Formulário MDIO 2020/2021

1 PL: Definição matricial

$$\begin{array}{|c|c|}\hline B^{-1} & \tilde{0} \\ \hline c_B B^{-1} & 1 \\ \hline \end{array} * \begin{array}{|c|c|c|}\hline A & I & b \\ \hline -c & \tilde{0} & 0 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline B^{-1}A & B^{-1} & B^{-1}b \\ \hline c_B B^{-1}A - c & c_B B^{-1} & c_B B^{-1} b \\ \hline \end{array}$$

2 PL: Simplex Dual (prob. min)

- Vértice dual admissível inicial (todos os coeficientes da linha da função objectivo são não-positivos, *i.e.*, $c \le \tilde{0}$) $^{(*)}$
- Repetir
  - Selecção da linha pivô:
    - Coeficiente mais negativo do lado direito
    - (em caso de empate, escolha arbitrária)
    - Se não existir coef.<0, solução óptima.
  - Selecção da coluna pivô:
    - Menor valor absoluto da razão (f.objectivo/linha pivô) **negativa** (coef.linha<0) $^{(**)}$
    - Se não existir coef.linha <0, problema é impossível.
  - Fazer eliminação de Gauss
- Enquanto (solução não for óptima)

- nota: o elemento pivô tem sempre valor **negativo**.

(*) O contrário para problemas de maximização.

(**) Valor absoluto para se aplicar também em problemas de maximização.

2.1 Problema impossível

Um problema (primal) é impossível se existir uma linha com um coeficiente negativo do lado direito e com todos os coeficientes das variáveis não-básicas não-negativos (≥ 0).

3 Relax4

```
n
m
org dst custo cap (m vezes)
vert (n vezes)
```

- `n`: número de vértices
- `m`: número de arcos do grafo
- `org`: vértice de origem do arco
- `dst`: vértice de destino
- `custo`: custo de transporte
- `cap`: capacidade do arco
- `vert`: oferta/procura no vértice, + e − respetivamente

4 Transportes: Introdução

4.1 Caracterização das soluções básicas

A uma base podemos associar uma árvore (grafo com vértices não orientados) que suporta todos os vértices.

4.1.1 Propriedades da árvore de suporte de um grafo G = (V, A)

- é um grafo ligado (existe um caminho entre cada par de vértices)
- sem ciclos
- com $|A| = |V| - 1$ (nº de arcos = nº de vértices - 1)

4.2 Método dos multiplicadores

1. Fixar o valor de qualquer multiplicador em 0
2. Arcos básicos: $c_{ij} = u_i - u_j$
3. Arcos não-básicos: $\delta_{ij} = c_{ij} - (u_i - u_j)$

5 Transportes: Grafos Bipartidos

Um grafo $G = (V, A)$ é bipartido se o conjunto de vértices V puder ser dividido em dois conjuntos disjuntos, V_1 e V_2 (i.e., $V_1 \cup V_2 = V, V_1 \cap V_2 = \emptyset$), de tal modo que todos os arcos $(i, j) \in A$ tenham origem num vértice $i \in V_1$ e destino num vértice $j \in V_2$.

5.1 Solução inicial

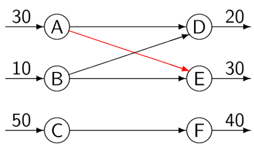
5.1.1 Método do canto NW / custo mínimo

1. Colocar a maior quantidade possível na casa mais a NW / com custo mínimo $\Longrightarrow$
   - ou a procura de um destino (coluna) é totalmente satisfeita,
   - ou a oferta de uma origem (linha) é totalmente usada,
   - ou ambas.
2. Cortar a linha ou a coluna (ou ambas)
3. Repetir se ainda houver uma casa

5.1.2 Seleção da variável básica com valor 0 (quando faltar uma var. básica)

- Nem todas as variáveis podem ser escolhidas!
- No seguinte exemplo, escolher a variável x_{AE} dá origem a um grafo que não é uma árvore.

	D	E	F
A	*	0	
B	*	*	
C			*



- Os arcos associados às variáveis formam um ciclo (*i.e.*, as colunas do modelo de PL são linearmente dependentes, e portanto não formam uma base)

6 Transportes: Redes c/ capacidades

6.1 Caracterização das soluções básicas

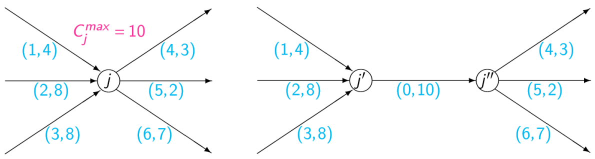
Iguais às referidas no 2.2, mas agora as variáveis no limite superior são também consideradas como não-básicas, para além das iguais a 0.

Uma variável não-básica é atrativa quando:

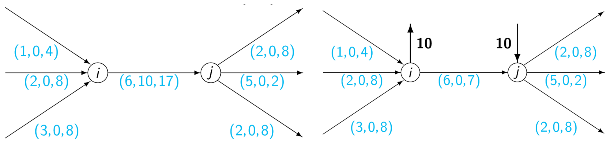
- $x_{ij} = 0$ (variável aumenta de valor) e $\delta_{ij} < 0$
- $x_{ij} = u_{ij}$ (variável decrementa de valor) e $\delta_{ij} > 0$

6.2 Transformações

6.2.1 Capacidade num vértice



6.2.2 Limite inferior num arco



7 Programação Inteira: Modelos

7.1 Expressões lógicas

Expressão lógica	Restrição binária
$a \Rightarrow b$	$a \leq b$
$\bar{b} \Rightarrow \bar{a}$	$(1-b) \leq (1-a)$
$\bar{b} \Rightarrow \bar{a}$	$a \leq b$
$a \Rightarrow \bar{b}$	$a+b \leq 1$
$b \Rightarrow \bar{a}$	$a+b \leq 1$
$a \dot{\vee} b$ (ou exclusivo)	$a+b=1$
seleccionar *exactamente* uma das opções	$a+b+\ldots+z=1$
seleccionar, *no máximo.* uma das opções	$a+b+\ldots+z \leq 1$
$a.b \Rightarrow c$	$a+b-1 \leq c$

7.2 Restrições ativas e redundantes

- A variável binária y_1 pode ser usada para tornar uma restrição *activa* ou *redundante* (M deve ter um valor adequado):

$$A^1 x \leq b^1 + M(1-y_1)$$

8 PI: Planos de corte

8.1 Algoritmo de planos de corte

1. Otimizar *relaxação linear*
2. Enquanto a solução não for inteira:
   - identificar um plano de corte
   - adicionar plano de corte ao conjunto de restrições
   - reotimizar (usando o método simplex dual)

8.2 Plano de corte de Chvátal-Gomory

1. Pegar na restrição da var. básica com a maior mantissa
2. Pegar na parte fracionária da restrição e meter $\geq$

8.2.1 Exemplo

1.
$$x_1 + \frac{1}{4}s_1 + \frac{3}{2}s_2 = \frac{5}{2}$$

2.
$$\frac{1}{4}s_1 + \frac{1}{2}s_2 \geq \frac{1}{2}$$

9 PI: Partição e avaliação

9.1 Partição do domínio

Dado um pai com uma solução fracionária:

1. selecionar variável x_j fracionária
2. criar 2 nós filhos, $x_j \leq \lfloor x_j \rfloor$ e $x_j \geq \lceil x_j \rceil$

9.2 Solução incumbente

É a melhor solução inteira encontrada até um dado passo da pesquisa (x_{SI}) com valor de função objetivo z_{SI}.

9.3 Avaliação do nó (prob. maximização)

9.3.1 [Início] Determinar sol. ótima PL ($x_{\mathrm{PL}} \rightarrow z_{\mathrm{PL}}$)

- se for inteira, é a melhor sol. inteira no domínio do nó
- se não, pode haver na subárvore uma sol. inteira $\leq z_{\mathrm{PL}}$

9.3.2 [Opção 1] Abandonar o nó (podar a subárvore) se:

- o problema for impossível (domínio vazio)
- a solução x_{PL} for inteira (atualizar incumbente se $z_{\mathrm{PL}} > z_{\mathrm{SI}}$)
- a solução x_{PL} for fracionária e não puder haver na subárvore uma solução inteira melhor do que a solução incumbente $z_{\mathrm{PL}} \leq z_{\mathrm{SI}}$

9.3.3 [Opção 2] Fazer partição (explorar a subárvore):

- se a solução x_{PL} for fracionária e ainda puder haver na subárvore uma solução inteira melhor do que a solução incumbente $z_{\mathrm{PL}} > z_{\mathrm{SI}}$

9.4 Limites para o valor do ótimo

$z_I^* \in [\mathrm{LI}, \mathrm{LS}]$, $z_I^* \rightarrow$ solução ótima inteira

9.5 Limite superior

Num problema de maximização o LS é apenas um valor de referência, não está associado a nenhuma solução inteira admissível. O valor do ótimo da relaxação linear (z_{RL}) é um limite superior para o valor do ótimo inteiro z_I^*:

$$z_I^* \leq z_{\mathrm{RL}}$$

Para problemas de minimização é o oposto (lim. inferior).

9.6 Limite inferior

O valor de qualquer solução inteira admissível é um limite inferior para z_I^*.

Para problemas de minimização é o oposto (lim. superior).