

IA881 – Otimização Linear

Aula: Métodos de Pontos Interiores em Programação Linear

Ricardo C. L. F. Oliveira

Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas

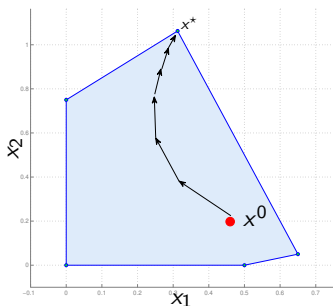
1º Semestre 2019

Mudança de paradigma

- Em 1984, Narendra K. Karmarkar propôs um novo algoritmo de complexidade polinomial para programação linear. A novidade era como uma solução evoluía em direção ao ótimo: **pele interior** da região factível.

- Diferentemente do método do elipsoide, esse novo algoritmo era um concorrente “à altura” do tradicional Simplex em problemas reais de grandes dimensões.

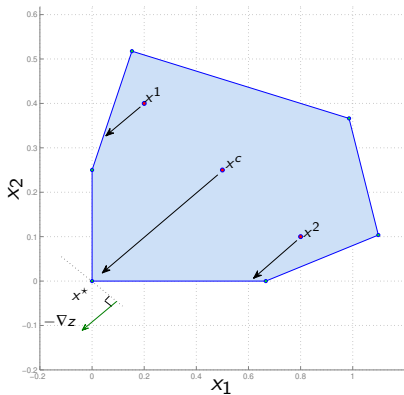
- Outra vantagem da técnica é que outras classes de problemas também podem ser tratadas, inclusive problemas de otimização não lineares.



Centro do poliedro

■ A primeira observação importante de Karmarkar é ilustrada na figura ao lado. Note que em x^c , próximo ao centro do poliedro, é possível ter a melhor melhoria na função objetivo ao caminhar pela direção de descida mais acentuada ($-\nabla z$). Em x^1 e x^2 a descida será menor (para não sair da região factível).

■ O segundo *insight*, necessário para viabilizar o primeiro, foi a chamada “transformação projetiva”, que “distorce” o espaço de busca de modo que a solução atual esteja no centro do espaço distorcido. Essa transformação preserva as propriedades fundamentais do problema.



Resumo do método de Karmarkar

- De modo resumido, o algoritmo de Karmarkar pode ser concebido da seguinte forma: Tome uma solução interior inicial e transforme o espaço de busca de modo que a solução atual esteja no centro do espaço transformado. Move-se na direção de descida mais acentuada, mas ainda permanecendo dentro da região factível. Aplique a transformação inversa na nova solução obtida e verifique algum critério de parada. Repete-se o procedimento até convergir.
- Originalmente Karmarkar apresentou sua ideia para um PL na seguinte forma

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.a} \quad & Ax = 0 \\ & e^T x = 1, \quad e = (1, 1, \dots, 1)^T \\ & x_i \geq 0 \end{aligned}$$

Uma das razões (existem outras) para o método de Karmarkar logo ter sido deixado em segundo plano foi a sua forma não padronizada para tratar um PL. A proposição da transformação *afim* em substituição à transformação projetiva de Karmarkar deu origem a métodos que tratam o PL na forma padrão. Por esse motivo (principalmente), não estudaremos a fundo os detalhes dos passos necessários para construir o algoritmo baseado em pontos interiores no método de Karmarkar, mas sim nos métodos baseados na transformação afim.

PL via pontos interiores

- Seja o PL na forma padrão

$$\begin{aligned} \min \quad & c^T x \\ \text{s.a} \quad & Ax = b \\ & x_i \geq 0 \end{aligned} \tag{1}$$

com $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ de rank completo de linhas e $x \in \mathbb{R}^n$ as variáveis do problema. A região de factibilidade do problema é definida pelo conjunto

$$\mathcal{P} = \{x \in \mathbb{R}^n \mid Ax = b, x_i \geq 0, i = 1, \dots, n\}$$

Também define-se o *interior* relativo a \mathcal{P} como sendo

$$\mathcal{P}_{int} = \{x \in \mathcal{P} \mid x_i > 0, i = 1, \dots, n\}$$

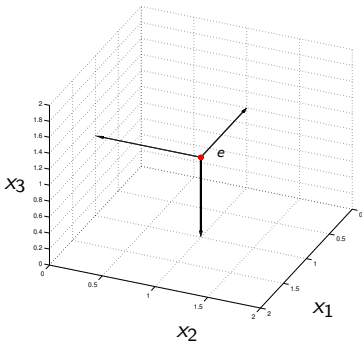
Um vetor $x \in \mathbb{R}^n$ é chamado **ponto interior** factível (ou solução interior) de (1) se $x \in \mathcal{P}_{int}$. Como hipótese básica, assumimos que \mathcal{P}_{int} não é vazio. Por enquanto assumimos também que um ponto interior factível inicial está disponível (discutiremos isso depois) e na sequência estudaremos todos os detalhes do algoritmo **primal afim escala**.

Nova transformação

- Ao contrário da formulação de PL usada por Karmarkar, que possui uma região factível limitada (definida pela restrição $e^T x = 1$), na formulação padrão podemos ter um poliedro ilimitado. A estrutura a ser considerada é a interseção do hiperplano $Ax = b$ com as restrições de não negatividade $x_i \geq 0$, que em princípio não fornece um “centro”.

- Uma saída é considerar o ponto e , que apesar de não ser o centro, possui a mesma distância para todas as “paredes” do octante não negativo. Desse modo, caso o movimento a partir de e tenha tamanho menor do que 1, garante-se que o próximo ponto continua no interior do octante não negativo.

- Com uma transformação apropriada para mapear um ponto qualquer em e , pode-se adaptar a estratégia de Karmarkar para trabalhar com o PL padrão.



Transformação afim escala

- Seja $x^k \in \mathbb{R}_+^n$ um ponto interior do octante não negativo, isto é, $x_i^k > 0$, $i = 1, \dots, n$, e a seguinte matriz diagonal

$$X_k = \text{diag}(x^k) = \begin{bmatrix} x_1^k & 0 & \dots & 0 \\ 0 & x_2^k & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & x_n^k \end{bmatrix}$$

Note que X_k é não singular e sua inversa pode ser facilmente determinada (invertendo os elementos da diagonal).

Definição 1

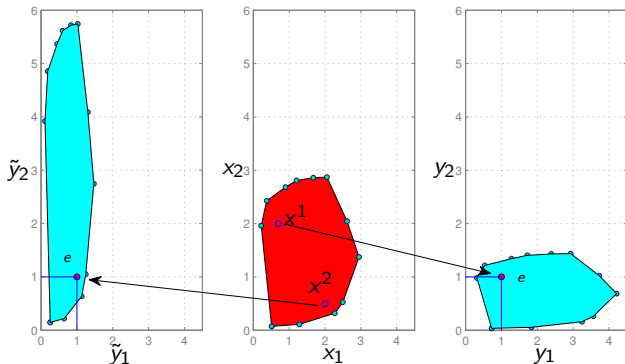
A *transformação afim escala* é definida como um mapa $\mathbb{R}_+^n \mapsto \mathbb{R}_+^n$ e dada na forma

$$y = T_k(x) = X_k^{-1}x$$

Note que transformação consiste basicamente numa *mudança de escala*, dividindo cada componente do vetor x_i por uma constante positiva (x_i^k).

Propriedades da transformação

- A figura abaixo mostra uma ilustração gráfica de uma transformação afim escala aplicada em um poliedro de 12 arestas (definido por 12 desigualdades em x_1 e x_2)



- Note que o politopo transformado não é uma representação completa, uma vez que serão introduzidas 12 variáveis de folga nas desigualdades originais, que estarão presentes da transformação.

Propriedades da transformação

- As seguintes propriedades podem ser facilmente reconhecidas
 - T_k é um mapa bem definido se x^k é um ponto interior de \mathcal{R}_+^n .
 - $T_k(x^k) = e$
 - $T_k(x)$ é um vértice de \mathcal{R}_+^n se x é um vértice.
 - $T_k(x)$ está sobre a borda de \mathcal{R}_+^n se x está sobre a borda.
 - $T_k(x)$ é um ponto interior de \mathcal{R}_+^n se x é um ponto interior.
 - T_k é um mapa bijetivo com uma transformação inversa T_k^{-1} tal que

$$T_k^{-1}(y) = X_k y, \quad y \in \mathbb{R}_+^n$$

PL transformado

- Adotando a relação $x = X_k y$, o PL original pode ser reescrito na forma

$$\begin{aligned} \min \quad & (c^k)^T y \\ \text{s.a} \quad & A_k y = b \\ & y_i \geq 0 \end{aligned} \tag{2}$$

com $c^k = X_k c$ e $A_k = A X_k$.

- O próximo passo é investigar as propriedades necessárias para garantir a evolução da solução inicial mantendo a factibilidade do problema e melhorando a função objetivo.
- Com relação à factibilidade, é necessário garantir que a próxima solução

$$y^{k+1} = e + \alpha d_y^k$$

permaneça no interior da região factível de (2), sendo que d_y^k é uma direção e α o tamanho do passo. Primeiro investigaremos como construir a direção d_y^k .

Direção factível

- Em um método de ponto interior, uma **direção factível** para uma solução atual é tal que se $z^k \in \mathcal{P}_{int}$ então $z^{k+1} \in \mathcal{P}_{int}$ com

$$z^{k+1} = z^k + \alpha d_z^k$$

Para manter a factibilidade, é necessário que

$$Az^{k+1} = b \Rightarrow Az^k + \alpha Ad_z^k = b \Rightarrow Ad_z^k = 0$$

ou seja, d_z^k precisa pertencer ao espaço nulo de A .

Definição 2

O **espaço nulo** de uma matriz $A \in \mathbb{R}^{m \times n}$ é dado por

$$\mathcal{N}(A) = \{x \in \mathbb{R}^n \mid Ax = 0\}$$

isto é, consiste no conjunto de vetores (direções) x tais que $Ax = 0$. A dimensão do espaço nulo é chamada de nulidade da matriz A é dada por $v(A) = n - \text{rank}(A)$.

Direção factível – Projecção

- Dada uma direção \hat{d} favorável à melhoria da função objetivo, é preciso projetar essa direção no subespaço gerado por $Ax = 0$, ou seja, no espaço nulo da matriz A . A projecção que garante a menor distância entre \hat{d} e a sua projecção pode ser computada pelo seguinte problema de otimização

$$\begin{array}{ll} \min_x & \|\hat{d} - x\|^2 \\ \text{s.a} & Ax = 0 \end{array}$$

que tem solução ótima dada por

$$x^* = \underbrace{(I - A^T(AA^T)^{-1}A)}_P \hat{d}$$

sendo P a matriz de [projecção ortogonal no espaço nulo](#) de A .

Projeção Via Álgebra Linear I

- Seja uma matriz $A \in \mathbb{R}^{m \times n}$, e seus quatro subespaços fundamentais [8, 3]: $\mathcal{C}(A)$ (espaço coluna), $\mathcal{C}(A')$ (espaço linha), $\mathcal{N}(A)$ (espaço nulo) e $\mathcal{N}(A')$ (espaço nulo à esquerda). Do teorema fundamental da álgebra linear, sabe-se que $\mathcal{C}(A')$ e $\mathcal{N}(A)$ são complementos ortogonais (são subespaços ortogonais entre si e juntos geram todo espaço vetorial, nesse caso o \mathbb{R}^n).

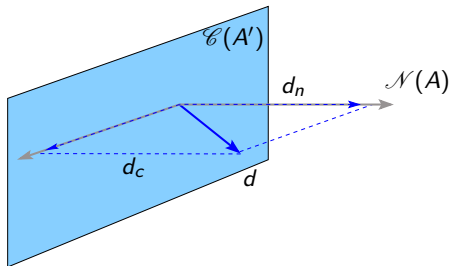


Figura 1: Espaços $\mathcal{C}(A')$ e $\mathcal{N}(A)$.

Projeção Via Álgebra Linear II

- Considere a ilustração apresentada na Figura 1. O objetivo é determinar d_n , que é a projeção do vetor d em $\mathcal{N}(A)$ (problema de projeção em subespaço). Inicialmente, note que $d = d_n + d_c$, com $d_c \in \mathcal{C}(A')$ e $d_n \in \mathcal{N}(A)$. Multiplicando $d_n = d - d_c$ à esquerda por A e notando que $Ad_n = 0$, temos

$$Ad_n = A(d - d_c) = 0 \Rightarrow Ad = Ad_c \quad (3)$$

Mas como d_c pertence a $\mathcal{C}(A')$, sabemos que $d_c = A'x$, com x desconhecido. Substituindo d_c em (3), temos

$$Ad = AA'x \Rightarrow x = (AA')^{-1}Ad.$$

Note que, como A tem rank completo de linhas, a inversa $(AA')^{-1}$ é bem definida. Portanto, a projeção de d em $\mathcal{C}(A')$ é $d_c = A'(AA')^{-1}Ad$, e a projeção de d em $\mathcal{N}(A)$ é $d_n = (I - A'(AA')^{-1}A)d$

Exemplo de projeção

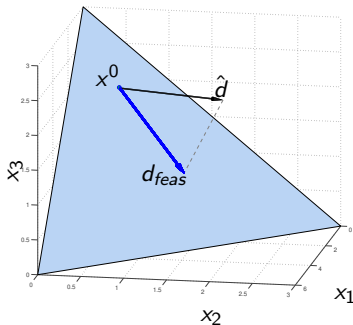
- Projete a direção $\hat{d} = (\frac{3}{2}, \frac{5}{2}, 0)$ de modo a garantir a factibilidade de $Ax = b$ com $A = [\frac{1}{2} \quad 1 \quad 1]$, $b = 3$.

$$AA^T = \frac{9}{4}$$

$$A^T(AA^T)^{-1}A = \begin{bmatrix} \frac{1}{9} & \frac{2}{9} & \frac{2}{9} \\ \frac{2}{9} & \frac{4}{9} & \frac{4}{9} \\ \frac{2}{9} & \frac{4}{9} & \frac{4}{9} \end{bmatrix}$$

$$P = \begin{bmatrix} \frac{8}{9} & -\frac{2}{9} & -\frac{2}{9} \\ -\frac{2}{9} & \frac{5}{9} & -\frac{4}{9} \\ -\frac{2}{9} & -\frac{4}{9} & \frac{5}{9} \end{bmatrix}$$

$$d_{feas} = \begin{bmatrix} \frac{14}{18} \\ \frac{16}{18} \\ \frac{26}{18} \end{bmatrix}$$



Direção de melhoria

- No PL (2), uma direção é considerada de **melhoria** para a solução atual se a mesma fornece uma nova solução de **custo menor**. Matematicamente temos a seguinte implicação (omitindo a dependência de c em k)

$$\begin{aligned} c^T y^{k+1} &\leq c^T y^k \Rightarrow \\ c^T y^k + \alpha c^T d_y^k &\leq c^T y^k \Rightarrow c^T d_y^k \leq 0 \end{aligned}$$

- Uma direção candidata que atende essa implicação é $-c$, ou seja, a direção contrária a do gradiente (descida mais acentuada). Desse modo, a direção d_y^k que **melhora** ($c^T d_y^k \leq 0$) e **mantém a factibilidade** ($A_k d_y^k = 0$) é dada pela projeção ortogonal de $-c$ no espaço nulo de A_k , ou seja

$$d_y^k = P_k(-c^k)$$

com

$$P_k = I - A_k^T (A_k A_k^T)^{-1} A_k = I - X_k A^T (A X_k^2 A^T)^{-1} A X_k$$

fornecendo

$$d_y^k = - \left(I - X_k A^T (A X_k^2 A^T)^{-1} A X_k \right) X_k c$$

- Note que a matriz de projeção P_k é bem definida se A tem rank completo de linhas (hipótese inicial) e se $x_i > 0$.

O quanto caminhar?

- Uma vez determinada a direção (factível e de melhoria) resta determinar qual será o **tamanho** do passo $\alpha > 0$ tal que

$$y^{k+1} = y^k + \alpha_k d_y^k > 0$$

Primeiramente note que se $d_y^k \geq 0$ (só possui componentes não negativas), então α_k pode ser qualquer número positivo, garantindo que a próxima solução $y^{k+1} \in \mathcal{P}_{int}$. Por outro lado, caso uma ou mais componentes de d_y^k sejam negativas, α_k deverá ser menor que

$$\frac{y_i^k}{-(d_y^k)_i} = \frac{1}{-(d_y^k)_i}$$

Portanto, podemos escolher $0 < \alpha < 1$ e aplicar o teste (menor taxa)

$$\alpha_k = \min_i \left\{ \frac{\alpha}{-(d_y^k)_i} \mid (d_y^k)_i < 0 \right\}$$

garantindo que $y^{k+1} \in \mathcal{P}_{int}$.

- Se α for próximo de 1, y^{k+1} tenderá a se aproximar das paredes do octante não negativo.

Mapeamento inverso

- Para mapear a solução y^{k+1} no espaço original para obter uma solução melhorada x^{k+1} , podemos aplicar a transformação inversa T_k^{-1} em y^{k+1} , isto é

$$\begin{aligned}
 x^{k+1} &= T_k^{-1} y^{k+1} = X_k y^{k+1} \\
 &= x^k + \alpha_k X_k d_y^k \\
 &= x^k - \alpha_k X_k P_k X_k c \\
 &= x^k - \alpha_k X_k \left(I - X_k A^T (A X_k^2 A^T)^{-1} A X_k \right) X_k c \\
 &= x^k - \alpha_k X_k^2 \left(c - A^T (A X_k^2 A^T)^{-1} A X_k^2 c \right) \\
 &= x^k - \alpha_k X_k^2 \left(c - A^T w^k \right)
 \end{aligned}$$

com $w^k = (A X_k^2 A^T)^{-1} A X_k^2 c$.

- Note que a direção de movimento no espaço original é $d_x^k = -X_k^2 (c - A^T w^k)$. Na sequência são apresentadas algumas observações importantes.

Observações

- Considerando que $d_y^k = -P_k c^k = -P_k X_k c$, $d_x^k = X_k d_y^k$ e $P = P^2$ (propriedade de toda matriz de projeção), tem-se

$$\begin{aligned}c^T x^{k+1} &= c^T (x^k + \alpha_k d_x^k) \\&= c^T x^k + \alpha_k c^T X_k d_y^k \\&= c^T x^k + \alpha_k c^T X_k (-P_k X_k c) \\&= c^T x^k - \alpha_k c^T X_k P_k P_k X_k c \\&= c^T x^k - \alpha_k \|P_k X_k c\|^2 \\&= c^T x^k - \alpha_k \|d_y^k\|^2\end{aligned}$$

Consequentemente $c^T x^{k+1} < c^T x^k$ se $d_y^k \neq 0$. Os casos $d_y^k > 0$ e $d_y^k = 0$ dão origem aos próximos lemas.

Lema 1

Se existir um $x^k \in \mathcal{P}_{int}$ que fornece $d_y^k > 0$ (todas componentes positivas), então o PL (1) é *ilimitado*.

Observações

Lema 2

Se existir um $x^k \in \mathcal{P}_{int}$ que fornece $d_y^k = 0$, então *toda solução* factível do PL (1) é *ótima*.

Prova: Se $d_y^k = -P_k X_k c = 0$ então $X_k c$ pertence ao complemento ortogonal de $\mathcal{N}(AX_k)$, isto é, $X_k c$ pertence ao range de $(AX_k)^T$. Assim, $\exists u^k$ tal que $(AX_k)^T u^k = X_k c$ ou $u^{kT} AX_k = c^T X_k$ que implica que $u^{kT} A = c^T$. Assim, para qualquer x factível, tem-se que $c^T x = u^{kT} Ax = u^{kT} b$, garantindo que o valor de $c^T x$ é sempre constante, e portanto qualquer x^k é ótimo.

Lema 3

Se o PL (1) é limitado inferiormente e sua função objetivo não é constante, então a sequência $\{c^T x^k, k = 1, 2, \dots\}$ é bem definida e estritamente decrescente.

Relembrando I

- Seja o par simétrico primal-dual

$$\begin{array}{ll} \min_x & c^T x \\ \text{s.a} & Ax \geq b, \\ & x_i \geq 0 \end{array}, \quad \begin{array}{ll} \max_w & b^T w \\ \text{s.a} & A^T w \leq c \\ & w_i \geq 0 \end{array} \quad (4)$$

e os vetores (de folga) $s = Ax - b \geq 0$ e $r = c - A^T w \geq 0$. Para x e w factíveis tem-se

$$\begin{aligned} 0 &\leq r^T x + s^T w \\ &= (c^T - w^T A)x + w^T (Ax - b) \\ &= c^T x - b^T w \end{aligned}$$

e a quantidade $c^T x - b^T w$ é o **gap de dualidade**, que vai a zero se e somente se $r^T x = 0$ e $s^T w = 0$.

Relembrando II

Teorema 1 (Folgas complementares)

Sejam x e w soluções factíveis dos PLs dados em (4). Então x e w são soluções ótimas se e somente se

$$\underbrace{(c^T - w^T A)_j}_{r_j} x_j = 0, \quad j = 1, \dots, n \quad \text{e} \quad w_i \underbrace{(Ax - b)_i}_{s_i} = 0, \quad i = 1, \dots, m$$

- Observação: Note que em $r_j x_j = 0$, ou $x_j = 0$ (não está na base do primal) ou $r_j = 0$ (restrição dual atendida na igualdade). De modo similar, em $w_i s_i = 0$, ou $w_i = 0$ ou $s_i = 0$ (restrição primal atendida na igualdade).
- Note que no PL (1), a restrição do primal é de igualdade ($Ax = b$), garantindo que $w_i s_i = 0$. Portanto, o *gap* de dualidade para o método de ponto interior em desenvolvimento é apenas $r^T x = 0$.
- Com essas observações, é possível enunciar as condições de otimalidade de Karush-Kuhn-Tucker (KKT) para o caso de programação linear.

Relembrando III

Teorema 2 (Karush-Kuhn-Tucker (KKT))

O ponto x é a solução ótima do PL (1) se e somente se existirem vetores w e r tais que

- $Ax = b, x \geq 0$ (*factibilidade primal*)
- $A^T w + r = c, r \geq 0$ (*factibilidade dual*)
- $r^T x = 0$, (*folga complementar*)

Estimativas duais I

■ Chamaremos

$$w^k = (AX_k^2 A^T)^{-1} AX_k^2 c$$

como o vetor de **estimativas duais** correspondente à solução primal x^k , e além disso

$$r^k = c - A^T w^k$$

como o vetor de **custos reduzidos** associados a x^k .

Se $r^k \geq 0$, então w^k torna-se uma solução dual factível e $x^{kT} r^k = e^T X_k r^k$ torna-se o *gap* de dualidade do par (x^k, w^k) . Quando $e^T X_k r^k = 0$ com $r^k \geq 0$ então obteve-se uma solução primal factível em x^k , dual factível em w^k e a condição das folgas complementares foi satisfeita, isto é, **otimalidade**.

Critério de parada

- As três condições presentes no teorema de KKT podem ser usadas como critérios de parada. Considerando um ambiente numérico não ideal de implementação, algumas relaxações podem ser aplicadas.

- Para testar a factibilidade primal, podemos considerar

$$\sigma_p = \frac{\|Ax^k - b\|}{\|b\| + 1}, \quad x^k \geq 0$$

A soma de 1 no denominador ajuda na estabilidade numérica.

- Para testar a factibilidade dual (não negatividade dos custos reduzidos), podemos considerar

$$\sigma_d = \frac{\|r^k\|}{\|c\| + 1}$$

sendo que $\|r^k\|$ e $\|c\|$ são calculados apenas para as componentes i tais que $r_i^k < 0$.

- Para testar folga complementar, podemos considerar

$$\sigma_c = c^T x^k - b^T w^k$$

Algoritmo

Algorithm 1 Algoritmo primal afim escala

- 1: Sejam A , b e c os dados de entrada.
 - 2: Encontre uma solução inicial $x^0 \in \mathcal{P}_{int}$ e faça $k = 0$.
 - 3: Especifique uma precisão ε e um limitador de passo $0 < \alpha < 1$
 - 4: **enquanto** não convergir **faça**
 - 5: Compute as estimativas duais $w^k = (AX_k^2 A^T)^{-1} AX_k^2 c$;
 - 6: Compute os custos reduzidos: $r^k = c - A^T w^k$;
 - 7: **se** $\sigma_d \leq \varepsilon$ e $\sigma_c \leq \varepsilon$ **então**
 - 8: Pare, encontrou o ótimo;
 - 9: **fim se**
 - 10: Compute a direção $d_y^k = -X_k r^k$
 - 11: **se** $d_y^k > 0$ ou $d_y^k = 0$ **então**
 - 12: Pare, a solução é ilimitada ou x^k é ótima e existem múltiplas soluções;
 - 13: **fim se**
 - 14: Determine o tamanho do passo $\alpha_k = \min_i \left\{ \frac{\alpha}{-(d_y^k)_i} \mid (d_y^k)_i < 0 \right\}$,
 - 15: Atualize a solução: $x^{k+1} = x^k + \alpha_k X_k d_y^k$,
 - 16: $k = k + 1$
 - 17: **fim enquanto**
-

Observações

- A cada iteração, o **gargalo** (maior fonte de esforço computacional) é o cômputo de w^k (inversão de uma matriz).
- Note que a factibilidade primal não é verificada no algoritmo, sendo implicitamente considerada como sempre válida por construção. Contudo, imprecisões numéricas (truncamentos e arredondamentos) podem ocasionar a perda de factibilidade. Portanto, é recomendável que a mesma seja testada (não necessariamente em toda iteração), e caso necessário, a factibilidade pode ser restaurada pela Fase-I.
- É comum observar o seguinte comportamento para o valor da função objetivo: decresce rapidamente nas primeiras iterações e desacelera nas proximidades do ótimo.
- O algoritmo **pode** mostrar sensibilidade (baixo desempenho) à degeneração do primal.
- Em um experimento numérico realizado em [11], foram reportadas as seguintes complexidades (levantadas com base em 137 problemas) para o número de iterações para os métodos Simplex e Afim escala, $0.7159m^{0.9522}n^{0.3109}$ e $7.3385m^{-0.01872}n^{0.1694}$, respectivamente.

Exemplo

- Frannie vende 3 cordas de lenha (1 corda equivale a $3.5m^3$) todo final de ano. Pode vender meia corda a R\$ 90 ou uma corda a R\$ 150. Como maximizar o lucro?

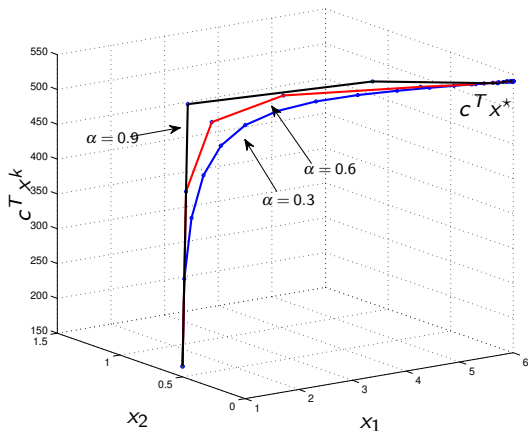
$$\text{Modelo : } \left\{ \begin{array}{l} \max \quad 90x_1 + 150x_2 \\ \text{s.a} \quad \frac{1}{2}x_1 + x_2 \leq 3 \\ \quad \quad x_1, x_2 \geq 0 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \min \quad -90x_1 - 150x_2 \\ \text{s.a} \quad \frac{1}{2}x_1 + x_2 + x_3 = 3 \\ \quad \quad x_1, x_2, x_3 \geq 0 \end{array} \right.$$

$x_3 =$ variável de folga, $(1, \frac{1}{2}, 2) =$ ponto inicial interior factível. Resultado para $\varepsilon = 10^{-3}$ e $\alpha = 0.95$

k	x_1	x_2	x_3	σ_c	σ_d	$c^T x^k$
0	1.00000	0.50000	2.00000	-110.00000	8.782e-01	-165.00000
1	3.09432	1.35284	0.10000	18.40552	7.359e-02	-481.41477
2	5.80562	0.06764	0.02955	7.24340	1.919e-04	-532.65184
3	5.91075	0.04315	0.00148	1.54111	3.538e-05	-538.43957
4	5.99331	0.00216	0.00119	0.27862	2.412e-07	-539.72125
5	5.99681	0.00154	0.00006	0.05681	4.374e-08	-539.94317
6	5.99975	0.00008	0.00005	0.01065	3.444e-10	-539.98935
7	5.99988	0.00006	0.00000	0.00212	5.945e-11	-539.99788
8	5.99999	0.00000	0.00000	0.00040	4.944e-13	-539.99960

Exemplo

- A figura mostrada a seguir ilustra a evolução da solução para três diferentes situações: $\alpha = 0.3$, $\alpha = 0.6$ e $\alpha = 0.9$. O número de iterações foi 38, 15 e 10 respectivamente.



Convergência e complexidade

- Embora o algoritmo afim escala seja mais fácil e intuitivo de se programar, sua análise é mais complexa do que, por exemplo, o algoritmo de Karmarkar.
- O próximo teorema apresenta algumas propriedades sobre a convergência do algoritmo ([9, 5, 11, 10]).

Teorema 3

- *Se ambos os problemas primal e dual são não degenerados, então para qualquer $\alpha < 1$, a sequência gerada pelo algoritmo converge para a solução ótima.*
 - *Para $\alpha \leq \frac{2}{3}$, a sequência gerada pelo algoritmo converge para uma solução ótima (degenerado ou não).*
 - *Existe um exemplo com um $\alpha < 1$ associado para o qual o algoritmo converge para uma solução não ótima.*
- Para uma tolerância ε especificada, até o presente momento não foi apresentada uma prova que demonstra que o algoritmo afim escala fornece a solução ótima (dentro dessa tolerância) em um número de iterações que é limitada polinomialmente em n (existem evidências contrárias a esse fato, podendo no pior caso também apresentar complexidade exponencial, como o Simplex).

Solução inicial via Big-M

- Assim como no método Simplex, uma alternativa para encontrar uma solução inicial factível é usar o método do Big-M.
- A técnica consiste em adicionar uma variável artificial x^a associada a um número positivo grande M no problema original (no caso, no PL (1)) de modo que $(x, x_a) = (1, 1, \dots, 1, 1)$ ($n+1$ componentes) torne-se um ponto interior factível para o seguinte PL

$$\begin{aligned} \min_{x, x^a} \quad & c^T x + Mx_a \\ \text{s.a} \quad & [A \quad b - Ae] \begin{bmatrix} x \\ x_a \end{bmatrix} = b \\ & x^a, x_i \geq 0 \end{aligned} \tag{5}$$

- Note que o número total de variáveis é $n+1$ (no Big-M para o Simplex é $n+m$).
- Ao aplicar o algoritmo primal escala no PL (5) (garantidamente factível), ou chega-se em uma solução ótima ou conclui-se que o problema é ilimitado.
- Similarmente ao método Simplex, se ao terminar a otimização x_a permanece positiva, então o PL original é infactível.

Exemplo

- Aplicando o método do Big-M no exemplo de Frannie (já na forma padrão), temos a formulação

$$\text{Big-M : } \begin{cases} \min & -90x_1 - 150x_2 + Mx_a \\ \text{s.a} & \frac{1}{2}x_1 + x_2 + x_3 + 0.5x_a = 3 \\ & x_i, x_a \geq 0 \end{cases}$$

- Adotando $\alpha = 0.95$, $\varepsilon = 10^{-3}$ e $M = 1000$, tem-se a seguinte solução

k	x_1	x_2	x_3	x_a	σ_c	σ_d	$c^T x^k$
0	1.0000	1.0000	1.0000	1.00000	394.0	1.899e+00	760.0
1	1.1527	1.2751	1.1234	0.05000	36.68127	4.019e-01	-245.0271
2	1.6679	2.0966	0.0561	0.02643	23.60498	1.432e-01	-438.1813
3	2.8783	1.5202	0.0399	0.00132	6.59636	8.726e-02	-485.7596
4	5.7900	0.0760	0.0283	0.00124	8.62056	2.089e-04	-531.2653
5	5.9086	0.0437	0.0014	0.00093	2.55931	3.673e-05	-537.4206
6	5.9930	0.0021	0.0011	0.00025	0.54564	2.560e-07	-539.4542
7	5.9959	0.0016	0.0003	0.00001	0.12099	6.322e-08	-539.8789
8	5.9994	0.0002	0.0000	0.00001	0.02113	1.290e-09	-539.9788
9	5.9998	0.0000	0.0000	0.00000	0.00494	1.235e-10	-539.9950
10	5.9999	0.0000	0.0000	0.00000	0.00080	5.764e-13	-539.9992

Solução inicial via Fase-I

- Inicialmente escolhe-se um ponto arbitrário $x^0 > 0$ e calcula-se $v = b - Ax^0$. Se $v = 0$, então x^0 é um ponto interior inicial factível. Caso contrário considera-se o seguinte PL de Fase-I com $n + 1$ variáveis

$$\begin{aligned} \min_{x,u} \quad & u \\ \text{s.a} \quad & [A \quad v] \begin{bmatrix} x \\ u \end{bmatrix} = b \\ & u, x_i \geq 0 \end{aligned} \tag{6}$$

- É imediato verificar que

$$\hat{x}^0 = \frac{x^0}{u^0} = \frac{x^0}{1} > 0$$

é um ponto interior factível para o PL (6).

- Como o PL (6) é limitado inferiormente por 0, o algoritmo primal escala terminará com uma solução ótima (x^*, u^*) . Se $u^* > 0$, então o PL original é infactível. Caso contrário x^* é uma solução inicial factível para o PL original.
- Explorando as estruturas das matrizes do PL (6), é possível adaptar o algoritmo afim escala para trabalhar no espaço de dimensão n , fornecendo vantagens. Por exemplo, torna o código menos suscetível a imprecisões numéricas.

Exemplo

- Aplicando o método da Fase-I no exemplo de Frannie (já na forma padrão) e considerando $x^0 = (1, 1, 1)^T$ (portanto $v = b - Ax^0 = 0.5$), temos a formulação

$$\text{Fase-I : } \begin{cases} \min_u & u \\ \text{s.a} & \frac{1}{2}x_1 + x_2 + x_3 + 0.5u = 3 \\ & x_i, u \geq 0 \end{cases}$$

- Adotando $\alpha = 0.95$, $\varepsilon = 10^{-3}$, tem-se a seguinte solução

k	x_1	u	σ_p	σ_d	σ_c	$c^T x^k$
0	1.00000	1.00000	0.00e+00	3.00000e-01	0.40000	1.000000
1	1.10556	0.05000	1.11e-16	5.78745e-04	0.04884	0.050000
2	1.11004	0.00250	1.11e-16	1.42306e-06	0.00250	0.002500
3	1.11026	0.00012	1.11e-16	3.55468e-09	0.00012	0.000125
Fim Fase-I / Início Fase-II						
0	1.11026		1.56e-05	4.58766e-01	28.83846	-283.283887
1	1.64781		1.56e-05	1.43728e-01	4.01540	-465.539772
2	5.23591		1.56e-05	3.38377e-03	10.08160	-528.070861
3	5.95616		1.56e-05	7.61757e-06	1.09983	-538.896007
9	5.99987		1.56e-05	9.21363e-15	0.01128	-539.988724
10	5.99987		1.56e-05	0.00000e+00	0.01126	-539.988743
11	5.99987		1.56e-05	0.00000e+00	0.01126	-539.988742
100	5.99987		1.56e-05	0.00000e+00	0.01126	-539.988738

Exemplo - continuação

- O algoritmo não convergiu e foi terminado em função do número máximo de iterações. Note que ao iniciar-se a fase-II, com a exclusão da variável artificial u , a factibilidade primal foi comprometida, ficando com um erro da ordem de 10^{-5} .
- Executando o algoritmo novamente, mas considerando $\varepsilon = 10^{-6}$ na fase-I (mantendo $\varepsilon = 10^{-3}$ na fase-II), obteve-se o resultado (convergente)

k	x_1	u	σ_p	σ_d	σ_c	$c^T x^k$
0	1.00000	1.00000	0.00e+00	3.00000e-01	0.40000	1.000000
1	1.10556	0.05000	1.11e-16	5.78745e-04	0.04884	0.050000
2	1.11004	0.00250	1.11e-16	1.42306e-06	0.00250	0.002500
3	1.11026	0.00012	1.11e-16	3.55468e-09	0.00012	0.000125
4	1.11027	0.00001	1.11e-16	8.88632e-12	0.00001	0.000006
5	1.11027	0.00000	1.11e-16	2.22158e-14	0.00000	0.000000
Fim Fase-I / Início Fase-II						
0	1.11027		3.91e-08	4.58767e-01	28.84443	-283.289174
1	1.64783		3.91e-08	1.43729e-01	4.02510	-465.549110
2	5.23592		3.91e-08	3.38479e-03	10.07160	-528.080310
3	5.95627		3.91e-08	7.61939e-06	1.08871	-538.907125
4	5.99738		3.91e-08	3.07334e-08	0.09329	-539.906694
5	5.99886		3.91e-08	5.63820e-09	0.01986	-539.980140
6	5.99992		3.91e-08	3.76502e-11	0.00350	-539.996502
7	5.99996		3.91e-08	6.95083e-12	0.00074	-539.999258

“Empurrão potencial”

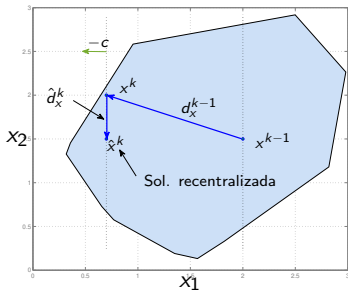
■ Uma maneira de permanecer longe das “paredes” do octante não negativo é por meio do “empurrão potencial”. A técnica consiste em “empurrar” a solução atual para longe da fronteira, mantendo o mesmo valor da função objetivo. Primeiramente define-se a função potencial ($x > 0$)

$$p(x) = - \sum_{i=1}^n \log(x_i)$$

que torna-se grande à medida que x se aproxima da fronteira ($x_i \rightarrow 0$). Portanto a minimização de $p(x)$ deve aplicar um “empurrão” em x^k para longe da fronteira. Matematicamente temos o seguinte problema de otimização

$$\begin{aligned} \min_x \quad & p(x) \\ \text{s.a} \quad & Ax = b \\ & c^T x = c^T x^k \\ & x_i \geq 0 \end{aligned}$$

A restrição $c^T x = c^T x^k$ garante que o valor da função objetivo permanecerá igual



Barreira logarítmica I

- Um outra alternativa para permanecer longe das fronteiras do octante não negativo é por meio da incorporação de uma função de “barreira” na função objetivo, que fornece valores grandes nas proximidades da fronteira. A minimização desta função objetivo modificada automaticamente levará a solução para longe da fronteira.
- O método da **barreira logarítmica** consiste na solução do seguinte problema (não linear) de otimização

$$\begin{aligned} \min_x \quad & F_\mu(x) = c^T x - \mu \sum_{i=1}^n \log(x_i) \\ \text{s.a} \quad & Ax = b \\ & x_i \geq 0, \mu > 0 \end{aligned}$$

- Se $x^*(\mu)$ é uma solução ótima para este problema, e $x^*(\mu)$ tende para um valor x^* à medida que μ se aproxima de zero, então x^* é uma solução ótima para o problema original.

Barreira logarítmica II

- Uma direção de busca pode ser determinada por meio do seguinte problema de otimização quadrático

$$\begin{array}{ll} \min_x & \frac{1}{2}d^T \nabla^2 F_\mu(x)d + (\nabla F_\mu(x))^T d \\ \text{s.a} & Ad = 0 \end{array}$$

sendo $\nabla F_\mu(x) = c - \mu X^{-1}$ e $\nabla^2 F_\mu(x) = \mu X^{-2}$. Ou seja, direção de busca (direção de Newton) pertence ao espaço nulo de A e minimiza a aproximação quadrática de $F_\mu(x)$. A solução ótima é dada por

$$d_\mu = \frac{1}{\mu} d_x^k + X_k \left(I - X_k A^T (A X_k^2 A^T)^{-1} A X_k \right) e$$

A segunda componente pode ser vista como uma “força” que move a solução para longe da fronteira. Ela também pode ser chamada de “força centralizadora”, e o algoritmo de “primal afim escala com força centralizadora”.

- Com a barreira, é possível mostrar que o algoritmo termina com o número de iterações limitado por uma curva polinomial [2, 7, 6].

Dual Afim Escala

- Considere o problema dual associado ao PL original dado em (1)

$$\begin{aligned} \max_w \quad & b^T w \\ \text{s.a} \quad & A^T w + s = c \\ & s_i \geq 0 \end{aligned} \tag{7}$$

- Similarmente ao dual Simplex, o algoritmo dual afim escala parte de uma solução dual factível e caminha em direção à otimalidade aumentando progressivamente a função objetivo e mantendo a factibilidade.
- O algoritmo pode ser concebido do seguinte modo: A partir de uma solução inicial factível interior (w^k, s^k) , isto é, que satisfaz

$$A^T w^k + s^k = c, \quad s^k > 0$$

encontrar as direções (d_w^k, d_s^k) e um tamanho de passo β_k tais que

$$\begin{aligned} w^{k+1} &= w^k + \beta_k d_w^k \\ s^{k+1} &= s^k + \beta_k d_s^k \end{aligned}$$

continuam sendo soluções interiores e $b^T w^{k+1} \geq b^T w^k$ (função objetivo melhora).

Transformação

- Com relação à manutenção da factibilidade de $s^{k+1} > 0$, pode-se usar a mesma estratégia utilizada no algoritmo primal, isto é, aplicando uma transformação, “centralizando” s^k em e . A matriz de transformação é dada por

$$S_k = \text{diag}(s^k) = \begin{bmatrix} s_1^k & 0 & \dots & 0 \\ 0 & s_2^k & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & s_n^k \end{bmatrix}, \quad S_k^{-1} s^k = e$$

Adotaremos $u = S^{-1}s$ para denotar a variável s transformada ($s = S_k u$). Portanto, se d_u^k é uma direção de melhoria no espaço transformado, então a direção correspondente no espaço original é $d_s^k = S_k d_u^k$.

- Com essa transformação a distância para as paredes do octante não negativo é conhecida, e qualquer movimento de distância menor que um preserva a factibilidade.
- Note que a variável w é irrestrita, não necessitando de um tratamento similar.

Direção

- Usando as definições de w^{k+1} e s^{k+1} , a garantia da factibilidade de

$$A^T w^{k+1} + s^{k+1} = c$$

pode ser realizada por meio da imposição

$$A^T d_w^k + d_s^k = 0$$

- No espaço transformado, tem-se

$$A^T d_w^k + d_s^k = 0 \Rightarrow A^T d_w^k + S_k d_u^k = 0 \Rightarrow S_k d_u^k = -A^T d_w^k \Rightarrow d_u^k = -S_k^{-1} A^T d_w^k$$

Com mais alguma manipulações e lembrando que A tem rank completo de linhas, chega-se a

$$d_w^k = -(AS_k^{-2}A^T)^{-1}AS_k^{-1}d_u^k$$

Definindo $Q_k = (AS_k^{-2}A^T)^{-1}AS_k^{-1}$, tem-se $d_w^k = -Q_k d_u^k$

Melhoria

- Para garantir $b^T w^{k+1} \geq b^T w^k$, é necessário que

$$b^T w^{k+1} = b^T w^k + \beta_k b^T d_w^k \Rightarrow \beta_k b^T d_w^k \geq 0$$

- Considere $d_u^k = -Q_k^T b$, que fornece

$$b^T d_w^k = -b^T Q_k d_u^k = b^T Q_k Q_k^T b = \|b^T Q_k\|^2 \geq 0$$

Portanto, temos que as seguintes direções de busca em w e em s (note que $d_s^k = -A^T d_w^k$) são dadas por

$$d_w^k = (AS_k^{-2}A^T)^{-1}b, \quad d_s^k = -A^T(AS_k^{-2}A^T)^{-1}b$$

Tamanho do passo

- Com as direções de caminhada (factíveis e de melhoria) $(d_w^k; d_s^k)$ em mãos, o tamanho do passo β_k é definido pela positividade de s^{k+1} . Assim temos
 - Se $d_s^k = 0$, então o problema dual tem uma função objetivo constante em sua região factível e toda solução factível é ótima.
 - Se $d_s^k \geq 0$ (mas $\neq 0$), então o problema é ilimitado.
 - Caso contrário:

$$\beta_k = \min_i \left\{ \alpha \frac{s_i^k}{-(d_s^k)_i} \mid (d_s^k)_i < 0 \right\}, \quad 0 < \alpha < 1$$

- Similarmente ao caso primal afim escala, se definirmos

$$x^k = -S_k^{-2} d_s^k \Rightarrow Ax^k = AS_k^{-2} A^T d_w^k = b$$

Portanto, x^k pode ser visto como uma “estimativa primal” no algoritmo dual afim escala. Assim, se $x^k \geq 0$, então ele é uma solução factível para o primal, com o *gap* de dualidade dado por $c^T x^k - b^T w^k$. Se o *gap* for zero, então $(w^k; s^k)$ e x^k são soluções duais e primais ótimas, respectivamente.

Critério de parada

- As condições de KKT também podem ser utilizadas para testar a otimalidade do problema. Note que a factibilidade dual é mantida durante a execução do algoritmo, e apenas a factibilidade primal e a folga complementar precisam ser testadas.
- Como observado anteriormente, a restrição $Ax = b$ está automaticamente satisfeita para qualquer

$$x^k = S_k^{-2} A^T (A S_k^{-2} A^T)^{-1} b$$

Portanto, se $x \geq 0$, então ela é primal factível. Uma observação interessante é que se convertermos o PL dual (7) em um PL na forma padrão e aplicarmos o método primal afim escala, a expressão acima será uma estimativa dual para o problema.

- A folga complementar pode ser testada impondo que $\sigma_c = c^T x^k - b^T w^k$ seja menor que uma certa quantidade especificada *a priori*.

Algoritmo

Algorithm 2 Algoritmo dual afim escala

- 1: Sejam A , b e c os dados de entrada.
 - 2: Encontre uma solução inicial $(w^0; s^0)$ interior e faça $k = 0$.
 - 3: Especifique uma precisão ε e um limitador de passo $0 < \alpha < 1$
 - 4: **enquanto** não convergir **faça**
 - 5: Compute as direções $d_w^k = (AS_k^{-2}A^T)^{-1}b$ e $d_s^k = -A^T d_w^k$
 - 6: **se** $d_s^k \geq 0$ ou $d_s^k = 0$ **então**
 - 7: Pare, o problema dual é ilimitado ou $(w^k; s^k)$ é ótima e existem múltiplas soluções;
 - 8: **fim se**
 - 9: Compute a estimativa primal: $x^k = -S_k^{-2}d_s^k$
 - 10: **se** $x^k \geq 0$ e $\sigma_c \leq \varepsilon$ **então**
 - 11: Pare, encontrou o ótimo;
 - 12: **fim se**
 - 13: Determine o tamanho do passo $\beta_k = \min_i \left\{ \alpha \frac{s_i^k}{-(d_s^k)_i} \mid (d_s^k)_i < 0 \right\}$
 - 14: Atualize a solução: $(w^{k+1}; s^{k+1}) = (w^k; s^k) + \beta_k (d_w^k; d_s^k)$;
 - 15: $k = k + 1$
 - 16: **fim enquanto**
-

Exemplo

- Seja o PL dual para o problema de Frannie

$$\text{Primal: } \begin{cases} \min & -90x_1 - 150x_2 \\ \text{s.a} & \frac{1}{2}x_1 + x_2 + x_3 = 3 \\ & x_1, x_2, x_3 \geq 0 \end{cases}, \quad \text{Dual: } \begin{cases} \max & 3w_1 \\ \text{s.a} & \frac{1}{2}w_1 + s_1 = -90 \\ & w_1 + s_2 = -150 \\ & w_1 + s_3 = 0 \\ & s_1, s_2, s_3 \geq 0 \end{cases}$$

- Seja o ponto inicial interior factível $(w^0, s^0)^T = (-250, 35, 100, 250)^T$. O resultado para $\varepsilon = 10^{-3}$ e $\alpha = 0.95$ é:

k	s_1	s_2	s_3	w_1	$\min(x_k)$	σ_c	$b^T w^k$
0	35.000	100.0	250.000	-250.000	1.50e-01	2.65e+02	-750.0000
1	1.750	33.500	183.500	-183.500	1.08e-03	1.17e+01	-550.5000
2	0.088	30.175	180.175	-180.175	2.83e-06	5.29e-01	-540.5250
3	0.004	30.009	180.009	-180.009	7.09e-09	2.63e-02	-540.0262
4	0.000	30.000	180.000	-180.000	1.77e-11	1.31e-03	-540.0013
5	0.000	30.000	180.000	-180.000	4.43e-14	6.56e-05	-540.0000

Observações

- Observando que o algoritmo dual afim escala é equivalente ao primal afim escala aplicado ao problema dual, propriedades de convergência similares podem ser obtidas.
- O gargalo do esforço computacional é o mesmo, mais precisamente, na inversão $(AS_k^{-2}A^T)^{-1}$ (no primal era $(AX_k^{-2}A^T)^{-1}$).
- O algoritmo dual possui uma tendência geral em convergir mais rápido que o primal.
- O problema de perda de factibilidade é menos sério no dual do que no primal. Mesmo que no cômputo de d_w^k ocorram erros de truncamento e arredondamento (principalmente na inversa de $AX_k^{-2}A^T$), a direção $d_s^k = -A^T d_w^k$ ainda pode ser factível.
- O algoritmo dual é menos sensível à degeneração do primal, mas ainda sensível à degeneração do dual.
- O algoritmo dual melhora rapidamente a função objetivo. Contudo, a factibilidade primal é lenta.

Solução inicial via Big-M

■ No caso especial de $c > 0$, então temos que $w^0 = 0$ e $s^0 = c$ é uma solução interior factível inicial. Caso contrário podemos aplicar o método do Big-M usando uma variável artificial w_a e um custo associado de elevado valor M , gerando a formulação dada em (8) e com o vetor $p \in \mathbb{R}^n$ definido em (9).

$$\begin{array}{ll} \max_{w, w_a} & b^T w + M w_a \\ \text{s.a} & A^T w + p w_a + s = c \\ & s \geq 0 \end{array} \quad (8)$$

$$p_i = \begin{cases} 1 & \text{se } c_i \leq 0 \\ 0 & \text{se } c_i > 0 \end{cases} \quad (9)$$

■ Computando $\bar{c} = \max_i |c_i|$, escolhendo $\theta > 1$, é possível construir a seguinte solução inicial factível para o PL (8)

$$(w^0, w_a^0, s^0)^T = (0, \dots, 0, -\theta \bar{c}, c + \theta \bar{c} p)^T$$

■ Note que w_a começa com $-\theta \bar{c} < 0$ e é forçado a aumentar de valor no procedimento iterativo pois M é grande. Em algum momento é esperado que o valor de w_a torne-se não negativo, caso contrário o PL original é infactível.

■ Quando w_a estiver muito próximo ou cruzar o zero, toma-se $\bar{w} = w^k$ e $\bar{s} = s^k + p w_a^k$ como soluções iniciais para o problema original.

Exemplo

- Aplicando o Big-M na forma dual do problema de Frannie (já na forma padrão), tem-se

$$\text{Frannie Dual Big-M : } \begin{cases} \max & 3w_1 + Mw_a \\ \text{s.a} & \frac{1}{2}w_1 + w_a + s_1 = -90 \\ & w_1 + w_a + s_2 = -150 \\ & w_1 + w_a + s_3 = 0 \\ & s_1, s_2, s_3 \geq 0 \end{cases}$$

- Escolhendo $\theta = 2$, $\varepsilon = 10^{-3}$, $\alpha = 0.95$, $M = 100$, tem-se o ponto inicial factível $(w^0, w_a^0, s^0)^T = (0, -300, 210, 150, 300)^T$. O resultado é:

k	s_1	s_3	w_1	w_a	$\min(x_k)$	σ_c	$b^T w^k$
0	210.000	300.000	0.000	-300.0	-7.5e+01	2.3e+04	-30000.000
1	10.500	339.455	-477.910	138.4	-7.1e+01	1.9e+04	12411.777
Fim Big-M / Início Fase-II							
0	148.955	477.910	-477.910		5.2e-01	1.0e+03	-1433.7304
1	7.448	194.896	-194.896		1.5e-02	5.6e+01	-584.6865
2	0.372	180.745	-180.745		5.0e-05	2.3e+00	-542.2343
3	0.019	180.037	-180.037		1.2e-07	1.1e-01	-540.1117
4	0.001	180.002	-180.002		3.2e-10	5.5e-03	-540.0055
5	0.000	180.000	-180.000		8.0e-13	2.7e-04	-540.0002

Barreira logarítmica

■ Assim como no caso primal, o algoritmo dual afim escala também pode ser modificado com a inserção de uma barreira logarítmica, que fornecerá valores altos nas proximidades das fronteiras, e conseqüentemente “empurrando” a solução para longe da fronteira.

■ Adotando um desenvolvimento similar ao caso primal, a direção de busca resultante é

$$d_\mu = \frac{1}{\mu}(AS_k^{-2}A^T)^{-1}b - (AS_k^{-2}A^T)^{-1}AS_k^{-1}e$$

■ Note que $d_w^k = (AS_k^{-2}A^T)^{-1}b$ e a segunda parcela é o termo adicional que produz a “força” que leva a solução para longe da fronteira. O método dual afim escala com essa adaptação é conhecido como “dual afim escala com força centralizadora”.

■ Com essa modificação, é possível mostrar que, com a escolha apropriada do valor de μ e do tamanho do passo, o algoritmo termina em um número de iterações limitado por uma curva polinomial [7].

Algoritmo Primal-Dual – Concepção

- Considere o PL primal dado em (1) com a seguinte modificação na função objetivo

$$\begin{array}{ll} \min_x & -\mu \sum_{i=1}^n \log_e(x_i) \\ \text{s.a} & Ax = b \end{array} \quad (10)$$

- Considerando uma região factível poliedral, o PL (10) tenderá a produzir uma solução próxima do centro do poliedro, pois soluções próximas das fronteiras (onde sempre algumas variáveis x_i se aproximam de zero) “penalizam” a função objetivo (chamada de barreira logarítmica).
- “Mesclando” a função objetivo original com a função objetivo de (10), teríamos um PL em que podemos criar um compromisso entre otimalidade ($\mu \rightarrow 0$) e “centralidade” (μ crescendo indefinidamente). O algoritmo primal-dual é concebido a partir dessa estratégia, denominando o caminho de soluções produzidas pela variação de μ como caminho central.
- O caminho central fornece uma espécie de “caminho seguro” para a otimalidade, pois impede uma aproximação rápida das fronteiras da região factível. Os métodos de pontos interiores que utilizam essa técnica são conhecidos como métodos “seguidores de caminho” (em inglês, *path-following*).

Algoritmo Primal-Dual – Preliminares I

- Considere os PLs primal e dual dados em (1) e (7), respectivamente.
- O desenvolvimento do algoritmo primal-dual requer as seguintes hipóteses
 - H1 O conjunto $\mathcal{S} = \{x \in \mathbb{R}^n \mid Ax = b, x > 0\}$ é não vazio
 - H2 O conjunto $\mathcal{T} = \{(w; s) \in \mathbb{R}^m \times \mathbb{R}^n \mid A^T w + s = c, s > 0\}$ é não vazio.
 - H3 A matriz A tem rank completo de linhas.
- Considere agora o PL primal (1) com a barreira logarítmica (e levando em conta $x > 0$)

$$\begin{aligned} \min \quad & c^T x - \mu \sum_{i=1}^n \log_e(x_i) \\ \text{s.a} \quad & Ax = b \\ & x > 0 \end{aligned} \tag{11}$$

e o problema dual (7) também com a barreira logarítmica (também levando em conta $s > 0$)

$$\begin{aligned} \max \quad & b^T w + \mu \sum_{i=1}^n \log_e(s_i) \\ \text{s.a} \quad & A^T w + s = c \\ & s > 0 \end{aligned} \tag{12}$$

Algoritmo Primal-Dual – Preliminares II

- As soluções ótimas dos PLs (11) e (12), parametrizadas em μ , podem ser caracterizadas pelas condições de otimalidade de KKT

$$Ax = b \quad x > 0 \quad (\text{factibilidade primal}) \quad (13a)$$

$$A^T w + s = c \quad s > 0 \quad (\text{factibilidade dual}) \quad (13b)$$

$$XSe - \mu e = 0 \quad (\text{folga complementar}) \quad (13c)$$

com $S = \text{diag}(s)$ e $X = \text{diag}(x)$. Além disso, utilizando as hipóteses estabelecidas, é possível mostrar que, à medida que $\mu \rightarrow 0$, a solução $x(\mu)$ de (11) tende para a solução ótima x^* de (1), e a solução $(w(\mu); s(\mu))$ de (12) tende para a solução ótima $(w^*; s^*)$ de (7).

Considerando $\mu > 0$, seja

$$\Gamma = \{(x(\mu); w(\mu); s(\mu)) \mid (x(\mu); w(\mu); s(\mu)) \text{ resolve (13) para algum } \mu > 0\}$$

um conjunto que denota o *caminho* (em inglês, *path*) das soluções de (13). À medida que $\mu \rightarrow 0$, o caminho Γ leva às soluções ótimas primais duais $(x^*; w^*; s^*)$.

Algoritmo Primal-Dual – Preliminares III

- De modo resumido, o algoritmo primal-dual parte de uma solução inicial $(x^0; w^0; s^0) \in \mathcal{S} \times \mathcal{T}$ e produz uma sequência de pontos $(x^k; w^k; s^k)$ escolhendo adequadamente uma **direção de movimento** $(d_x^k; d_w^k; d_s^k)$ e um **tamanho de passo** β^k a cada iteração $(x^k; w^k; s^k)$,
- Antes de investigarmos a direção de movimento, apresentaremos (de modo resumido) o **Método de Newton**.

Método de Newton

■ O método de Newton é uma das técnicas mais utilizadas para computar as raízes de um sistema não linear de equações, consistindo basicamente em aplicar aproximações lineares iterativamente.

■ Seja $F(z)$ um mapeamento não linear $\mathbb{R}^n \rightarrow \mathbb{R}^n$ e o objetivo é encontrar um $z^* \in \mathbb{R}^n$ tal que $F(z^*) = 0$. Para um dado ponto inicial z^0 , o método consiste em encontrar uma direção Δ_z tal que $F(z^0 + \Delta_z) = 0$. Adotando uma expansão em série de Taylor para $F(z)$ (uma vez que trabalhar com a $F(z)$ original é inviável no caso geral), tem-se

$$F(z^0 + \Delta_z) \approx F(z^0) + J(z^0)\Delta_z, \quad J(z^0) = \left[\frac{\partial F_i(z)}{\partial z_j} \right]_{z=z^0}$$

sendo $J(z^0)$ a matriz **Jacobiana**. Igualando essa aproximação a zero, tem-se o **sistema linear** de equações (numa iteração k genérica)

$$J(z^k)\Delta_z = -F(z^k)$$

cuja solução permite a determinação de um nova solução $z^{k+1} = z^k + d_z^k$, com $d_z^k = \Delta_z$. Quando $J(z^k)$ é não singular e o ponto inicial z^0 é "próximo o suficiente" da solução z^* , o método de Newton converge quadraticamente para z^* .

Cômputo das direções I

- Reescrevendo as condições dadas em (13) em termos de uma função, tem-se

$$F((x^k; w^k; s^k)) = \begin{bmatrix} Ax^k - b \\ A^T w^k + s^k - c \\ X_k S_k e - \mu e \end{bmatrix}$$

A partir do cômputo do Jacobiano no ponto $(x^k; w^k; s^k)$, as direções de Newton $(d_x^k; d_w^k; d_s^k)$ podem ser determinadas por meio do seguinte sistema linear de equações

$$\underbrace{\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S_k & 0 & X_k \end{bmatrix}}_{J(\cdot)} \begin{bmatrix} d_x^k \\ d_w^k \\ d_s^k \end{bmatrix} = \begin{bmatrix} b - Ax^k \\ c - A^T w^k - s^k \\ \mu^k e - X_k S_k e \end{bmatrix}$$

Criando os termos auxiliares

$$t^k = b - Ax^k, \quad u^k = c - A^T w^k - s^k, \quad v^k = \mu^k e - X_k S_k e$$

Cômputo das direções II

é possível reescrever o sistema de equações na forma

$$Ad_x^k = t^k \quad (14a)$$

$$A^T d_w^k + d_s^k = u^k \quad (14b)$$

$$S_k d_x^k + X_k d_s^k = v^k \quad (14c)$$

- Manipulando estas expressões e definido $X_k^{-1} v^k = \mu^k X_k^{-1} e - S_k e = p^k$ tem-se as seguintes direções de Newton

$$d_x^k = X_k S_k^{-1} (p^k - d_s^k) \quad (15a)$$

$$d_w^k = (AX_k S_k^{-1} A^T)^{-1} (AX_k S_k^{-1} (u^k - p^k) + t^k) \quad (15b)$$

$$d_s^k = u^k - A^T d_w^k \quad (15c)$$

- Elaborando um pouco mais a direção d_x^k , podemos obter

$$d_x^k = \underbrace{\mu^k \hat{D}_k P_k \hat{D}_k X_k^{-1} e}_{d_{ctr}^k} \underbrace{- \hat{D}_k P_k \hat{D}_k c}_{d_{obj}^k} + \underbrace{\hat{D}_k^2 A^T (A \hat{D}_k^2 A^T)^{-1} t^k}_{d_{feas}^k}$$

Cômputo das direções III

com $\hat{D}_k^2 = X_k S_k^{-1}$ e $P_k = I - \hat{D}_k A^T (A \hat{D}_k^2 A^T)^{-1} A \hat{D}_k$ (matriz de projeção no espaço nulo de $A \hat{D}_k$)

- O termo d_{ctr}^k é chamado de **direção de centralização** (evita a proximidade da fronteira primal). O segundo termo d_{obj}^k é chamado de **direção de redução da função objetivo** (menos gradiente primal projetado). O terceiro termo d_{feas}^k é chamado de **direção de factibilidade**, pois t^k mede a factibilidade primal.
- Se x^0 estiver longe da região factível, t^0 será alto, e o esforço inicial do algoritmo é encontrar um ponto factível o mais próximo possível da trajetória central.
- Análises similares podem ser realizadas com d_w^k e d_s^k .
- Caso pontos iniciais factíveis $x^k \in \mathcal{S}$ e $(w^k; s^k) \in \mathcal{T}$ (implicando que $u^k = 0$ e $t^k = 0$) estejam disponíveis, é possível simplificar as direções de movimento, e poderíamos construir um algoritmo primal-dual que parte das factibilidades primal e dual (para mais detalhes ver [1, Seção 7.3.2]) e as mantém durante a execução

Tamanho do Passo

- Tamanho do passo: Uma vez que as direções de movimento estão calculadas, produz-se novos pontos

$$x^{k+1} = x^k + \beta_P d_x^k \quad (16a)$$

$$w^{k+1} = w^k + \beta_D d_w^k \quad (16b)$$

$$s^{k+1} = s^k + \beta_D d_s^k \quad (16c)$$

tais que $x^{k+1} > 0$ e $s^{k+1} > 0$. Os passos primal (β_P) e dual (β_D) podem ser determinados da seguinte maneira (dado $\alpha < 1$)

$$\beta_P = \frac{1}{\max\{1, -d_{x_i}^k / \alpha x_i^k\}} \quad (17a)$$

$$\beta_D = \frac{1}{\max\{1, -d_{s_i}^k / \alpha s_i^k\}} \quad (17b)$$

Ajuste da penalidade e do critério de parada

- Embora várias iterações do algoritmo sejam necessárias para que a solução convirja para a trajetória central definida por um valor fixo de μ^k , em implementações práticas em geral o valor de μ^k muda (diminui) a cada iteração pois, para que a condição de folga complementar seja atendida, ele precisa ser levado a zero.
- Um critério que pode ser adotado para reduzir o valor de μ^k é

$$\mu^k = \frac{\sigma_\mu}{n} (s^k)^T x^k, \quad \sigma_\mu < 1$$

em que σ_μ é um fator utilizado para acelerar o decréscimo de μ^k .

- Com relação aos critérios de parada, podemos usar a factibilidade primal (t^k), a factibilidade dual (u^k) e a folga complementar v^k .

Algoritmo

Algorithm 3 Algoritmo primal-dual

- 1: Sejam A , b e c os dados de entrada .
- 2: Seja uma solução inicial $(x^0; w^0; s^0)$ tal que $x^0 > 0$ e $s^0 > 0$.
- 3: Especifique as precisões ε_1 , ε_2 e ε_3 , as constante, $\alpha, \sigma_\mu < 1$ e faça $k = 0$.
- 4: **enquanto** não convergir **faça**
- 5: Compute $\mu^k = \sigma_\mu (x^k)^T s^k / n$, $t^k = b - Ax^k$, $u^k = c - A^T w^k - s^k$, $v^k = \mu^k e - X_k S_k e$, $p^k = X_k^{-1} v^k$.
- 6: Compute as violações $\sigma_p = \|t\| / (\|b\| + 1)$ e $\sigma_d = \|u\| / (\|c\| + 1)$;
- 7: **se** $\mu^k < \varepsilon_1$ e $\sigma_p < \varepsilon_2$ e $\sigma_d < \varepsilon_3$ **então**
- 8: Pare, encontrou o ótimo;
- 9: **fim se**
- 10: Compute as direções d_x^k , d_w^k e d_s^k usando (15).
- 11: **se** $(t^k = 0$ e $d_x^k > 0$ e $c^T d_x^k < 0)$ ou $(u^k = 0$ e $d_s^k > 0$ e $b^T d_w^k > 0)$ **então**
- 12: O problema primal (dual) é ilimitado; Abandone.
- 13: **fim se**
- 14: Determine os tamanhos de passo β_P e β_D usando (17).
- 15: Atualize a solução usando (16).
- 16: $k = k + 1$
- 17: **fim enquanto**

Exemplo

■ Aplicando o algoritmo primal-dual no problema de Frannie com as seguintes escolhas: $x^0 = (1, 1, 1)^T$, $w^0 = 0$, $s^0 = (1, 1, 1)^T$, $\alpha = 0.99$, $\sigma_\mu = 0.85$ e $\varepsilon = 10^{-3}$, tem-se os seguintes resultados

k	x_1	w_1	μ^k	σ_p	σ_d	$b^T w^k$	$c^T x^k$
0	1.000	0.000	8.50e-01	1.25e-01	1.00e+0	0.000	-240.0000
1	1.539	-1.360	1.27e-01	1.24e-01	9.87e-01	-4.080	-397.4498
2	3.007	-3.935	6.78e-03	1.21e-01	9.69e-01	-11.806	-422.2117
3	5.008	-4.094	2.87e-04	1.21e-01	9.69e-01	-12.280	-452.2634
4	5.029	-83.112	2.62e-03	1.21e-01	5.33e-01	-249.336	-452.6738
5	5.557	-180.001	1.59e-03	5.53e-02	4.50e-17	-540.003	-500.1913
6	5.999	-180.000	1.31e-03	0.00e+0	4.71e-17	-540.001	-539.9968
7	5.999	-180.000	1.12e-03	1.11e-16	8.88e-17	-540.001	-539.9973

Exemplo do sapateiro I

- Seja o problema do sapateiro

$$\begin{array}{ll}
 \max_x & x_1 + x_2 \\
 \text{s.a} & 2x_1 + x_2 \leq 8 \\
 & x_1 + 2x_2 \leq 7 \\
 & x_2 \leq 3 \\
 & x_i \geq 0
 \end{array}
 \Rightarrow
 \begin{array}{ll}
 \min_x & -x_1 - x_2 \\
 \text{s.a} & 2x_1 + x_2 + x_3 = 8 \\
 & x_1 + 2x_2 + x_4 = 7 \\
 & x_2 + x_5 = 3 \\
 & x_i \geq 0
 \end{array}$$

- Aplicaremos os três algoritmos apresentados nesta aula, utilizando os seguintes parâmetros: $\varepsilon = 10^{-3}$, $\alpha = 0.95$ e

- PAE: $x^0 = (0.1, 0.1, 7.7, 6.7, 2.9)^T$.
- DAE: $w^0 = (-2, -2, -1)^T$, $s^0 = c - A^T w^0$.
- PDAE: x^0 usado no PAE e w^0 e s^0 usando no DAE, $\sigma = 0.4$.

- O resultado do algoritmo PAE é mostrado a seguir

k	σ_p	σ_d	σ_c	$c^T x^k$
0	0.00e+00	5.84410e-01	0.18774	-0.200000
1	1.65e-16	8.88564e-02	0.27010	-4.444289
2	3.77e-15	8.23370e-04	0.08202	-4.920450
3	2.75e-14	9.17983e-05	0.01651	-4.983268
4	4.50e-14	1.27150e-06	0.00300	-4.997003
5	5.60e-13	1.12401e-07	0.00061	-4.999300

Exemplo do sapateiro II

- Na sequência é apresentado o resultado do algoritmo DAE

k	$\min(x_k)$	σ_c	$b^T w^k$
0	1.14e+00	3.02e+01	-33.000000
1	-5.01e-01	4.25e+00	-9.061404
2	4.00e-02	6.05e-01	-5.561431
3	-2.58e-03	1.17e-01	-5.116246
4	-2.07e-03	3.23e-02	-5.032653
5	3.66e-07	7.78e-03	-5.007769
6	-5.09e-08	1.94e-03	-5.001940
7	-3.95e-07	4.70e-04	-5.000470
8	5.67e-10	1.15e-04	-5.000115

Exemplo do sapateiro III

- E finalmente é apresentado o resultado do algoritmo PDAE

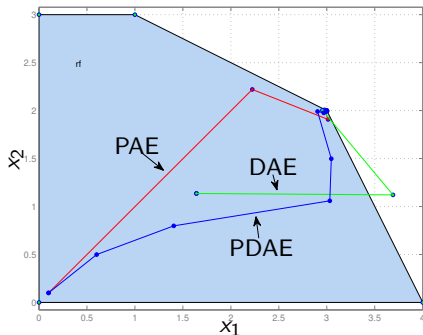
k	μ^k	σ_p	σ_d	$b^T w^k$	$c^T x^k$
0	2.62e+00	0.00e+00	0.00e+00	-33.000000	-0.200000
1	1.05e+00	3.69e-17	0.00e+00	-14.219930	-1.099930
2	4.20e-01	7.37e-17	0.00e+00	-7.451449	-2.203449
3	1.68e-01	0.00e+00	3.25e-17	-6.189671	-4.090471
4	6.72e-02	3.38e-16	4.02e-17	-5.384614	-4.544934
5	2.69e-02	2.12e-16	8.02e-17	-5.230569	-4.894697
6	1.07e-02	1.30e-15	3.86e-17	-5.078878	-4.944529
7	4.30e-03	1.54e-15	6.31e-17	-5.032202	-4.978463
8	1.72e-03	2.45e-16	9.16e-17	-5.012900	-4.991405

Exemplo do sapateiro IV

■ A figura mostrada a seguir ilustra a sequência de pontos geradas pelos três algoritmos. Note que o algoritmo PDAE apresenta uma trajetória mais “central”.

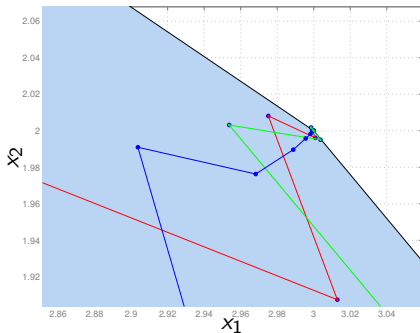
■ Note que o algoritmo PAE se aproxima rapidamente da fronteira, e que o ponto fora da região factível produzido pelo algoritmo DAE é coerente, pois o DAE trabalha com a factibilidade da restrição dual

(em termos de w e s) e a figura apresentada é mostrada em termos de x (mais precisamente em termos de x_1 e x_2).



Exemplo do sapateiro V

- A figura ao lado mostra um “zoom” próximo do ponto ótimo. Note que os algoritmos PAE e DAE apresentam o característico comportamento de *zig-zag* produzidos pelos métodos baseados no gradiente.
- O método PDAE se aproxima de ótimo de uma maneira mais “central”.



Referências I



S.-C. Fang and S. Puthenpura.

Linear optimization and Extensions: Theory and Algorithms.

Prentice Hall, Englewood Cliffs, New Jersey, 1993.



C. C. Gonzaga.

An algorithm for solving linear programming problems in $o(n^3l)$ operations.

In N. Megiddo, editor, *Progress in Mathematical Programming*, pages 1–28.

Springer-Verlag, New York, 1989.



A. Grigorev.

The fundamental theorem of linear algebra.

Technical report, Technische Universität Berlin.

<http://alexeygrigorev.com/projects/imsem-ws14-lina/imsem.pdf>.



V. Klee and G. J. Minty.

How good is the simplex algorithm?

In O. Shisha, editor, *Inequalities III*, pages 159–175. Academic Press, New York, 1972.

Referências II



W. F. Mascarenhas.

The affine scaling algorithm fails for stepsize 0.999.

SIAM Journal on Optimization, 7(1):34–46, February 1997.



R. D. C. Monteiro and I. Adler.

Interior path following primal-dual algorithms. part i: Linear programming.

Mathematical programming, 44(1-3):27–41, 1989.



C. Roos and J.-P. Vial.

Long steps with the logarithmic penalty barrier function in linear programming.

In J. Gabszewicz, J. F. Richard, and L. Wolsey, editors, *Economic Decision-Making: Games, Economics, and Optimisation*, pages 433–441. Elsevier Science Publishers, Amsterdam, Holland, 1990.






G. Strang.

The fundamental theorem of linear algebra.

American Mathematical Monthly, 100(9):848–855, November 1993.

Referências III

-  T. Tsuchiya and M. Muramatsu.
Global convergence of a long-step affine scaling algorithm for degenerate linear programming problems.
SIAM Journal on Optimization, 5(3):525–551, August 1995.
-  R. J. Vanderbei.
Linear Programming: Foundations and Extensions.
Springer, 3 edition, 2008.
-  R. J. Vanderbei, M. S. Meketon, and B. A. Freedman.
A modification of karmarkar's linear programming algorithm.
Algorithmica, 1(1-4):395–407, 1986.