

CI1238 - Otimização

Aula 04 - Programação Linear: Modelagem

Professor Murilo V. G. da Silva

Departamento de Informática
Universidade Federal do Paraná

2026 / Primeiro Semestre

Programação Linear

Notação: Dados vetores $\mathbf{a}, \mathbf{b}$ de dimensão n , escrevemos $\mathbf{a} < \mathbf{b}$ se $a_i < b_i, i = 1, \dots, n$.

PROGRAMAÇÃO LINEAR (PL):

► **Instância:** Uma tripla $(A, \mathbf{b}, \mathbf{c})$ onde A é uma matriz $m \times n$ sobre $\mathbb{R}$, $\mathbf{b}$ é um vetor sobre $\mathbb{R}^m$ e $\mathbf{c}$ é um vetor sobre $\mathbb{R}^n$.

► **Solução:** Um vetor $\mathbf{x}^* \in \mathbb{R}^n$ satisfazendo

► $A\mathbf{x}^* \leq \mathbf{b},$

► $\mathbf{c}^T \mathbf{x}^* \geq \mathbf{c}^T \mathbf{x}, \forall \mathbf{x} \mid A\mathbf{x} \leq \mathbf{b}$

Obs: vetores são tratados como matrizes com apenas uma coluna; Às vezes também como tuplas.

Em alguns contextos será dito também da seguinte forma:

“obter $\mathbf{x}$ que maximize $\mathbf{c}^T \mathbf{x}$, sujeito a $A\mathbf{x} \leq \mathbf{b}$ ”

“maximizar $\mathbf{c}^T \mathbf{x}$, sujeito a $A\mathbf{x} \leq \mathbf{b}$ ”, entre outras formas...

Programação Linear: Modelagem e algoritmos

PL: Dado $(A, \mathbf{b}, \mathbf{c})$, encontrar $\mathbf{x}$ tal que $\mathbf{c}^T \mathbf{x}$ seja máximo, respeitando $A\mathbf{x} \leq \mathbf{b}$

- ▶ Nesta e nas próximas aulas: Vamos estudar o problema em si
 - ▶ Trata-se de um problema muito estudado e útil
- ▶ Depois: veremos o *Algoritmo Simplex* para solucionar o problema
 - ▶ Algoritmo Simplex: eficiente na prática
complexidade suavizada polinomial
(complexidade de pior caso exponencial)
 - ▶ Existem outros algoritmos para o problema
(e.g., Método Elipsoidal que é polinomial mesmo no pior caso)

Um problema envolvendo vetores reais:

Suponha que queiramos encontrar um par $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ tal que $x_1 + x_2$ seja **máximo** e que satisfaça as desigualdades:

- ▶ $x_1 \geq 0$ e $x_2 \geq 0$
- ▶ $x_2 - x_1 \leq 1$
- ▶ $x_1 + 6x_2 \leq 15$
- ▶ $4x_1 - x_2 \leq 10$

Podemos modelar isto como um problema de PL, pois podemos reescrever

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ & -x_2 & \leq 0 \\ -x_1 & +x_2 & \leq 1 \\ x_1 & +6x_2 & \leq 15 \\ 4x_1 & -x_2 & \leq 10 \end{array} \quad \text{i.e.,} \quad \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -1 & 1 \\ 1 & 6 \\ 4 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 15 \\ 10 \end{pmatrix} \text{ maximizando } (1 \ 1) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Encontrar $\mathbf{x}$ significa encontrar uma solução para a instância $(A, \mathbf{b}, \mathbf{c})$ de PL:

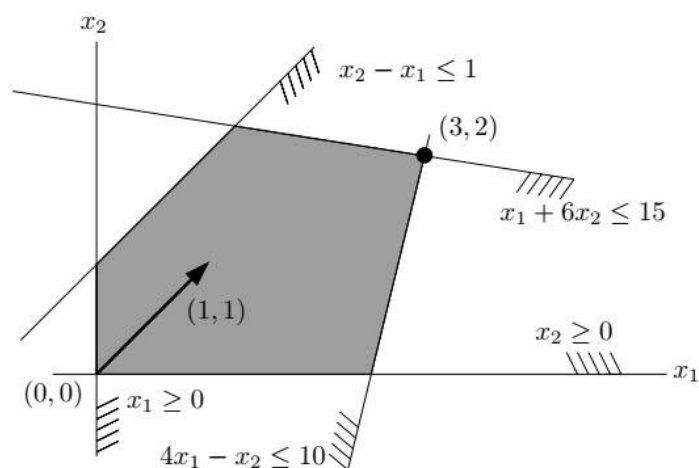
$$A = \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ -1 & 1 \\ 1 & 6 \\ 4 & -1 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 15 \\ 10 \end{pmatrix} \text{ e } \mathbf{c} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Programação Linear: Interpretação geométrica

Maximizar: $x_1 + x_2$

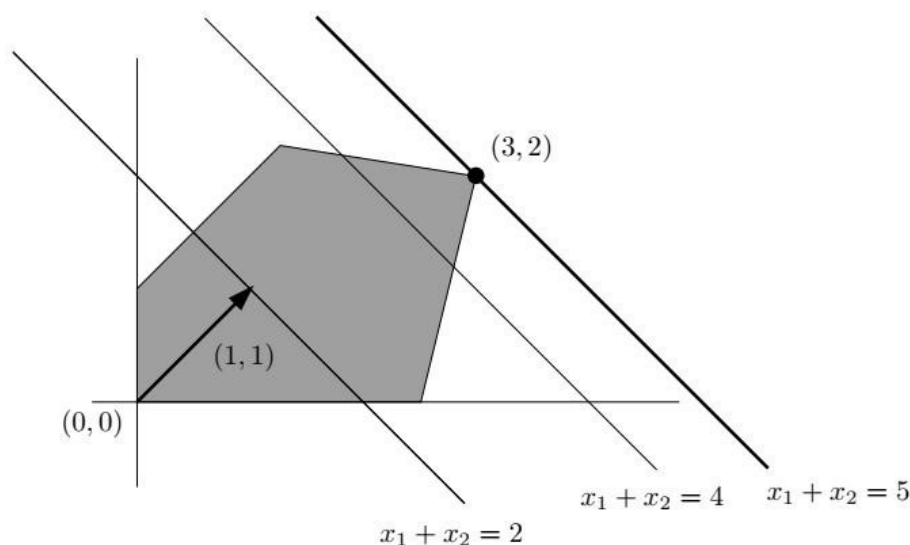
Sujeito a:

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ -x_2 & \leq & 0 \\ -x_1 + x_2 & \leq & 1 \\ x_1 + 6x_2 & \leq & 15 \\ 4x_1 - x_2 & \leq & 10 \end{array}$$



Programação Linear

Maximizando a função objetivo $f(x_1, x_2) = x_1 + x_2$



Observação: Considere a equação reta $x_1 + x_2 = s$ (no exemplo acima, $s = 2$)

Note que para qualquer s , $\mathbf{c} = (1,1)$ é o vetor normal a esta reta

Para pensar: Se tivéssemos n variáveis? Qual seria a interpretação geométrica?

Programação Linear

Relembrando:

PROGRAMAÇÃO LINEAR (PL):

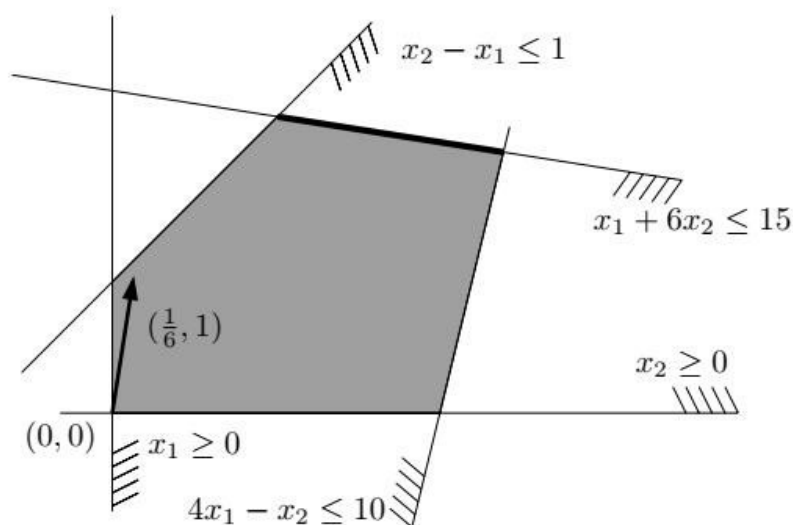
- ▶ **Instância:** Uma tripla $(A, \mathbf{b}, \mathbf{c})$ onde A é uma matriz $m \times n$ sobre $\mathbb{R}$, $\mathbf{b}$ é um vetor sobre $\mathbb{R}^m$ e $\mathbf{c}$ é um vetor sobre $\mathbb{R}^n$.
- ▶ **Solução:** Um vetor $\mathbf{x}^* \in \mathbb{R}^n$ satisfazendo
 - ▶ $A\mathbf{x}^* \leq \mathbf{b}$,
 - ▶ $\mathbf{c}^T \mathbf{x}^* \geq \mathbf{c}^T \mathbf{x}, \forall \mathbf{x} \mid A\mathbf{x} \leq \mathbf{b}$

A matriz A junto com o vetor $\mathbf{b}$ define as restrições;
O vetor $\mathbf{c}$ define função objetivo.

- ▶ E se tivéssemos, por exemplo, a restrição $2x_1 + x_2 \geq 5$?
 - ▶ Multiplique a equação por (-1)
- ▶ E se tivéssemos uma restrição do tipo $2x_1 + x_2 = 5$?
 - ▶ Crie duas restrições: $2x_1 + x_2 \leq 5$ e $2x_1 + x_2 \geq 5$. (depois multiplique esta por (-1))
- ▶ E se quisermos minimizar?
 - ▶ Multiplique a $\mathbf{c}$ por (-1)

Programação Linear

Considere a função objetivo dada por $(\frac{1}{6}, 1)$

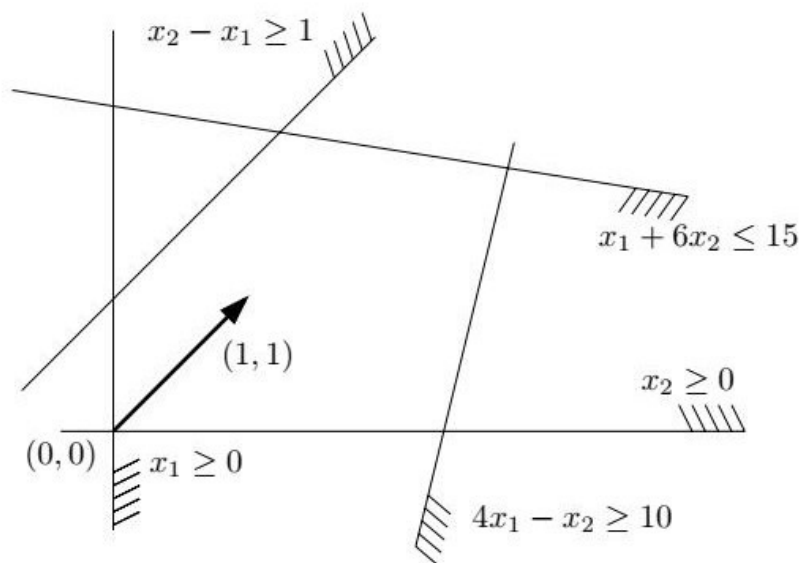


Quantas soluções temos?

- ▶ Infinitas (todos pontos da reta destacada)

Programação Linear

Invertendo restrições no exemplo original

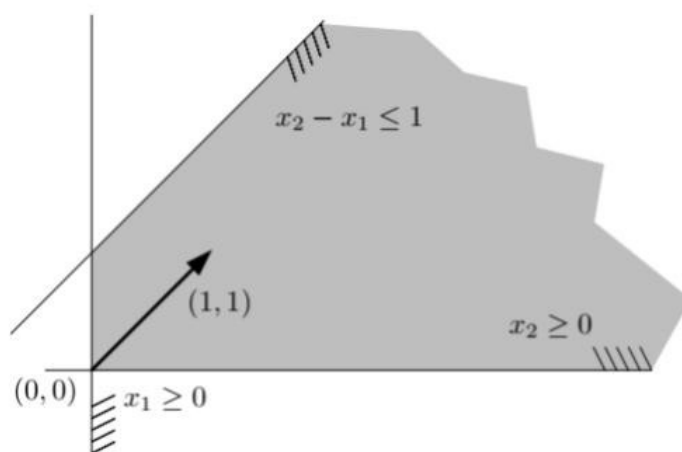


Quantas soluções temos?

- Nenhuma solução

Programação Linear

Removendo restrições no problema original



Quantas soluções temos?

- Nenhuma solução

Programação Linear: Mais exemplos

Minimizar custos com dieta equilibrada:

Food	Carrot, Raw	White Cabbage, Raw	Cucumber, Pickled	Required per dish
Vitamin A [mg/kg]	35	0.5	0.5	0.5 mg
Vitamin C [mg/kg]	60	300	10	15 mg
Dietary Fiber [g/kg]	30	20	10	4 g
price [€/kg]	0.75	0.5	0.15*	—

*Residual accounting price of the inventory, most likely unsaleable.

Queremos a quantidade x_1 de cenoura, x_2 de repolho e x_3 de pepino (em kg)

Minimizar: $0.75x_1 + 0.5x_2 + 0.15x_3$

Sujeito a:

$$x_1 \geq 0$$

$$x_2 \geq 0$$

$$x_3 \geq 0$$

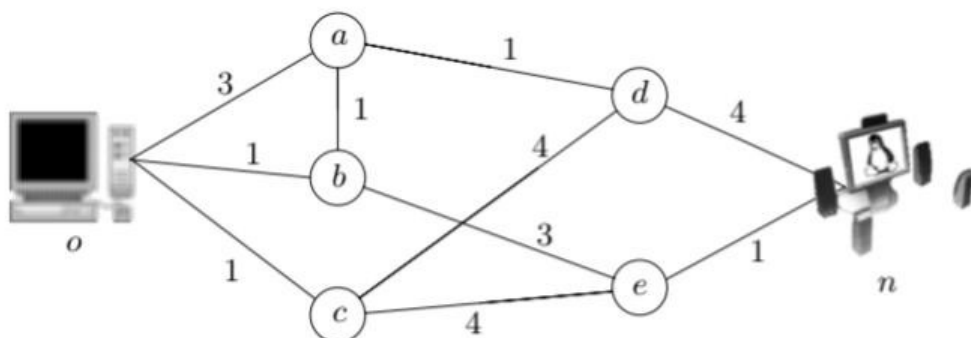
$$35x_1 + 0.5x_2 + 0.5x_3 \geq 0.5$$

$$60x_1 + 300x_2 + 10x_3 \geq 15$$

$$30x_1 + 20x_2 + 10x_3 \geq 4$$

Programação Linear: Mais exemplos

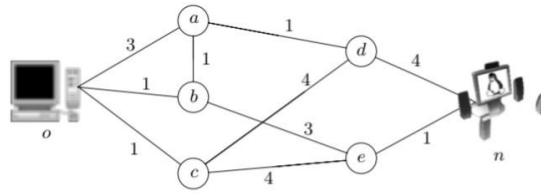
Maximizar fluxo em uma rede:



Variáveis: Para cada aresta $\{u, v\}$, uma variável x_{uv}
Maximizar o quê? Quais restrições?

Programação Linear: Mais exemplos

Maximizar fluxo em uma rede:



Variáveis: Para cada aresta $\{u, v\}$, uma variável x_{uv}

Maximizar $x_{oa} + x_{ob} + x_{oc}$

Sujeito a

$$\begin{aligned} -3 \leq x_{oa} \leq 3, & \quad -1 \leq x_{ob} \leq 1, & \quad -1 \leq x_{oc} \leq 1 \\ -1 \leq x_{ab} \leq 1, & \quad -1 \leq x_{ad} \leq 1, & \quad -3 \leq x_{be} \leq 3 \\ -4 \leq x_{cd} \leq 4, & \quad -4 \leq x_{ce} \leq 4, & \quad -4 \leq x_{dn} \leq 4 \\ -1 \leq x_{en} \leq 1 \end{aligned}$$

$$x_{oa} = x_{ab} + x_{ad}$$

$$x_{ob} + x_{ab} = x_{be}$$

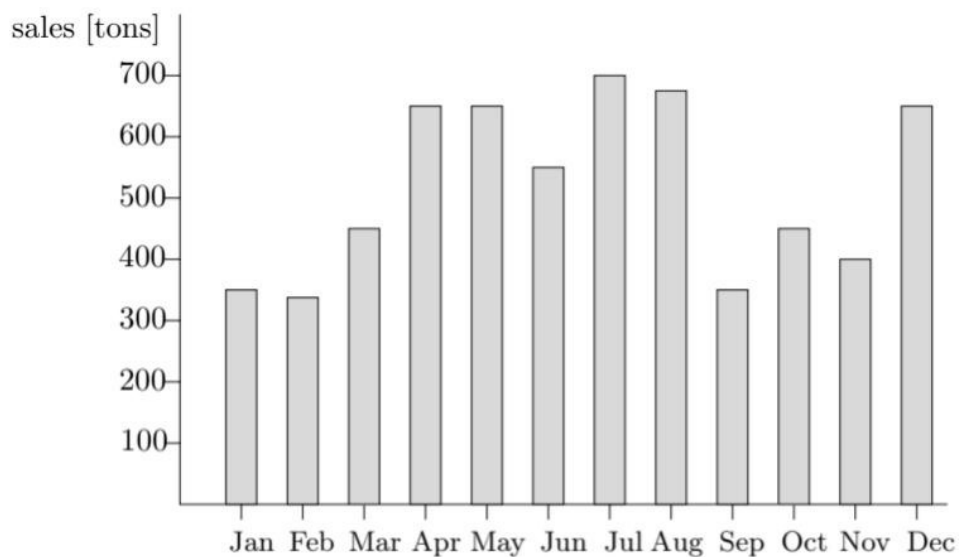
$$x_{oc} = x_{cd} + x_{ce}$$

$$x_{ad} + x_{cd} = x_{dn}$$

$$x_{be} + x_{ce} = x_{en}$$

Programação Linear: Mais exemplos

Minimizar custos em fábrica de sorvetes:



- ▶ Produção “uniforme” tem custos (estoque)
- ▶ Produção “variável” tem custos também

PL - Modelagem: Fábrica de sorvetes

Variáveis e desigualdades:

- ▶ Produção no mês i : x_i , $i = 1, \dots, 12$
- ▶ Demanda e estoque no mês i : d_i e s_i
 - ▶ Produzindo o suficiente:
 $x_i + s_i \geq d_i$
 - ▶ Estoque para mês seguinte: $x_i + s_{i-1} - d_i = s_i$. Portanto:
 $x_i + s_{i-1} - s_i = d_i$
 - ▶ Estoque no começo e fim do ano:
 $s_0 = 0$ e $s_{12} = 0$
- ▶ Custos para ter produção variável:
 - ▶ Mudar produção do mês $i - 1$ para mês i : \$50 (por tonelada)
 - ▶ Custo de armazenamento no mês i : \$20 (por tonelada)
 - ▶ Portanto, custo total:
 - ▶ $50 \sum_{i=1}^{12} |x_i - x_{i-1}| + 20 \sum_{i=1}^{12} s_i$ (função objetivo)
 - ▶ Para o caso $i = 0$, $x_0 = 0$.
 - ▶ Há algo errado com a função objetivo? **NÃO É LINEAR!**

PL - Modelagem: Fábrica de sorvetes

- ▶ Função objetivo: $50 \sum_{i=1}^{12} |x_i - x_{i-1}| + 20 \sum_{i=1}^{12} s_i$
- ▶ Observe: $|x_i - x_{i-1}| \geq 0$ ou ≤ 0 . i.e., mudança na produção:
 - ▶ aumenta em y_i , ou diminui em z_i (ambos valores ≥ 0)
 - ▶ na função objetivo, trocamos $|x_i - x_{i-1}|$ por $y_i + z_i$.e criamos uma nova equação: $x_i - x_{i-1} = y_i - z_i$

Com isso, o programa linear é:

- ▶ Minimizar: $50 \sum_{i=1}^{12} y_i + 50 \sum_{i=1}^{12} z_i + 20 \sum_{i=1}^{12} s_i$
- ▶ Sujeito a:

$$\begin{aligned}x_i + s_{i-1} - s_i &= d_i \text{ for } i = 1, 2, \dots, 12 \\x_i - x_{i-1} &= y_i - z_i \text{ for } i = 1, 2, \dots, 12 \\x_0 &= 0 \\s_0 &= 0 \\s_{12} &= 0 \\x_i, s_i, y_i, z_i &\geq 0 \text{ for } i = 1, 2, \dots, 12.\end{aligned}$$

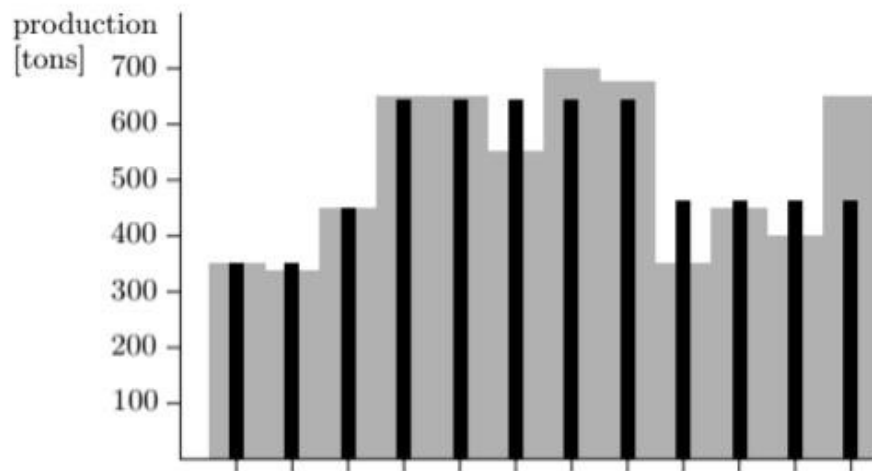
A resposta para uma dada instância é um vetor (x^*, y^*, z^*, s^*)

Para $i = 1, \dots, 12$, observe que y_i^* ou z_i^* é zero (caso contrário, solução não é ótima)

- ▶ Lembrando que $x_i^* = y_i^* + z_i^*$

PL - Modelagem: fábrica de sorvetes

Demanda vs Produção:



PL - Modelagem: fábrica de papel

Fábrica produz rolos de papel com 3 metros de largura.

- ▶ Compradores querem larguras diversas.

Por exemplo, 3m pode ser cortado em:

- ▶ dois rolos de 93cm
- ▶ um de 108
- ▶ sobra: 6cm

PL - Modelagem: fábrica de papel

Considere o pedido:

- ▶ 97 de 135cm 610 de 108cm
- ▶ 395 de 93cm 211 de 42cm

Quantos rolos de 3 metros devem ser produzidos? Possibilidades de cortes:

P1:	2×135	P7:	$108 + 93 + 2 \times 42$
P2:	$135 + 108 + 42$	P8:	$108 + 4 \times 42$
P3:	$135 + 93 + 42$	P9:	3×93
P4:	$135 + 3 \times 42$	P10:	$2 \times 93 + 2 \times 42$
P5:	$2 \times 108 + 2 \times 42$	P11:	$93 + 4 \times 42$
P6:	$108 + 2 \times 93$	P12:	7×42

- ▶ x_i : quantidade de rolos cortados de acordo com P_i
- ▶ Por exemplo, para restrição “395 de 93cm”, temos
 - ▶ $x_3 + 2x_6 + x_7 + 3x_9 + 2x_{10} + x_{11} \geq 395$
- ▶ Para as demais restrições, análogo
- ▶ **Objetivo:** $\sum_{j=1}^{12} x_j$

Solução: $x_1 = 48.5$, $x_5 = 206.25$, $x_6 = 197.5$ (demais $x_i = 0$)

Pergunta: soluções fracionárias são aceitas?

Soluções inteiras

Para algumas aplicações de PL queremos soluções inteiras, mas a solução ótima de um PL pode não ser inteira.

- ▶ Para o problema da fábrica de papel, da aula passada, poderíamos usar a solução ótima arredondada como resposta?
 - ▶ Solução encontrada: $x_1 = 48.5$, $x_5 = 206.25$, $x_6 = 197.5$
 - ▶ Arredondando, teríamos $x_1 = 49$, $x_5 = 207$, $x_6 = 198$
 - ▶ Assim, a resposta fica com 454 rolos de papel (lembrando: a função objetivo é $\sum_{j=1}^{12} x_j$)
- ▶ Entretanto, existe uma solução **viável inteira** com **453 rolos**:
 - ▶ $x_1 = 49$, $x_5 = 207$, $x_6 = 196$, $x_9 = 1$

E agora? Será que PL pode ser útil em problemas cuja solução é inteira?

- ▶ Às vezes sim, às vezes não.

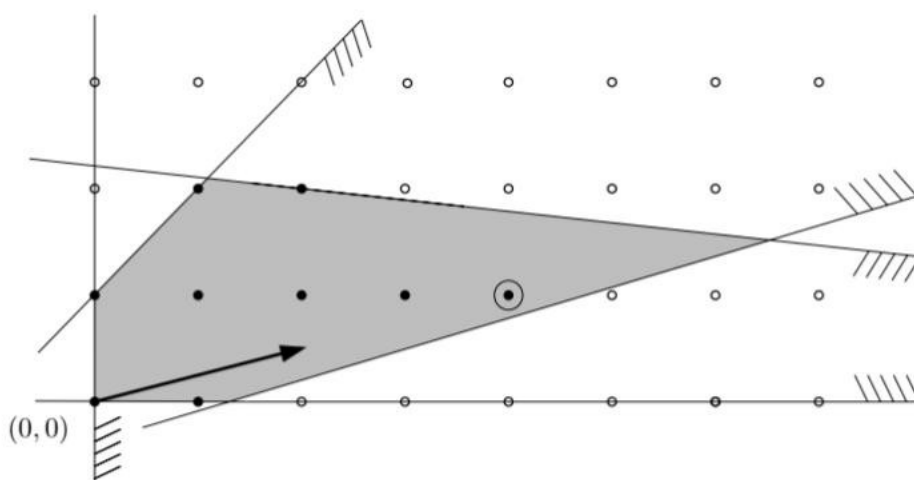
Programação Linear Inteira

PROGRAMAÇÃO LINEAR (PLI):

- ▶ **Instância:** Uma tripla $(A, \mathbf{b}, \mathbf{c})$ onde A é uma matriz $m \times n$ sobre $\mathbb{R}$, $\mathbf{b}$ é um vetor sobre $\mathbb{R}^m$ e $\mathbf{c}$ é um vetor sobre $\mathbb{R}^n$.
- ▶ **Resposta:** Um vetor $\mathbf{x}^* \in \mathbb{Z}^n$ satisfazendo
 - ▶ $A\mathbf{x}^* \leq \mathbf{b}$,
 - ▶ $\mathbf{c}\mathbf{x}^* \geq \mathbf{cx}$, $\forall \mathbf{x} \mid A\mathbf{x} \leq \mathbf{b}$

Programação Linear Inteira (PLI)

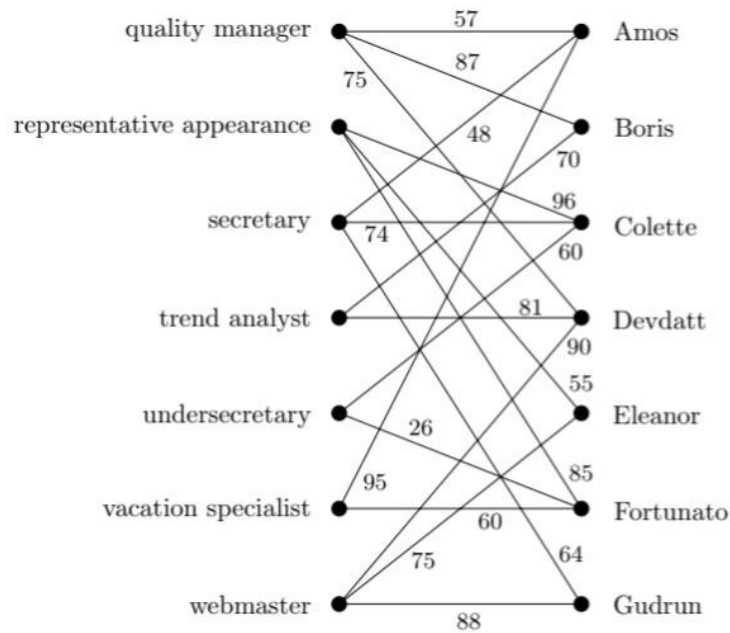
Exemplo para PLI com duas variáveis



- ▶ Espaço de busca: pontos de $\mathbb{Z}^2$
- ▶ Soluções viáveis e ótimas: subconjunto destes pontos

Obs: Como em PL, por enquanto estamos focando apenas no problema de PLI
Depois veremos como solucionar PLI (em geral são problemas difíceis)

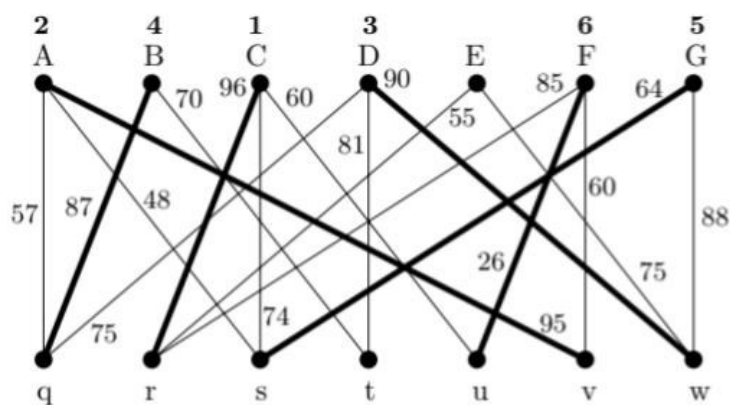
Programação Linear Inteira (PLI): Emparelhamento



- ▶ Queremos preencher **todas as vagas**, maximizando as preferências.
- ▶ Será que um algoritmo guloso funciona?

Programação Linear Inteira (PLI): Emparelhamento

Solução via algoritmo *guloso*:



- ▶ Solução obtida não é viável.
- ▶ Obs: Mesmo se quiséssemos retirar a restrição de preenchimento de todas as vagas, o algoritmo não funcionaria.

Definições auxiliares: grafos bipartidos

Seja $G = (V, E)$ um grafo.

Grafo Bipartido: O grafo G é *bipartido* se existe uma partição (X, Y) de V tal que X e Y são conjuntos independentes

Emparelhamento perfeito: $M \subseteq E$ é um *emparelhamento perfeito* em G se $\forall v \in V, |\partial(v)| = 1$, onde $\partial(v)$ é o conjunto de arestas incidentes a v , ou seja,

- ▶ **nenhum par de arestas** de M tem pontas em comum
- ▶ cada vértice de V é ponta de **alguma** aresta de M

Exemplo de PLI

EMPARELHAMENTO PERFEITO MÁXIMO EM GRAFOS BIPARTIDOS (EPMB)

Instância: Grafo ponderado *bipartido* $G = (V, E)$

Resposta: Um *emparelhamento perfeito* de peso máximo em G , caso exista

Idéia importante: Dado G ,

- ▶ Queremos construir uma instância $(A_G, \mathbf{b}_G, \mathbf{c}_G)$ de PLI tal que
- ▶ A resposta da instância nos fornece a resposta para G

Variáveis: $\forall e \in E$, temos uma variável x_e

- ▶ Queremos variáveis binárias: $x_e = 1$ ou 0 indica se $e \in M$ ou $e \notin M$
- ▶ Para isso, temos a restrição: $0 \leq x_e \leq 1$ e $x_e \in \mathbb{Z}$

Equações: Para cada vértice $v \in V$ temos uma equação: $\sum_{e \in \partial(v)} w(e)x_e = 1$

Função objetivo: maximizar $\sum_{e \in E(G)} w(e)x_e$