PCC173/BCC463 - Otimização em Redes

Marco Antonio M. Carvalho

Departamento de Computação
Instituto de Ciências Exatas e Biológicas
Universidade Federal de Ouro Preto

Conteúdo

Aviso

Fonte

Este material é baseado no material

- Top Coder. *Minimum cost flow part I: key concepts*. Disponível em https://www.topcoder.com/community/competitive-programming/tutorials/minimum-cost-flow-part-one-key-concepts/.
- Top Coder. *Minimum cost flow part II: algorithms*. Disponível em https://www.topcoder.com/community/data-science/data-science-tutorials/minimum-cost-flow-part-two-algorithms/.

Licença

Fluxo de Custo Mínimo

Introdução

O **Problema de Fluxo de Custo Mínimo** é uma generalização importante do problema de fluxo máximo, visto anteriormente.

Trata-se de um problema de programação linear com diversas aplicações práticas e que pode ser solucionado de maneira extremamente eficiente.

Entre os problemas que podem ser modelados como o problema de fluxo de custo mínimo, cita-se o problema de **transporte** e o problema de **atribuição**, entre outros.

Custos

Os custos são associados à utilização de cada arco, **por unidade de fluxo**.

Tipos de Vértices

As redes de problemas de fluxo de custo mínimo, possuem pelo menos um **vértice de oferta** e um **vértice de demanda**, embora seja o caso comum em que existam mais de um destes vértices.

Vértices de oferta **produzem** fluxo a uma certa taxa fixa, ao passo que vértices de demanda **consomem** fluxo também a uma determinada taxa fixa.

Vértices intermediários que não produzem ou consomem fluxo são denominados **vértices de transbordo**.

Fluxo de Custo Mínimo

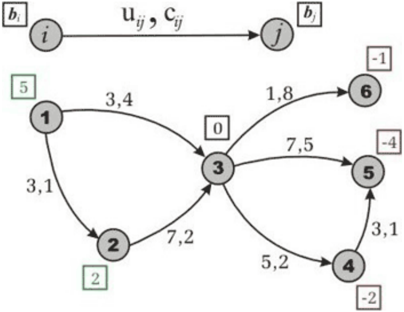
Definição

Dada uma rede $R = (V, A, U, B, C)$, em que:

- V é o conjunto de n vértices;
- A é o conjunto de m arcos;
- U é o conjunto das capacidades u_{ij} não negativas em cada um dos arcos $(i, j) \in A$;
- B é conjunto das demandas/ofertas de fluxo b_i de cada um dos vértices $i \in V$
  - um valor b_i positivo indica oferta;
  - um valor b_i negativo indica demanda;
  - um valor b_i nulo indica transbordo.
- C é conjunto dos custos c_{ij} por unidade de fluxo em cada um dos arcos $(i, j) \in A$.

O objetivo é determinar o menor custo do fluxo F que satisfaça a demanda B dos vértices da rede e respeite as capacidades U de cada arco utilizado.

Fluxo de Custo Mínimo



Exemplo de rede com 2 vértices de oferta (1 e 2), 3 vértices de demanda (6, 5 e 4) e um vértice de transbordo (3).

Formulação Geral

Se denominarmos por x_{ij} o fluxo em cada arco $(i,j) \in A$, então podemos escrever a formulação geral do problema de Fluxo de Custo Mínimo da seguinte forma:

$$min\ z = \sum_{(i,j) \in A} c_{ij} x_{ij} \tag{1}$$

$$sujeito\ a:$$

$$\sum_{(i,j) \in A} x_{ij} - \sum_{(k,i) \in A} x_{ki} = b_i \quad i = 1, \ldots, n \tag{2}$$

$$0 \leq x_{ij} \leq u_{ij}\ \forall\ (i,j) \in A \tag{3}$$

Transformação

Podemos transformar a rede R para termos apenas um vértice de oferta e apenas um vértice de demanda.

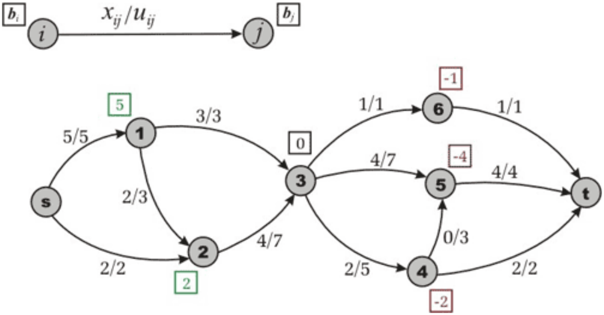
Para isto, adicionamos dois vértices artificiais e também arcos artificiais de custo zero ligando-os aos verdadeiros vértices de oferta e demanda.

Os arcos que ligam o vértice de oferta artificial aos vértices i de oferta possuem capacidade b_i, ao passo que os arcos que ligam os vértices j de demanda possuem capacidade $-b_j$.

Com a rede transformada, podemos determinar se há solução viável para o problema: basta executar algum algoritmo de fluxo máximo e verificar se os arcos artificiais foram esgotados.

Adicionalmente, podemos usar este fluxo máximo como fluxo viável posteriormente na rede original R (sem vértices e arcos artificiais, portanto).

Fluxo de Custo Mínimo



Rede transformada. Por simplicidade, os custos dos arcos foram omitidos.

Fluxo de Custo Mínimo vs. Fluxo Máximo

Claramente, o Problema de Fluxo Máximo é um caso particular do Problema de Fluxo de Custo Mínimo.

Em uma rede R do Problema de Fluxo Máximo:

- Há apenas um vértice de oferta;
- Há apenas um vértice de demanda;
- Todos os demais vértices são de transbordo;
- O custo de cada arco é zero;
- A função objetivo deve ser maximizada.

Fluxo de Custo Mínimo vs. Problema de Caminho Mais Curto

Problema de Caminho Mais Curto também é um caso particular do Problema de Fluxo de Custo Mínimo.

Em um grafo G do Problema de Caminho Mais Curto (*Single-Pair Shortest Path Problem*):

- Há apenas um vértice de oferta;
- Há apenas um vértice de demanda;
- Todos os demais vértices são de transbordo;
- Os arcos não possuem restrições de capacidade.

Fluxo de Custo Mínimo

Assunções

Ao resolver problemas de fluxo de custo mínimo, fazemos algumas assunções, embora às vezes isto possa levar à perda de generalidade:

- Todos os dados u_{ij}, c_{ij} e b_i são inteiros;
- A rede é direcionada;
- Todos os custos são não negativos;
- As ofertas e demandas dos vértices satisfazem $\sum_{i \in V} b_i = 0$ e o problema possui solução viável.

Rede Residual

Extendemos o conceito de **rede residual** (ou grafo de aumento de fluxo) do problema de fluxo máximo para utilização no problema de fluxo de custo mínimo.

Supondo um fluxo x_{ij} em um arco $(i, j) \in A$ de uma rede R, sua **capacidade residual** é definida como $r_{ij} = u_{ij} - x_{ij}$.

Isto significa dizer que é possível enviar r_{ij} unidades adicionais de fluxo do vértice i para o vértice j e também que é possível cancelar x_{ij} unidades de fluxo entre i e j.

Cada unidade adicional de fluxo no arco (i, j) aumenta a função objetivo em c_{ij} unidades, ao passo que cada unidade de fluxo cancelada diminui a função objetivo no mesmo montante.

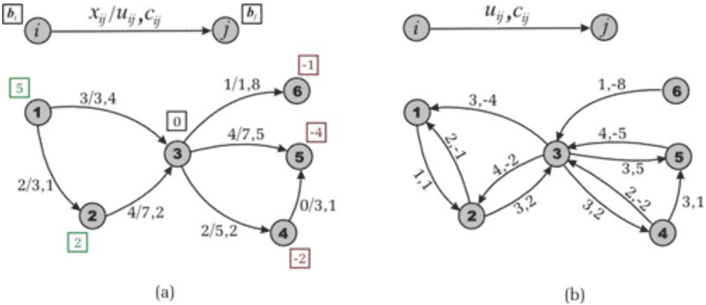
Fluxo de Custo Mínimo

Rede Residual

Desta maneira, dada uma rede R, uma solução viável gera uma uma rede residual $R_f = (V, A_f)$, em que A_f é o conjunto de arcos residuais, tal que cada arco $(i, j) \in A$ é substituído por dois arcos:

- arco (i, j) com custo c_{ij} e capacidade $r_{ij} = u_{ij} - x_{ij}$;
- arco (j, i) com custo $-c_{ij}$ e capacidade $r_{ij} = x_{ij}$.

Adicionalmente, podemos encontrar situações em que temos diversos arcos paralelos, os quais **não poderão** ser unificados como no problema de fluxo máximo.



Rede do exemplo anterior, com uma solução viável (a) e rede residual correspondente.

Algoritmo de Cancelamento de Ciclos

Histórico

O Algoritmo de Cancelamento de Ciclos foi proposto em 1964 por Balinski e Gomery.

Este algoritmo é "dual" ao algoritmo Húngaro, e foi originalmente proposto para os problemas de atribuição e transporte.

Princípio

Este algoritmo utiliza redes residuais e se baseia em dois teoremas sobre a existência de solução viável e condições de otimalidade.

Algoritmo de Cancelamento de Ciclos

Teorema 1: Existência de Solução

Seja R uma rede. Suponha que R não contém ciclos não capacitados de custo negativo e que existe uma solução viável para o problema de fluxo de custo mínimo. Então existe uma solução ótima.

Teorema 2: Condição de Otimalidade de Ciclos Negativos

Seja x^* uma solução viável de um problema de fluxo de custo mínimo. Então x^* será uma solução ótima se e somente se a rede residual R_f não contiver um ciclo (direcionado) de custo negativo.

Algoritmo de Cancelamento de Ciclos

Princípio

O algoritmo de Cancelamento de Ciclos utiliza algum algoritmo de fluxo máximo para estabelecer um fluxo viável na rede R.

Após isto, o algoritmo tenta melhorar o valor da função objetivo buscando ciclos de custo negativo na rede residual R_f e aumentando o fluxo nestes ciclos.

Terminologia

- R_f: rede residual;
- W: ciclo de custo negativo em R_f;
- δ: menor capacidade residual de um arco pertencente a W.

Algoritmo de Cancelamento de Ciclos

Entrada: Rede R

Determine um fluxo viável F em R;
Construa R_f a partir de R;
enquanto R_f *contiver ciclos de custo negativo* W **faça**
    Identifique o ciclo W;
    $\delta \leftarrow min\{r_{ij} : (i,j) \in W\}$
    Aumente o fluxo em W em δ unidades de fluxo;
    Atualize a rede residual R_f;
fim

Algoritmo de Cancelamento de Ciclos

Complexidade

Seja a capacidade máxima de um arco denotada por U_{max} e o maior valor de custo de um arco C_{max}.

Em um problema de fluxo de custo mínimo, o valor da função objetivo é limitado por $mC_{max}U_{max}$.

Qualquer ciclo cancelado diminui o valor da função objetivo por um valor positivo inteiro e, dado que todos os valores são inteiros, o algoritmo executa $O(mC_{max}U_{max})$ iterações.

Se utilizarmos um algoritmo $O(nm)$ (como o algoritmo de correção de rótulos) para identificar ciclos negativos, teremos complexidade total $O(nm^2C_{max}U_{max})$.

Algoritmo de Cancelamento de Ciclos

Complexidade Aprimorada

Em 1997, Sokkalingam, Ahuja e Orlin propuseram um algoritmo de cancelamento de ciclos de complexidade **fortemente polinomial**: $O(m(m + nlogn)min\{log(nU_{max}), mlogn)\}$.

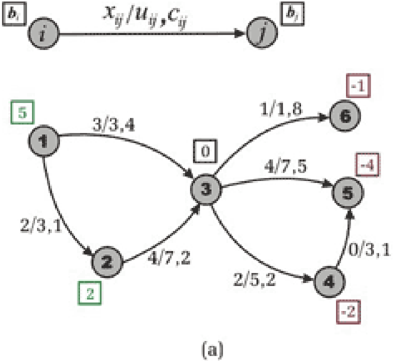
Complexidade Fortemente Polinomial

Os conceitos de fortemente e fracamente polinomial são relevantes apenas para problemas cujas entradas consistem de números inteiros.

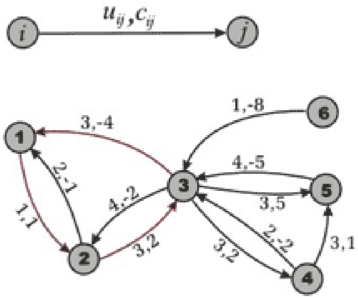
Considerando o modelo computacional aritmético, em que as operações básicas (mais a comparação de valores) são realizadas em uma unidade de tempo, independente do tamanho dos operandos, um algoritmo é executado em tempo fortemente polinomial se:

- O número de operações realizadas é limitado por um polinômio no número de inteiros da entrada; e
- O espaço utilizado pelo algoritmo é limitado por um polinômio no tamanho da entrada.

Algoritmo de Cancelamento de Ciclos



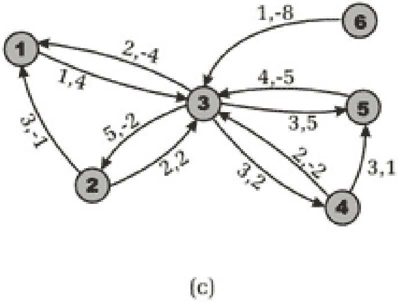
Rede com solução viável de custo 54.



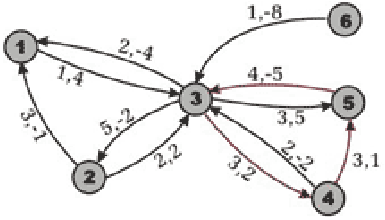
(b)

Ciclo de custo negativo 1-2-3-1 detectado na rede residual, com custo -1 e capacidade mínima 1.

Algoritmo de Cancelamento de Ciclos

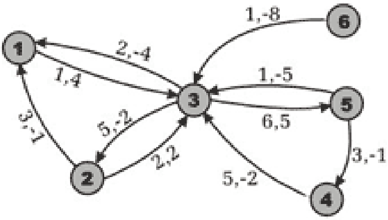


Rede residual após aumento do fluxo no ciclo.



(d)

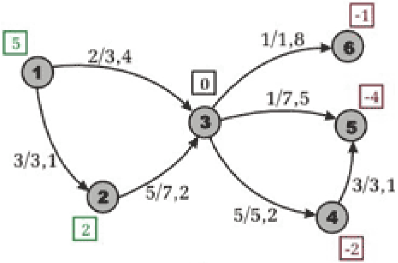
Outro ciclo de custo negativo 3-4-5-3 detectado, com custo -2 e capacidade mínima 3.



(e)

Rede residual após aumento do fluxo no ciclo, sem nenhum ciclo de custo negativo.

Algoritmo de Cancelamento de Ciclos



Rede original e fluxo de custo ótimo, igual a 47.

Dúvidas?