abordou a gestão eficiente de redes de computadores, destacando a importância da topologia na determinação do desempenho do sistema. O foco foi o problema do corte mínimo de arestas em grafos, fundamental na otimização de redes. Foram comparadas abordagens heurísticas, centradas em vértices de menor e maior grau, com o algoritmo exato de Stoer-Wagner. Os resultados indicaram que heurísticas com vértices de menor grau apresentaram soluções mais eficazes, enquanto Stoer-Wagner ofereceu os cortes mínimos mais precisos, embora com tempos de execução mais longos. Experimentos com grafos reais sugeriram a aplicabilidade prática desses algoritmos, proporcionando *insights* valiosos para a análise de redes de computadores reais.**

***Index Terms*—Redes de Computadores, Teoria dos Grafos, Corte Mínimo de Arestas**

I. Introdução

A gestão eficiente de redes de computadores tornou-se uma preocupação crucial na era da conectividade digital. Nesse cenário, a topologia da rede desempenha um papel fundamental na determinação do desempenho e da robustez do sistema. Um dos desafios centrais enfrentados pelos administradores de redes é otimizar a conectividade entre os diversos nós, minimizando custos e maximizando a eficiência.

A alta conectividade é um dos pré-requisitos em um mundo globalizado, onde as informações devem propagar em milissegundos para todo o planeta. Porém, a queda de links em uma rede de computadores podem provocar uma desconexão na rede, fazendo com que seus usuários percam acesso a recursos. Neste contexto, uma importante medida para mensurar a estabilidade da rede é determinar o número mínimo de links que podem cair e causar uma desconexão. Quanto maior for esse valor menor a probabilidade de haver uma desconexão na rede e mais eficiente a mesma será [1].

Além disso, esta medida pode ser utilizada de forma a ser monitorada dentro de uma rede fazendo com que os administradores da rede possam ter como métrica para tomada de decisão, tais como remover ou adicionar novos links gerando redundâncias e garantindo estabilidade da rede.

Este problema de determinar o número de quedas de links necessários para desconectar uma rede é um problema estudado pela área de Teoria dos Grafos. O mesmo pode ser modelado a partir de um grafo, onde os vértices representam *hosts* e as arestas são os links (conexões) entre os *hosts*. Neste caso, queremos determinar o número mínimo de arestas que precisam ser removidas para desconectar o grafo. Este problema é bastante explorado na literatura. Existem trabalhos que exploram algoritmos para o problema e outros se concentram em determinar o valor exato deste parâmetro para famílias de grafos [3].

Neste trabalho exploramos o algoritmo de Corte Mínimo / Fluxo Máximo de Ford-Fulkerson para resolver o problema. O objetivo deste trabalho é verificar o desempenho do algoritmo de Ford-Fulkerson quando fixamos os parâmetros s e t do s,t-corte a ser realizado utilizando duas estratégias gulosas e comparando com o algoritmo de Stoer-Wagner. Além disso, avaliar o corte mínimo de arestas em uma rede de computadores real afim de avaliar a viabilidade destes algoritmos em situações reais.

II. Corte Mínimo de Arestas

O grafo é uma estrutura matemática que nos permite modelar diferentes problemas. Esta estrutura consiste em um par ordenado $G(V,E)$, onde V é o conjunto de vértices e E, o conjunto de arestas. Um vértice $u \in V$ representa um objeto e uma aresta $uv \in E$ representa uma relação entre dois objetos do conjunto V.

Um caminho em um grafo consiste em uma sequência de vértices $v_1 v_2 \cdots v_i$ de forma que para todo $j \in [1, i-1]$, temos que $v_j v_{j+1} \in E$, ou seja, temos uma aresta que conecta dois vértices consecutivos. Um grafo é dito conexo quando para todo par de vértices $u, v \in V$ existe um caminho $P \subseteq E$ que conecta estes vértices. Caso contrário, o grafo é dito desconexo.

Um corte de arestas em um grafo consiste um conjunto de arestas $M \subseteq E$ tal que quando removemos estas arestas transformamos o grafo em desconexo. O corte mínimo de arestas em um grafo G consiste em determinar $M \subseteq E$ tal que para todo corte de aresta $M' \subseteq E$, temos que $|M| \leq |M'|$. A Figura 1 apresenta um grafo inicial conexo. As arestas destacadas representam o corte mínimo de arestas, que no caso é igual a 2, deste grafo. A Figura 2 apresenta o mesmo grafo após a remoção do corte de arestas.

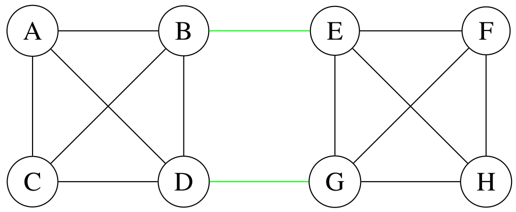

Fig. 1. Grafo G conexo

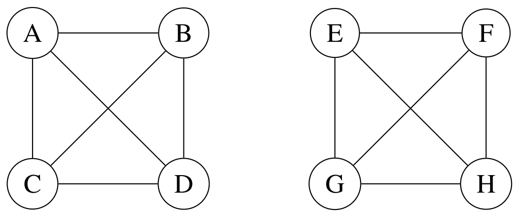

Fig. 2. Grafo G após corte mínimo de arestas

O algoritmo de Stoer-Wagner é um algoritmo utilizado na teoria dos grafos para calcular o número mínimo de arestas de corte em um grafo. A ideia básica do algoritmo é contrair repetidamente o grafo até que ele se reduza a um único vértice, mantendo o controle das arestas que são contraídas. O custo mínimo de corte é então encontrado considerando as arestas que foram contraídas durante esse processo [2].

O algoritmo Stoer-Wagner utilizado é implementado pela biblioteca Networkx. A versão utilizada neste trabalho possui complexidade $O(n(n+m)\log n)$, onde $n = |V|$ e $m = |E|$.

Um s,t-corte em um grafo consiste em um conjunto de arestas $M \subseteq E$ tal que ao remover M desconectamos os vértices s e t, ou seja, não existirá mais um caminho que interligue os vértices s e t no grafo. Um s,t-corte $M \subseteq E$ é dito mínimo quando para todo s,t-corte $M' \subseteq E$, temos que $|M| \leq |M'|$. Para determinar o corte mínimo de aresta, basta calcular para cada par de vértices $s,t \in V$ o seu s,t-corte mínimo e retornar o menor valor encontrado. Quando fixamos os vértices s e t e calculamos o s,t-corte, temos que este valor é um limite superior para o corte mínimo de arestas, ou seja, temos um procedimento heurístico para o problema.

O algoritmo de Ford-Fulkerson é amplamente conhecido por resolver o problema do fluxo máximo em uma rede, mas sua aplicabilidade vai além disso [4]. Pode ser adaptado para encontrar um corte mínimo entre dois vértices específicos, s e t, em um grafo.

A ideia central do algoritmo envolve o aumento iterativo do fluxo ao longo dos caminhos disponíveis até que não seja mais possível encontrar um caminho adicional. Durante esse processo, os vértices são categorizados em dois conjuntos distintos, S e T, onde $s \in S$ e $t \in T$ [5]. O s,t-corte mínimo é, então, composto pelas arestas que conectam os vértices de S aos de T. O algoritmo de Ford-Fulkerson é apresentado no Algoritmo 1.

Neste trabalho implementamos o algoritmo de Ford-Fulkerson para corte mínimo utilizando a linguagem de programação Python. O algoritmo implementado possui complexidade $O(nm^2)$, onde $n = |V|$ e $m = |E|$.

Algorithm 1 Algoritmo de Ford-Fulkerson para um s,t-corte

```
Procedimento FordFulkerson(grafo, s, t):
    Inicialize residual com as mesmas
    arestas do grafo original

    Enquanto existir um caminho P-aumentante:
        Encontre a capacidade mínima c_f
        ao longo do caminho P

        Aumente o fluxo ao longo do caminho P
        por c_f

        Atualize o grafo residual

    Calcule o corte mínimo minCut usando o
    grafo residual final

    Retorne minCut
```

III. Metodologia

Os experimentos computacionais realizados consistem na comparação de algoritmos para calcular o corte mínimo em grafos e a análise do algoritmo exato em uma rede de computadores real. Neste trabalho foi utilizado dois métodos para resolução do problema de corte mínimo:

- Algoritmos Heurísticos (*Heuristic Cut*): Foram implementados dois algoritmos heurísticos baseados no algoritmo Ford-Fulkerson fixando os vértices s e t para calcular o corte mínimo no grafo. Uma das abordagens toma s e t como os vértices de maior grau e a outra abordagem adotam estes vértices como os de menor grau. É esperado que abordagem de menor grau apresente resultados melhores que a abordagem de maior grau. Estas escolhas foram para avaliar como uma escolha ruim destes vértices pode piorar consideravelmente a solução.
- Algoritmo Exato (*Exact Cut*): O algoritmo exato utilizado é o stoer_wagner da biblioteca NetworkX, que é uma implementação do algoritmo de Stoer-Wagner para encontrar o corte mínimo em um grafo.

Os experimentos iniciais foram conduzidos em grafos gerados com diferentes números de vértices $\{10, 20, 50, 100, 200, 500, 1000\}$ e densidades $\{0.3, 0.5, 0.7, 0.9\}$. A densidade representa a probabilidade de existência de uma aresta entre dois vértices.

Após isso, foi realizado o uso do *dataset* Cisco Secure Workload Networks of Computing Hosts disponível na plataforma UC Irverine Machine Learning Repository. Este conjunto de dados consiste em 22 grafos distintos que representam a comunicação de rede de aplicativos distribuídos.

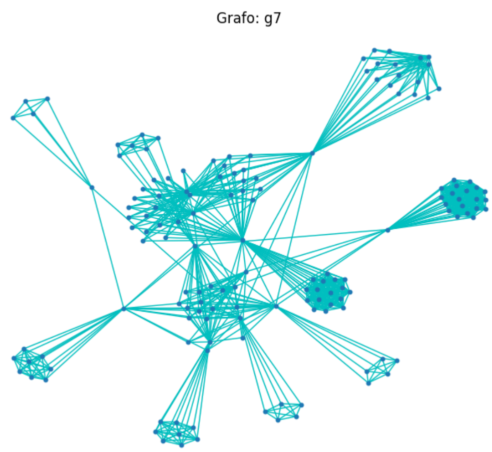

Fig. 3. Grafo $g7$.

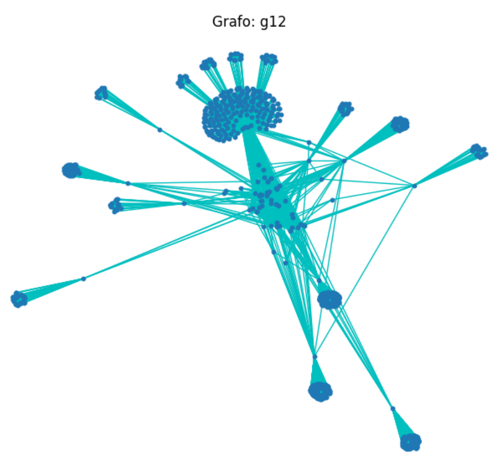

Fig. 4. Grafo $g12$.

Os grafos abrangem diferentes períodos de coleta, alguns ao longo de vários dias em diferentes meses. Foi realizado a extração das arestas por grafos unindo os diversos dias de coleta. Embora o *dataset* contenha 22 grafos, apenas dois deles ($g7$ e $g12$) são conexos e, por tanto, são considerados viáveis para os nossos testes. Os grafos $g7$ e $g12$ são apresentados nas Figuras 3 e 4, respectivamente.

A metodologia deste trabalho pode ser dividida da seguinte forma:

- **Geração de Grafos Aleatórios**: Um grafo foi gerado usando o modelo Watts-Strogatz da biblioteca NetworkX que garante que o grafo retornado é conexo. O modelo de Watts–Strogatz, proposto por Duncan J. Watts e Steven H. Strogatz, é um modelo de geração de grafos que combina estruturas regulares com elementos aleatórios para capturar propriedades de redes complexas.
- **Comparação entre Algoritmos**: Os algoritmos heurísticos foram aplicados ao grafos aleatórios gerados, considerando os vértices de maior e menor grau. O tamanho do corte mínimo e o tempo de execução foram registrados. O mesmo foi realizado com o algoritmo exato de stoer_wagner da biblioteca NetworkX.
  Os resultados dos experimentos são apresentados para cada combinação de número de vértices e densidade, mostrando o tamanho do corte mínimo e o tempo de execução para os algoritmos heurísticos e exato. Isso permite a análise comparativa do desempenho desses algoritmos em diferentes tipos de grafos. Os experimentos foram projetados para fornecer *insights* sobre a eficácia dos algoritmos heurísticos em comparação com o algoritmo exato, considerando diferentes características dos grafos.
- **Obtenção de Dados Reais**: Nesta etapa foi realizado a extração e processamentos dos grafos do *dataset* Cisco Secure Workload Networks of Computing Hosts. Inicialmente extraímos informações de arestas de cada grafo durante o período de tempo disponibilizado. Por fim, verificamos quais deles eram viáveis para nosso experimento, ou seja, quais deles são conexos. Por fim, armazenamos os grafos em aquivos .csv para uma posterior utilização destes grafos.
- **Análise dos Grafos Reais**: Por fim, utilizamos o algoritmo exato para verificar a viabilidade de se calcular um corte mínimo de arestas em uma rede de computadores real. Para isto, foi utilizado os grafos filtrados na etapa anterior.

Os experimentos foram implementados na linguagem Python e executados em computador com processador Intel® Core™ i5-4210U CPU @ 1.70GHz $\times$ 4 com o sistema operacional Ubuntu 20.04.

IV. RESULTADOS

A Tabela I apresenta os resultados obtidos nos experimentos computacionais que tiveram como objetivo comparar as abordagens heurística com a abordagem exata tendo como entrada grafos gerados de forma aleatória seguindo o modelo de Watts-Strogatz.

Os resultados obtidos nestes experimentos computacionais oferecem uma análise perspicaz do desempenho de diferentes abordagens para resolver o problema do corte mínimo em grafos. Inicialmente, ao examinar os cortes gerados pela heurística baseada nos vértices de maior grau, observamos, de maneira geral, que essa estratégia produziu soluções com tamanhos maiores em comparação com a heurística que emprega os vértices de menor grau. Esse padrão sugere que a abordagem centrada em vértices de maior grau pode resultar em cortes menos eficazes na minimização do número de arestas de corte.

Por outro lado, a heurística que utiliza vértices de menor grau resultou em soluções de corte mínimo mais eficazes, in-

n	p	Menor Grau		Maior Grau		Exato	
		Sol	Temp	Sol	Temp	Sol	Temp
10	0.3	3	0.000268	5	0.000344	3	0.209403
10	0.5	2	0.000252	5	0.000336	2	0.000735
10	0.7	2	0.000248	5	0.000283	2	0.000735
10	0.9	3	0.000260	5	0.000283	3	0.000779
20	0.3	5	0.000780	10	0.000959	5	0.002957
20	0.5	6	0.000866	11	0.001247	6	0.002851
20	0.7	6	0.000818	10	0.000962	6	0.002918
20	0.9	5	0.000776	10	0.000958	5	0.003003
50	0.3	16	0.005917	23	0.007263	16	0.032681
50	0.5	15	0.005793	24	0.007393	15	0.031826
50	0.7	14	0.005439	26	0.007901	14	0.034608
50	0.9	13	0.005342	25	0.008365	13	0.033418
100	0.3	34	0.033971	46	0.041497	34	0.212553
100	0.5	31	0.031933	48	0.042681	31	0.216776
100	0.7	30	0.031298	49	0.044820	30	0.218098
100	0.9	31	0.031930	47	0.042043	31	0.217758
200	0.3	69	0.208754	91	0.258591	69	1.706527
200	0.5	68	0.214308	91	0.269237	68	1.732046
200	0.7	68	0.208417	93	0.264862	68	1.795014
200	0.9	66	0.201100	94	0.265188	66	1.792013
500	0.3	179	3.126223	218	3.747193	179	30.155173
500	0.5	177	3.088967	224	3.886657	177	30.397614
500	0.7	170	2.983887	224	3.865233	170	31.008552
500	0.9	178	3.142392	222	3.819204	178	30.713651
1000	0.3	364	26.005245	428	30.992192	364	233.659388
1000	0.5	368	25.940372	436	30.330534	368	243.166811
1000	0.7	360	25.395654	439	30.734200	360	248.161657
1000	0.9	356	25.279258	439	30.726058	356	248.674356

TABLE I
RESULTADOS COMPUTACIONAIS OBTIDOS

dicando sua possível superioridade na minimização do número de corte de arestas em um grafo. Além disso, os tempos de execução foram muito inferiores quando comparados à abordagem exata. Essa eficiência na obtenção de soluções de corte mínimo em um intervalo de tempo relativamente curto destaca a praticidade dessa heurística, especialmente em contextos onde a precisão da solução é crucial, mas o tempo de computação deve ser minimizado.

Ao compararmos as heurísticas com a solução exata do problema, utilizando o algoritmo Stoer-Wagner, fica evidente que as soluções exatas proporcionam os menores cortes, demonstrando a superioridade desse algoritmo em termos de qualidade da solução. No entanto, essa precisão vem acompanhada de tempos de execução consideravelmente mais longos. Portanto, a escolha entre abordagens heurísticas e exatas dependerá das necessidades específicas do problema, equilibrando a precisão desejada em relação ao tempo disponível para a computação.

Após esta etapa, foi realizado uma avaliação experimental sobre grafos reais de redes de computadores. Para isto, foi utilizado o algoritmo exato de Stoer Wagner para determinar o corte mínimo de arestas dos grafos $g7$ e $g12$. As Figuras 5 e 6 apresentam o corte obtido nestes grafos, onde as partições dos vértices são representadas pelas cores azul e amarelo, enquanto o corte mínimo determinado é destacado em vermelho.

A Tabela II apresenta informações relevantes sobre os grafos, tais como: número de vértices (n), número de arestas (m), grau máximo (Δ), grau mínimo (δ), raio e diâmetro. Além disso, são apresentados a solução e o tempo de execução

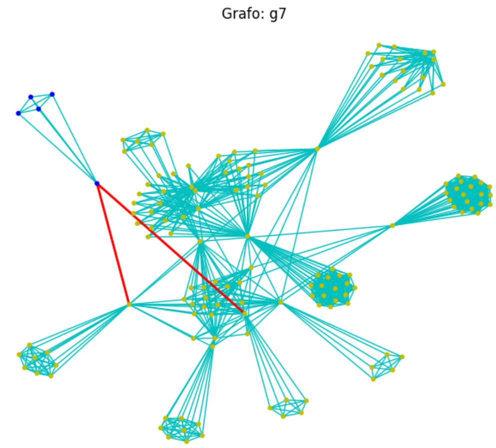

Fig. 5. Corte de Arestas Mínimo no Grafo $g7$.

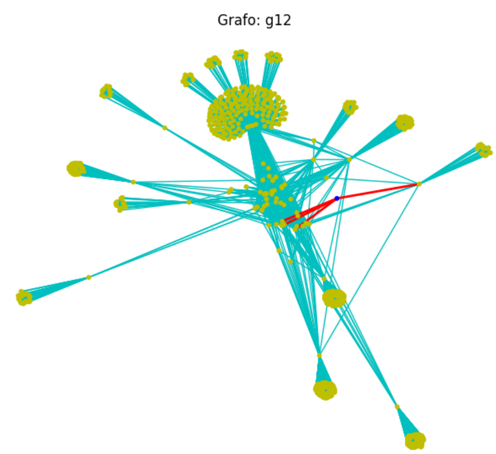

Fig. 6. Corte de Arestas Mínimo no Grafo $g12$.

para obter o corte mínimo de arestas em cada grafo.

Informações	**g7**	**g12**
n	148	1733
m	788	247300
Δ	50	486
δ	4	4
d	0.0724	0.1648
Raio	4	3
Diâmetro	6	5
Corte Mínimo	2	4
Tempo (s)	0.1929	483.2761

TABLE II
RESULTADOS DOS CORTES MÍNIMOS E TEMPOS DE EXECUÇÃO PARA G7 E G12.

A análise dos parâmetros coletados revela que ambos os grafos possuem propriedades distintas, sendo o grafo $g12$ é substancialmente maior e mais complexo que o grafo $g7$. O tempo de execução para os cortes mínimos é destacado, demonstrando eficiência computacional com valores inferiores a 10 minutos. Esses resultados sugerem a aplicabilidade prática dessa técnica em cenários do mundo real, proporcionando *insights* valiosos para a análise e otimização de redes complexas.

V. Conclusões

O presente trabalho abordou a problemática do corte mínimo de arestas em grafos, uma questão fundamental na otimização de redes de computadores. A gestão eficiente dessas redes é crucial na era da conectividade digital, onde a topologia do sistema desempenha um papel vital. Os objetivos do trabalho foi avaliar o desempenho de diferentes abordagens para resolver esse problema e verificar a viabilidade da utilização de corte mínimo de arestas em redes de computadores reais.

Os experimentos computacionais revelaram *insights* valiosos. Heurísticas centradas em vértices de menor grau demonstraram eficácia, proporcionando soluções mais eficientes em termos de corte mínimo. Embora heurísticas centradas em vértices de maior grau tenham apresentado soluções ligeiramente maiores, destacou-se pela rapidez de execução. Por outro lado, o algoritmo exato Stoer-Wagner proporcionou os cortes mínimos mais precisos, embora com tempos de execução mais longos.

É interessante notar que, embora a precisão do algoritmo exato seja alta, os tempos de execução mais longos podem representar um desafio em termos de escalabilidade em cenários de redes de grande escala. No entanto, os resultados dos experimentos com grafos reais sugerem que os algoritmos de corte mínimo de arestas têm aplicabilidade prática, permitindo a antecipação e a identificação de possíveis falhas ou vulnerabilidades na rede.

Dessa forma, o estudo não apenas avaliou diferentes abordagens para resolver o problema do corte mínimo de arestas, mas também ressaltou sua relevância e viabilidade no contexto das redes de computadores reais. Estes resultados podem servir de base para o desenvolvimento e implementação de estratégias mais eficazes na gestão e otimização de redes de computadores, visando a melhoria da sua robustez e desempenho.

VI. Trabalhos Futuros

Como trabalhos futuros sugerimos:

- Uma maior investigação da precisão da abordagem heurística de utilizar definir o corte mínimo de arestas através de um s, t-corte, onde s e t são vértices de menor grau do grafo.
- A implementação de um sistema que permita avaliar o desempenho de uma rede que esteja em constantes mudanças, tais como, adição/remoção de hosts e adição/remoção de links baseado nos algoritmos de corte mínimo de arestas abordados neste trabalho.

References

[1] David R. KARGER. Minimum cuts in near-linear time. Journal of the ACM (JACM), v. 47, n. 1, p. 46-76, 2000.

[2] Srinivasa R. ARIKATI.; Kurt. MEHLHORN. A correctness certificate for the Stoer–Wagner min-cut algorithm. Information Processing Letters, v. 70, n. 5, p. 251-254, 1999.

[3] David R. KARGER. Using randomized sparsi cation to approximate minimum cuts. In: Proc. 5th Symp. on Discrete Algorithms. p. 432. 1994.

[4] Thomas H. CORMEN et al. Algoritmos: teoria e prática. Editora Campus, v. 2, p. 296, 2002.

[5] Noraini ABDULLAH; Ting K. HUA. Using ford-fulkerson algorithm and max flow-min cut theorem to minimize traffic congestion in kota kinabalu, sabah. J. Inf, v. 2, n. 4, p. 18-34, 2017.

VII. Link Repositório GitHub

https://github.com/eplaie/O-Problema-do-Corte-M-nimo-de-Arestas-como-Medida-de-Desempenho-de-uma-Rede-de-Computadores.