Análise e Síntese de Algoritmos

Programação Linear [CLRS, Cap. 29]

2011/2012

Contexto

- Revisão [CLRS, Cap.1-13]
  - Fundamentos; notação; exemplos
- Algoritmos em Grafos [CLRS, Cap.21-26]
  - Algoritmos elementares
  - Árvores abrangentes
  - Caminhos mais curtos
  - Fluxos máximos
- Programação Linear [CLRS, Cap.29]
  - Algoritmos e modelação de problemas com restrições lineares
- Técnicas de Síntese de Algoritmos [CLRS, Cap.15-16]
  - Programação dinâmica
  - Algoritmos greedy
- Tópicos Adicionais [CLRS, Cap.32-35]
  - Emparelhamento de Cadeias de Caracteres
  - Complexidade Computacional
  - Algoritmos de Aproximação

Resumo

Motivação

Exemplo

	Urbanos	Suburbanos	Rurais
Estradas	-2	5	3
Liberalização da Droga	8	2	-5
Subsídios Agricultura	0	0	10
Imposto sobre Gasolina	10	0	-2

- Cada entrada representa o número de votos (em milhares) ganhos por cada 1000 Euros gastos em campanhas
- Objectivo a atingir:
  - Queremos ganhar pelo menos 50% dos votos (100.000 urbanos, 200.000 suburbanos e 50.000 rurais)
  - Minimizar o total a gastar nas campanhas

Motivação

Exemplo

	Urbanos	Suburbanos	Rurais
Estradas	-2	5	3
Liberalização da Droga	8	2	-5
Subsídios Agricultura	0	0	10
Imposto sobre Gasolina	10	0	-2

- Considere que as seguintes variáveis denotam a quantia a gastar em campanha para os diferentes temas:
- x_1 = estradas; x_2 = droga; x_3 = subsídios; x_4 = imposto

$$
\begin{array}{lllll}
\text{minimizar} & \sum_{j=1}^{4} x_j & & \\
\text{sujeito a} & -2x_1 + 8x_2 + 0x_3 + 10x_4 & \geq & 50 \\
 & 5x_1 + 2x_2 + 0x_3 + 0x_4 & \geq & 100 \\
 & 3x_1 - 5x_2 + 10x_3 - 2x_4 & \geq & 25 \\
 & x_1, x_2, x_3, x_4 \geq 0 & &
\end{array}
$$

Formulações

Formulação Geral

Programação Linear

- Optimizar (minimizar ou maximizar) função linear sujeita a conjunto de restrições lineares
- Função linear (função objectivo):

$$f(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} c_j x_j$$

- Restrições Lineares:

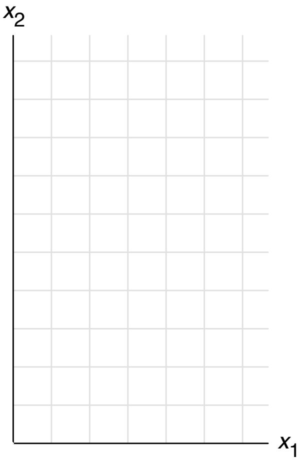
$$g_1(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} a_{ij} x_j \quad \begin{array}{c} \geq \\ = \\ \leq \end{array} b_i$$

Formulações

Exemplo

maximize

subject to



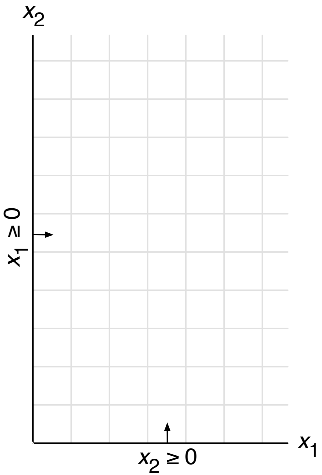
Formulações

Exemplo

maximize

subject to

$$x_1, x_2 \geq 0$$

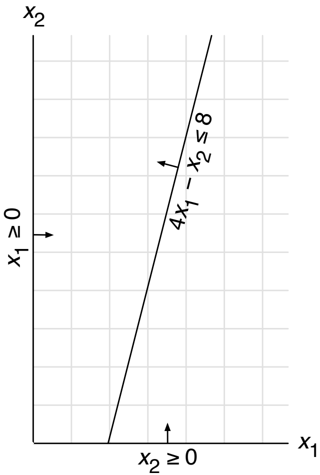


Formulações

Exemplo

maximize
subject to

$$
\begin{array}{rcrcl}
4x_1 & - & x_2 & \leq & 8 \\
 & & & & \\
 & x_1, x_2 & & \geq & 0
\end{array}
$$



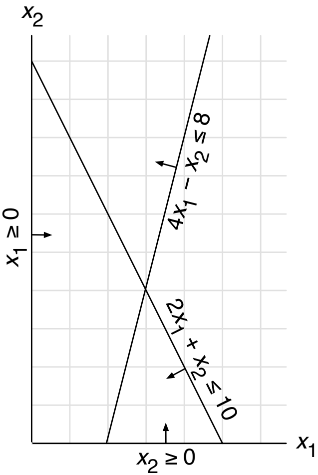
Formulações

Exemplo

maximize

subject to

$$
\begin{array}{rcrcr}
4x_1 & - & x_2 & \leq & 8 \\
2x_1 & + & x_2 & \leq & 10 \\
\\
& x_1, x_2 & & \geq & 0
\end{array}
$$



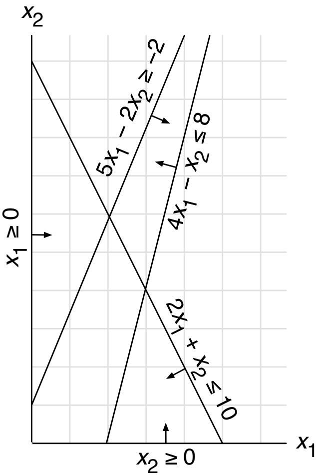
Formulações

Exemplo

maximize

subject to

$$\begin{array}{rcrcr} 4x_1 & - & x_2 & \leq & 8 \\ 2x_1 & + & x_2 & \leq & 10 \\ 5x_1 & - & 2x_2 & \geq & -2 \\ & x_1, x_2 & & \geq & 0 \end{array}$$



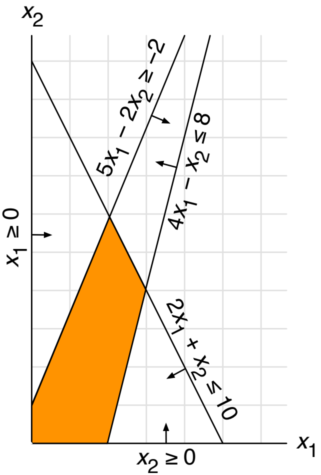
Formulações

Exemplo

maximize

subject to

$$\begin{array}{rcrcr} 4x_1 & - & x_2 & \leq & 8 \\ 2x_1 & + & x_2 & \leq & 10 \\ 5x_1 & - & 2x_2 & \geq & -2 \\ & x_1, x_2 & & \geq & 0 \end{array}$$

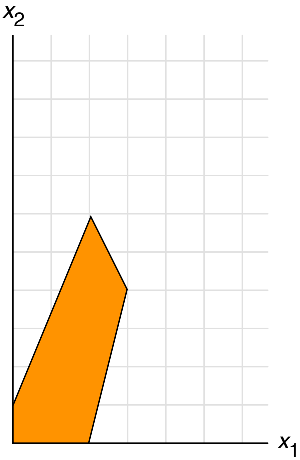


Formulações

Exemplo

maximize
subject to

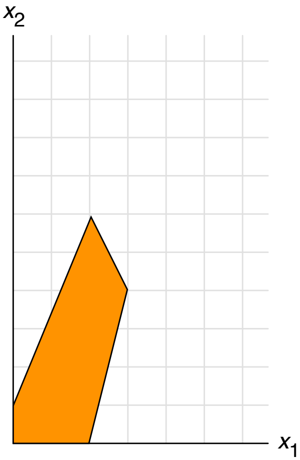
$$
\begin{array}{rcrcr}
4x_1 & - & x_2 & \leq & 8 \\
2x_1 & + & x_2 & \leq & 10 \\
5x_1 & - & 2x_2 & \geq & -2 \\
 & x_1, x_2 & & \geq & 0
\end{array}
$$



Formulações

Exemplo

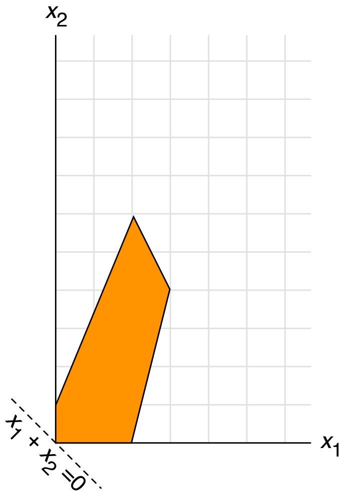
$$
\begin{array}{lccccc}
\text{maximize} & x_1 & + & x_2 & & \\
\text{subject to} & & & & & \\
 & 4x_1 & - & x_2 & \leq & 8 \\
 & 2x_1 & + & x_2 & \leq & 10 \\
 & 5x_1 & - & 2x_2 & \geq & -2 \\
 & & x_1, x_2 & & \geq & 0
\end{array}
$$



Formulações

Exemplo

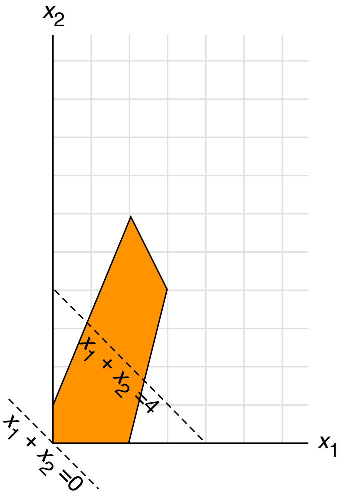
$$
\begin{array}{lccccc}
\text{maximize} & x_1 & + & x_2 & & \\
\text{subject to} & & & & & \\
 & 4x_1 & - & x_2 & \leq & 8 \\
 & 2x_1 & + & x_2 & \leq & 10 \\
 & 5x_1 & - & 2x_2 & \geq & -2 \\
 & & x_1, x_2 & & \geq & 0
\end{array}
$$



Formulações

Exemplo

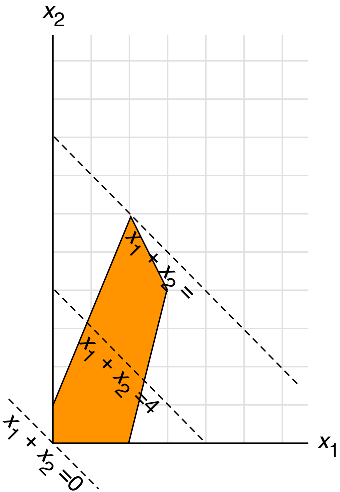
$$
\begin{array}{lccccc}
\text{maximize} & x_1 & + & x_2 & & \\
\text{subject to} & & & & & \\
& 4x_1 & - & x_2 & \leq & 8 \\
& 2x_1 & + & x_2 & \leq & 10 \\
& 5x_1 & - & 2x_2 & \geq & -2 \\
& & x_1, x_2 & & \geq & 0
\end{array}
$$



Formulações

Exemplo

$$
\begin{array}{lrcrcr}
\text{maximize} & x_1 & + & x_2 & & \\
\text{subject to} & & & & & \\
 & 4x_1 & - & x_2 & \leq & 8 \\
 & 2x_1 & + & x_2 & \leq & 10 \\
 & 5x_1 & - & 2x_2 & \geq & -2 \\
 & & x_1, x_2 & & \geq & 0
\end{array}
$$



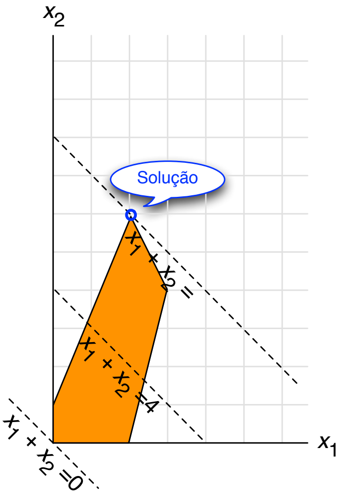
Formulações

Exemplo

$$\begin{array}{lrcrcr} \text{maximize} & x_1 & + & x_2 & & \\ \text{subject to} & & & & & \\ & 4x_1 & - & x_2 & \leq & 8 \\ & 2x_1 & + & x_2 & \leq & 10 \\ & 5x_1 & - & 2x_2 & \geq & -2 \\ & & x_1, x_2 & & \geq & 0 \end{array}$$

Solução

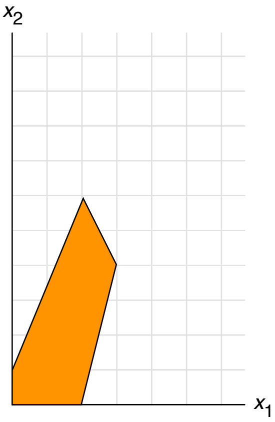
$x_1 = 2, \ x_2 = 6$



Formulações

Região Exequível

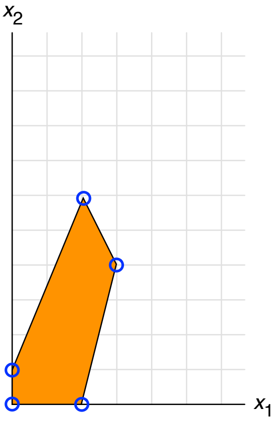
- Qualquer solução que satisfaça o conjunto de restrições designa-se por solução exequível
- A cada solução exequível corresponde um valor (custo) da função objectivo
- O conjunto de soluções exequíveis é designado por região exequível
- A região exequível é um conjunto convexo no espaço n-dimensional
  - Conjunto convexo S: qualquer ponto de um segmento que liga quaisquer dois pontos em S está também em S
  - S é designado por simplex



Formulações

Região Exequível

- Qualquer solução que satisfaça o conjunto de restrições designa-se por solução exequível
- A cada solução exequível corresponde um valor (custo) da função objectivo
- O conjunto de soluções exequíveis é designado por região exequível
- A região exequível é um conjunto convexo no espaço *n*-dimensional
  - Conjunto convexo *S*: qualquer ponto de um segmento que liga quaisquer dois pontos em *S* está também em *S*
  - *S* é designado por simplex
- A solução óptima encontra-se num vértice do simplex



Formulações

Algoritmos

- Algoritmo Simplex
  - Exponencial no pior caso; eficiente na prática e muito utilizado
- Algoritmo da Elipsóide
  - Polinomial; normalmente ineficiente
- Métodos de Ponto Interior

Formulações

Perspectiva sobre Programação Linear

Conceitos a reter:

- Solução: exequível ou não exequível
- Valor da função objectivo: valor objectivo
- Valor máximo/mínimo: valor objectivo óptimo
- Se formulação não tem soluções exequíveis diz-se não exequível; caso contrário diz-se exequível
- Se formulação é exequível, mas sem solução óptima, diz-se não limitado

Formulações

Caminhos Mais Curtos

Caminhos mais curtos entre s e t:

$$\begin{array}{ll} \text{maximizar} & d[t] \\ \text{sujeito a} & d[v] \leq d[u] + \omega(u,v), \forall (u,v) \in E \\ & d[s] = 0 \end{array}$$

Fluxo Máximo

$$\begin{array}{lll} \text{maximizar} & \sum_{v \in V} f(s,v) & \\ \text{sujeito a} & f(u,v) \leq c(u,v) & \forall u,v \in V \\ & f(u,v) = -f(v,u) & \forall u,v \in V \\ & \sum_{v \in V} f(u,v) = 0 & \forall u \in V - \{s,t\} \end{array}$$

Formulações

Forma Standard

$$\begin{array}{lll} \text{maximizar} & \sum_{j=1}^{n} c_j x_j & \\ \text{sujeito a} & \sum_{j=1}^{n} a_{ij} x_j \leq b_i & i = 1, 2, \ldots, m \\ & x_j \geq 0 & j = 1, 2, \ldots, n \end{array}$$

- Todos os valores c_j, a_{ij}, b_i são valores reais
- Representação Matricial

$$\begin{array}{ll} \text{maximizar} & \mathbf{c}^T \mathbf{x} \\ \text{sujeito a} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0 \end{array}$$

Em que $\mathbf{A} = (a_{ij}), \mathbf{b} = (b_i), \mathbf{c} = (c_j)$ e $\mathbf{x} = (x_j)$

Formulações

Problemas na Conversão para a Forma Standard

- Minimização em vez de maximização
- Variáveis sem restrição de serem não negativas
- Restrições com igualdade
- Restricões com $\geq$

Formulações

Problemas na Conversão para a Forma Standard

- Minimização vs. Maximização:
  - Multiplicar coeficientes por -1
- Variáveis sem restrição de serem não negativas:
  - Substituir cada ocorrência de x_i por $(x_{i1} - x_{i2})$, em que x_{i1} e x_{i2} são novas variáveis
- Restrições com igualdade:
  - Introduzir duas restrições, uma com $\leq$ e outra com $\geq$
- Restrições com $\geq$:
  - Multiplicar restrição por –1

Formulações

Conversão para a Forma Slack

- Objectivo é trabalhar apenas com igualdades
- Todas as restrições, excepto as restrições das variáveis serem não negativas, são igualdades
- Para cada restrição introduzir uma nova variável s_i (variável de slack)

$$s_i = b_i - \sum_{j=1}^{n} a_{ij} x_j$$
$$s_i \geq 0$$

- Conversão de forma standard para forma slack:

$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j$$
$$x_{n+i} \geq 0$$

Formulações

Conversão para a Forma Slack

Nas expressões: $x_{n+i} = b_i - \sum_{j=1}^{n} a_{ij} x_j$

- Variáveis expressas em função de outras variáveis designam-se por variáveis básicas
- As variáveis que definem as variáveis básicas designam-se por variáveis não-básicas
- A solução básica é obtida quando se colocam as variáveis não-básicas com valor 0

Na forma slack, a função objectivo é definida como:

$$z = \sum_{j=1}^{n} c_j x_j$$

Formulações

Conversão para a Forma Slack

- N: Conjunto de índices das variáveis não básicas, $|N| = n$
- B: Conjunto de índices das variáveis básicas, $|B| = m$
  - $N \cup B = \{1, 2, \ldots, n+m\}$
- Forma slack descrita por: (N, B, A, b, c, v)
  - v: constante na função objectivo

$$
\begin{aligned}
z &= v + \sum_{j=1}^{n} c_j x_j \\
x_{n+i} &= b_i - \sum_{j=1}^{n} a_{ij} x_j \quad i = 1, 2, \ldots, m
\end{aligned}
$$

Formulações

$$\begin{array}{rrcrcr} \text{minimize} & -2x_1 & + & 3x_2 & & \\ \text{subject to} & & & & & \\ & x_1 & + & x_2 & = & 7 \\ & x_1 & - & 2x_2 & \leq & 4 \\ & x_1 & & & \geq & 0 \end{array}$$

$$\begin{array}{rrcrcr} \text{maximize} & 2x_1 & - & 3x_2 & & \\ \text{subject to} & & & & & \\ & x_1 & + & x_2 & = & 7 \\ & x_1 & - & 2x_2 & \leq & 4 \\ & x_1 & & & \geq & 0 \end{array}$$

Formulações

$$
\begin{array}{lrcrcl}
\text{maximize} & 2x_1 & - & 3x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & + & x_2 & = & 7 \\
& x_1 & - & 2x_2 & \leq & 4 \\
& x_1 & & & \geq & 0
\end{array}
$$

$$
\begin{array}{lrcrcrcl}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & = & 7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

Formulações

$$
\begin{array}{lrcrcrcl}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
 & x_1 & + & x_2' & - & x_2'' & = & 7 \\
 & x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
 & & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

$$
\begin{array}{lrcrcrcl}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
 & x_1 & + & x_2' & - & x_2'' & \leq & 7 \\
 & x_1 & + & x_2' & - & x_2'' & \geq & 7 \\
 & x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
 & & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

Formulações

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & \leq & 7 \\
& x_1 & + & x_2' & - & x_2'' & \geq & 7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & \leq & 7 \\
& -x_1 & - & x_2' & + & x_2'' & \leq & -7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

Formulações

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & \leq & 7 \\
& -x_1 & - & x_2' & + & x_2'' & \leq & -7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0
\end{array}
$$

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2 & + & 3x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & - & x_3 & \leq & 7 \\
& -x_1 & - & x_2 & + & x_3 & \leq & -7 \\
& x_1 & - & 2x_2 & + & 2x_3 & \leq & 4 \\
& & & x_1, x_2, x_3 & & & \geq & 0
\end{array}
$$

Formulações

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2 & + & 3x_3 & & \\
\text{subject to} & & & & & & & \\
 & x_1 & + & x_2 & - & x_3 & \leq & 7 \\
 & -x_1 & - & x_2 & + & x_3 & \leq & -7 \\
 & x_1 & - & 2x_2 & + & 2x_3 & \leq & 4 \\
 & & & x_1, x_2, x_3 & & & \geq & 0
\end{array}
$$

$$
\begin{array}{rcrcrcrcr}
z & = & & & 2x_1 & - & 3x_2 & + & 3x_3 \\
x_4 & = & 7 & - & x_1 & - & x_2 & + & x_3 \\
x_5 & = & -7 & + & x_1 & + & x_2 & - & x_3 \\
x_6 & = & 4 & - & x_1 & + & 2x_2 & - & 2x_3
\end{array}
$$

Algoritmo Simplex

Operação Pivot

Operação central do algoritmo Simplex

- Escolher variável não básica x_e para passar a básica
  - Variável de entrada
- Escolher variável básica x_l para passar a não básica
  - Variável de saída
- Calcular nova forma slack do problema
  - $N' = N - \{x_e\} \cup \{x_l\}$
  - $B' = B - \{x_l\} \cup \{x_e\}$
  - (N', B', A, b, c, v)

Algoritmo Simplex

Algoritmo Simplex

- Calcular forma slack inicial
  - Para a qual solução básica inicial é exequível
  - Caso contrário reporta problema não exequível (*unfeasible*) e termina
- Enquanto existir $c_e > 0$ (i.e. valor de z pode aumentar)
  - x_e define variável de entrada (i.e. nova variável básica)
  - Seleccionar x_l
    - x_l corresponde a linha i que minimiza b_i / a_{ie}, para $a_{ie} > 0$
    - Se $a_{ie} < 0$ para todo o i, retornar "unbounded"
- Aplicar pivoting com (N, B, A, b, c, v, l, e)
- No final, retornar solução básica
  - $\overline{x}_i \leftarrow b_i$, se $i \in B$ (variáveis básicas)
  - $\overline{x}_i \leftarrow 0$, se $i \in N$ (variáveis não-básicas)

Algoritmo Simplex

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 3x_1 & + & x_2 & + & 2x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & + & 3x_3 & \leq & 30 \\
& 2x_1 & + & 2x_2 & + & 5x_3 & \leq & 24 \\
& 4x_1 & + & x_2 & + & 2x_3 & \leq & 36 \\
& & & x_1, x_2, x_3 & & & \geq & 0
\end{array}
$$

$$
\begin{array}{rcrcrcrcr}
z & = & & & 3x_1 & + & x_2 & + & 2x_3 \\
x_4 & = & 30 & - & x_1 & - & x_2 & - & 3x_3 \\
x_5 & = & 24 & - & 2x_1 & - & 2x_2 & - & 5x_3 \\
x_6 & = & 36 & - & 4x_1 & - & x_2 & - & 2x_3
\end{array}
$$

Algoritmo Simplex

$$
\begin{aligned}
z &= && 3x_1 + x_2 + 2x_3 \\
x_4 &= 30 &-& x_1 - x_2 - 3x_3 \\
x_5 &= 24 &-& 2x_1 - 2x_2 - 5x_3 \\
x_6 &= 36 &-& 4x_1 - x_2 - 2x_3
\end{aligned}
$$

$$
\begin{aligned}
z &= 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4} \\
x_1 &= 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \\
x_4 &= 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4} \\
x_5 &= 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2}
\end{aligned}
$$

Algoritmo Simplex

$$
\begin{aligned}
z &= 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4} \\
x_1 &= 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \\
x_4 &= 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4} \\
x_5 &= 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2}
\end{aligned}
$$

$$
\begin{aligned}
z &= \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16} \\
x_1 &= \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16} \\
x_3 &= \frac{3}{2} - \frac{3x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \\
x_4 &= \frac{69}{4} + \frac{3x_2}{16} + \frac{5x_5}{8} - \frac{x_6}{16}
\end{aligned}
$$

Algoritmo Simplex

$$
\begin{aligned}
z &= \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16} \\
x_1 &= \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16} \\
x_3 &= \frac{3}{2} - \frac{3x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \\
x_4 &= \frac{69}{4} + \frac{3x_2}{16} + \frac{5x_5}{8} - \frac{x_6}{16}
\end{aligned}
$$

$$
\begin{aligned}
z &= 28 - \frac{x_3}{6} - \frac{x_5}{6} - \frac{2x_6}{3} \\
x_1 &= 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3} \\
x_2 &= 4 - \frac{8x_3}{3} - \frac{2x_5}{3} + \frac{x_6}{3} \\
x_4 &= 18 - \frac{x_3}{2} + \frac{x_5}{2}
\end{aligned}
$$

Algoritmo Simplex

Solução Exequível Inicial

Um programa linear pode ser exequível, mas solução básica inicial pode não ser exequível

- Seja L um programa linear na forma standard, e seja L_{aux} definido da seguinte forma:

$$
\begin{array}{lll}
\text{maximizar} & -x_0 & \\
\text{sujeito a} & \sum_{j=1}^{n} a_{ij} x_j - x_0 \leq b_i & i = 1, 2, \ldots, m \\
 & x_j \geq 0 & j = 0, 1, 2, \ldots, n
\end{array}
$$

- Então L é exequível se e só se o valor objectivo óptimo de L_{aux} é 0
  - Se L tem solução, então L_{aux} tem solução com $x_0 = 0$, o valor óptimo
  - Se o valor óptimo de x_0 é 0, então solução é solução para L

Algoritmo Simplex

Solução Exequível Inicial

Se solução básica inicial for não exequível:

- A partir de L construir L_{aux}
- Determinar índice l com menor b_i
  - Aplicar operação pivot entre x_l e x_0
  - A solução básica calculada é exequível para L_{aux}
- Aplicar passos do Simplex para calcular solução óptima
  - Se solução óptima verifica $x_0 = 0$, retornar solução calculada, sem x_0
  - Caso contrário L não é exequível

Algoritmo Simplex

Solução Exequível Inicial

- Após a primeira aplicação de pivot, a solução básica é exequível para L_{aux}
  - $e = 0$
  - l tal que $b_l < b_i$, $i = 1, \ldots, m$
    - $b_l < 0$, pois solução inicial exequível se $bi \geq 0$
  - Após aplicar operação pivot tem-se:
    - $x_0 = b_l / a_{l0}$
    - $x_i = b_i - a_{i0}(b_l / a_{l0})$, $i \neq 0$
  - Como $a_{i0} = -1$ para todo o i,
    - $x_0 = -b_l > 0$
    - $x_i = b_i - b_l > 0$

Algoritmo Simplex

$$
\begin{array}{llcrcr}
\text{maximize} & 2x_1 & - & x_2 & & \\
\text{subject to} & & & & & \\
& 2x_1 & - & x_2 & \leq & 2 \\
& x_1 & - & 5x_2 & \leq & -4 \\
& & x_1, x_2 & & \geq & 0
\end{array}
$$

$$
\begin{array}{llcrcrcr}
\text{maximize} & & & -x_0 & & & & \\
\text{subject to} & & & & & & & \\
& 2x_1 & - & x_2 & - & x_0 & \leq & 2 \\
& x_1 & - & 5x_2 & - & x_0 & \leq & -4 \\
& & & x_1, x_2, x_0 & & & \geq & 0
\end{array}
$$

Algoritmo Simplex

Resultados Formais

Dado um programa linear (A, b, c):

- Se o algoritmo Simplex retorna uma solução, a solução é exequível
- Se o algoritmo Simplex retorna "unbounded", o programa é não limitado
- Dado um programa linear (A, b, c) na forma standard, e B um conjunto de variáveis básicas, a forma slack é única

Algoritmo Simplex

Resultados Formais

Variação do valor da função objectivo após pivoting:

- Valor da função objectivo não pode diminuir
  - Variável escolhida tem coeficiente positivo
  - Valor da variável é não negativo, pelo que novo valor da função de custo não pode diminuir
- Valor da função objectivo pode não aumentar
  - Degenerescência
  - Mas é sempre possível assegurar que algoritmo termina

Algoritmo Simplex

Resultados Formais

O Simplex está em ciclo se existem formas slack idênticas para duas iterações do algoritmo

- Se o algoritmo Simplex não termina após C_m^{n+m} iterações, então o algoritmo está em ciclo
  - Cada conjunto B determina unicamente a forma slack
  - Existem $n+m$ variáveis e $|B| = m$
  - Número de modos de escolher B: C_m^{n+m}
  - Número de formas slack distintas: C_m^{n+m}
  - Se algoritmo executar mais de C_m^{n+m} iterações, então está em ciclo
- Eliminar ciclos:
  - Regra de Bland: desempates na escolha de variáveis através da escolha da variável com o menor indíce

Dualidade

Dualidade

- Conceito essencial em optimização
  - Normalmente associado com existência de algoritmos polinomiais
  - E.g., fluxo máximo corte mínimo
- Programa linear dual:

$$
\begin{array}{lll}
\text{minimizar} & \sum_{i=1}^{m} b_i y_i & \\
\text{sujeito a} & \sum_{i=1}^{m} a_{ij} y_i \geq c_j & j = 1, 2, \ldots, n \\
 & y_i \geq 0 & i = 1, 2, \ldots, m
\end{array}
$$

- A formulação original é conhecida como o programa primal

Dualidade

Dualidade Fraca em Programação Linear

Seja x uma qualquer solução exequível do programa primal e seja y uma qualquer solução exequível do programa dual. Nestas condições:

$$\sum_{j=1}^{n} c_j x_j \leq \sum_{i=1}^{m} b_i y_i$$

Prova:

$$\begin{aligned}
\sum_{j=1}^{n} c_j x_j &\leq \sum_{j=1}^{n} \Big(\sum_{i=1}^{m} a_{ij} y_i \Big) x_j \\
&= \sum_{i=1}^{m} \Big(\sum_{j=1}^{n} a_{ij} x_j \Big) y_i \\
&\leq \sum_{i=1}^{m} b_i y_i
\end{aligned}$$

Dualidade

Dualidade em Programação Linear

Seja x uma qualquer solução pelo algoritmo Simplex, e sejam N e B os conjuntos de variáveis para a forma slack final.
Seja c' o vector dos coeficientes da forma slack final e seja $y_i = -c'_{n+i}$ para $(n+i) \in N$; 0 caso contrário.
Nestas condições:

- x é solução óptima para o programa primal
- y é a solução óptima para o programa dual
- e, $\sum_{j=1}^{n} c_j x_j = \sum_{i=1}^{m} b_i y_i$

Dualidade

Teorema Fundamental da Programação Linear

Qualquer programa linear na forma standard:

- Ou tem solução óptima com valor finito,
- Ou não é exequível,
- Ou não é limitado.
- Se L não é exequível, o algoritmo Simplex retorna "infeasible"
- Se L não é limitado, o algoritmo Simplex retorna "unbounded"
- Caso contrário, o algoritmo Simplex retorna uma solução óptima com um valor objectivo finito