Trabalho Prático 2: Rede Elétrica da Metalmax

João Correia Costa (2019029027)

Dezembro de 2024, Belo Horizonte

1 Introdução

A Metalmax é uma siderúrgica nacional especializada na produção de aços de alta qualidade. Para transformar ferro e carbono em aço, a empresa depende de processos físicos que exigem um fornecimento elevado e constante de energia. Por isso, os engenheiros da Metalmax precisam projetar uma rede elétrica capaz de garantir o fluxo ininterrupto e suficiente de energia para atender à planta industrial.

Essa rede é composta por três elementos principais: geradores, consumidores e conexões. Os geradores possuem capacidade ilimitada de fornecimento de energia. Os consumidores, por outro lado, apresentam demandas específicas que devem ser atendidas para o funcionamento adequado dos equipamentos. Já as conexões, que interligam esses elementos, possuem restrições de capacidade, limitando a quantidade de energia que podem transmitir.

O objetivo deste trabalho é analisar a rede elétrica projetada para identificar se os consumidores estão recebendo energia suficiente, além de localizar conexões críticas e gargalos no sistema. Essa análise permitirá aos engenheiros reorganizar a estrutura da rede, otimizando o fluxo de energia. O futuro da Metalmax depende de um algoritmo acertivo!

2 Modelagem

O problema apresentado possui uma estrutura claramente combinatória. Geradores e consumidores estão interligados por conexões unidirecionais com capacidade limitada, formando uma rede que pode ser naturalmente representada por um grafo de arestas ponderadas. Nesse grafo, os geradores e consumidores correspondem aos vértices, enquanto as conexões são representadas pelas arestas direcionadas.

A implementação do grafo será realizada por meio de uma matriz de adjacência ponderada, como ilustrado na Figura 1. Trata-se de uma matriz quadrada, onde cada posição $M[i][j]$ indica a capacidade da conexão de transportar energia do vértice i para o vértice j. Caso não exista uma conexão entre i e j, o valor de $M[i][j]$ será 0.

A escolha pela matriz de adjacência ponderada, em vez de uma lista de adjacência, justifica-se pela natureza do algoritmo empregado na análise da rede. Será aplicada uma adaptação do algoritmo de Ford-Fulkerson, que demanda a criação dinâmica de arestas durante a execução, uma operação consideravelmente mais eficiente com o uso de matrizes de adjacência. A adaptação inclui a inserção de um **super-gerador**, conectado aos geradores do grafo por arestas de capacidade infinita, representando a energia ilimitada fornecida. Para as conexões originadas nos super-geradores, a capacidade "infinita" será representada pelo valor máximo de `std::numeric_limits<size_t>::max()`. Além disso, para considerar a demanda energética de cada componente da rede, todos os consumidores são conectados a um **super-sumidouro**, com arestas de capacidade igual ao consumo especificado para cada componente. Na prática, a matriz quadrada aumenta em duas dimensões.

Com essa modelagem, é possível aplicar algoritmos de fluxo para determinar o fluxo máximo da rede, identificar conexões críticas e calcular a energia perdida.

3 Solução

Nesta seção iremos detalhar os algoritmos implementados para se determinar o fluxo máximo no grafo, a energia perdida na rede e as conexões críticas, isto é, com um fluxo corrente igual a capacidade.

3.1 Fluxo Máximo

Adaptamos o algoritmo de Ford-Fulkerson para incorporar dois vértices artificiais: o super-gerador e o super-sumidouro.

O super-gerador funciona como uma fonte única no grafo, conectada a todos os geradores originais por arestas de capacidade infinita, garantindo que o fornecimento de energia não seja limitado. Já o super-sumidouro é um vértice artificial que atua como destino único, ligado a todos os consumidores. As arestas conectando os consumidores ao super-sumidouro possuem capacidade igual à demanda de cada vértice consumidor, representando a quantidade de energia necessária para atender cada equipamento.

Além desses vértices artificiais, o grafo mantém suas conexões originais, onde cada aresta possui um peso representando a capacidade de transmissão de energia entre dois vértices. Com essa estrutura, o algoritmo busca determinar o maior fluxo

$$
\begin{array}{c|cccc}
 & 0 & 1 & 2 & 3 \\
\hline
0 & 0 & 5 & 3 & 0 \\
1 & 0 & 0 & 0 & 0 \\
2 & 0 & 2 & 0 & 4 \\
3 & 0 & 0 & 0 & 0
\end{array}
$$

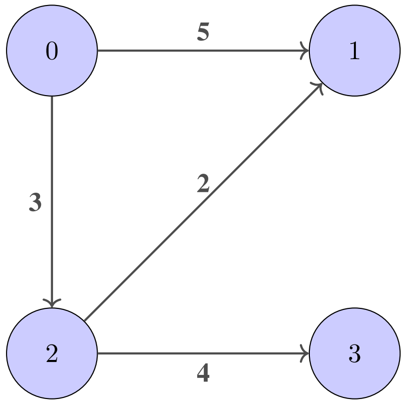

Figure 1: *Matriz de Adjacência Ponderada e Grafo Direcionado*

possível que pode ser transmitido do super-gerador ao super-sumidouro, respeitando as restrições impostas pelas capacidades das arestas.

A solução inicia com a definição do fluxo total, que é inicializado como zero. Em seguida, o algoritmo executa uma Busca em Largura (BFS) no grafo residual, com o objetivo de identificar um caminho aumentante que conecte o super-gerador ao super-sumidouro.

Se um caminho aumentante for encontrado, o próximo passo é determinar o bottleneck, que corresponde à menor capacidade ao longo do caminho identificado. Após calcular o valor do bottleneck, o grafo residual é atualizado para refletir o uso parcial da capacidade das arestas. O valor do bottleneck é subtraído das capacidades das arestas diretas e adicionado às capacidades das arestas reversas, permitindo a representação de possíveis fluxos de retorno no futuro.

O valor do bottleneck é então somado ao fluxo total acumulado. Esse processo é repetido continuamente, com novas execuções da Busca em Largura e atualizações do grafo residual, até que não seja possível encontrar mais caminhos aumentantes conectando o super-gerador ao super-sumidouro.

Quando não há mais caminhos aumentantes disponíveis, o algoritmo é encerrado. Nesse momento, o valor acumulado no fluxo total representa o fluxo máximo da rede. Esse resultado corresponde à maior quantidade de energia que pode ser transmitida do super-gerador ao super-sumidouro, respeitando todas as restrições de capacidade impostas pelas arestas do grafo.

Abaixo, apresentamos o pseudocódigo do algoritmo adaptado:

Algorithm 1 Ford-Fulkerson com Super-Sumidouro e Super-Gerador

Input: `Grafo residual`
Output: `Fluxo máximo`
Inicializar `fluxoTotal` $\leftarrow 0$
while *`BFS(Grafo residual)` encontra caminho aumentante* **do**
  Determinar o `bottleneck` ao longo do caminho aumentante Atualizar o `grafo residual`:
  - Subtrair `bottleneck` das arestas diretas
  - Somar `bottleneck` às arestas reversas

  Adicionar `bottleneck` ao `fluxoTotal`
end
return `fluxoTotal`

3.2 Energia Perdida

Após determinar o fluxo máximo na rede elétrica, foi possível calcular a energia perdida utilizando a capacidade total dos geradores comuns. Esse cálculo considera que cada gerador contribui inicialmente com sua capacidade máxima e que a diferença entre essa capacidade e o fluxo efetivamente transmitido pela rede representa a energia perdida.

Para isso, percorremos todos os vértices do grafo e, para cada gerador ordinário, somamos as capacidades de suas conexões de saída. Essa soma reflete o fluxo total teórico que seria fornecido pelos geradores se não houvesse restrições impostas pelas capacidades das conexões ou pela topologia do grafo. A energia perdida, então, é obtida subtraindo o fluxo máximo calculado anteriormente desse fluxo teórico.

Formalmente, a energia perdida é dada por:

$$\text{Energia Perdida} = \sum_{\text{u é gerador}} \sum_{\text{v}} \text{capacidade}(u, v) - \text{fluxo máximo}$$

Abaixo está o pseudocódigo do cálculo da energia perdida:

Algorithm 2 Cálculo da Energia Perdida

```
Input: grafo, fluxo máximo, vértices geradores
Output: energia perdida
Inicializar energiaTotal ← 0
for u em grafo.vértices do
    if u é gerador then
        for v em grafo.vértices do
            Adicionar capacidade(u, v) a energiaTotal
        end
    end
end
Calcular energiaPerdida ← energiaTotal − fluxo máximo
return energiaPerdida
```

Com esse procedimento, obtemos um indicador importante para a análise de eficiência da rede elétrica, permitindo identificar o volume de energia gerada que não é utilizado devido às limitações estruturais ou operacionais da rede.

3.3 Arestas Críticas

Além do cálculo do fluxo máximo e da energia perdida, realizamos a identificação das arestas críticas na rede elétrica. Arestas críticas são aquelas que tiveram toda a sua capacidade original utilizada durante o cálculo do fluxo máximo. Identificar essas arestas é fundamental para localizar possíveis gargalos na transmissão de energia e planejar melhorias na rede.

O procedimento para encontrar as arestas críticas baseia-se na análise do grafo residual e da capacidade original do grafo. Para cada aresta (u, v) do grafo original, verificamos se sua capacidade residual no grafo residual é igual a zero. Quando isso ocorre, significa que todo o fluxo permitido pela capacidade original da aresta foi utilizado, classificando-a como crítica.

Formalmente, uma aresta (u, v) é crítica se:

$$\text{Arestas Críticas} = \{(u, v) \mid u \neq v,\ g_{\text{residual}}[u][v] = 0,\ g_{\text{original}}[u][v] > 0\}.$$

O algoritmo percorre todas as arestas do grafo, excluindo as conexões com os vértices artificiais (super-gerador e super-sumidouro), e verifica as condições acima. As arestas identificadas como críticas são armazenadas em uma lista, que é ordenada por capacidade para facilitar sua análise.

O pseudocódigo do algoritmo é apresentado a seguir:

Algorithm 3 Identificação de Arestas Críticas

```
Input: grafo original, grafo residual
Output: lista de arestas críticas
Inicializar arestasCríticas como lista vazia
for u em grafo.vértices exceto super-gerador e super-sumidouro do
    for v em grafo.vértices exceto super-gerador e super-sumidouro do
        if u ≠ v e grafoResidual(u, v) = 0 e grafoOriginal(u, v) > 0 then
            Adicionar aresta(u, v) a arestasCríticas
        end
    end
end
Ordenar arestasCríticas por ordem decrescente de capacidade
return arestasCríticas
```

4 Análise de Complexidade de Tempo e Espaço

Nesta seção, discutimos a análise de complexidade de tempo e espaço para os algoritmos de fluxo máximo, energia perdida e identificação de arestas críticas, assumindo que o grafo residual e o grafo original são representados como matrizes de adjacências.

4.1 Fluxo Máximo

O algoritmo Ford-Fulkerson opera sobre o grafo residual, que é representado por uma matriz de adjacências. Em cada iteração, o algoritmo realiza uma busca em largura para encontrar um caminho aumentante. Quando o grafo é representado como uma matriz de adjacências, acessar os vizinhos de um vértice exige $O(N)$ operações por vértice, onde N é o número total de vértices. Portanto, o custo de uma única busca é $O(N^2)$. O número de iterações do algoritmo é limitado pelo fluxo máximo total C, que corresponde à soma das capacidades das arestas. Logo, a complexidade de tempo total do algoritmo é $O(C \cdot N^2)$.

Quanto à complexidade de espaço, uma matriz de adjacências para representar tanto o grafo residual quanto o grafo original requer $O(N^2)$ espaço. Portanto, a complexidade de espaço do algoritmo é $O(N^2)$.

4.2 Energia Perdida

O cálculo da energia perdida envolve somar as capacidades das arestas conectadas aos vértices geradores na matriz de adjacências do grafo original e subtrair o fluxo máximo previamente calculado. Esse cálculo é feito iterando apenas sobre os vértices geradores. Para cada vértice gerador, são exploradas as arestas conectadas a ele, e a complexidade depende do número de vértices geradores. Como o número de vértices geradores é G, e para cada vértice gerador é necessário explorar todos os seus vizinhos, o que leva $O(N)$ operações por vértice, a complexidade de tempo para o cálculo da energia perdida é $O(G \cdot N)$, onde N é o número total de vértices no grafo. Caso o número de vértices geradores seja pequeno em relação ao número total de vértices, a complexidade pode ser consideravelmente mais eficiente que $O(N^2)$. No entanto, se a maioria dos vértices for geradora, a complexidade se aproxima de $O(N^2)$.

Em relação ao espaço, o cálculo não exige armazenamento adicional significativo além do valor da energia perdida e dos somatórios intermediários. Portanto, a complexidade de espaço é $O(1)$.

4.3 Arestas Críticas

A identificação de arestas críticas requer a verificação de todas as arestas do grafo original e a comparação com o grafo residual, ambos representados como matrizes de adjacências. Para cada vértice u, é necessário iterar sobre seus N vizinhos, o que resulta em $O(N^2)$ operações para acessar todas as arestas do grafo original. A verificação das arestas críticas consiste em comparar os valores das matrizes de adjacências do grafo residual e do grafo original, também com custo $O(N^2)$. Portanto, a complexidade de tempo total para a identificação de arestas críticas é $O(N^2)$.

Além disso, a etapa de ordenação das arestas críticas identificadas, que pode conter até N^2 arestas no pior caso, adiciona uma complexidade de $O(N^2 \log N^2)$, que é equivalente a $O(N^2 \log N)$. No total, a complexidade de tempo para identificar as arestas críticas é $O(N^2 \log N)$.

Em relação à complexidade de espaço, as arestas críticas são armazenadas em uma lista, que no pior caso pode conter até N^2 elementos. Assim, a complexidade de espaço para armazenar as arestas críticas é $O(N^2)$.

5 Conclusão

Neste trabalho, foi realizada uma análise detalhada de uma rede elétrica industrial da Metalmax, com o objetivo de determinar o fluxo máximo de energia, identificar conexões críticas e calcular a energia perdida. A modelagem da rede foi feita através de um grafo direcionado, utilizando uma matriz de adjacência ponderada para representar as conexões entre geradores e consumidores, e adaptamos o algoritmo de Ford-Fulkerson para incorporar os vértices artificiais de super-gerador e super-sumidouro.

A análise permitiu obter o fluxo máximo da rede, identificando as arestas críticas, aquelas que estão com toda sua capacidade utilizada, e calcular a energia perdida devido às limitações estruturais da rede. Essas informações são cruciais para otimizar a rede elétrica, identificar gargalos e melhorar a eficiência energética da Metalmax.

A complexidade dos algoritmos, tanto em termos de tempo quanto de espaço, foi discutida, destacando a viabilidade de sua implementação para redes de grande porte e densas, com o uso de matrizes de adjacência e a aplicação de busca em largura para encontrar os caminhos aumentantes.

Em suma, este trabalho oferece uma solução eficiente para o problema de análise de redes elétricas industriais, sendo fundamental para o planejamento e otimização de redes de energia, garantindo o fornecimento adequado e ininterrupto para as operações industriais da Metalmax.

6 Bibliografia

1. Chaimowicz, L. and Prates, R. (2020). Slides virtuais da disciplina de estruturas de dados. Disponibilizado via moodle. Departamento de Ciência da Computação. Universidade Federal de Minas Gerais. Belo Horizonte.

2. Introduction to Algorithms, Thomas H. Cormem, Charles E. Leiserson, Ronald L. Rivest.