

CAPÍTULO IV

IV.1 - ALGORITMOS NÃO-POLINOMIAIS (Introdução):

Trataremos neste capítulo de dois algoritmos de fluxo de custo mínimo: o algoritmo de ciclos de custo negativo e o algoritmo de caminhos mínimos sucessivos. O primeiro é consequência imediata das condições dos ciclos de custo negativo e o segundo (embora não imediato) das condições de otimalidade de custos reduzidos. A abordagem é um pouco diversa já que no algoritmo dos ciclos de custo negativo mantemos sempre a viabilidade primal buscando obter viabilidade dual. No algoritmo de caminhos mínimos sucessivos o processo é justamente o contrário, ou seja, mantemos viabilidade dual buscando obter viabilidade primal. Os dois algoritmos apresentados são não polinomiais.

IV.2 - ALGORITMO DE CICLOS DE CUSTO NEGATIVO:

Este algoritmo mantém a viabilidade primal x e busca obter viabilidade dual. O fluxo inicial x_0 pode ser obtido através da solução de um problema de fluxo máximo (seção II.2). O algoritmo procura determinar ciclos direcionados na rede residual $G(x)$ e aumentar o fluxo nestes ciclos. Ele termina quando não existirem mais ciclos de custo negativo na rede residual (satisfazendo portanto as condições de otimalidade).

No algoritmo abaixo r_{ij} representa a capacidade do arco (i,j) na rede residual:

Algoritmo: Ciclos Negativos;

Início

Obter fluxo x viável; r_{ij}

Enquanto $G(x)$ possuir algum ciclo negativo **faça**

Identificar ciclo w de custo negativo;

$d = \min\{r_{ij} : (i, j) \in w\}$;

Aumentar δ unidades de fluxo no ciclo w e atualizar $G(x)$;

fim;

fim.

A determinação dos ciclos de custo negativo é discutida na seção II.6. Para maiores detalhes vide Ahuja, seção 5.5. Abaixo representamos um exemplo ilustrando a execução do algoritmo:

Exemplo IV.1) Seja o grafo abaixo com 4 nós e 6 arcos. Desejamos enviar 2 unidades de fluxo de **s** a **t** com o menor custo possível. Os pares ordenados ao lado de cada arco representam o custo e a capacidade respectivamente:

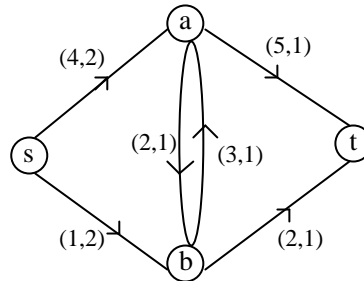


Figura IV.1: Rede Original

A primeira coordenada de cada par (i,j) representa o custo, a segunda representa a capacidade de cada arco.

Suponha que exista uma unidade de fluxo ao longo do caminho (s,a,t) e uma unidade ao longo de (s,a,b,t) . Observe que o custo total será 17. Logo, a rede residual correspondente ao fluxo: $x_{sa} = 2, x_{at} = x_{ab} = x_{bt} = 1, x_{sb} = x_{ba} = 0$ será:

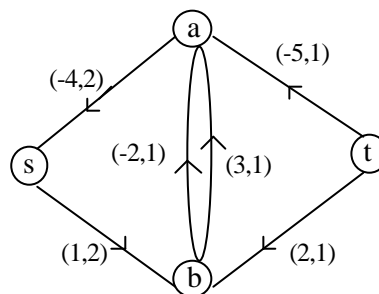


Figura IV.2: Rede Residual c/ Ciclo Negativo

Note que podemos enviar um fluxo de uma unidade ($d = \min\{r_{ij} : (i,j) \in w\}$) ao longo do ciclo (s,b,a,s) . Observe que o custo correspondente será -5. Chamando de x_0 o fluxo inicial e y o fluxo unitário ao longo do ciclo (s,b,a,s) , o novo fluxo x será dado por $x = x_0 + y$.

Desta forma, no novo fluxo enviaremos uma unidade no caminho (s,a,t) e uma unidade no caminho (s,b,t) . A rede residual correspondente será:

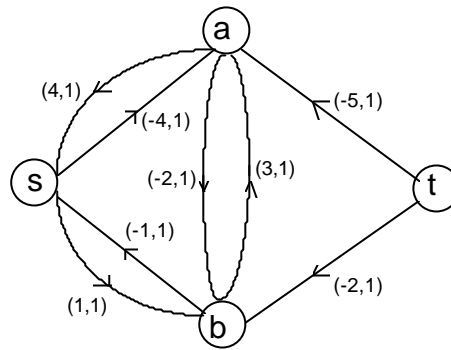


Figura IV.3: Rede Residual s/ Ciclo Negativo

Neste caso não temos mais nenhum ciclo de custo negativo, chegando portanto a uma solução ótima.

IV.2.1 - Análise de Complexidade:

Para o cálculo de complexidade do algoritmo acima, é necessário inicialmente que se analise a complexidade de um algoritmo para a determinação do ciclo de custo negativo na rede residual. Como mencionado no capítulo II, o algoritmo de correção de rótulos FIFO (que requer no pior caso $O(nm)$ passos) poderia ser implementado. Uma outra alternativa seria utilizar o simplex de rede (de complexidade pseudo-polinomial, vide Ahuja seção 11.7).

Note que mCU é uma cota superior para o fluxo inicial do problema de fluxo de custo mínimo ($c_{ij} \leq C$ e $x_{ij} \leq U, \forall (i, j) \in A$). A cada passo do algoritmo o valor da função objetivo sofre uma alteração de:

$$\left(\sum_{(i,j) \in w} c_{ij} \right) d$$

unidades (observe que este valor é estritamente negativo). Como, por hipótese, todos os dados do problema são inteiros o algoritmo termina em no máximo $O(mCU)$ iterações totalizando portanto $O(nm^2CU)$ unidades de tempo. Note que estamos supondo que o algoritmo de correção de rótulos FIFO está sendo utilizado na determinação dos ciclos de custo negativo.

IV.3 - ALGORITMO DE CAMINHOS MÍNIMOS SUCESSIVOS:

No algoritmo de ciclos de custo negativo mantínhamos a viabilidade da solução (viabilidade primal) buscando satisfazer as condições de otimalidade (viabilidade dual). No algoritmo

de caminhos mínimos sucessivos mantemos a viabilidade dual ($\bar{c}_{ij} \geq 0, \forall (i, j) \in G(x)$ onde x não é necessariamente viável) e tentamos obter viabilidade primal.

Nosso problema de fluxo de custo mínimo, como vimos, pode ser formulado como:

$$\begin{aligned} \min z(x) &= \sum_{(i,j) \in A} c_{ij} x_{ij} \\ \text{s.a} \quad &\sum_{\{j:(i,j) \in A\}} x_{ij} - \sum_{\{k:(k,i) \in A\}} x_{ki} = b(i), \quad \forall i \in N \quad (I) \\ &0 \leq x_{ij} \leq u_{ij}, \quad \forall (i,j) \in A \end{aligned}$$

Em nosso caso o vetor x deve satisfazer apenas as restrições de capacidade e não-negatividade, violando portanto, as restrições de conservação de fluxo em cada nó. Temos assim as seguintes definições:

Definição IV.1: Um *pseudofluxo* é uma função $f: x \in \Re^m \rightarrow \Re^+$ satisfazendo somente as restrições de capacidade e não-negatividade (m representa o número total de arcos da rede original). •

Definição IV.2: Para cada pseudofluxo x definimos o *desajuste* de cada nó i como:

$$e(i) = b(i) + \sum_{\{j:(j,i) \in A\}} x_{ji} - \sum_{\{k:(i,k) \in A\}} x_{ik}; \quad \forall i \in N \quad \bullet$$

Se $e(i) > 0$ para algum i , nos referimos a $e(i)$ como *excesso* do nó i , se $e(i) < 0$ nos referimos a $-e(i)$ como *déficite* do nó i . Um nó estará *balanceado* quando $e(i) = 0$. Sejam $E = \{i: e(i) > 0\}$ e $D = \{i: e(i) < 0\}$ conjunto de nós com excesso e déficite respectivamente.

Importante: A rede residual $G(x)$ correspondente a um pseudofluxo x é definida de maneira idêntica a que definimos anteriormente para um fluxo viável x qualquer. •

Utilizando o conceito de pseudofluxo e as condições de otimalidade de custo reduzido provamos o importante resultado que será utilizado na elaboração do algoritmo de caminhos mínimo sucessivos:

Teorema IV.1: Suponha que um pseudofluxo (ou fluxo) x satisfaça as condições de otimalidade com respeito às variáveis duais π , ou seja, $\bar{c}_{ij} = c_{ij} - \pi(i) + \pi(j) \geq 0 \quad \forall (i, j) \in G(x)$, ou seja, π é

um vetor dual viável em $G(x)$. Seja \mathbf{d} o caminho mais curto de s aos demais nós em $G(x)$ com respeito aos custos reduzidos \bar{c}_{ij} . Assim, temos o seguinte resultado:

- a) O pseudofluxo x também satisfaz as condições de otimalidade com respeito às variáveis duais $\pi' = \pi - \mathbf{d}$, ou seja $\bar{c}_{ij}' \geq 0, \forall (i, j) \in G(x)$.
- b) Os custos reduzidos $\bar{c}_{ij}' = c_{ij} - \mathbf{p}'(i) + \mathbf{p}'(j)$ são zero para todos os arcos (i, j) pertencentes ao caminho mais curto do nó s a um outro nó qualquer de $G(x)$.

Demonstração: Como x satisfaz as condições de otimalidade de custo reduzido com respeito a π , temos $\bar{c}_{ij} \geq 0, \forall (i, j) \in G(x)$.

O vetor \mathbf{d} representa o caminho mais curto do nó s a todos os outros nós sendo \bar{c}_{ij} o comprimento de cada arco (i, j) . Neste caso, como não temos ciclos de custo negativo o vetor \mathbf{d} irá satisfazer as condições de otimalidade, ou seja:

$$d(j) \leq d(i) + \bar{c}_{ij}, \quad \forall (i, j) \in G(x)$$

Substituindo $\bar{c}_{ij} = c_{ij} - \mathbf{p}(i) + \mathbf{p}(j)$ na desigualdade acima temos: $d(j) \leq d(i) + c_{ij} - \mathbf{p}(i) + \mathbf{p}(j)$. Ou ainda: $c_{ij} - (\mathbf{p}(i) - d(i)) + (\mathbf{p}(j) - d(j)) \geq 0$. Como $\mathbf{p}'(i) = \mathbf{p}(i) - d(i)$ e $\mathbf{p}'(j) = \mathbf{p}(j) - d(j)$ temos $\bar{c}_{ij}' \geq 0$. Terminamos portanto a parte **a**.

Para demonstrar a parte **b** basta observar que $d(j) = d(i) + \bar{c}_{ij}$ no caminho mínimo de **a** a um nó qualquer **l** (note que i e j devem pertencer a este caminho mínimo). Substituindo $\bar{c}_{ij} = c_{ij} - \mathbf{p}(i) + \mathbf{p}(j)$ na equação acima temos $d(j) = d(i) + c_{ij} - \mathbf{p}(i) + \mathbf{p}(j)$. Portanto: $\bar{c}_{ij}' = c_{ij} - \mathbf{p}'(i) + \mathbf{p}'(j) = 0, \forall (i, j)$ no caminho mínimo de s a outro nó qualquer da rede. •

O próximo resultado nos fornece uma maneira de atualizar o pseudofluxo x (ou fluxo x) mantendo as condições de otimalidade de custo reduzido.

Teorema IV.2: Suponha que um pseudofluxo (ou fluxo x) satisfaça as condições de otimalidade de custo reduzido, e seja x' um pseudofluxo (ou fluxo) obtido com envio de fluxo através do caminho mais curto da fonte s a um outro nó qualquer k . Então x' também satisfaz as condições de otimalidade de custo reduzido.

Demonstração: Sejam π e π' variáveis duais obtidas como no teorema anterior. Temos também que qualquer arco (i, j) pertencente ao caminho mínimo P de s a k satisfaz $\bar{c}_{ij}' = 0$.

Aumentando-se o fluxo neste caminho a rede residual sofrerá alteração somente ao longo de P permanecendo constante nos demais arcos da rede residual.

Observe que aumentando o fluxo em um arco $(i,j) \in P$ poderíamos criar um arco contrário (j,i) na rede residual. Como $\bar{c}_{ij}' = c_{ij} - \mathbf{p}(i) + \mathbf{p}(j) = 0 \quad \forall (i,j) \in P$ teremos $\bar{c}_{ji}' = 0$, assim, o arco (j,i) irá satisfazer também as condições de otimalidade com respeito ao novo fluxo •

Note que esse dois teoremas nos dão as condições necessárias para elaboração de um novo algoritmo de fluxo de custo mínimo. Tomamos primeiramente um pseudofluxo x_0 obtendo portanto uma rede residual $G(x_0)$. Escolhemos uma solução dual inicial \mathbf{p}_0 onde $\bar{c}_{ij} = c_{ij} - \mathbf{p}_0(i) + \mathbf{p}_0(j) \geq 0 \quad \forall (i,j) \in G(x_0)$. Calculamos agora \mathbf{d} com respeito aos custos reduzidos \bar{c}_{ij} e atualizamos π segundo o teorema IV.1.a, ou seja, fazemos $\pi' = \pi - \mathbf{d}$, obtendo desta forma, uma nova solução dual viável (ou seja, $\bar{c}_{ij}' \geq 0$). Calculamos agora todos os custos reduzidos \bar{c}_{ij}' em $G(x_0)$. Do teorema IV.1.b, $\bar{c}_{ij}' = 0$ qualquer que seja o arco (i,j) ao longo do caminho entre s (nó com excesso, $e(s) > 0$) ao nó t (nó com déficit, $e(t) < 0$). Enviamos o maior fluxo possível neste caminho. Segundo o teorema IV.2, este fluxo também satisfaz as condições de otimalidade com respeito aos custos reduzidos \bar{c}_{ij}' . Calculamos agora a nova rede residual com respeito a este fluxo e repetimos o processo até que todos os nós estejam balanceados.

O algoritmo de caminhos mínimos sucessivos é descrito abaixo. Analogamente ao algoritmo de ciclos de custo negativo, r_{ij} representa a capacidade dos arcos (i,j) na rede residual:

Algoritmo: Caminhos Mínimos Sucessivos;

Início

$x := 0; \pi := 0;$

$e(i) := b(i)$ para todo $i \in N$;

Inicializar $E = \{i : e(i) > 0\}$ e $D = \{i : e(i) < 0\}$

Enquanto $E \neq \emptyset$ **faça**

 Selecione $k \in E$ e $l \in D$;

 Determine o caminho mais curto de s a todos os outros nós em $G(x)$ com respeito a \bar{c}_{ij} ;

$\mathbf{p} = \mathbf{p} - \mathbf{d}$;

 Calculamos $\bar{c}_{ij}' \quad \forall (i,j) \in G(x)$;

 Seja P o caminho mais curto de k até l ;

$\mathbf{d} = \min\{e(k), -e(l), \min\{r_{ij} : (i,j) \in P\}\}$

 Aumente δ unidades de fluxo ao longo do caminho P ;

 Atualize $x, G(x), E, D$ e os custos reduzidos;

fim enq;

fim.

Como observado acima, no algoritmo de caminhos mínimos sucessivos selecionamos $s \in E = \{i: e(i) > 0\}$ e $t \in D = \{i: e(i) < 0\}$ e enviamos um fluxo de s a t ao longo do caminho mínimo entre estes vértices (na rede residual). O algoritmo termina quando todos os nós da rede estiverem balanceados (isto é: $e(i)=0$).

Ao inicializarmos o algoritmo fazemos $x=0$ (pseudofluxo inicial). Temos neste caso que $G(x)=G$. Observe que fazendo $\pi=0$ as condições de otimalidade de custo reduzido são satisfeitas, $\bar{c}_{ij} = c_{ij} \geq 0 \quad \forall (i, j) \in G(x)$ (lembre-se que supomos $c_{ij} \geq 0 \quad \forall (i,j) \in A$ (capítulo III)).

Note que enquanto E e D forem não-vazios não teremos alcançado a viabilidade primal, ou ainda, como mantemos sempre a viabilidade dual a cada passo, para E e D vazios teremos uma solução ótima.

No capítulo III supomos a existência de um caminho direcionado ente cada par de nós de G . Logo, os rótulos $d(\cdot)$ estão bem definidos. A convergência do algoritmo está garantida já que a cada atualização fazemos um decréscimo de E e D respectivamente, na mesma proporção.

Exemplo IV.2: Seja G o seguinte grafo com 4 vértices e 5 arestas como abaixo. Os pares ordenados (\bar{c}_{ij}, r_{ij}) representam custo e capacidade respectivamente na rede residual. Inicialmente temos: $E = \{1\}$ e $D = \{4\}$ ($s=1$ e $t=4$). Para $x=0$ e $\pi=0$ (solução dual viável inicial) temos:

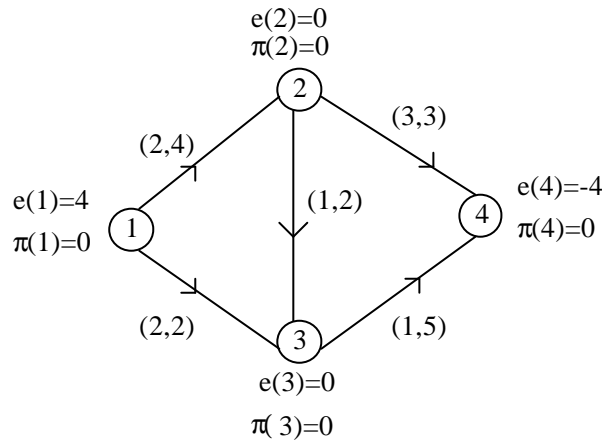


Figura IV.4: Rede Residual Inicial $c/ x=0$ e $\pi=0$

Observe que $x=0$ não é viável pois $e(1)=4$ e $e(4)=-4$. Note que no primeiro passo temos a rede residual igual à rede original. Calculando d com respeito aos custos \bar{c}_{ij} e fazendo $\mathbf{p}' = \mathbf{p} - d$, temos:

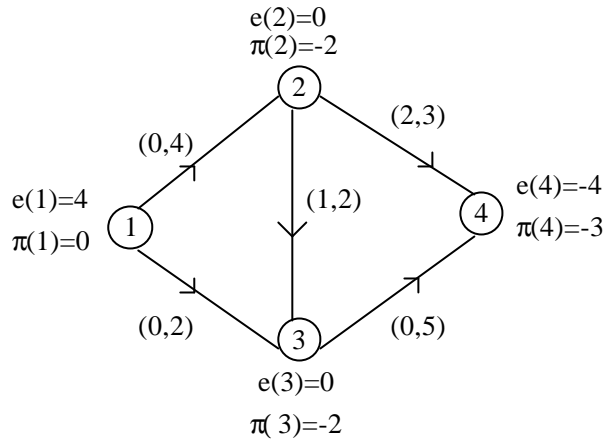


Figura IV.5: Rede Residual após a atualização de π

Observe que $\bar{c}_{13} = \bar{c}_{34} = 0$ no caminho mínimo entre 1 e 4 (vide teorema IV.2). Fazemos agora $\mathbf{d} = \min\{e(1), -e(4), r_{13}, r_{34}\}$ (variação máxima do fluxo neste caminho). Assim $\mathbf{d} = \min\{4, 4, 2, 5\} = 2$. O novo fluxo x' será portanto $x'_{12} = x'_{24} = x'_{23} = 0$, $x'_{13} = x'_{34} = 2$. Abaixo calculamos $G(x')$ e fazemos atualização de E e D :

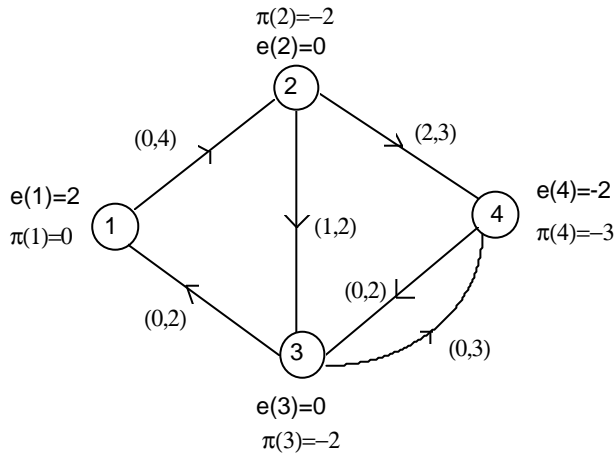


Figura IV.6: Rede Residual após o acréscimo de 2 unidades no caminho 1-3-4.

Repetindo o processo entre os vértices 1 e 4 temos $d(1)=0$, $d(2)=0$, $d(3)=1$ e $d(4)=1$. Assim, calculando o novo π e fazendo a atualização de cada \bar{c}_{ij} , obtemos:

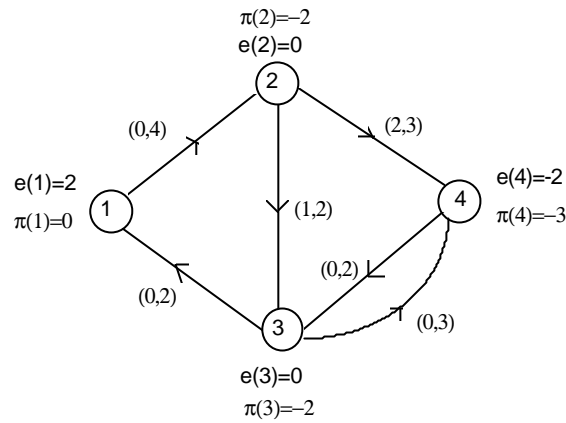


Figura IV.7: Rede após a atualização de π .

Note que $\bar{c}_{12} = \bar{c}_{23} = \bar{c}_{34} = 0$ no caminho mínimo 1-2-3-4 entre 1 e 4. Temos agora que $\mathbf{d} = \min\{e(1), -e(4), r_{12}, r_{23}, r_{34}\} = \min\{2, 2, 4, 2, 3\} = 2$. O novo fluxo x' será portanto: $x'_{12} = x'_{23} = x'_{34} = 2, x'_{24} = 0$ e $x'_{34} = 4$. Após a atualização de E e D a rede residual será:

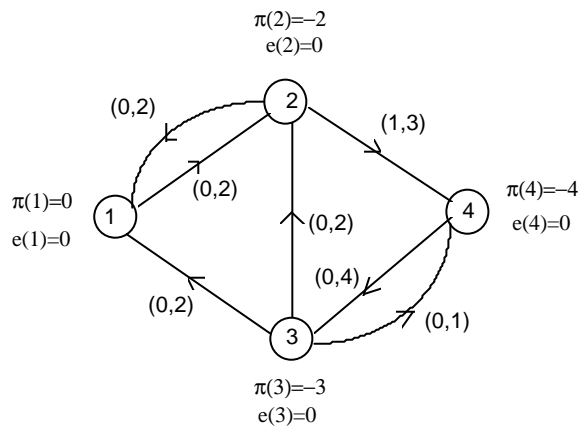


Figura IV.8: Rede Residual após o acréscimo de fluxo em 1-2-3-4

Como todos os nós estão balanceados terminamos portanto o algoritmo.

IV.3.1 - ANÁLISE DE COMPLEXIDADE:

A cada iteração do algoritmo resolvemos um problema de caminho mínimo (com arcos não-negativos) e descrevemos estritamente o excesso de cada nó em E (e obviamente

também de cada nó em D). Como U (definido anteriormente) é um limitante superior para $b(i) \forall i \in N$ o algoritmo termina em no máximo nU iterações, sendo portanto $O(nU)$ iterações. Se $S(n,m,C)$ representa a complexidade do problema de caminho mínimo com arcos não-negativos, a complexidade final será de $O(nUS(n,m,nC))$ passos, sendo portanto pseudo-polinomial. Observe que utilizamos nC ao invés de C na expressão da complexidade acima, pois os custos na rede residual são limitados por nC .