

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA
CÂMPUS DE ILHA SOLTEIRA**

Matheus Nascimento Fernandes

**Análise de Desempenho de Algoritmos de Busca em Grafos para
Processamento de Topologia em Redes de Distribuição de Energia**

Ilha Solteira - SP
2023

PROGRAMA DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

MATHEUS NASCIMENTO FERNANDES

**ANÁLISE DE DESEMPENHO DE ALGORITMOS DE BUSCA EM GRAFOS PARA
PROCESSAMENTO DE TOPOLOGIA EM REDES DE DISTRIBUIÇÃO DE
ENERGIA**

Trabalho de conclusão de curso
apresentado à Faculdade de Engenharia de
Ilha Solteira – Unesp como parte dos
requisitos para obtenção do título de
bacharel em Engenharia Elétrica.

Jonatas Boas Leite
Orientador

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

F363a Fernandes, Matheus Nascimento.
Análise de desempenho de algoritmos de busca em grafos para
processamento de topologia de redes de distribuição de energia / Matheus
Nascimento Fernandes. -- Ilha Solteira: [s.n.], 2023
47 f. : il.

Trabalho de conclusão de curso (Graduação em Engenharia Elétrica) -
Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira, 2023

Orientador: Jonatas Boas Leite

Inclui bibliografia

1. Sistemas de distribuição. 2. Sistema de Gerenciamento de distribuição. 3.
Processamento de topologia. 4. Grafos.

ATA DE DEFESA DE TRABALHO DE GRADUAÇÃO

Aos trinta e um dias do mês de junho do ano de dois mil e vinte e três, o discente **Matheus Nascimento Fernandes**, matriculado sob o nº 162053657, tendo como banca examinadora o seu orientador, o Prof. Dr. Jonatas Boas Leite, o Prof. Dr. Renzo Amilcar Vargas Peralta e o Doutorando Cleberton Reiz, apresentou o Trabalho de Graduação intitulado "Análise de Desempenho de Algoritmos de Busca em Grafos para Processamento de Topologia em Redes de Distribuição de Energia", obtendo a nota 10,0 (DEZ) e conceito APROVADO.



Prof. Dr. Jonatas Boas Leite

- Orientador -



Matheus Nascimento Fernandes

- Discente -



Prof. Dr. Renzo Amilcar Vargas Peralta

- Membro da Banca -



Doutorando Cleberton Reiz

- Membro da Banca -

AGRADECIMENTOS

Primeiro quero agradecer a Deus, pela oportunidade, sabedoria e tudo a qual tive a chance de viver. Também gostaria de agradecer em especial aos meus pais, Roosevelt e Edna, por terem sonhado juntamente comigo, me apoiado durante esta árdua caminhada e terem feito tudo o que foi possível por mim. Aos meus irmãos, cunhados e sobrinhos, pois cada um, em algum momento contribuiu para a realização deste sonho. E a minha namorada, a qual dividiu comigo os últimos anos dessa caminhada, me apoiando e torcendo pelo meu sucesso, sempre.

Aos meus amigos, de turma, de república e colegas da faculdade que tive o prazer de conhecer e compartilhar momentos marcantes. Foram vocês que durante essa jornada estiveram mais próximos fisicamente, tornarem-se a minha família, foram vocês que deixaram essa caminhada mais leve e divertida, enfim fizemos esses anos de faculdade se tornarem um dos melhores períodos de minha vida.

Ao meu orientador de estágio e de trabalho de graduação, Jonatas, que me ensinou, instruiu e me direcionou para que nessa etapa final da jornada de graduação eu alcançasse êxito e sucesso, sempre com muita paciência e dedicação, assim como todos os outros docentes do curso de Engenharia Elétrica da UNESP de Ilha Solteira a qual contribuíram e muito para minha formação.

E em especial, agradeço a memorável UNESP da cidade de Ilha Solteira, por ter proporcionado momentos admiráveis, de felicidade e de desafios pessoais, por ter proporcionado ensino de altíssima qualidade a milhares de estudantes, um dos quais tive a imensa felicidade de ser. E que com grande carinho e admiração irei carregar comigo o nome desta incrível instituição, fica aqui o meu muito obrigado!

“A vida é o exercício mágico e transcendente de se reinventar a cada sofrimento”. Ana Beatriz Barbosa Silva.

RESUMO

Nos últimos anos a energia elétrica vem sendo cada vez mais utilizada e difundida na sociedade moderna, tornando-se um recurso essencial ao bem-estar do ser humano. A ampla utilização desse recurso traz inúmeros desafios a todo o sistema elétrico de potência. Particularmente, no caso dos sistemas de distribuição de energia, a alta demanda por esse recurso exige sistemas de distribuição com altos níveis de confiabilidade e disponibilidade. Isso significa que a estrutura de operação e controle dos sistemas de distribuição necessita ser cada vez mais robusta e confiável. Esse é o papel de um sistema de gerenciamento da distribuição (DMS) que deve garantir confiabilidade da operação, qualidade da energia fornecida, níveis aceitáveis de tensão, perdas mínimas e provisão de qualquer ação normal ou emergencial para garantir a integridade do sistema de distribuição. Para atingir esses requisitos, o DMS é dotado de vários aplicativos, cada um deles com funções únicas e específicas. Esses aplicativos vêm de um conjunto de ferramentas de modelagem e análise do sistema de distribuição de energia elétrica, sendo uma delas a ferramenta de análise de topologia, ou processador de topologia. Portanto, esse trabalho de graduação busca estudar e analisar algoritmos de busca em grafos aplicados às redes de distribuição, visando o desenvolvimento do processamento de topologia, e melhorar o desempenho dessa ferramenta em todo o gerenciamento dos sistemas de distribuição.

Palavras-chave: Sistemas de distribuição, Sistema de gerenciamento da distribuição, Processamento de topologia, Grafos.

ABSTRACT

Currently, electrical energy has been increasingly used and spread in modern society, become an essential resource for human well-being. The wide use of this resource brings numerous challenges to the entire electrical power system. Particularly, in the case of energy distribution systems, the high demand for this resource requires a distribution system with high levels of reliability and availability. This means that the structure operating and controlling the distribution systems needs to be increasingly robust and reliable. This is the role of a distribution management system (DMS), which must guarantee the reliability of the operation, the quality of the energy supplied, acceptable voltage levels, minimum losses and provide any normal or emergency action to guarantee the integrity of the distribution system. To meet these requirements, the DMS is equipped with several applications, each with unique and specific functions. These applications come from a set of power distribution system modeling and analysis tools, one of which is the topology analysis tool, or topology processor. Therefore, this graduation work seeks to study and analyze search algorithms in graphs applied to distribution networks, aiming at the development of topology processor and to improve the performance of this tool throughout the management of distribution systems.

Keywords: Distribution systems, Distribution management system, Topology Processing, Graphs.

LISTA DE FIGURAS

Figura 1	- Representação do sistema elétrico de potência.....	16
Figura 2	- Matriz de geração energética no Brasil.....	17
Figura 3	- Estrutura do sistema de gerenciamento de distribuição (DMS)....	19
Figura 4	- Exemplo prático de um grafo.....	21
Figura 5	- Diagrama de classe de rede de distribuição.....	26
Figura 6	- Rede de distribuição com 136 barras.....	29
Figura 7	- Variação do tempo de execução de acordo com quantidade de execuções do algoritmo para método de busca em largura.....	30
Figura 8	- Variação do tempo de execução de acordo com quantidade de execuções do algoritmo para método de busca em profundidade	32
Figura 9	- Comparação entre o tempo de processamento de ambos os métodos.....	33
Figura 10	- Diagrama de classe para obtenção das seções da rede	34
Figura 11	- Identificação das chaves seccionadoras presentes na rede de distribuição.....	35
Figura 12	- Diagrama de seções da rede de distribuição.....	37
Figura 13	- Seções da rede de distribuição e alimentadores laterais	38
Figura 14	- Diagrama de classes com chave de transferência	39
Figura 15	- Identificação do estado das seções da rede	41

LISTA DE TABELAS

Tabela 1	- Tempo de Execução do Algoritmo de Busca em Largura	30
Tabela 2	- Tempo de Execução do Algoritmo de Busca em Profundidade ...	31
Tabela 3	- Comparação entre Carga e Capacidade dos Alimentadores	40
Tabela 4	- Seções interrompidas e não interrompidas de acordo com a falta	42

LISTA DE ALGORITMOS

Algoritmo 1	- Pseudocódigo do método de busca em profundidade	25
Algoritmo 2	- Algoritmo para o método de Busca em Profundidade	26
Algoritmo 3	- Pseudocódigo do método de busca em largura	27
Algoritmo 4	- Algoritmo para o método de Busca em Largura	28
Algoritmo 5	- Algoritmo para obtenção das seções da rede de distribuição	35
Algoritmo 6	- Algoritmo para obtenção do estado das seções	39

LISTA DE SIGLAS E ABREVIATURAS

CC	<i>Corrente Contínua</i>
CA	<i>Corrente Alternada</i>
GCOI	<i>Grupo de Coordenação de Operações Interligadas</i>
SIN	<i>Sistema Interligado Nacional</i>
ONS	<i>Operador Nacional do Sistema</i>
SEP	<i>Sistema Elétrico de Potência</i>
DMS	<i>Distribution Management System</i>
LAPSEE	<i>Laboratório de Planejamento de Sistemas de Energia Elétrica</i>
TP	<i>Processador de topologia</i>
DER	<i>Recursos energéticos distribuídos</i>

SUMÁRIO

1	INTRODUÇÃO.....	14
2	SISTEMA ELÉTRICO DE POTÊNCIA	16
2.1	Sistema de Geração de Energia	16
2.2	Sistema de Transmissão de Energia	17
2.3	Sistema de Distribuição de Energia	18
2.3.1	Sistema de Gerenciamento de Distribuição (DMS)	19
3	REPRESENTAÇÃO DE SISTEMAS DE DISTRIBUIÇÃO POR GRAFOS	20
3.1	Teoria de Grafos	20
4	PROCESSAMENTO DE TOPOLOGIA DE REDES DE DISTRIBUIÇÃO	23
4.1	Algoritmos de Busca em Grafos	24
4.1.1	Busca em Profundidade	24
4.1.2	Busca em Largura	27
5	RESULTADOS E DISCUSSÕES	29
5.1	Análise do Desempenho dos Algoritmos de Busca em Grafos	29
5.2	Aplicação do Algoritmo de Busca em Profundidade e Obtenção das Seções	33
5.3	Alimentadores Laterais	37
6	CONCLUSÃO	43
7	REFERÊNCIAS	46

1 INTRODUÇÃO

A sociedade atual atingiu uma taxa de desenvolvimento nunca visto na história da humanidade. Todo o avanço tecnológico dos últimos 10 anos supera o do século anterior. Todo esse desenvolvimento foi possibilitado em sua grande parte pela disseminação da eletricidade desde os primórdios do século XIX.

A versatilidade que a energia elétrica oferece, podendo ser facilmente convertida em outras formas de trabalho, empregada em diversas e inúmeras atividades, levou a humanidade a uma gigantesca utilização desta forma de energia. Atualmente, a eletricidade é considerada vital para o desenvolvimento de qualquer nação ao redor do mundo.

A utilização da energia elétrica de forma comercial surgiu em meados do final do século XIX. (BICHELS, 2018) Na Alemanha, ela era utilizada basicamente para iluminação pública. Com a invenção da lâmpada elétrica por Thomas Edison nos EUA, essa evolução mostrou-se ser muito mais eficiente do que as antigas lâmpadas a gás utilizadas na época, sendo esse o principal propulsor da utilização da energia elétrica em larga escala.

Os anos seguintes foram dedicados a viabilizar a utilização em grande escala dessa então nova tecnologia. A energia elétrica sob a forma de corrente contínua (CC) era o que prevalecia naquela época. Nikola Tesla trabalhou em propagar as vantagens dos sistemas em corrente alternada (CA) com seus diversos experimentos, contribuindo para o desenvolvimento e estabelecimento desses conceitos. (ZIVIC, 2023) A transmissão de energia na forma CA minimiza as perdas em relação a CC e dessa forma possibilitando que a eletricidade alcançasse maiores distâncias a partir da fonte geradora.

No Brasil, já no findar do século XIX, a cidade de Campos no Rio de Janeiro foi a primeira a possuir iluminação pública elétrica, sendo até a primeira cidade da América do Sul a ter tal forma de iluminação. A partir do fim da Segunda Guerra Mundial, o sistema elétrico nacional ganhou impulsos com a construção da primeira grande usina hidroelétrica do País, a usina de Paulo Afonso I na Bahia. Nas décadas seguintes, também foram construídas as usinas de Furnas e Três Marias. (BICHELS, 2018)

Com o desenvolvimento, crescimento e a popularização dos sistemas de energia elétrica, surgiu então a necessidade da interligação dos sistemas para

melhor aproveitamento dos recursos hídricos dos diversos sistemas isolados que, passaram a representar cargas significativas. Nessa época, diversas regiões do país tinham seus sistemas independentes e cada um deles com características distintas. O sistema do Rio de Janeiro, assim como parte do de Minas Gerais e do Espírito Santo e, também, o Rio Grande do Sul operavam com frequência de 50 Hz, enquanto os sistemas de São Paulo operavam com frequência de 60 Hz, juntamente com Belo Horizonte e o sudoeste de Minas Gerais e ainda parte do Rio de Janeiro, o Paraná e Santa Catarina. (ALBUQUERQUE, 1982)

Para regulação de um setor que era relativamente novo e estava em rápida expansão, no final da década de 1960, foi criado o Grupo de Coordenação de Operação Interligada (GCOI). Esta organização desempenhou um papel estratégico no setor de energia elétrica do Brasil. Mas somente em 1961, aconteceu uns dos marcos mais expressivos até então no setor energético nacional, foi instituída uma comissão de uniformização de frequência, a qual recomendou a adoção de 60 Hz para todo o país. (ELETROBRAS, 1977)

Efetivamente, só a partir da década de 70 que o Sistema Interligado Nacional (SIN) foi implantado. (ALBUQUERQUE, 1982) Esse processo teve início com a integração dos diversos sistemas isolados, que cada estado possuía até então, seguindo com a interligação dos sistemas de uma mesma região. O SIN foi devidamente concluído com a ligação entre as grandes regiões Sul, Sudeste, Centro-Oeste, Norte e Nordeste, por meio de grandes linhas de transmissão.

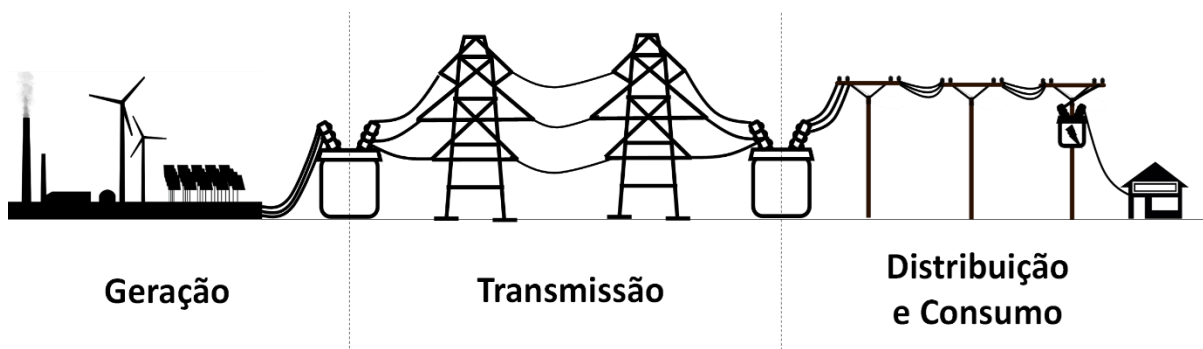
Até o ano de 2022, o Brasil detinha um gigantesco número de linhas de transmissão com mais de 175 mil quilômetros de distância de acordo com o Operador Nacional do Sistema (ONS) com tensões de 230 a 800kV em corrente alternada e frequência de 60Hz, bem como linhas de corrente contínua em 600kV e 800kV ligando algumas grandes hidroelétricas do país, como Itaipu e Belo Monte. Ao total o SIN possui mais de 4.700 usinas interconectadas e com uma capacidade instalada de mais de 180.000 MW. (ONS, 2023)

2 SISTEMA ELÉTRICO DE POTÊNCIA

O Sistema Elétrico de Potência (SEP) é o conjunto de todos os elementos, instalações e equipamentos destinados à geração, transmissão e distribuição de energia elétrica. Esse conjunto de instalações tem por objetivo primário levar a energia gerada aos consumidores. Isso é necessário pois as fontes geradoras estão predominantemente em regiões remotas, mas de grande potencial energético, bem como os grandes centros de consumo estão localizados nas cidades e metrópoles. Dessa forma, faz-se necessária a transmissão de energia entre esses pontos (BICHELS, 2018).

Na Figura 1 há uma representação dos três subsistemas do SEP que são abordados a seguir com maiores detalhes e com ênfase no subsistema de distribuição que é o enfoque deste trabalho.

Figura 1 – Representação do sistema elétrico de potência.



Fonte: Elaborada pelo Autor.

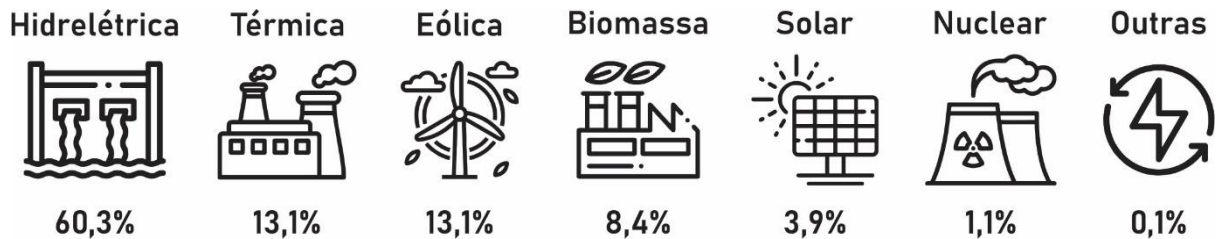
2.1 Sistema de Geração de Energia

O sistema de geração é responsável por coletar e converter uma forma de energia primária em energia elétrica. Essa conversão de energia pode ser realizada de diversas maneiras, tais como usinas hidroelétricas, termelétricas, nucleares, solares, entre outras. Particularmente, no Brasil a matriz de geração energética é predominantemente hidráulica, sendo que essa representa mais de 60% da matriz energética nacional. (ONS, 2023)

Além de usinas hidroelétricas, o Brasil também possui usinas térmicas de carvão, óleo e gás, usinas solares, eólicas, de biomassa e nuclear que compõem a

matriz energética nacional. Na Figura 2 é mostrado o percentual que cada fonte primária representou no ano de 2022.

Figura 2 – Matriz de geração energética no Brasil.



Fonte: Elaborado pelo Autor.

A operação de todas as usinas geradoras de energia é feita pelo ONS que controla a geração de energia de acordo com a atual demanda de carga do sistema elétrico e determina quais os horários e períodos que cada geradora deve operar e despachar energia no sistema.

2.2 Sistema de Transmissão de Energia

Neste subsistema, a energia gerada nas usinas é levada até os consumidores. Nesta parte do SEP, a tensão da energia é elevada. Isso ocorre, para que não haja grandes perdas energéticas durante o caminho percorrido até as unidades consumidoras.

No Brasil, as linhas de transmissão também chamada de rede básica, é formada por linhas de transmissão aéreas entre 230kV e 750kV em corrente alternada e de até 800kV de corrente contínua. Esta rede de alta tensão compõe uma malha que interliga cada usina ou subestação a outras usinas e subestações, e consequentemente as cargas consumidoras.

Atualmente a rede básica de transmissão possui aproximadamente 180 mil quilômetros de linhas de transmissão, e segundo estimativas da ONS a previsão é que até o ano de 2027 a rede básica de transmissão tenha mais de 270 mil quilômetros de linhas, o que significaria um aumento de aproximadamente 50% na malha de transmissão. (ONS, 2023)

O sistema de transmissão de energia elétrica se finda nas subestações abaixadoras, que geralmente ficam localizadas próximos aos centros de consumo, em uma cidade ou até mesmo dentro de parques industriais ou em alguns casos grandes indústrias que consomem energia elétrica diretamente da rede transmissão com suas subestações dedicadas.

2.3 Sistema de Distribuição de Energia

O sistema de distribuição é definido como a parcela do sistema elétrico de potência que entrega a energia elétrica aos consumidores e/ou usuários finais. A distribuição é realizada pelas concessionárias por meio de suas linhas áreas em média tensão após a redução da voltagem nas subestações abaixadoras.

As redes de distribuição operam com tensões trifásicas de 13,8 kV até 34,5 kV, essas são denominadas de alimentadores. Na rede de distribuição há a presença de transformadores que abaixam a tensão dos alimentadores para os níveis de consumo em 380 V, 220 V ou 127 V, que são normalmente trifásicos, mas podem existir, em algumas localidades, transformadores monofásicos.

Na composição da rede de distribuição, geralmente são utilizados alimentadores dispostos radialmente, com chaves seccionadoras em pontos estratégicos e com interligação entre eles por meio de conexão através de chave nas extremidades dos alimentadores. No entanto essas seccionadoras nas extremidades dos alimentadores operam normalmente abertas.

A conexão entre as chaves intermediárias, em pontos estratégicos, e a conexão no final dos alimentadores permite a transferência de cargas de um alimentador para outro quando ocorrem falhas nos alimentadores. Isso possibilita isolar a parte com falha sem que grande parte dos consumidores sejam afetados.

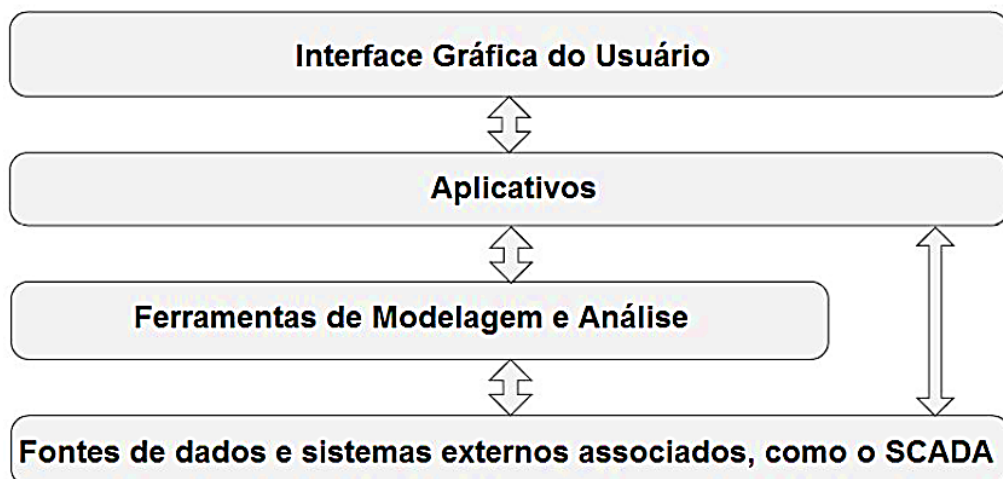
Para a operação de um sistema de distribuição de energia é utilizado um sistema de gerenciamento da distribuição (DMS), do Inglês *distribution management system*. O DMS é responsável pela confiabilidade geral da operação, qualidade do fornecimento de energia, economia da rede e todos os controles normais e de emergência do sistema de distribuição, incluindo a manutenção de um nível aceitável de tensão.

2.3.1 Sistema de Gerenciamento de Distribuição

Um DMS é um conjunto de aplicativos, cada um com funcionalidades específicas, usados pelos operadores da concessionária da rede de distribuição para monitorar e controlar o desempenho do sistema de distribuição assim como sua operação. Esse sistema tem o propósito de permitir que o sistema de distribuição proporcione confiabilidade e qualidade do serviço de fornecimento, eficiência e eficácia da operação do sistema. Além disso, o DMS deve possibilitar uma melhor gestão de ativos da rede, provisão de novos serviços ao sistema de distribuição e uma maior satisfação do cliente consumidor.

O DMS tem como estrutura fundamental uma interface gráfica com o usuário, uma série de aplicativos simples em um sistema computacional. O suporte a esses aplicativos vem de um conjunto de ferramentas de modelagem e análise do sistema de distribuição de energia elétrica, e fontes de dados e subsistemas que coletam dados em tempo real do *status* da rede de distribuição e de suas variáveis.

Figura 3 – Estrutura do sistema de gerenciamento de distribuição.



Fonte: Adaptado de Ekanayake (2012).

Uma das ferramentas de modelagem e análise do sistema de distribuição, é a ferramenta de análise de topologia, ou processamento de topologia, a qual é o objeto de estudo deste trabalho.

3 REPRESENTAÇÃO DOS SISTEMAS DE DISTRIBUIÇÃO POR GRAFOS

É possível descrever os grafos como um ótimo modelo matemático para estudo de relações entre objetos discretos de qualquer tipo. Esse princípio tem sido aplicado em diversas áreas do conhecimento, tais como, circuitos elétricos, física, química, genética, computação, arquitetura e até mesmo na análise de sistemas elétricos de potência.

A utilização de grafos para representação de sistemas de distribuição tem se mostrado muito eficiente, uma vez que os sistemas de distribuição têm grande similaridade com os princípios dessa teoria, como formação de árvores em configurações radiais.

3.1 Teoria de Grafos

De forma geral entende-se como grafo, um modelo matemático que é amplamente utilizado para determinar relações entre objetos. Esses objetos podem ser aplicados no entendimento ou solução de problemas reais, como dito anteriormente, de diversas áreas do conhecimento.

Na teoria de grafos, existem alguns elementos que são fundamentais, os quais são designados por vértices e arestas. O vértice, representa os objetos ou entidades em um grafo, também pode ser chamado de nó ou nódulo. A aresta, faz as ligações entre os vértices, demonstrando como esses vértices estão relacionados entre si, as arestas também podem ser designadas por arco.

Da literatura (MELO, 2014), tem-se a seguinte definição para Grafo:

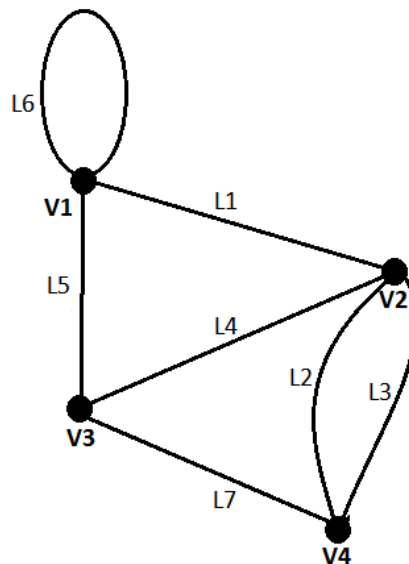
“Um grafo G , consiste em um conjunto finito e não vazio $V(G)$ de elementos chamados vértices (ou nós), e um conjunto finito $A(G)$ de pares não ordenados de elementos distintos de $V(G)$ chamados arestas (ou arcos). Denotaremos por simplicidade o grafo G como o par ordenado $G = (V; A)$, onde $V = V(G)$ e $A = A(G)$ ”

Assumindo que u e v sejam dois nós de um grafo, ou seja, $u, v \in V$, e se o par $\{u, v\} \in A$, este é uma aresta de G , dessa forma essa aresta é nomeada de L , sendo $\{u, v\}$. Dessa forma, $L = \{u; v\} \in A$, assim pode-se dizer que:

- ✓ u e v são vértices ligados pela aresta L ;
- ✓ os vértices u e v são adjacentes;
- ✓ os vértices u e v são vizinhos;
- ✓ u e L são incidentes (respectivamente para v);
- ✓ u é uma extremidade de L (respectivamente para v).

Na Figura 4 é representada de forma simplificada um grafo para exemplificar e facilitar o entendimento desse conceito. Dessa forma, o grafo $G = (V; A)$ é dado conforme a definição anterior.

Figura 4 – Exemplo prático de um grafo.



Fonte: Elaborado pelo Autor.

Sendo,

$$V = \{v1; v2; v3; v4\};$$

$$A = \{L1; L2; L3; L4; L5; L6; L7\};$$

$$L1 = \{v1; v2\}; L2 = \{v2; v4\}; L3 = \{v2; v4\}; L4 = \{v2; v3\}; L5 \\ = \{v1; v3\}; L6 = \{v1; v1\}; L7 = \{v3; v4\}.$$

Seja, então, um determinado grafo $G = (V; A)$, representando uma árvore. É conveniente considerar um vértice especial, denominado de raiz da árvore. O significado do nó raiz depende de qual é a abordagem do problema modelado pelo grafo. O nó raiz é escolhido como o "nó inicial" na árvore. De forma geral, esse nó raiz é o início da árvore ou ponto de origem.

Portanto, um sistema de distribuição é equivalente a uma árvore, onde a subestação alimentadora é equivalente a raiz dessa árvore. Em um primeiro nível, os vértices correspondem às barras de conexão das unidades consumidoras e as arestas são os circuitos de conexão entre as barras. Em um nível acima os setores, ou seções dos alimentadores representam os nós e os ramais da rede, que podem conter chaves seccionadoras, são equivalentes as arestas de um grafo.

4 PROCESSAMENTO DE TOPOLOGIA EM REDES DE DISTRIBUIÇÃO

O processador de topologia (TP), do Inglês *topology processor*, é uma ferramenta de modelagem e análise que atua em segundo plano dentro do sistema de gerenciamento da distribuição. Esse processador estabelece com precisão a atual topologia da rede de distribuição, baseando-se no modo como estão os dispostos os dispositivos de comutação, também exibindo a coloração do circuito alimentador. Isso significa, o *status* de energização, e os caminhos de fornecimento.

Uma vez que a topologia é determinada, a direção do fluxo de energia também é devido à configuração radial. Os dados de topologia são a base de muitos outros aplicativos DMS. Ele também é usado para oferecer suporte ao processamento inteligente de alarmes com base na estrutura da topologia. Em alguns casos, uma rede de distribuição pode ter cenários de operação em malha intencional. TP é responsável por determinar os circuitos em malha e fazer marcas especiais para destacá-los mediante solicitação dos operadores. Em geral, o aplicativo de TP pode realizar as seguintes funções:

- ✓ Localizar um elemento da rede de distribuição (por exemplo, transformador, seção) por nome ou ID;
- ✓ Localizar e marcar caminhos de fornecimento de elementos de rede;
- ✓ Determinar e destacar o *status* de energização dos elementos da rede,
- ✓ Localizar e destacar as malhas de rede;
- ✓ Localizar e destacar todos os elementos de rede à jusante de um elemento selecionado;
- ✓ Localizar e destacar alimentadores vizinhos de um alimentador selecionado que pode servir como um fornecimento alternativo para o alimentador;
- ✓ Colorir individualmente os alimentadores;
- ✓ Colorir por nível de tensão;
- ✓ Colorir segmentos de linha com magnitudes de tensão menores do que os limites especificados;
- ✓ Colorir segmentos de linha com carregamento maior do que os limites especificados;

- ✓ Localizar e destacar as partes do alimentador de distribuição que estão isoladas da rede elétrica da concessionária e estão sendo energizadas por microrredes ou recursos energéticos distribuídos (DER).

4.1 Algoritmos de Busca em Grafos

As redes de distribuição são compostas pelo que se denominam de ramos e barras. Os ramos são todos os equipamentos elétricos da rede de distribuição que estão conectados em séries, como linhas, chaves e reguladores de tensão. As barras são equipamentos que estão conectados em *shunt*, ou seja, em derivação, e são cargas, capacitores e geradores.

Assumindo que um grafo simples seja descrito por $G = (V, A)$ composto por um conjunto V não vazio de vértices e um possível conjunto vazio A de arestas, e que cada aresta é um conjunto de dois vértices (DEO, 2017).

Adotando que as barras da rede de distribuição são como vértices e os ramos dessa rede são como arestas, uma rede de distribuição pode ser representada por um grafo. Dessa forma, a análise de topologia normalmente é realizada usando algoritmos de busca de grafos tais como o método de busca em profundidade, do Inglês *depth-first search*, e em largura, do Inglês *breadth-first search* (EKANAYAKE, 2012).

4.1.1 Busca em Profundidade

No algoritmo de busca em profundidade, existe a seguinte lógica, cada vértice v é visitado, em seguida cada vértice não visitado adjacente ao vértice v é visitado. Caso algum vértice v não tenha vértices adjacentes a ele ou todos os vértices adjacentes tenham sido visitados, retornar-se ao predecessor de v .

A busca é concluída se esse processo de visita e retrocesso levar ao primeiro vértice onde a travessia, ou busca começou. Caso ainda haja alguns vértices não visitados no grafo, a busca continua reiniciando para um dos vértices não visitados (DROZDEK, 2012).

Adicionalmente, mas embora não seja necessário para o resultado adequado deste método, o algoritmo atribui um número único a cada vértice visitado de forma

que os vértices sejam renumerados. Isso é útil para aplicações posteriores do algoritmo.

Algoritmo 1 – Pseudocódigo do método de busca em profundidade.

1- BEP(v)	10- BuscaEmProfundidade(v)
2- <i>num(v) = i+1;</i>	11- <i>para todos vértices v</i>
3- <i>para todos vértices u adjacentes à v</i>	12- <i>num(v) = 0</i>
4- <i>se num(u) é igual a 0</i>	13- <i>fim para</i>
5- <i>Adicionar a aresta (uv) em arestas</i>	14- <i>arestas = vazio</i>
6- BEP(u)	15- <i>i = 1</i>
7- <i>fim se</i>	16- <i>enquanto existir v tal que num(v) é 0</i>
8- <i>fim para</i>	17- BEP(v)
9- <i>fim BEP()</i>	18- <i>fim enquanto</i>
	19- <i>fim BuscaEmProfundidade()</i>

Fonte: adaptado de DROZDEK (2012)

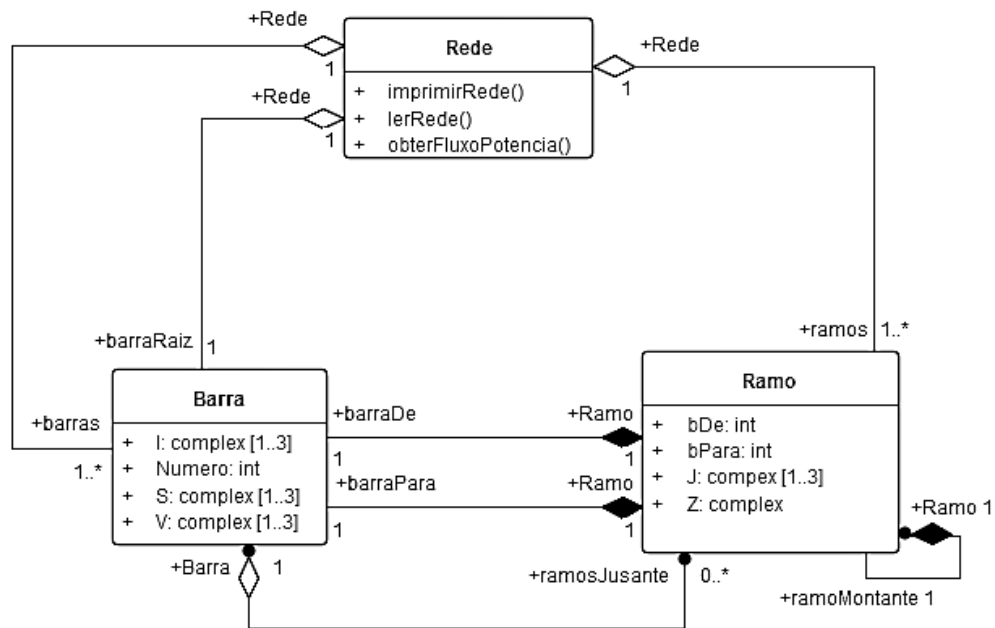
Este algoritmo garante a geração de uma árvore que inclui ou se estende por todos os vértices do grafo original. A árvore que atende a essa condição é chamada de árvore geradora. Uma árvore gerada é verificada pelo fato de que o algoritmo não inclui na árvore resultante nenhuma aresta que conduza do vértice atualmente analisado a um vértice já analisado.

Através da utilização do Visual Studio uma ferramenta de desenvolvimento de programação voltada a objeto, esse pseudocódigo foi desenvolvido a linguagem de programação C# (C Sharp).

Os diagramas de classe são ferramentas essenciais na modelagem de sistemas de programação orientada a objetos. Eles representam graficamente as classes, objetos e as relações entre eles. Cada classe é representada por um retângulo dividido em três compartimentos: o primeiro contém o nome da classe, o segundo lista seus atributos e o terceiro descreve seus métodos. As relações entre as classes são representadas por linhas, indicando a associação, agregação, composição, herança, entre outras. A associação mostra que um objeto de uma classe está relacionado com um ou mais objetos de outra classe. Essa ferramenta auxilia a visualizar a estrutura e o comportamento dos sistemas, facilitando o desenvolvimento e a compreensão dos programas orientados a objetos.

A partir do diagrama de classes de uma rede de distribuição representado na Figura 5, é possível observar quais são as relações entre cada uma das classes e objetos que podem ser utilizados no desenvolvimento do código fonte.

Figura 5 – Diagrama de classe de rede de distribuição.



Fonte: Ferreira (2012)

A partir da definição de todas as classes com seus respectivos atributos e do seu código fonte em linguagem de programação, tem-se, como mostrado no Algoritmo 2, o recorte do trecho a qual diz respeito a Busca em Profundidade.

Algoritmo 2 – Código em C# para o método de Busca em Profundidade.

```

//Metodo Busca em Profundidade
public void BuscaEmProfundidade ()
{
    List<Ramo> Arestas = new List<Ramo> ();
    int Contvertc = 1;

    void BEP (Barra v)
    {
        v.num = Contvertc;
        Contvertc++;

        for (int k = 0; k < v.ramosJusante.Count(); k++)
        {
            Arestas.Add(v.ramosJusante[k]);
            BEP(v.ramosJusante[k].paraBarra);
        }
    }
}
  
```

Fonte: Elaborado pelo Autor.

4.1.2 Busca em Largura

Diversos algoritmos de busca em grafos são baseados em modelo BEP(.), porém, alguns algoritmos são mais eficientes que outros se for levado em conta que a travessia do grafo subjacente não for em profundidade, mas sim em largura.

Os algoritmos em profundidade contam com o uso de uma pilha (explícita ou implicitamente, na recursão) e a busca em largura usa uma fila como estrutura de dados básica. Essa ideia também pode ser estendida para grafos, conforme mostrado no pseudocódigo do Algoritmo 3.

Algoritmo 3 – Pseudocódigo do método de busca em largura.

1- BuscaEmLargura()	10- enquanto <i>a fila não está vazia</i>
2- para <i>todos vértices v</i>	11- <i>v = desenfileirar()</i>
3- <i>num(v) = 0</i>	12- para <i>todos vértices u adjacentes à v</i>
4- fim para	13- se <i>num(u) é igual à 0</i>
5- <i>arestas = vazio</i>	14- <i>num(u) = i + 1</i>
6- <i>i = 1</i>	15- <i>enfileirar(u)</i>
7- enquanto <i>existir v tal que num(v) é 0</i>	16- <i>adicionar a aresta (vu) em arestas</i>
8- <i>num(v) = i+1</i>	17- fim se
9- <i>enfileirar(v)</i>	18- fim para
	19- fim enquanto
	20- fim enquanto
	21- fim BuscaEmLargura()

Fonte: DROZDEK (2012)

O algoritmo de busca em largura primeiro tenta marcar todos os vizinhos de um vértice *v* antes de prosseguir para outros vértices, enquanto BEP(.) escolhe um vizinho de um *v* e então prossegue para um vizinho deste vizinho antes de processar quaisquer outros vizinhos de *v*.

Da mesma maneira que no algoritmo de busca em profundidade, o algoritmo de busca em largura é codificado em linguagem C# no Visual Studio® como mostrado no Algoritmo 4.

Algoritmo 4 – Código em C# para o método de Busca em Largura.

```

//Metodo Busca em Largura
public void BuscaEmLargura ()
{
    for (int i = 0; i < barras.Length; i++)
    {
        barras[i].numP = 0;
    }
    List<Ramo> ArestasP = new List<Ramo> ();
    int iLargura = 1;
    List<Barra> Vertices = new List<Barra> ();

    barraRaiz.numP = iLargura;
    Vertices.Add(barraRaiz);

    while(Vertices.Count() > 0)
    {
        Barra v = new Barra ();
        v = Vertices[0];
        Vertices.RemoveAt(0);

        for (int j = 0; j < v.ramosJusante.Count(); j++)
        {
            if (v.ramosJusante[j].paraBarra.numP == 0)
            {
                v.ramosJusante[j].paraBarra.numP = ++iLargura;
                Vertices.Add(v.ramosJusante[j].paraBarra);
                ArestasP.Add(v.ramosJusante[j]);
            }
        }
    }
}

```

Fonte: Elaborado pelo Autor.

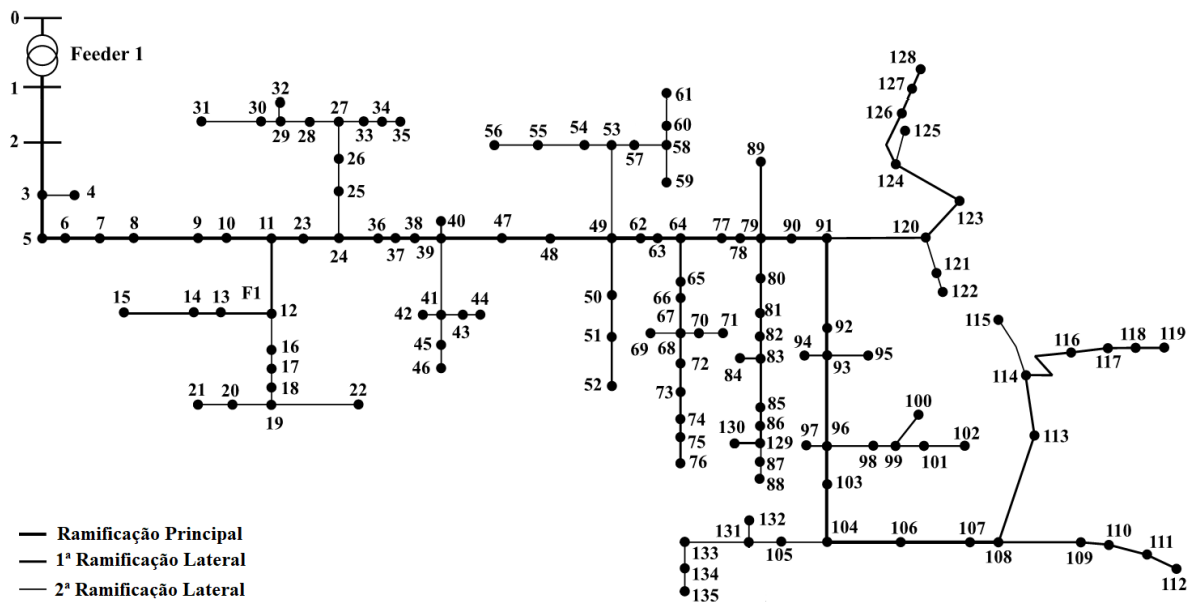
5 RESULTADOS E DICUSSÕES

5.1 Análise de Desempenho dos Algoritmos de Busca em Grafos.

Para que uma técnica de solução seja viável de implementação, ela deve atender alguns requisitos de eficiência computacional, dessa forma, é comparado o desempenho de ambos os algoritmos de travessia em grafos para uma mesma rede de distribuição genérica.

Esses dois algoritmos fazem a leitura dos dados de entrada, que corresponde a uma rede de distribuição física, descrita através de um arquivo *.xml seguindo uma estrutura padrão como apresentada na Figura 5 com informações das classes “rede”, “barra” e “ramo”. Esses dados utilizados são referentes a uma rede de distribuição real como representada na Figura 6.

Figura 6 – Rede de distribuição com 136 barras.



Fonte: LAPSEE (2020)

Nesta rede de distribuição, os vértices ou barras são representados pelos pontos numerados, no caso de 0 até 135. Já as arestas ou ramos são descritos pelas linhas que conectam os vértices.

Para a realização da análise de cada um dos algoritmos, eles foram submetidos a diversas e repetidas iterações com os dados dessa rede, de forma que fosse possível processar a topologia e medir o tempo que foi necessário para esse

processamento. Cada um dos métodos foi executado de mil a um milhão de vezes, esse procedimento foi repetido 10 vezes, a fim de mensurar o tempo de duração da execução de cada algoritmo. Na Tabela 1 é mostrado os resultados medidos para o método de busca em largura.

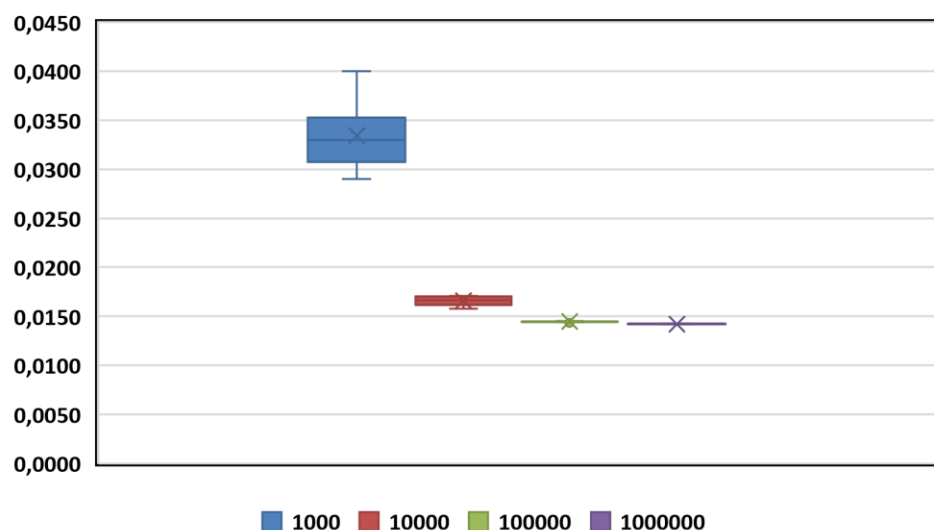
Tabela 1 – Tempo de execução do algoritmo de busca em largura.

Amostra	Número de Repetições do Método			
	10^3	10^4	10^5	10^6
1	0,0330	0,0163	0,01448	0,014235
2	0,0360	0,0158	0,01445	0,014226
3	0,0400	0,0166	0,01447	0,014244
4	0,0310	0,0171	0,01448	0,014259
5	0,0350	0,0170	0,01453	0,014180
6	0,0350	0,0158	0,01450	0,014202
7	0,0290	0,0167	0,01444	0,014263
8	0,0330	0,0163	0,01444	0,014199
9	0,0300	0,0171	0,01436	0,014234
10	0,0320	0,0170	0,01443	0,014200
Média	0,0334	0,01657	0,014458	0,0142242
Desvio Padrão	0,00248	0,000416	0,000034	0,0000232

Fonte: Elaborado pelo Autor.

A partir dos resultados obtidos na Tabela 1, foi elaborado o gráfico de caixas, ou *boxplot* apresentado na Figura 7.

Figura 7 – Variação do tempo de execução de acordo com quantidade de execuções do algoritmo para método de busca em largura.



Fonte: Elaborado pelo Autor.

A partir do gráfico de *boxplot* é perceptível que quanto maior o número de repetições do método, mais preciso fica o seu tempo de execução. Para poucas repetições as sub-rotinas de outros programas que podem estar executando em segundo plano no computador afetam o resultado e o tempo de execução do método.

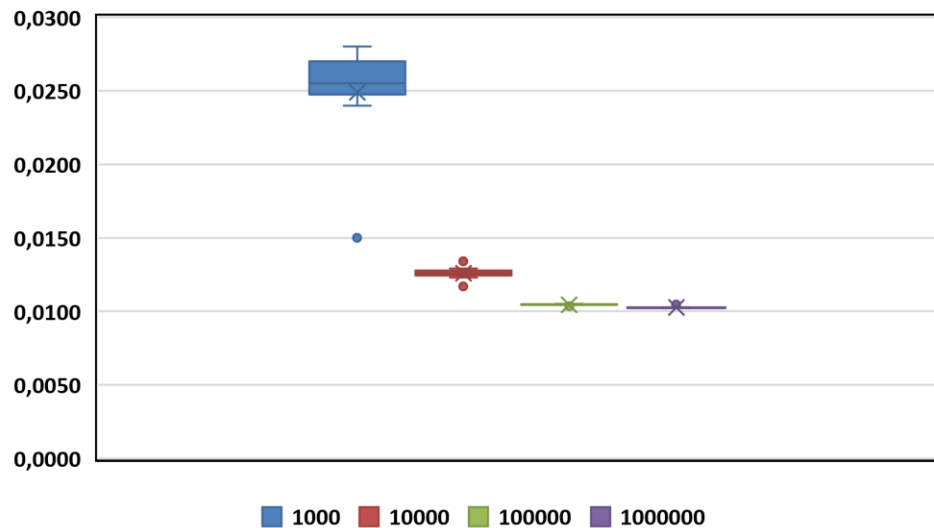
Tabela 2 – Tempo de execução do algoritmo de busca em profundidade.

Amostra	Número de Repetições do Método			
	10^3	10^4	10^5	10^6
1	0,0270	0,0126	0,01043	0,010470
2	0,0270	0,0127	0,01047	0,010239
3	0,0250	0,0127	0,01053	0,010231
4	0,0250	0,0123	0,01050	0,010233
5	0,0260	0,0125	0,01046	0,010259
6	0,0150	0,0134	0,01047	0,010250
7	0,0280	0,0117	0,01045	0,010237
8	0,0250	0,0126	0,01044	0,010233
9	0,0240	0,0129	0,01036	0,010246
10	0,0270	0,0126	0,01044	0,010232
Média	0,0249	0,0126	0,01046	0,010263
Desvio Padrão	0,00216	0,000260	0,000031	0,0000414

Fonte: Elaborado pelo Autor.

A partir dos resultados obtidos na Tabela 2, foi elaborado o gráfico *boxplot* apresentado na Figura 8.

Figura 8 – Variação do tempo de execução de acordo com quantidade de execuções do algoritmo para método de busca em profundidade.



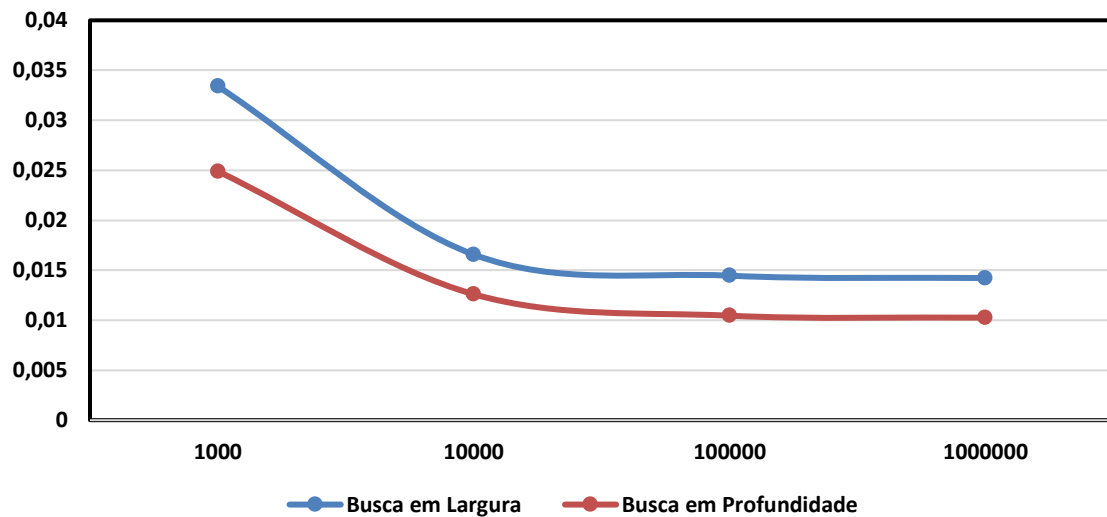
Fonte: Elaborado pelo Autor.

O mesmo raciocínio é válido para o método de busca em profundidade, com um maior número de execuções do método, menor fica o desvio entre as medidas de tempo obtidas e assim obtemos uma melhor acurácia no resultado do método.

A partir desses resultados obtidos é possível determinar sob esse parâmetro qual o é método mais eficiente, ou seja, aquele que obtém o resultado de processar a topologia da rede com um menor tempo de processamento computacional.

Na Figura 9 são mostradas as curvas de comparação entre os dois métodos de busca em grafos.

Figura 9 – Comparação entre o tempo de processamento de ambos os métodos.



Fonte: Elaborado pelo Autor.

A partir dessa análise pode-se constatar que o método de busca em profundidade é mais eficiente do que o método de busca em largura para a obtenção da topologia de uma rede de distribuição, uma vez que para qualquer número de execuções o método de busca em profundidade levou menos tempo do que o de busca em largura.

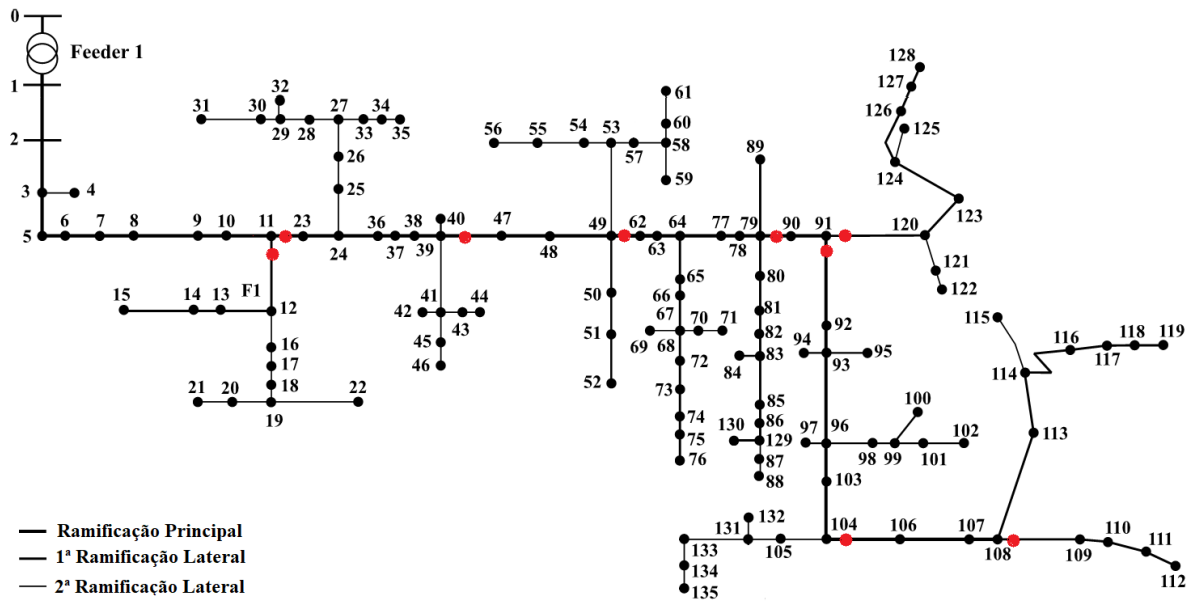
5.2 Aplicação do Algoritmo de Busca em Profundidade e Obtenção das Seções

Partindo do método mais eficiente para processamento de topologia de uma rede de distribuição, foram incrementadas algumas funcionalidades ao algoritmo, de forma que agora ele não apenas obtém a topologia da rede, mas também retorne quais são as seções da rede.

Antes de exibirmos os resultados referentes ao algoritmo atualizado para segmentar a rede de distribuição, é importante conceituar a seção de uma rede de distribuição como uma porção da estrutura dividida por elementos seccionadores, como chaves seccionadoras ou disjuntores..

Essa nova funcionalidade do algoritmo segue a estrutura do diagrama de classes apresentado na Figura 10.

Figura 11 – Identificação das chaves seccionadoras presentes na rede de distribuição.



Fonte: Adaptado do LAPSEE (2023)

O código fonte elaborado para a obtenção das seções é descrito no Algoritmo 5.

Algoritmo 5 – Algoritmo para obtenção das seções da rede de distribuição.

```
//Obtenção das seções da rede de distribuição
public void ObterSecoes ()
{
    void BEP (Barra v, Secao SecaoAtual)
    {
        SecaoAtual.cargaTotal += v.S[0].Magnitude + v.S[1].Magnitude +
v.S[2].Magnitude;
        for (int i = 0; i < chaveTransferencia.Count(); i++)
        {
            if (chaveTransferencia[i].Para == v.Numero)
            {
                SecaoAtual.TS = chaveTransferencia[i];
                break;
            }
        }
        for (int k = 0; k < v.ramosJusante.Count(); k++)
        {
            if (v.ramosJusante[k].Chave == 0)
            {
                SecaoAtual.ArestasSecao.Add(v.ramosJusante[k]);
                BEP(v.ramosJusante[k].paraBarra, SecaoAtual);
            }
            else
            {
                SecaoAtual.ramoSeccionamento.Add(v.ramosJusante[k]);
            }
        }
    }
}
```

```

    }
    Secoes = new List<Secao>();
    for (int i = 0; i < ramos.Length; i++)
    {
        if (ramos[i].Chave == 1)
        {
            Secoes.Add (new Secao() {ramoRaiz = ramos[i]});
        }
    }
    for (int z = 0; z < Secoes.Count(); z++)
    {
        Secoes[z].cargaTotal = 0;
        Secoes[z].ArestasSecao = new List<Ramo> ();
        Secoes[z].ramoSeccionamento = new List<Ramo> ();
        BEP(Secoes[z].ramoRaiz.paraBarra, Secoes[z]);
    }
    //Seção Montante
    for (int j = 0; j < Secoes.Count(); j++)
    {
        for (int k = 0; k < Secoes[j].ramoSeccionamento.Count(); k++)
        {
            for (int l = 0; l < Secoes.Count(); l++)
            {
                if (Secoes[l].ramoRaiz ==
Secoes[j].ramoSeccionamento[k])
                    Secoes[l].secaoMontante = Secoes[j];
            }
        }
    }
    //Seções a Jusante
    for (int d = 0; d < Secoes.Count(); d++)
    {
        Secoes[d].secoesJusante = new List<Secao>();
        for (int e = 0; e < Secoes[d].ramoSeccionamento.Count(); e++)
        {
            for (int f = 0; f < Secoes.Count(); f++)
            {
                if (Secoes[d].ramoSeccionamento[e] ==
Secoes[f].ramoRaiz)
                {
                    Secoes[d].secoesJusante.Add(Secoes[f]);
                }
            }
        }
    }
}

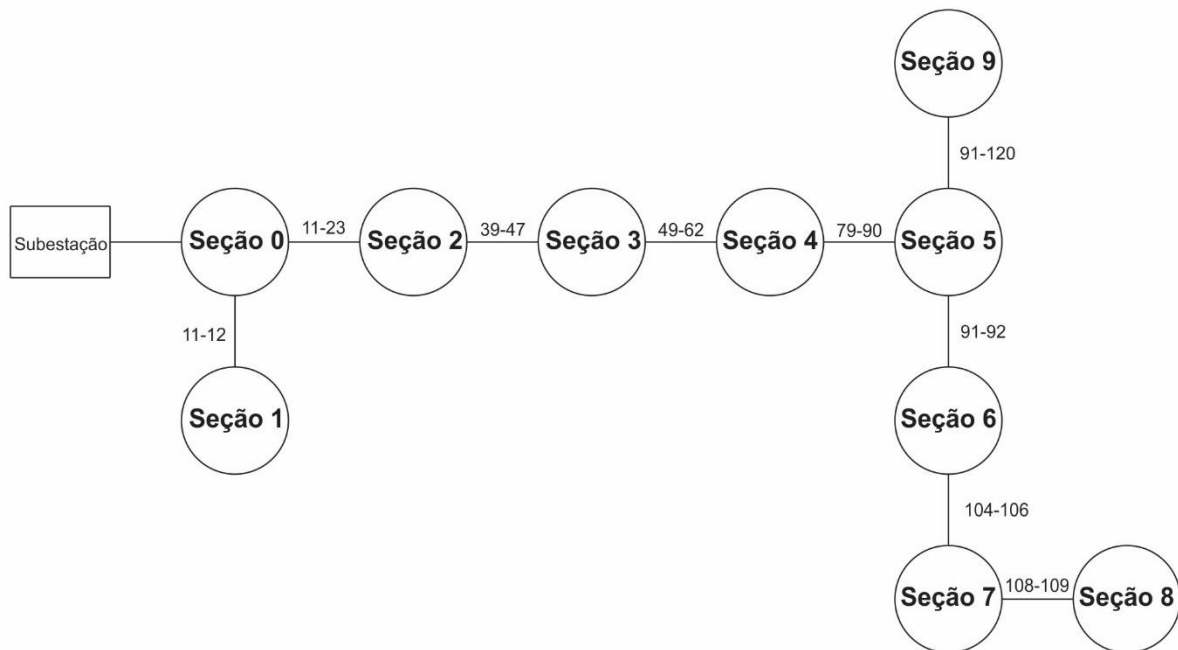
```

Fonte: Elaborado pelo Autor.

Executando o novo código, o resultado que o algoritmo retornou foi a topologia desta rede de distribuição juntamente com as seções e os respectivos elementos que compõem cada uma delas, ou seja, esse algoritmo criou listas da nova classe, a qual foi denominada de “seções”, e dentro dessas foram adicionados os objetos, “ramos” e “barras” que pertencem respectivamente a cada seção, além de calcular a carga total de cada uma das seções.

A Figura 12 é um diagrama que ilustra o resultado obtido pelo algoritmo, as divisões de seções desta rede de distribuição.

Figura 12 – Diagrama de seções da rede de distribuição.



Fonte: Elaborada pelo Autor.

5.3 Alimentadores Laterais

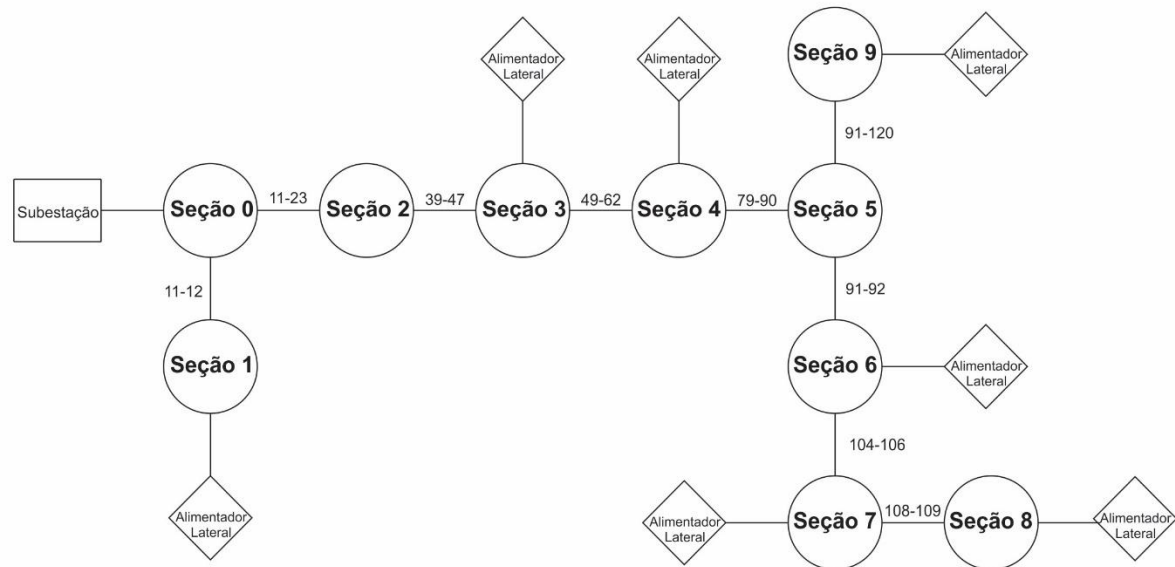
Uma das principais características das redes de distribuição é a presença de alimentadores laterais, esses são alimentadores vizinhos conectados na mesma subestação, ou em outra subestação, que forcem caminho alternativos para transferência de carga e reestabelecimento do fornecimento de energia para parte da rede.

Esses alimentadores podem atuar em momentos de falha em alguma seção da rede, de forma que poucas seções sejam afetadas com essa falha.

A rede de distribuição descrita na Figura 11 possui alguns alimentadores laterais, esses são descritos no mesmo arquivo **.xml* desta rede, onde cada um desses alimentadores está alocado e que é especificado através da capacidade de transferência de carga para cada um desses alimentadores laterais.

Na Figura 13 é mostrada a disposição desses alimentadores laterais dispostos na rede de distribuição já identificado pelo algoritmo anterior.

Figura 13 – Seções da rede de distribuição e alimentadores laterais.

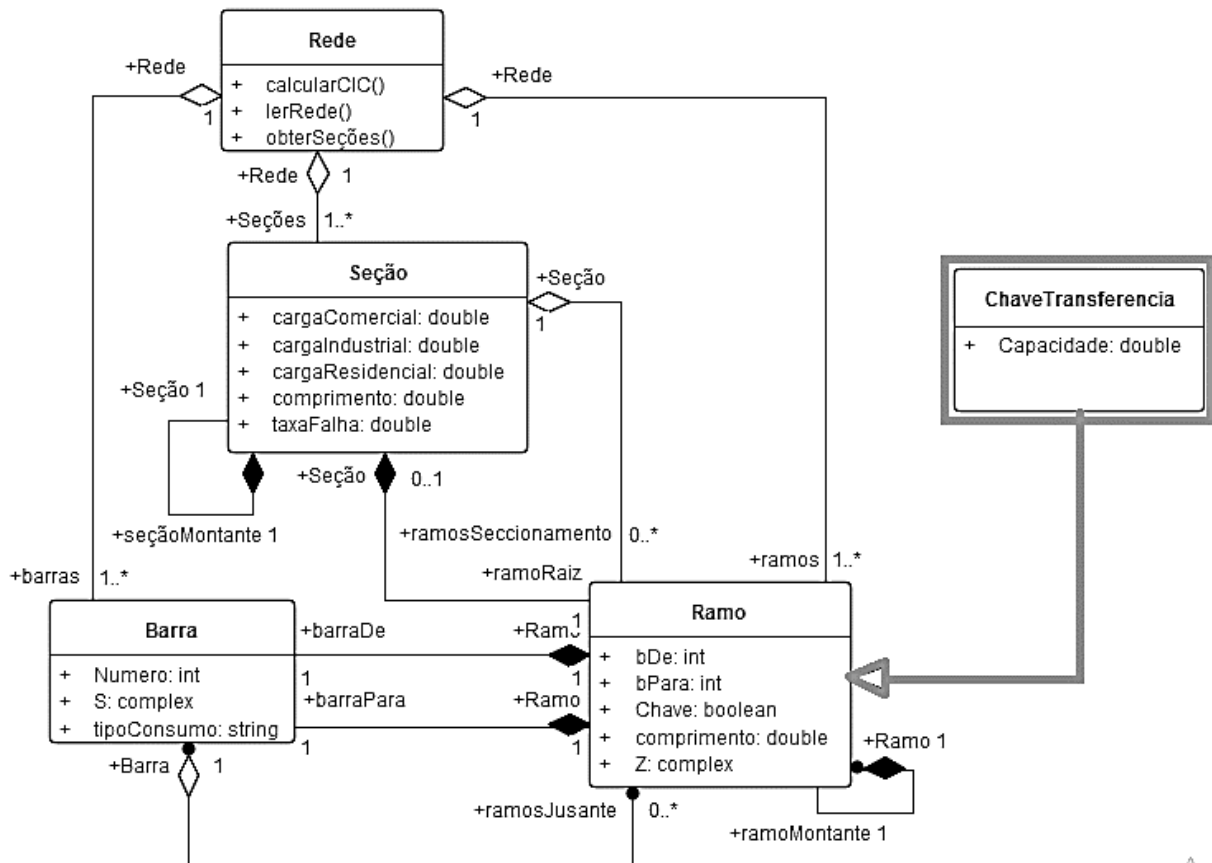


Fonte: Elaborado pelo Autor.

Para que essa ferramenta de processamento de topologia fique mais completa, foi integrado a funcionalidade de *Estado da Seção*. A ideia aqui é que a partir de uma falta em um ramo qualquer da rede de distribuição, primeiro o algoritmo identifique onde essa falha ocorreu. Segundo, compare a carga total de cada seção com a capacidade do alimentador lateral, caso a seção possua alimentador lateral. Por fim, caso a carga da seção seja menor do que a capacidade do alimentador lateral, o algoritmo pode fazer a transferência de alimentação para o respectivo alimentador.

Essa transferência de carga é realizada através do chamado no diagrama de classes da rede de “Chave de Transferência”, que são chaves seccionadoras normalmente abertas em conexão com alimentadores laterais, com uma certa capacidade de fornecimento de energia, quando necessário. Na Figura 14 é mostrado o novo diagrama de classes utilizado para implementar essa funcionalidade na ferramenta de processamento de topologia.

Figura 14 – Diagrama de classes com chave de transferência.



Fonte: Elaborado pelo Autor.

O código fonte no Algoritmo 6 mostra como essa funcionalidade foi implementada. Além disso, mostra que o algoritmo exige inicialmente a especificação de um ramo qualquer com uma falha para que a funcionalidade seja testada.

Algoritmo 6 – Algoritmo para obtenção do estado das seções.

```

ramos[48].falha = true;

// Estado da Seção
for (int a = 0; a < Secoes.Count(); a++)
{
    for (int b = 0; b < Secoes[a].ArestasSecao.Count(); b++)
    {
        if (Secoes[a].ArestasSecao[b].falha == true)
        {
            Secoes[a].interrompida = true;
            Secoes[a].emFalta = true;

            void iBEP (Secao s)
            {
                if (s.TS == null)
                {

```

```

        s.interrompida = true;
    }
    else if (s.TS.capacidade < s.cargaTotal)
    {
        s.interrompida = true;
    }

    for (int i = 0; i < s.secoesJusante.Count(); i++)
    {
        iBEP(s.secoesJusante[i]);
    }
    }

    iBEP (Secoes[a]);

    break;
    break;
}
}
}

```

Fonte: Elaborado pelo Autor.

Na Tabela 3 é mostrada a carga das seções e dos respectivos alimentadores laterais.

Tabela 3 – Comparação entre carga e capacidade dos alimentadores

Seção	Carga da Seção	Capacidade do Alimentador Lateral
0	570000	-
<u>1</u>	<u>728600</u>	<u>900000</u>
2	1420899	-
<u>3</u>	<u>803499</u>	<u>900000</u>
4	1595799	1200000
5	75000	-
6	997998	600000
7	378500	300000
<u>8</u>	<u>333500</u>	<u>450000</u>
<u>9</u>	<u>273500</u>	<u>450000</u>

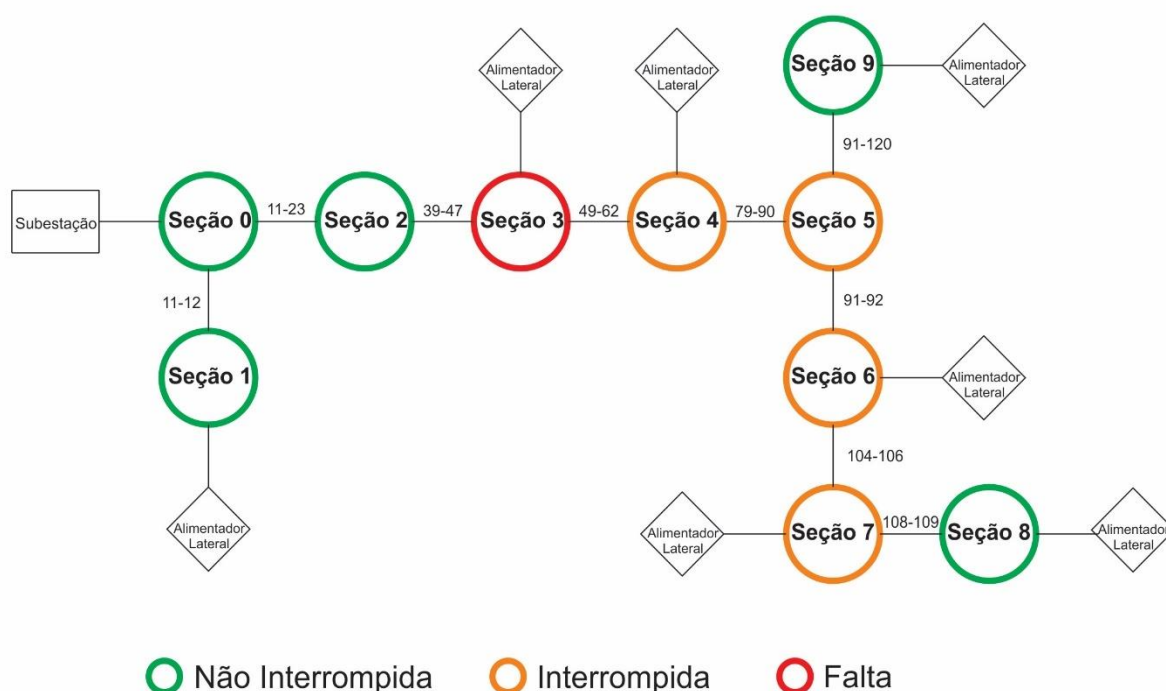
Fonte: Elaborado pelo Autor.

A partir da Tabela 3 é possível determinar quais seções podem ser transferidas para seu respectivo alimentador lateral caso uma falta aconteça na rede em um ramo que esteja a montante desta seção, ou seja, as seções que tem menor carga do que a capacidade do alimentador são menos impactados pela falta, pois podem ser transferidos para o alimentador lateral.

Inicialmente para o teste deste algoritmo, é suposta uma falta no ramo #48 desta rede, como está destacado no Algoritmos 6.

A partir desta falta indicada, espera-se que o algoritmo retorne qual a seção em falta, quais as seções interrompidas e quais as seções não interrompidas. Na Figura 15, é mostrado o diagrama de como ficou o estado das seções após a identificação desta falta no ramo #48.

Figura 15 – Identificação do estado das seções da rede.



Fonte: Elaborado pelo Autor.

Como mostrado na Figura 15, as seções 0, 1 e 2 são alimentadas pela subestação principal, a falta ocorre no ramo 48, que pertence a seção 3 e esta seção está em falta, ou seja, isolada e sem alimentação. As seções 4, 5, 6 e 7 que estão a jusante da falta estão interrompidas, no caso das seções 4, 6 e 7 seus alimentadores laterais não têm capacidade suficiente que alimentar a seção e no caso da seção 5 ela não possui alimentador lateral. As seções 8 e 9 podem operar normalmente pois seus alimentadores laterais têm capacidade de alimentá-las.

Movendo a falta para cada uma das seções, obteve-se um resultado distinto de quais seções ficariam interrompidas e quais não ficariam. Na Tabela 4 são

mostradas quais seriam as seções interrompidas e não interrompidas de acordo com qual seção estivesse em falta.

Tabela 4 – Seções interrompidas e não interrompidas de acordo com a falta.

Seção em Falta	Seção interrompida	Seção não interrompida
0	0; 2; 4; 5; 6; 7	1; 3; 8; 9
1	1	2; 3; 4; 5; 6; 7; 8; 9
2	2; 4; 5; 6; 7	0; 1; 3; 8; 9
3	3; 4; 5; 6; 7	0; 1; 2; 8; 9
4	4; 5; 6; 7	0; 1; 2; 3; 8; 9
5	5; 6; 7	0; 1; 2; 3; 4; 8; 9
6	6; 7	0; 1; 2; 3; 4; 5; 8; 9
7	7	0; 1; 2; 3; 4; 5; 6; 8; 9
8	8	0; 1; 2; 3; 4; 5; 6; 7; 9
9	9	0; 1; 2; 3; 4; 5; 6; 7; 8

Fonte: Elaborado pelo Autor.

A Tabela 4 evidência de forma clara o desempenho do algoritmo aplicado a esta rede de 136 barras, mostrando quais são as seções não interrompidas, tais como as interrompidas, sendo a falta em qualquer uma das seções.

6 CONCLUSÃO

O processamento de topologia é uma ferramenta indispensável ao bom funcionamento de todo o sistema de gerenciamento de distribuição de energia elétrica e a resolução de problemas que podem ocorrer na rede. Essa ferramenta, não só atua no mapeamento da atual topologia da rede como automatiza a tomada de decisão, visando menores impactos aos consumidores quando uma falha é identificada na rede de distribuição.

Como analisado, existem algumas maneiras de fazer o processamento de uma rede de distribuição, dois desses métodos foram abordados neste trabalho, o de busca em largura e o de busca em profundidade (DROZDEK, 2012). Esses dois métodos foram implementados sob a mesma ferramenta de programação em linguagem C# (C Sharp) de forma que fosse possível comparar o desempenho de ambos os métodos, uma vez que cada usando sua lógica atinge o mesmo objetivo no final, retornando uma estrutura de grafo que representa uma rede de distribuição de energia.

Entre esses métodos, o que obteve melhor desempenho computacional foi o método de busca em profundidade. Ter um melhor desempenho computacional significa processar a topologia da rede distribuição em menor tempo, e em média o método de busca em profundidade levou 26% menos tempo do que o método de busca em largura, conforme mostra a Figura 9. Para redes de distribuição de pequena escala, essa diferença pode não ser relevante, porém para rede mais complexa e com uma quantidade enorme de dados esse número se torna expressivo.

O método de busca em profundidade é mais eficiente em certos cenários de busca de caminhos ou soluções em árvores ou grafos. Ele explora o caminho o mais profundamente possível antes de retroceder. Essa abordagem economiza memória computacional, pois requer apenas uma pilha para armazenar os nós do caminho atual. Quando um nó é completamente explorado, ele é removido da pilha, liberando espaço de memória. Essa característica é vantajosa em árvores ou grafos com muitas ramificações ou ciclos profundos, pois a busca em largura tenderia a expandir muitos nós em níveis baixos, consumindo mais memória.

O método de busca em largura, por outro lado, é mais eficiente em grafos ou árvores que possuem soluções mais próximas da raiz. Ele explora todos os nós em

um nível antes de passar para o próximo nível. Enquanto isso, requer uma fila para armazenar os nós a serem explorados. Embora a busca em largura seja abrangente e encontre a solução mais próxima na maioria dos casos, pode ser ineficiente em termos de memória computacional em grafos com muitos níveis ou profundidades desconhecidas, pois precisa armazenar todos os nós de cada nível na fila, ocupando mais memória RAM.

Um fator que foi levado em conta na análise de desempenho dos algoritmos foi o número de execuções. Quanto maior o número de execuções dos métodos, menores foram as interferências de outras atividades que poderiam estar sendo executadas em segundo plano no computador. Isso porque quanto maior o número de execuções, menor era o desvio médio do tempo medido de cada amostra, como pode ser observado nas Figuras 7 e 8.

Independentemente do número de execuções de cada um dos métodos, o de busca em profundidade se mostrou mais eficiente.

Como o processamento de topologia é uma das ferramentas que compõem todo o sistema de gerenciamento de distribuição é desejável que por si só ela seja mais eficiente possível, para auxiliar o operador em tempo real a identificar e tomar decisões assertivas no gerenciamento da rede.

Uma dessas medidas que o algoritmo foi capaz de solucionar é a obtenção das seções da rede, e de seus alimentadores laterais. Dessa forma quando foi imputado uma falha em um ramo qualquer da rede, o algoritmo por si só identificou em qual seção estava a falta, quais eram as seções que ficariam interrompidas e quais seriam transferidas para os respectivos alimentadores laterais.

Essa agilidade em tomada de decisões traz grandes benefícios tanto para as concessionárias, uma vez que por meio da ferramenta de processamento sabe-se a localização da falta podendo deslocar uma equipe de manutenção com mais prontidão e também aos consumidores pois reduz a quantidade de seções afetadas pela falta fazendo a transferência de carga das seções que possuem alimentadores com capacidade suficiente de alimentação.

De forma ampla, a ferramenta de processamento de topologia desempenha um papel fundamental na eficiência e confiabilidade das redes de distribuição modernas. Compreender a topologia, ou seja, a configuração e interconexões dos elementos da rede, é essencial para o planejamento, operação e manutenção adequados do sistema. Essa ferramenta permite aos operadores visualizarem a

estrutura da rede, identificarem rapidamente a localização de falhas ou interrupções e otimizar a alocação de recursos. Além disso, o processamento de topologia facilita a análise de dados e a simulação de cenários, ajudando a antecipar possíveis problemas e a tomar decisões mais informadas. Em um cenário de crescente complexidade das redes elétricas, a ferramenta de processamento de topologia é indispensável para garantir um fornecimento de energia eficiente, seguro e confiável.

7 REFERÊNCIAS

- ALBUQUERQUE, M. M. História da energia elétrica no Brasil. São Paulo: Atual, 1982.
- BICHELS, A. Características de Sistemas Elétricos de Potência. *In: SISTEMAS Elétricos de Potência*. [S. l.: s. n.], 2018. cap. 2, p. 31-51.
- DEO, N., “*Graph Theory with Applications to Engineering and Computer Science*,” Estados Unidos: Dover Publications, 2017.
- DROZDEK, A. “*Data Structures and Algorithms in C++*,” Estados Unidos, Cengage Learning, 2012.
- EKANAYAKE, J., LIYANAGE, K., WU, J., YOKOYAMA, A., JENKINS, N., “*Smart Grid: Technology and Applications*”- Wiley, 2012.
- ELETROBRAS. A energia elétrica no Brasil: da primeira lâmpada à ELETROBRAS. Rio de Janeiro: Biblioteca do Exército, 1977.
- FERREIRA, D. S., Análise do algoritmo de fluxo de potência e da extensão do modelo de informação comum para redes de distribuição de energia elétrica. Universidade Estadual Paulista (Unesp), 2021. Disponível em: <<http://hdl.handle.net/11449/214695>>.
- IEC 61970. “*Energy Management System Application Program Interface (EMS-API), Part 556: CIM-Baswed Graphic Exchange Format (CIM/G)*,” Geneva, Switzerland, IEC, 2016.
- LABORATÓRIO DE PLANEJAMENTO DE SISTEMAS DE ENERGIA ELÉTRICA - LAPSEE. Sistema de distribuição para testes – 135 barras. Ilha Solteira: LAPSEE, 2020. Disponível em: <https://www.feis.unesp.br/#!/lapsee>.
- MELO, G. S. de. Introdução a Teoria de Grafos. 2014. Dissertação (Mestrado em Matemática) - Universidade Federal da Paraíba, 2014.

MOMOH, J. A. "*Electric Power Distribution, Automation, Protection, and Control*," Estados Unidos: CRC Press, 2017.

OPERADOR NACIONAL DO SISTEMA – ONS. Evolução da Capacidade Instalada no SIN – maio/2023 a dezembro/2027, 2023. Disponível em: <https://www.ons.org.br/paginas/sobre-o-sin/o-sistema-em-numeros>

SILVA, A. B. B. *Mentes Ansiosas: O Medo e a Ansiedade Nossos de Cada Dia*. 2. ed. rev. São Paulo: Principium, 2017. 296 p.

ZIVIC, Petra. Nikola Tesla: o pioneiro da energia elétrica que tornou nossa vida mais fácil. BBC World Service, [S. l.], p. 1-1, 21 jan. 2023. Disponível em: <https://www.bbc.com/portuguese/geral-64177253#:~:text=Em%201891%2C%20ele%20inventou%20a,televis%C3%A3o%20e%20outros%20equipamentos%20eletr%C3%B4nicos>. Acesso em: 18 abr. 2023.