PROGRAMAÇÃO INTEIRA

Marcus Ritt

INF 05010 – Otimização combinatória — *<2020-05-11 seg>*

Outline

1. Resumo

2. Programação inteira: introdução

3. Programação inteira: formulação

$t \in \mathbb{R}$ $\quad \hat{c} = c + t\Delta c \implies$

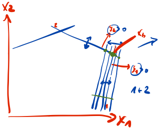
- Seleciona Δb, seja $\hat{b} = b + t\Delta b$ $\quad t=0$
- Solução dual y_N $= -\bar{c}$ e matriz $\bar{A}$ não variam (objetivo fixo!)
- Nova solução primal

$x_B = B^{-1}\hat{b}$ $\quad y_N \geq 0$ ✓ dualmente viável

$B^{-1}\hat{b} = B^{-1}(b + t\Delta b) = B^{-1}b + t \cdot B^{-1}\Delta b$

$$\hat{x}_B = B^{-1}\hat{b} = x_B + t\Delta x_B$$

com $\Delta x_B = B^{-1}\Delta b$



- Novo valor da função objetivo

$z = c_B^t B^{-1} b = c_B^t x_B$

$$\hat{z} = c_B^t \hat{x}_B = \bar{z} + t\Delta z$$

com $\Delta z = c_B^t B^{-1}\Delta b.$ $= c_B^t \Delta x_B.$

$\hat{z} = c_B^t \hat{x}_B = c_B^t(x_B + t\Delta x_B) = c_B^t x_B + t \cdot c_B^t \Delta x_B = \bar{z} + t \cdot \Delta z$

$t = 0$

Condição para otimalidade:

- $x_B \geq 0$: $\iff x_B + t\Delta x_B \geq 0 \to$ condições para t
- $y_N \geq 0$: sempre satisfeito

Para uma solução não mais ótima:

- Reotimizar com método Simplex dual

$B = \underbrace{(\,|\,|\,|\,|\,)}_{\text{va. básicas}} \Rightarrow B^{-1}$

$$B = \begin{pmatrix} 7 & 0 & 10 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$
$\quad p \quad w_3 \quad c$

$$B^{-1} = \begin{pmatrix} 0 & 1 & 0 \\ -1/10 & 7/10 & 1 \\ 1/10 & -7/10 & 0 \end{pmatrix}$$

$BB^{-1} = I$

$B^{-1}B = I$

$AB \neq BA$

A matriz B^{-1} é formada pelas colunas do dicionário final (na forma $\bar{A}x_N + Ix_B = \bar{b}$) que correspondem com as variáveis de folga.

max $25p + 30c$
s.a $7p + 10c \leq 56k$
$p \leq 6k$
$c \leq 4k$
$p, c \geq 0$

$$\begin{pmatrix} 7 & 10 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p \\ c \\ w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

I ; A

$\Rightarrow$

$z = 192k - 3w_1 - 4w_2$
$p = 6k - w_2$
$w_3 = 2.6k + 1/10\, w_1 - 7/10\, w_2$
$c = 1.4k - 1/10\, w_1 + 7/10\, w_2$

$B = \{p, w_3, c\}$

$\Rightarrow$

p	c	w_1	w_2	w_3	
p			$+w_2$		$= 6k$
		$-1/10\, w_1$	$+7/10\, w_2$	$+w_3$	$= 2.6k$
	c	$+1/10\, w_1$	$-7/10\, w_2$		$= 1.4k$

$Ax = b$

$B^{-1}A = B^{-1}[A' \; I]$
$= [B^{-1}A' \;\; B^{-1}]$

(Branco)

(Branco)

PL ⇒ Programa inteiro (puro) → todas var. inteiras.

$$
\begin{aligned}
\textbf{maximiza} \quad & c^t x \\
\textbf{sujeito a} \quad & Ax \leq b, \\
& \cancel{x \geq 0}.
\end{aligned}
$$

$\boxed{\text{var. inteiras}}$

$x \in \mathbb{Z}^n_+$.

$x \in \mathbb{R}^n_+$

PL ∈ P (∃ alg. polinom.)

PI ∈ NP-difícil

Otim. combinatória

$x_1 \in \mathbb{Z}$

$\underbrace{0 \leq x_1 \leq 1}$ Restr. lin.

$x_1 \in \{0, 1\} = \mathbb{B}$

~ booleano

Fluxos, transporte
nível de produção

Decisão

PIM

$$
\begin{aligned}
\textbf{maximiza} \quad & c^t x + h^t y \\
\textbf{sujeito a} \quad & Ax + Gy \leq b, \\
& x \in \mathbb{R}^n_+ \cancel{\geq 0},\ y \in \mathbb{Z}^m_+ \cancel{\geq 0};
\end{aligned}
$$

$$\begin{aligned} \textbf{maximiza} \quad & x_1 + x_2 \\ \textbf{sujeito a} \quad & 2x_1 + 7x_2 \le 49, \\ & 5x_1 + 3x_2 \le 50. \end{aligned}$$

$x_1, x_2 \in \mathbb{R}_+.$

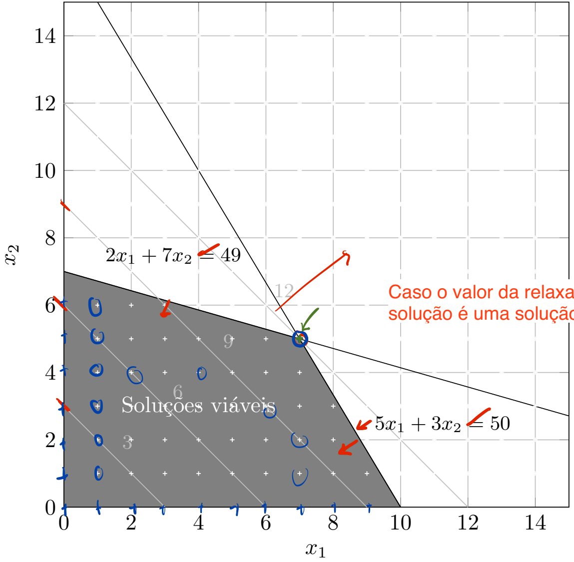
$\Rightarrow \quad x_1, x_2 \in \mathbb{Z}_+.$

$\max\{c^t x \mid Ax \le b, x \in \mathbb{Z}_+\} \le \max\{c^t x \mid Ax \le b, x \in \mathbb{R}_+\}$

11 → 11.2 Relaxação linear

O valor da relaxação linear é um limite superior para o valor do programa inteiro correspondente.

Consequências

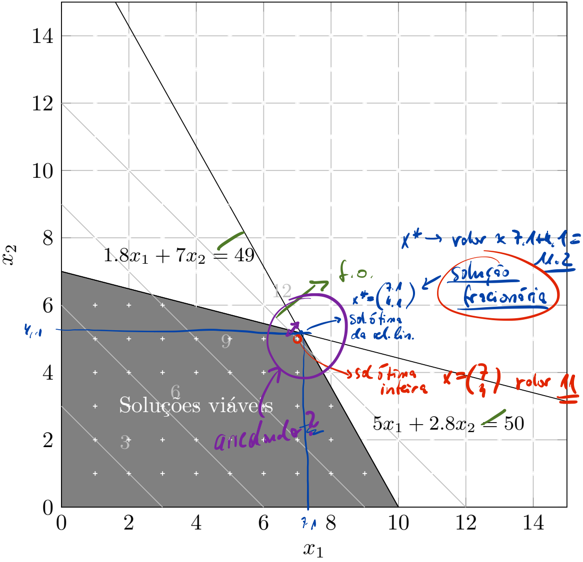


Caso o valor da relaxação é inteiro, a solução é uma solução ótima inteira.

Consequências

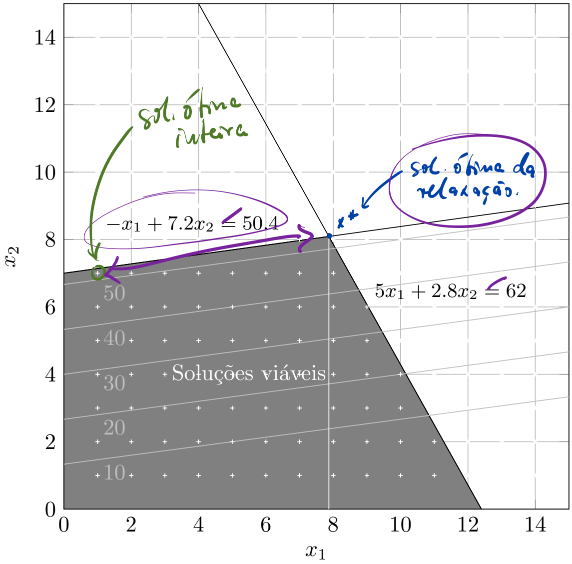
$$
\begin{aligned}
\textbf{maximiza} \quad & x_1 + x_2 \\
\textbf{sujeito a} \quad & \mathbf{1.8}x_1 + 7x_2 \leq 49, \\
& 5x_1 + \mathbf{2.8}x_2 \leq 50.
\end{aligned}
$$

Consequências



$$
\begin{array}{rl}
\textbf{maximiza} & -x_1 + 7.5x_2 \\
\textbf{sujeito a} & -x_1 + 7.2x_2 \leq 50.4, \\
& 5x_1 + 2.8x_2 \leq 62.
\end{array}
$$

Consequências



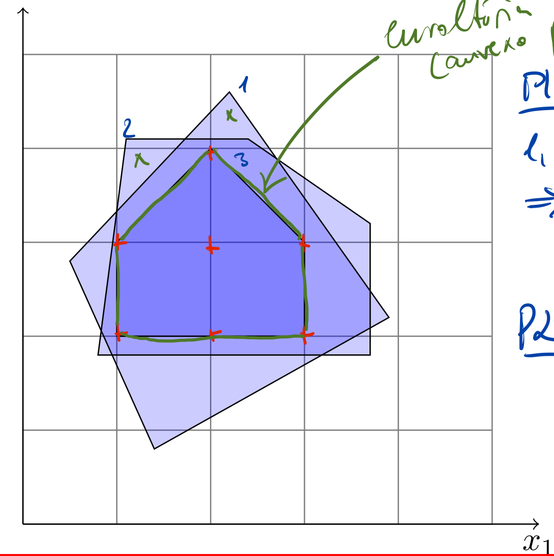
Técnicas de solução

1. Mostrar que a relaxação linear é inteira
2. Algoritmo de plano de cortes
3. Algoritmos branch-and-bound

solvers

- Criar (boas) formulações é uma arte.
- Algumas diretivas básicas:
  - escolha das variáveis **de decisão**.
  - escolha do objetivo.
  - ajuste das restrições.

Formulações



Envoltória convexa pontos inteiros

P1

1, 2, 3 equiv.
⇒ mesmo conj. soluções

P2 1, 2, 3 dif.

Mochila n itens
pesos $p_1, p_2, \dots, p_n$
mochila com peso máximo P
valores $v_1, v_2, \dots, v_n$

Problema: ∈ NP-c. seleciona um subconjunto dos itens $S \subseteq [n]$, tal que
$\boxed{\sum_{i \in S} p_i \leq P}$ e tal que o valor total
$\sum_{i \in S} v_i$ é máximo.

Programa inteiro? PI

Variáveis de decisão?

$$x_i = \begin{cases} 1, & \text{caso item } i \text{ sel.,} \\ 0, & \text{caso contrário.} \end{cases}$$

$$\max. \sum_{i \in [n]} v_i x_i$$
$$\text{s.a} \quad \underbrace{\sum_{i \in [n]} p_i x_i}_{\text{peso total}} \leq P$$

$n = 4$

i	1	2	3	4	
p_i	3	1	4	1	≤ 5
v_i	2	7	1	4	13 ✓
	$x_1 = 1$	$x_2 = 1$	$x_3 = 0$	$x_4 = 1$	⟸

$P = 5$

Modelo PI

Dados
Var. decisão

$$\max \sum_{i \in [n]} v_i x_i$$

$$\text{s.a} \quad \sum_{i \in [n]} p_i x_i \leq P$$

$$x_i \in \{0, 1\}, \quad \forall i \in [n].$$

Exemplos

Não posso selecionar itens 2 e 5 no mesmo tempo:

Restrição? $x_2 + x_5 \leq 1$

x_2	x_5	
0	0	✓
1	0	✓
0	1	✓
1	1	✗

Preciso levar pelo menos um dos itens 3, 7, ou 9:

Restrição? $x_3 + x_7 + x_9 \geq 1$

x_3	x_7	x_9	
0	0	0	✗
0	0	1	✓
0	1	0	✓
0	1	1	✓
1	0	0	✓
1	0	1	✓
1	1	0	✓
1	1	1	✓

Preciso levar pelo menos 3 dos itens 1,2,3,4,5?

Restrição? $x_1 + x_2 + x_3 + x_4 + x_5 \geq 3$

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos

Exemplos