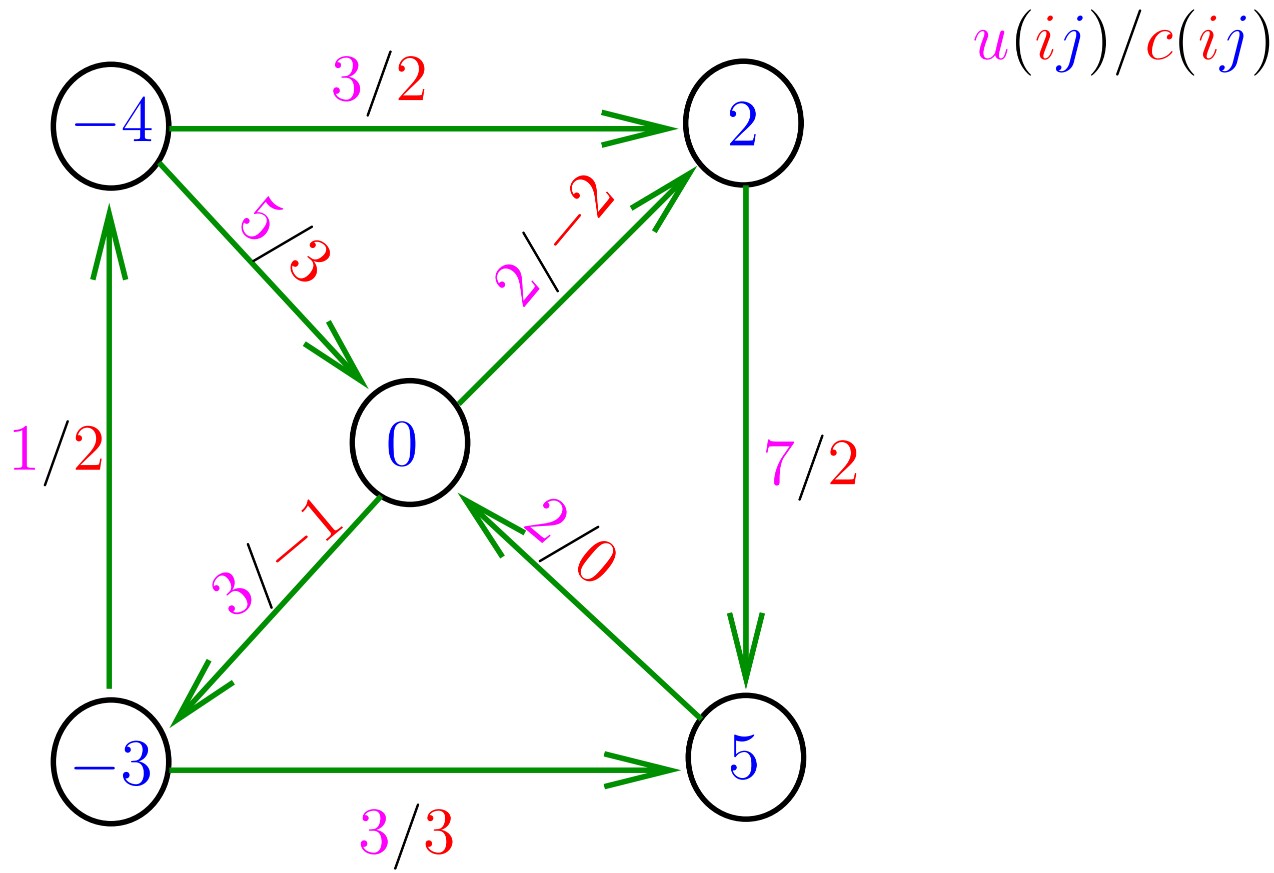
Melhores momentos

AULA PASSADA

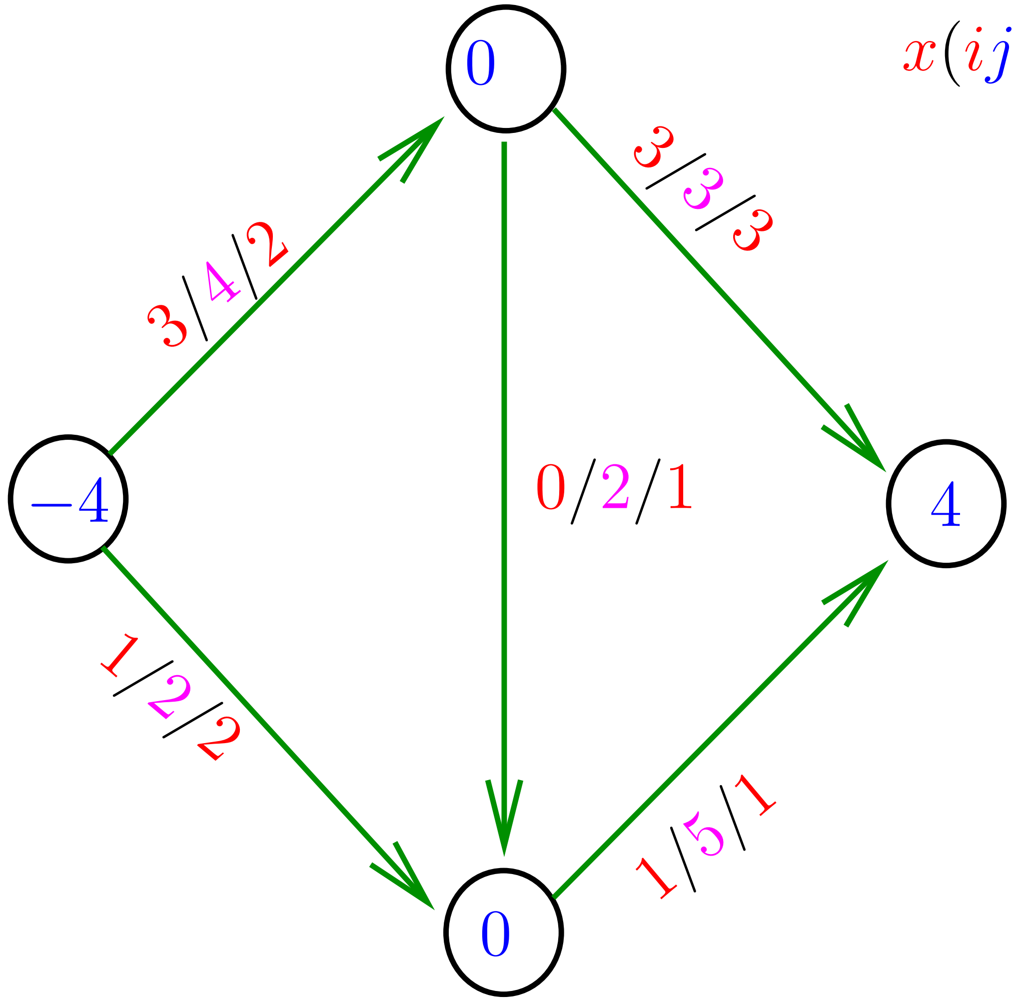
Problema do fluxo viável de custo mínimo

Dada uma rede (N, A, u, b, c) com função-capacidade u, função-demanda b e função-custo c, encontrar um fluxo de custo mínimo que satisfaça b e respeite u.



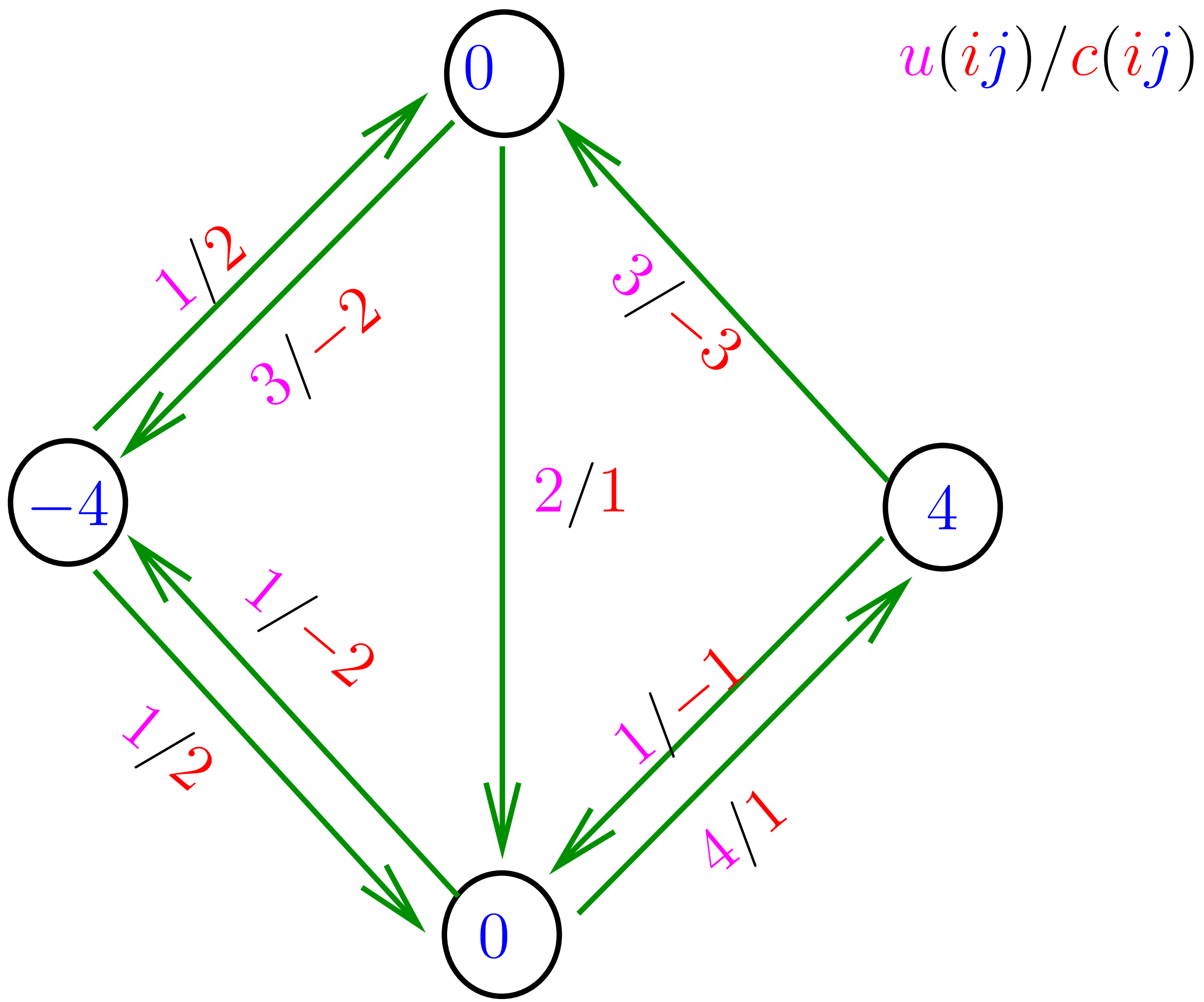
Klein: é fluxo de custo mínimo?

$x(ij)/u(ij)/c(ij)$

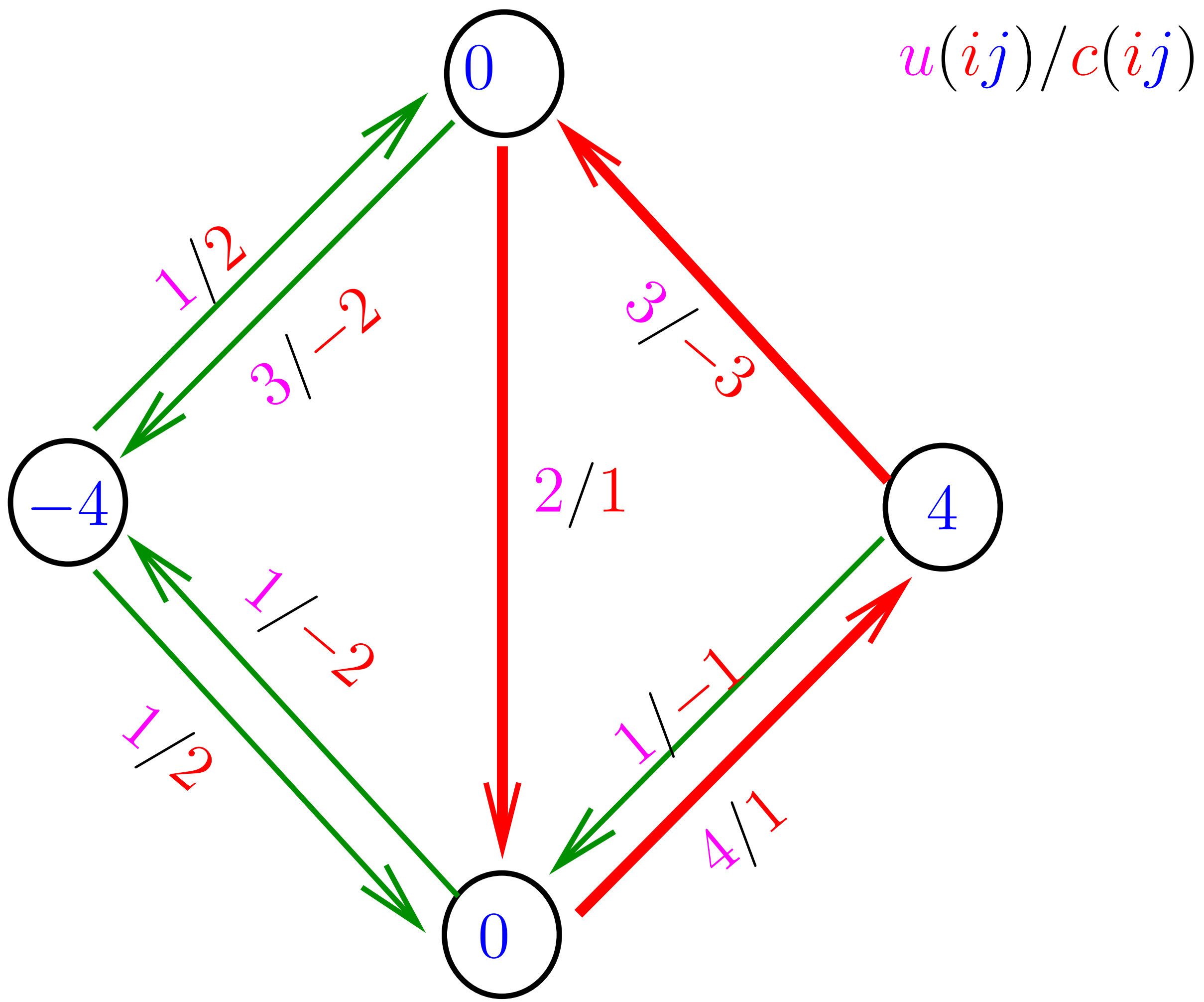


Custo $= 3 \times 2 + 1 \times 2 + 0 \times 1 + 3 \times 3 + 1 \times 1 = 18$

Klein: rede residual



Klein: ciclo negativo



Custo mínimo e ciclo negativo

Se x é um fluxo viável, então vale uma e apenas uma das afirmações:

- x é um fluxo viável de custo mínimo;
- existe um ciclo de custo negativo na rede residual.

Se x é um fluxo viável, então vale uma e apenas uma das afirmações:

- existe uma "certa" função potencial y na rede residual;
- existe um ciclo de custo negativo na rede residual.

Lema da dualidade

Se x é um fluxo viável então

$$cx \geq yb - wu.$$

para qualquer função-custo $w \geq 0$ e $(c+w)$-potencial y.

Chamaremos tal par (y, w) de solução dual-viável.

Conseqüência. Se x é um fluxo viável tal que $cx = yb - wu$ para alguma função-custo $w \geq 0$ e algum $(c+w)$-potencial y então x é um fluxo ótimo.

Folgas complementares (1)

Seja x um fluxo.

Seja y um potencial.

Diremos que as folgas de x e y são complementares se, para cada arco ij,

$$x(ij) > 0 \Rightarrow y(j) - y(i) \geq c(ij) \quad \text{e}$$

$$x(ij) < u(ij) \Rightarrow y(j) - y(i) \leq c(ij)\ .$$

Fato. Se x é um fluxo viável e suas folgas são complementares às de algum potencial y então x é ótimo.

Folgas complementares (2)

Seja x um fluxo.

Seja y um potencial.

Diremos que as folgas de x e y são complementares se, para cada arco ij,

$$y(j) - y(i) < c(ij) \Rightarrow x(ij) = 0 \quad \text{e}$$
$$y(j) - y(i) > c(ij) \Rightarrow x(ij) = u(ij) \; .$$

Fato. Se x é um fluxo viável e suas folgas são complementares às de algum potencial y então x é ótimo.

Fluxo viável de custo mínimo

Problema do fluxo viável de custo mínimo:

Dados

- uma matriz de incidências M de um grafo (N, A),
- uma função-demanda b ($N \to \mathbb{Z}$)
- uma função-custo c ($A \to \mathbb{Z}$) e
- uma função-capacidade u ($A \to \mathbb{Z}_{\geq}$)

encontrar um vetor x indexado por A que

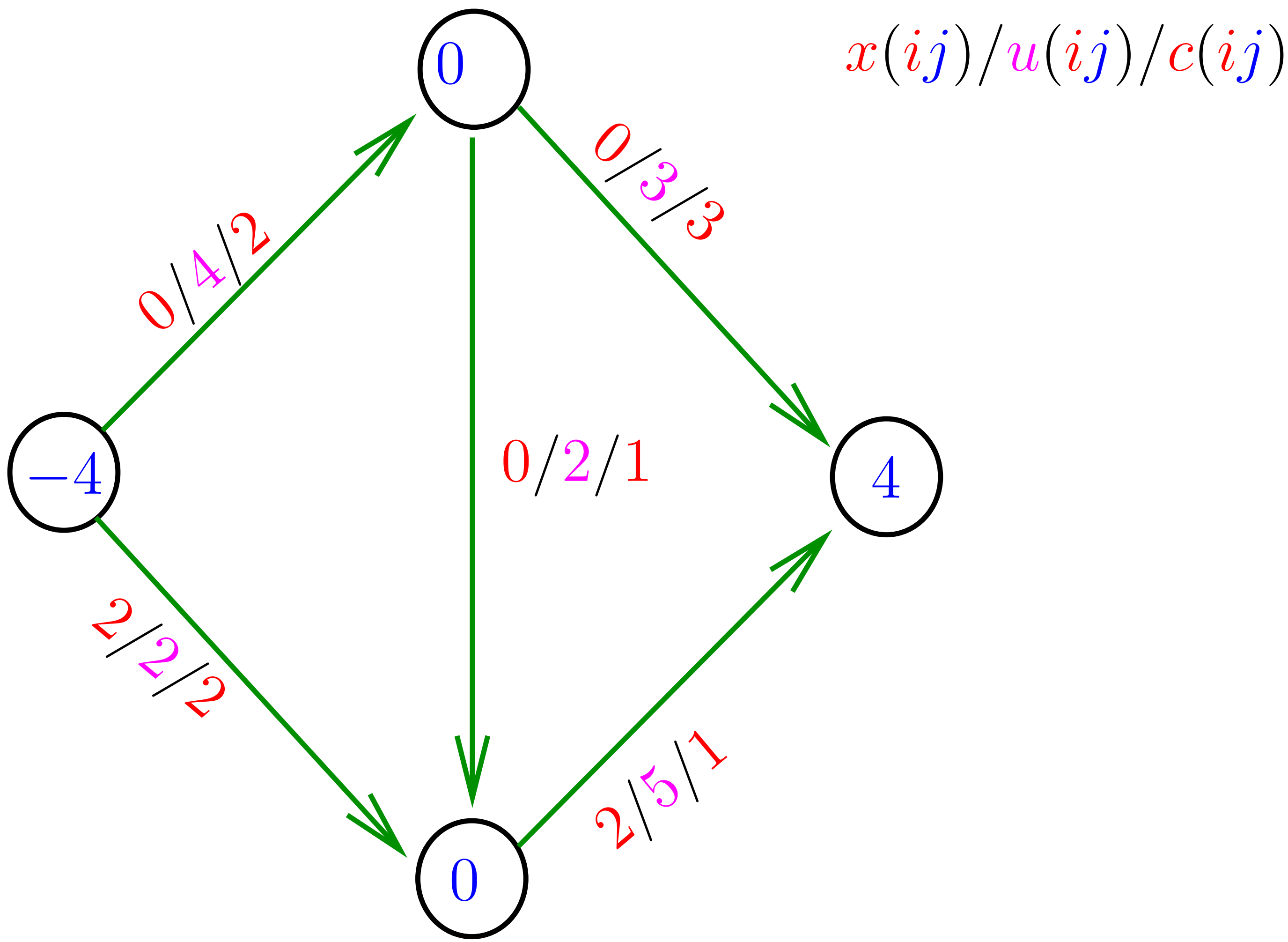
$$\begin{array}{rrcll} \text{minimize} & cx & & & \\ \text{sob as restrições} & Mx & = & b & \\ & x(ij) & \leq & u(ij) & \text{para cada } ij \text{ em } A \\ & x(ij) & \geq & 0 & \text{para cada } ij \text{ em } A. \end{array}$$

Problema dual

O correspondente problema dual é: encontrar vetores y indexado por N e w indexado por A que

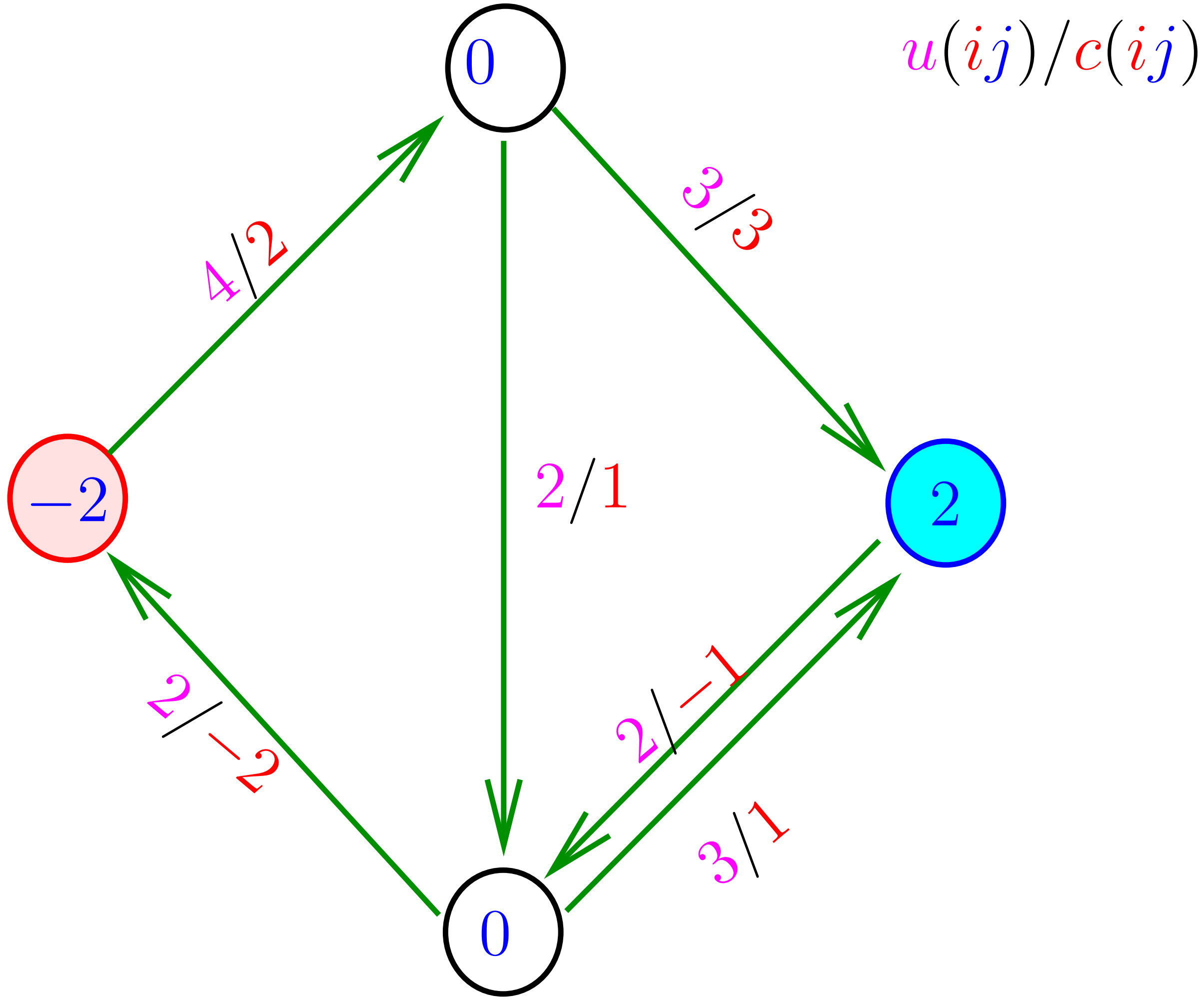
$$\begin{array}{rrcll} \text{maximize} & yb - wu & & & \\ \text{sob as restrições} & y(j) - y(i) + w(ij) & \geq & c(ij) & \text{para } ij \in A, \\ & w(ij) & \geq & 0 & \text{para } ij \in A. \end{array}$$

Jewell: rede

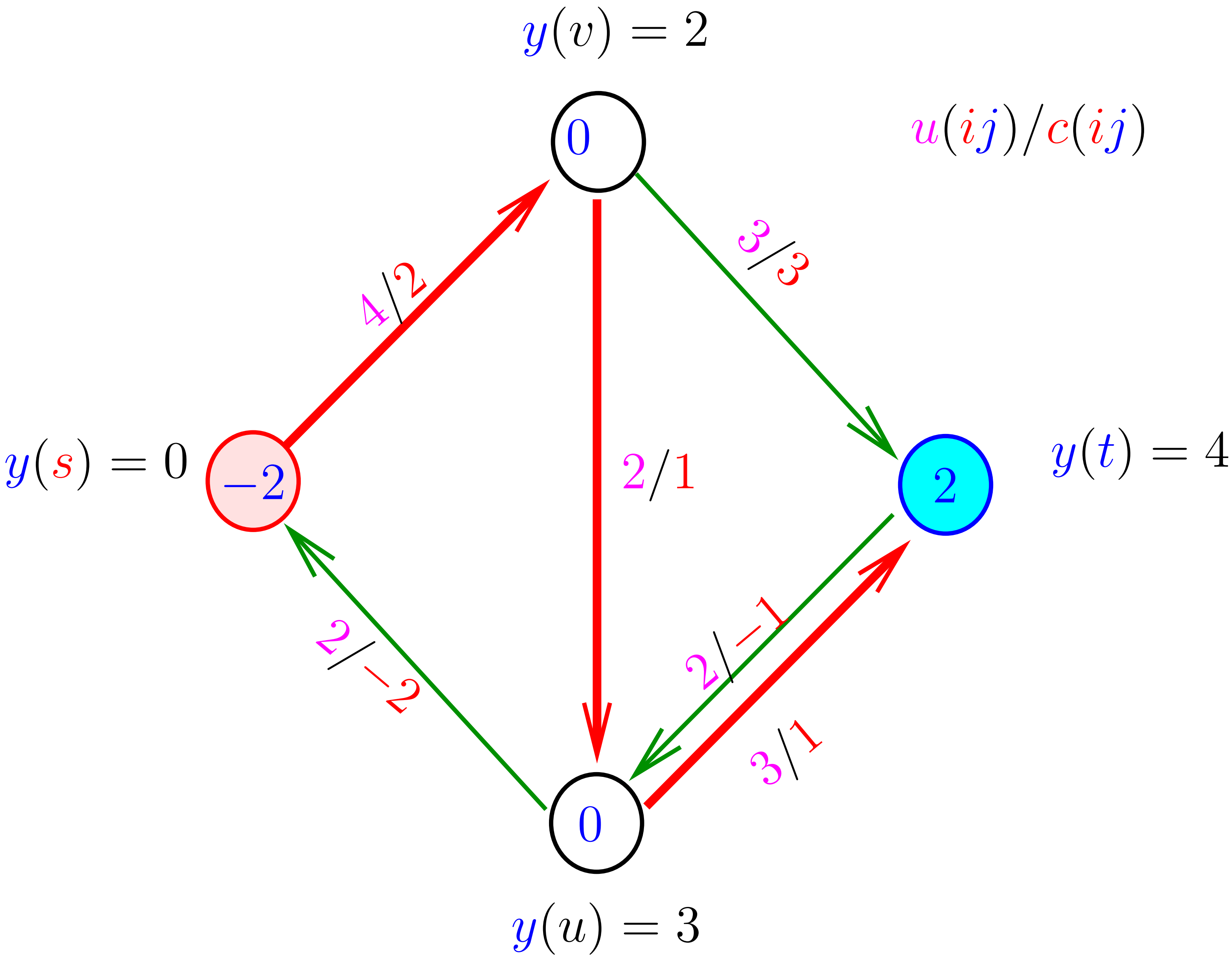


Custo $= 0 \times 2 + 2 \times 2 + 0 \times 1 + 0 \times 3 + 2 \times 1 = 6$

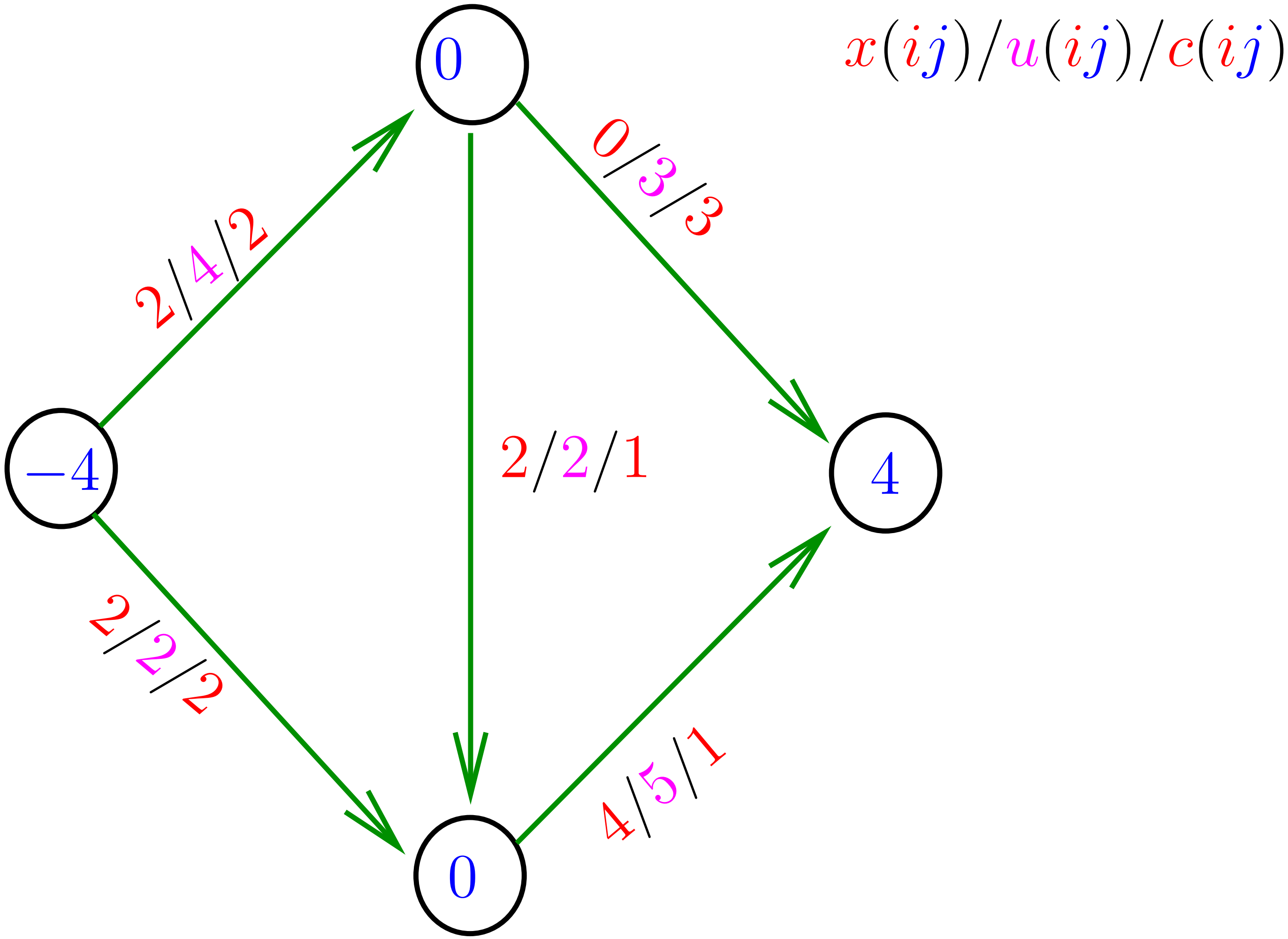
Jewell: rede residual



Jewell: caminho de custo mínimo



Jewell: rede



Custo $= 2 \times 2 + 2 \times 2 + 2 \times 1 + 0 \times 3 + 4 \times 1 = 14$

Algoritmos

- KLEIN: mantém um fluxo que respeita u e satisfaz b e em cada iteração procura ciclos negativos.
- JEWEEL: mantém um fluxo que respeita u e tem custo mínimo (dentre as fluxos que respeita u e satisfazem e) e em cada iteração procura um caminho de incremento de custo mínimo.

Algoritmo	consumo de tempo
KLEIN	$\mathrm{O}(nm^2UC)$
JEWEEL	$\mathrm{O}(n^3B)$

AULA 20

Algoritmo Cost Scaling

PF 24.1, 24.2, 24.3

Folgas complementares

Seja x um fluxo.

Seja y um potencial.

Diremos que as folgas de x e y são complementares se, para cada arco ij,

$$y(j) - y(i) < c(ij) \Rightarrow x(ij) = 0 \quad \text{e}$$
$$y(j) - y(i) > c(ij) \Rightarrow x(ij) = u(ij) \ .$$

Fato. Se x é um fluxo viável e suas folgas são complementares às de algum potencial y então x é ótimo.

Folgas ϵ-complementares (1)

Seja x um fluxo.

Seja y um potencial.

Diremos que as folgas de x e y são ϵ-complementares se, para cada arco ij em $A_{\check{x}}$,

$$y(j) - y(i) \leq c(ij) + \epsilon \; .$$

Folgas ϵ-complementares (2)

Seja x um fluxo.

Seja y um potencial.

Diremos que as folgas de x e y são ϵ-complementares se, para cada arco ij em A,

$$y(j) - y(i) < c(ij) - \epsilon \Rightarrow x(ij) = 0 \quad \text{e}$$

$$y(j) - y(i) > c(ij) + \epsilon \Rightarrow x(ij) = u(ij)\ .$$

Folgas ϵ-complementares e ciclos (1)

Fato. Seja x um fluxo e y um potencial. Se as folgas de x e y são ϵ-complementares, então

$$c(O) \geq -\epsilon n$$

para todo ciclo O na rede residual $(N, A_{\check{x}})$

Demonstração (esboço)

$$\begin{aligned} c(O) &= \sum(c(ij) : ij \text{ é arco de } O) \\ &= \sum(c(ij) - (y(j) - y(i)) : ij \text{ é arco de } O) \\ &\geq -\epsilon|O| \\ &\geq -\epsilon n \end{aligned}$$

Folgas ϵ-complementares e ciclos (2)

Consequência. Seja x um fluxo viável, y um potencial e $\epsilon < 1/n$. Se as folgas de x e y são ϵ-complementares então, x é um fluxo viável de custo mínimo.

Demonstração (esboço) Para todo ciclo O na rede residual $(N, A_{\breve{x}})$ vale que

$$\begin{aligned} c(O) &\geq -\epsilon n \\ &> -(1/n)n \\ &= -1\ . \end{aligned}$$

Como $c(ij)$ é um número inteiro para cada ij,

$$c(O) \geq 0\ .$$

para todo ciclo O na rede residual.

Algoritmo Cost-Scaling

COST-SCALING $(N, A, u, b, c))$

$\langle x, T\rangle \leftarrow$ FLUXO-VIÁVEL (N, A, u, b)
se x não está definido
    então devolva T

$y \leftarrow 0$
$\epsilon \leftarrow C$
enquanto $\epsilon > 1/n$ **faça**
    se $y(j) - y(i) < c(ij)$
        então $x(ij) \leftarrow 0$
        senão se $y(j) - y(i) > c(ij)$
                então $x(ij) \leftarrow u(ij)$
    PREFLOW-PUSH$_\epsilon(N, A, u, b, c, x, y)$
    $\epsilon \leftarrow \epsilon/2$

devolva x

Invariantes (1)

Na linha 6, antes do "**enquanto** $\epsilon > 1/n$ ...", vale que:

(i1) x respeita u;

(i2) x satisfaz b;

(i3) $y(j) - y(i) < c(ij) - \epsilon \Rightarrow x(ij) = 0$ para cada arco ij; e

(i4) $y(j) - y(i) > c(ij) + \epsilon \Rightarrow x(ij) = u(ij)$ para cada arco ij.

(i3)+(i4) é o mesmo que

x e y têm folgas ϵ-complementares.

Invariantes (2)

Na linha 11, antes da execução do PREFLOW-PUSH$_\epsilon$, vale que:

(i5) x respeita u; e

(i6) x e y tem folgas 0-complementares.

Veremos que na linha 12, antes da execução do "$\epsilon \leftarrow \epsilon/2$", vale que:

(i7) x respeita u;

(i8) x sastisfaz b; e

(i9) x e y tem folgas $\epsilon/2$-complementares.

Número de iterações

O número de iterações da linhas 6–12 é $< 1 + \lfloor \lg(nC) \rfloor$.

Se k é o número de iterações das linhas 6–12, então

$$C/2^{k-1} \geq 1/n > C/2^{k}$$

e portanto

$$1 + \lg(nC) \geq k > \lg(nC) \ .$$

Consumo de tempo

O consumo do algoritmo PREFLOW-PUSH$_\epsilon$ é $\mathrm{O}(n^2 m)$.

$$\boxed{\text{O consumo de tempo do algoritmo COST-SCALING é } \mathrm{O}(n^2 m \lg(nC)).}$$

Este consumo de tempo é polinomial.

Algoritmo Preflow-Push$_\epsilon$

PREFLOW-PUSH$_\epsilon$ (N, A, u, b, u, x, y)

```
PRÉ-PROCESSAMENTO(x)
enquanto e(i) > b(i) para algum i faça
    A(x̌) ← {ij ∈ A : x̌(ij) < u(ij)}
    se algum ij em A_x̌(i) é justo
        então PUSH(ij)
        senão RELABEL_ε(i)
x ← FLUXO(x̌)
devolva x
```

Aqui, diremos que um arco ij é justo se

$$y(j) - y(i) > c(ij) .$$

Pré-processamento

PRÉ-PROCESSAMENTO (x)

$\check{x} \leftarrow 0$

$e \leftarrow 0$

para cada ij em A **faça**

$\quad \check{x}(ij) \leftarrow x(ij)$

$\quad \check{x}(ij) \leftarrow -x(ij)$

$\quad e(j) \leftarrow e(j) + x(ij)$

$\quad e(i) \leftarrow e(i) - x(ij)$

Consumo de tempo: $\mathrm{O}(n+m)$.

Push

PUSH (ij)

1 $\delta \leftarrow \min\{e(i), u(ij) - \check{x}(ij))\}$

2 $\check{x}(ij) \leftarrow \check{x}(ij) + \delta$

3 $\check{x}(ji) \leftarrow \check{x}(ji) - \delta$

4 $e(i) \leftarrow e(i) - \delta$

5 $e(j) \leftarrow e(j) + \delta$

Consumo de tempo: $\mathrm{O}(1)$

Relabel$_\epsilon$

RELABEL$_\epsilon$ (i)

1 $\quad y(i) \leftarrow y(i) - \epsilon/2 \qquad \triangleright$ $y(i)$ decresce

Consumo de tempo: $\mathrm{O}(1)$

Invariantes

Na linha 1, antes do "**enquanto** $e(i) > 0 \ldots$" vale que

(i10) $x = \text{FLUXO}(\check{x})$ é um pré-fluxo;

(i11) $e(i) = \check{x}(\bar{i}, i) - \check{x}(i, \bar{i})$ para cada i em N;

(i12) x respeita u;

(i13) y é um potencial em $(N, A_{\check{x}})$ tal que

$$y(j) - y(i) \leq c(ij) + \epsilon/2 \ ;$$

(i14) se $e(i) > b(i)$ então existe um nó s com $e(s) < b(s)$ e um caminho de i a s em $(N, A_{\check{x}})$;

(i15) se

$$B = \{ij \in A_{\check{x}} : y(j) - y(i) > c(ij)\},$$

então a rede (N, B) é acíclica.

Demonstração de (i13)

No início da 1a. iteração, para cada arco ij em $A_{\check{x}}$

$$y(j) - y(i) \leq c(ij) < c(ij) + \epsilon/2 \ .$$

Consideremos agora o efeito de uma iteração.

Caso 1. A iteração realizou PUSH(ij)

O arco ij é justo e portanto

$$y(j) - y(i) > c(ij) \ .$$

Logo, para o arco ji que após a iteração faz parte da rede residual, temos que

$$\begin{aligned} y(i) - y(j) &= -(y(j) - y(i)) \\ &< -c(ij) \\ &= c(ji) < c(ji) + \epsilon/2 \ . \end{aligned}$$

Demonstração de (i13)

Caso 2. A iteração realizou RELABEL$_\epsilon(i)$

Seja y' a função potencial após a iteração.

Para cada arco da forma ij temos que

$$y(j) - y(i) < c(ij) \ .$$

Assim, para todo arco da forma ij vale que

$$\begin{aligned} y'(j) - y'(i) &= y(j) - (y(i) - \epsilon/2) \\ &= y(j) - y(i) + \epsilon/2 \\ &< c(ij) + \epsilon/2 \ . \end{aligned}$$

Demonstração de (i13)

Caso 2. A iteração realizou RELABEL$_\epsilon(i)$

Seja y' a função potencial após a iteração.

Para cada arco da forma ji temos que

$$y(i) - y(j) \leq c(ji) + \epsilon/2 \; .$$

Logo, para todo arco da forma ji vale que

$$\begin{aligned} y'(i) - y'(j) &= y(i) - \epsilon/2 - y(j) \\ &= y(i) - y(j) - \epsilon/2 \\ &\leq c(ji) \; . \end{aligned}$$

Demonstração de (i15)

No início da 1a. iteração, para cada arco $B = \emptyset$.
Consideremos agora o efeito de uma iteração.

Caso 1. A iteração realizou PUSH(ij)

O arco ij é justo e portanto

$$y(j) - y(i) > c(ij) \ .$$

Assim, para o arco ji que após a iteração faz parte da rede residual, temos que

$$\begin{aligned} y(i) - y(j) &= -(y(j) - y(i)) \\ &< -c(ij) \\ &= c(ji) \ . \end{aligned}$$

Logo, ji não é justo.

Demonstração de (i15)

Caso 2. A iteração realizou $\text{RELABEL}_\epsilon(i)$

Seja y' a função potencial após a iteração.

Para cada arco da forma ji temos que

$$y(i) - y(j) \le c(ji) + \epsilon/2 \ .$$

Logo, para todo arco da forma ji vale que

$$\begin{aligned} y'(i) - y'(j) &= y(i) - \epsilon/2 - y(j) \\ &= y(i) - y(j) - \epsilon/2 \\ &\le c(ji) \ . \end{aligned}$$

Logo, nenhum arco da forma ji é justo e portanto não há ciclo que contém i.

Número de iterações

Fato 0. Para cada nó i, RELABEL(i) é executado $< 3n$ vezes.

Fato 1. O algoritmo RELABEL é executado $< 3n^2$ vezes.

PUSH (ij) é saturante se $\check{x}(ij) = u(ij)$ após a execução.

Fato 2. Um PUSH saturante é executado $< (3/2)nm$ vezes.

Demonstração (rascunho): Entre duas execuções de um PUSH saturante de um arco ij o valor de $y(j)$ diminui de pelo menos ϵ.

Fato 3. Um PUSH não-saturante é executado $< 3n^2(m+2)$ vezes.

Número de iterações

Fato 3. Um PUSH não-saturante é executado $< 3n^2(m+2)$.

Demonstração (rascunho): $B := \{ij : y(j) - y(i) > c(ij)\}$.
Invariante (i15) $\Rightarrow (N, B)$ é rede acíclica.

Seja $h(i) =$ número de nós acessíveis a partir de i em (N, B) .

Considere o valor de

$$\Phi := \sum (h(i) : i \in N \text{ e } e(i) > b(i)) \; .$$

- É evidente que $\Phi \geq 0$ no início de cada iteração.
- No início da primeira iteração $\Phi = n$.

Número de iterações

Sejam Φ_1 e Φ_2 o valor de Φ no início de duas iterações consecutivas.

Se na iteração é executado um:

- RELABEL $\Rightarrow \Phi_2 = \Phi_1 + n$.
- PUSH saturante $\Rightarrow \Phi_2 < \Phi_1 + n$
- PUSH não-saturante $\Rightarrow \Phi_2 \leq \Phi_1 - 1$

Como no início da última iteração $\Phi = 0$, então o número de execuções de um PUSH não-saturante é

$$< n + 3n^2 + (3/2)n^2m < 3n^2(m+2) \ .$$

Consumo de tempo

Algoritmo	número máximo de execuções	consumo total de tempo
RELABEL	$< 3n^2$	$\mathrm{O}(n^2)$
PUSH saturante	$< (3/2)nm$	$\mathrm{O}(nm)$
PUSH não-saturante	$< 3n^2(m+2)$	$\mathrm{O}(n^2m)$

O consumo de tempo do algoritmo PREFLOW-PUSH$_\epsilon$ é $\mathrm{O}(n^2m)$.