

Pesquisa Operacional - Problema da Corrente de Equilíbrio

Eduardo César¹ Manassés Ferreira¹ Marzo Júnior¹
Thiago Linke¹

¹Universidade Federal de Minas Gerais, Brasil

Pesquisa Operacional, 2013

Resumo

1 Modelagem

- O Problema
- O Modelo

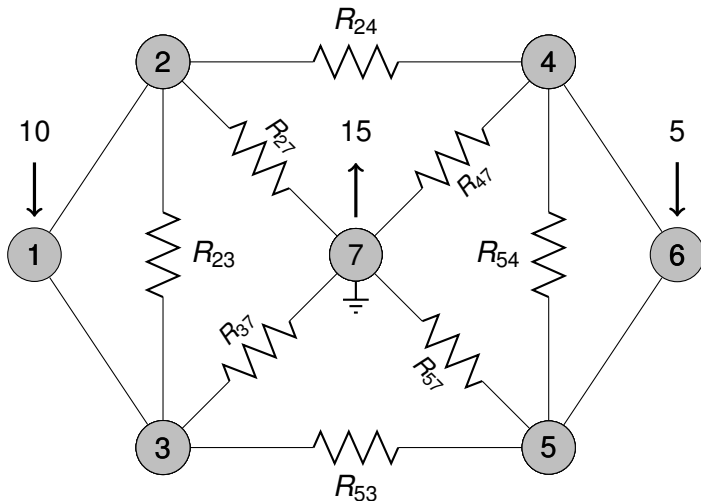
2 Resolução

- Resolução sistema quadrático
- Resolução Fluxo em Redes
- Minimização Quadrática - Octave
- Sistema Linear - Numerical Recipes
- netgen.c

3 Conclusões

Determinação da corrente de equilíbrio

Deseja-se determinar a corrente de equilíbrio que flui em um circuito elétrico como por exemplo:



Modelos

Existe mais de uma forma de se resolver o este problema. Três dessas formas serão mostradas aqui:

- 1 Modelagem como sistema de equações lineares.
- 2 Modelagem com sistema quadrático
- 3 Modelagem como fluxo em redes com custo convexo.

Modelagem como sistema linear

- Problema é resolvido introduzindo-se uma variável x_{ij} representando o fluxo de corrente no arco (i, j) do circuito elétrico e montar um sistema de equações de equilíbrio para estes fluxos. A solução para este sistema fornece a intensidade x_{ij} para cada arco respectivo.
- Baseia-se nos princípios físicos:
 - Lei de Ohm $\rightarrow V = R \times I$
 - 1ª Lei de Kirchhof $\rightarrow I_1 = I_2 + I_3$

Modelagem como fluxo em redes com custo convexo

Esta formulação se utiliza de um comportamento conhecido de que as correntes de equilíbrio nos resistores são os fluxos para qual os resistores dissipam a menor potência total suprida pelas fonte de tensão (ou seja, a corrente elétrica segue o caminho de menor resistência.)

$$\text{Minimizar } \sum_{(i,j) \in A} r_{ij} x_{ij}^2 \quad \text{sujeito a}$$

$$\sum_{j:(i,j) \in A} x_{ij} - \sum_{i:(j,i) \in A} x_{ji} = b_i \quad \text{para cada nodo } i \in N,$$

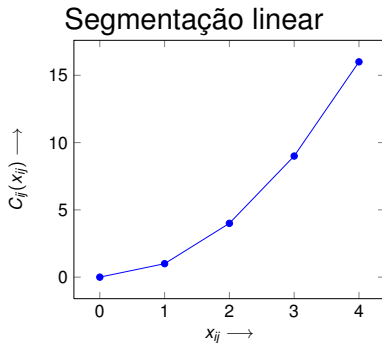
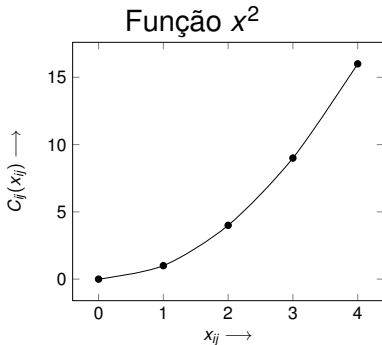
$$x_{ij} \geq 0 \quad \text{para cada arco } (i,j) \in A.$$

Linearização do custo convexo

Aproximação por segmentação:

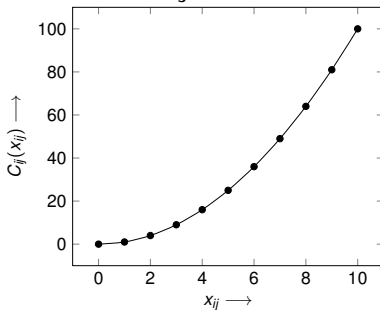
- Cada custo de arco $C_{ij}(x_{ij})$ possui p segmentos lineares: $0 = d_{ij}^0 < d_{ij}^1 < d_{ij}^2 < d_{ij} < \dots$, que denotam os pontos onde a função “quebra”.
- custo varia linearmente no intervalo $[d_{ij}^{k-1}, d_{ij}^k]$. Denotamos c_{ij}^k como o coeficiente de custo linear no intervalo $[d_{ij}^{k-1}, d_{ij}^k]$.
- Sendo assim, para especificar o a função aproximada, precisamos especificar os segmentos e a inclinação da função nesses segmentos.

Linearização do custo convexo

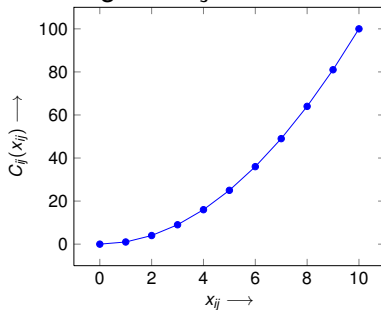


Linearização do custo convexo

Função x^2

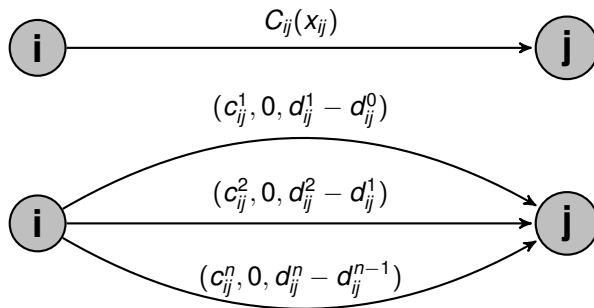


Segmentação linear

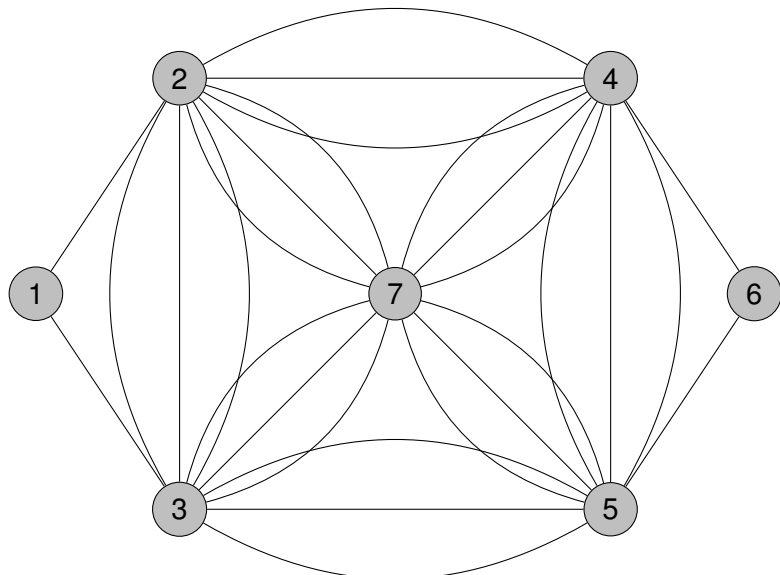


Linearização do custo convexo

Arco original e arcos correspondentes na nova rede:



Linearização do custo convexo



Resolução sistema quadrático

Sistema quadrático

- Problema 1: GLPK não resolve funções objetivo que não sejam lineares

Sistema quadrático

- Problema 1: GLPK não resolve funções objetivo que não sejam lineares
- Solução: O GNU Octave tem um solver para programação linear quadrática

Sistema quadrático

- Problema 1: GLPK não resolve funções objetivo que não sejam lineares
- Solução: O GNU Octave tem um solver para programação linear quadrática
- Problema 2: GNU Octave não aceita GMPL

Sistema quadrático

- Problema 1: GLPK não resolve funções objetivo que não sejam lineares
- Solução: O GNU Octave tem um solver para programação linear quadrática
- Problema 2: GNU Octave não aceita GMPL
- Solução: Programa simples converte grafo de entrada para entrada do GNU Octave

Resolução sistema quadrático

Sistema quadrático

- Para tornar as matrizes mais compactas, vamos usar uma representação baseada em arestas presentes no grafo

Sistema quadrático

- Para tornar as matrizes mais compactas, vamos usar uma representação baseada em arestas presentes no grafo
- Para tal, vamos definir um mapa $F : A \longrightarrow I \subset \mathbb{N}$ que mapeia arestas em números

Sistema quadrático

- Para tornar as matrizes mais compactas, vamos usar uma representação baseada em arestas presentes no grafo
- Para tal, vamos definir um mapa $F : A \longrightarrow I \subset \mathbb{N}$ que mapeia arestas em números
- Definindo uma ordem total nas arestas e preservando a ordem por F garante conversão e inversão fáceis

Sistema quadrático

- Em termos de $F[(i, j)]$ com $(i, j) \in A$, temos:

$$r_{ij} \longrightarrow H_{ab}$$

Tal que

$$H_{ab} = 0 \qquad a \neq b$$

$$H_{F[(i, j)], F[(i, j)]} = r_{ij} \qquad \text{caso contrário}$$

Sistema quadrático

- Em termos de $F[(i, j)]$ com $(i, j) \in A$, temos:

$$r_{ij} \longrightarrow H_{ab}$$

Tal que

$$H_{ab} = 0 \quad a \neq b$$

$$H_{F[(i,j)], F[(i,j)]} = r_{ij} \quad \text{caso contrário}$$



$$A_{i, F[(i,j)]} = 1 \quad \forall i, j : (i, j) \in A$$

$$A_{i, F[(i,j)]} = -1 \quad \forall i, j : (j, i) \in A$$

$$A_{a,b} = 0 \quad \text{Caso contrário}$$

Sistema quadrático

Tendo uma conversão organizada e codificada em um programa, podemos usar o GNU Octave para resolver o sistema quadrático.

Resolução Fluxo em Redes

Tabela de segmentação dos fluxos

Segmento	1->2	2->7	2->4	2->3	3->5	3->2	4->6	5->6	6->4	7->5	tempo (ms)
1	0	100	100	0	0	0	50	0	0	50	2.147
2	25	75	75	0	25	0	50	0	0	50	2.593
3	50	50	50	0	50	0	50	0	0	50	2.476
4	38	62	62	0	38	0	38	12	12	38	2.385
8	43	57	38	19	38	5	38	36	31	19	4.616

Minimização Quadrática - Octave

Minimização Quadrática - Octave

1->2	2->7	2->4	2->3	3->5	3->2	4->6	5->6	6->4	7->5	tempo (ms)
50.000	50.000	8.333	41.667	8.333	41.667	25.000	33.333	25.000	33.333	245

Sistema Linear - Numerical Recipes

Sistema Linear - Numerical Recipes

1->2	2->7	2->4	2->3	3->5	3->2	4->6	5->6	6->4	7->5	tempo (ms)
50.000	50.000	8.333	41.667	8.333	41.667	25.000	33.333	25.000	33.333	2

netgen.c

net generator Klingman

netgn151

método	tempo (ms)
nr	4
octave	86000

netgn101

método	tempo (s)
nr	900
octave	∞

Conclusões

- O algoritmo em redes é mais eficiente que a resolução em sistemas de equações lineares.

Conclusões

- O algoritmo em redes é mais eficiente que a resolução em sistemas de equações lineares.
- O equilíbrio está associado à condição de optimalidade do problema linear associado.

Dúvidas

