MODELO DE MULTI-PRODUTOS USANDO AFIM ESCALA DUAL

LUIS ERNESTO TORRES GUARDIA
Departamento de Engenharia de Produção
Universidade Federal Fluminense
Rua Passo da Pátria 156, São Domingos
24210-240, Niterói, R.J., Brasil
tepletg@vm.uff.br

Resumo
Neste trabalho, o método de pontos interiores afim escala dual é desenvolvido para resolver o problema linear de fluxos em rede para multi-produtos. Em cada iteração do algoritmo dual, a solução do sistema linear de equações é determinada usando um novo procedimento de decomposição da matriz, e incorporando o método do gradiente conjugado, objetivando uma implementação eficiente, de modo tal a explorar a estrutura da matriz de restrições, dispensando seu armazenamento. Experiências computacionais obtidos, para o problema dinâmico de distribuição/estoque, mostram que o algoritmo proposto é competitivo quando comparado com métodos já existentes.
Palavras-chave: Fluxos em rede. Pontos Interiores Afim Escala Dual. Problema de estoque.

Abstract
In this work, the dual affine scaling interior point method is developed for the multicommodity network flow problems. This specialization uses a specialized factorization to solve a linear system of equations at each iteration of the algorithm and a conjugate gradient solver to exploiting the particular structure of the constraint matrix. The computational experiments were carried out using a set of dynamic distribution/inventory problems, and the results indicate that the proposed algorithm is competitive with others existing methods.
Keywords: Network flows problem. Dual Affine Scaling Interior Point. Inventory Problem.

1. Introdução

O problema linear de fluxos em rede para multi-produtos usualmente tem muitas variáveis e restrições , a vezes difícil de ser resolvido por procedimentos gerais, digamos o método simplex.

Uma grande quantidade de aplicações práticas do problema de fluxos para multi-produtos pode ser encontrada na literatura, ver por exemplo, Assad [1978], Kennington [1978], Ali *et al*. [1980], [1984]. Para uma revisão geral da teoría, algoritmos e aplicaçòes de fluxos em rede, veja os livros de Ahuja *et al*. [1993] e Kennigton e Helgason [1980].

Desde o surgimento dos métodos de pontos interiores para resolver problemas de programação linear, códigos computacionais vêm apresentando-se como alternativas eficientes, principalmente, para solução de problemas de grande porte.

Em cada iteração dos métodos de pontos interiores, a resolução de um sistema linear é efetuada, e em termos computacionais, consiste no passo mais caro. A principal característica neste trabalho é a apresentação de um novo método de resolução do sistema linear, obtendo excelente desempenho computacional para o caso de um único produto.

Este trabalho é organizado como segue. Seção 2 apresenta a formulação matemática do problema de fluxos para multi-produtos a ser resolvido. O método de pontos interiores afim escala dual é apresentado na seção 3. Na seção 4 apresenta-se uma especialização do método de pontos interiores para o problema de fluxos em rede para multi-produtos. Na seção 5 aplica-se o método de pontos interiores para um modelo de distribuição/estoque de multi-produtos. A seção 6 descreve os experimentos computacionais, e em seguida, nas seções 7 e 8 apresentam-se as conclusões e referências bibliográficas respectivamente.

2. Formulação do Problema de Fluxos em Rede para Multi-produtos

O problema linear de fluxos em rede, para K produtos distribuídos em um grafo dirigido (N,**A**), onde N é o conjunto de m nós e **A** é o conjunto de n arcos, pode ser formulado como segue:

$$\text{minimizar} \quad \sum_{k=1}^{K} (\mathbf{c}^k)^t \mathbf{x}^k \tag{1}$$

$$\text{sujeito a} \quad \mathbf{A}\,\mathbf{x}^k = \mathbf{b}^k, \quad k = 1,\dots,K \tag{2}$$

$$\sum_{k=1}^{K} \mathbf{x}^k + \mathbf{x}^v = \mathbf{b}_{mc}, \tag{3}$$

$$\mathbf{x}^k \geq \mathbf{0},\ \mathbf{x}^v \geq \mathbf{0},\ k = 1,\dots,K.$$

onde $\mathbf{A} \in \mathbf{R}^{mxn}$ representa a matriz de incidência nó-arco correspondente à rede (cada coluna i está relacionada ao arco $\boldsymbol{a} \in \mathbf{A}$ com coeficientes +1 e –1), $\mathbf{c}^k \in \mathbf{R}^n$ representa o vetor de custo para cada produto k; e $\mathbf{b}^k$ representa o vetor requerimento de suprimento/procura para cada produto k, onde $b_i^k > 0$ indica o nó de suprimento i para o produto k, $b_i^k < 0$ indica o nó de procura i para o produto k. O vetor $\mathbf{b}_{mc} \in \mathbf{R}^n$ representa o vetor de capacidade correspondente a cada arco da rede. É suposto que **A** é uma matriz de posto completo.

Além disso, $\mathbf{x}^k \in \mathbf{R}^n$ representa o vetor de fluxos no arco para o produto k, $\mathbf{x}^v \in \mathbf{R}^n$ representa o vetor da variável de folga. A relação (1) representa a função objetivo a minimizar. A relação (2) expressa as restrições de conservação de fluxos da rede, enquanto a relação (3) é conhecida como a restrição de capacidade, especifica o fluxo total máximo de todos os produtos para cada arco. Problemas de fluxos para multi-produtos não satisfazem a propriedade de a solução ótima ser um número inteiro como no caso para um único produto.

O problema dual associado a (1) - (3) é escrito como segue:

$$\text{maximizar} \quad \sum_{k=1}^{K} (\mathbf{b}^k)^t \mathbf{y}^k - \mathbf{b}_{mc}\mathbf{y}^v$$

$$\text{sujeito a} \quad \mathbf{A}^t\,\mathbf{y}^k - \mathbf{y}^v + \mathbf{s}^k = \mathbf{c}^k$$

$$\mathbf{y}^v \geq \mathbf{0},\ \mathbf{s}^v \geq \mathbf{0}.$$

onde $\mathbf{y}^v$ é o vetor dual e $\mathbf{s}^k$ é o vetor de folga.

Nos últimos anos, vários métodos foram desenvolvidos para resolver problemas de fluxos para multi-produtos, assim como códigos computacionais foram implementados usando algoritmos especializados do simplex, ver por exemplo Farvolden *et al.*[1993], Jones *et al.* [1993], Schneur e Orlin [1998], Detlefsen e Wallace [2002] entre outros.

Karmarkar [1984] publicou um novo método polinomial para resolver problemas de programação linear, disposto a bater o método simplex. O método de Karmarkar e suas variantes, conhecidos como métodos de pontos interiores, foram usados eficientemente para resolver problemas de fluxos em rede. Os métodos de pontos interiores compartem a mesma estrutura básica computacional.

A partir de um ponto interior inicial, os algoritmos de pontos interiores realizam o progresso em direção à otimalidade através do interior da região factível, em vez de visitar seus vértices, como no caso do método simplex. Em cada iteração determina-se uma direção, digamos δ, que envolve a resolução de um sistema linear de equações do tipo:

$$(\mathbf{A}' \; \boldsymbol{\Theta} \; (\mathbf{A}')^t) \; \delta = \beta \tag{4}$$

onde $\mathbf{A}'$ é a matriz de restrições do programa linear, $\boldsymbol{\Theta}$ é uma matriz diagonal de escalonamento e β é um vetor qualquer.

Choi e Goldfarb [1990], Lustig e Li [1992] e Castro [2000] apresentam o método de ponto interior primal-dual para resolver problemas de fluxos em rede para multi-produtos. Chardaire e Lisser [2002] apresentam o método de ponto interior Afim Dual de escalonamento para o caso de multi-produtos aplicado a um caso especial de rede de telecomunicações.

3. Algoritmo de Pontos Interiores Afim Dual.

O algoritmo afim escala dual foi um dos primeiros métodos de pontos interiores que mostrou-se computacionalmente competitivo se comparado ao método simplex.

Apresentamos a continuação, uma breve descrição do algoritmo afim dual para resolver o seguinte problema de programação linear (primal) com restrições lineares:

$$\begin{array}{ll} \text{minimizar} & \mathbf{c}^t\mathbf{x} \\ \text{sujeito a} & \mathbf{A}'\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}, \end{array} \tag{5}$$

onde $\mathbf{x} \in \mathbf{R}^{n'}$, $\mathbf{c} \in \mathbf{R}^{n'}$, $\mathbf{b} \in \mathbf{R}^{m'}$ e $\mathbf{A}' \in \mathbf{R}^{m' x n'}$ uma matriz de posto de linha completo. O dual do problema (5) é dado por:

$$\begin{array}{ll} \text{maximizar} & \mathbf{b}^t \; \mathbf{y} \\ \text{sujeito a} & (\mathbf{A}')^t \, \mathbf{y} + \mathbf{s} = \mathbf{c} \\ & \mathbf{s} \geq \mathbf{0} \end{array} \tag{6}$$

onde $\mathbf{y} \in \mathbf{R}^{m'}$ representa a variável dual e $\mathbf{s} \in \mathbf{R}^{\mathbf{n}'}$ representa a variável de folga. Note que o problema de fluxos para multi-produtos (1) – (3) pode ser formulado como o problema linear (5).

Dado um ponto interior dual viável $\mathbf{y}^0$, o algoritmo afim dual pode ser descrito da seguinte maneira, ver Birge e Holmes [1992]:

1. j = 0.
2. parar se o critério de parada é satisfeito.
3. seja $\mathbf{s}^\mathbf{j} = \mathbf{c} - (\mathbf{A}')^t \, \mathbf{y}^\mathbf{j}$.
4. calcular a direção de busca
   (a) seja $\mathbf{D}^\mathbf{j}$ = diagonal$\{(1/s_1^j),\ldots,(1/s_n^j)\}$.
   (b) seja $\mathbf{dy} = [\mathbf{A}'(\mathbf{D}^\mathbf{j})^\mathbf{2}(\mathbf{A}')^\mathbf{t}]^{\mathbf{-1}} \, \mathbf{b}$.

(c) seja $\mathbf{ds} = -(\mathbf{A}')^t\,\mathbf{dy}$.

5. calcular o tamanho do passo

 seja $\alpha = \gamma \min \{ s_i^j / -(\mathbf{ds})_i : (\mathbf{ds})_i < 0, i = 1,...,n \}$, onde $0 < \gamma < 1$.

6. atualizar as variáveis primais e duais

 (a) seja $\mathbf{y}^{j+1} = \mathbf{y}^j + \alpha\,\mathbf{dy}$.

 (b) seja $\mathbf{x}^{j+1} = -(\mathbf{D}^j)^2\,\mathbf{ds}$

 (c) seja $j = j + 1$.

 (d) ir ao passo 2.

O maior esforço computacional requerido no procedimento anterior é o cálculo da solução do sistema linear no passo (4b), para o qual existem métodos diretos usando algum tipo de decomposição da matriz $[\mathbf{A}'(\mathbf{D})^2(\mathbf{A}')^t]$, usualmente uma decomposição do tipo (LU). Existem também métodos iterativos, que geram uma seqüência de aproximações a **dy** , através de multiplicações simples de matriz-vetor, digamos do tipo do gradiente conjugado.

4. Problema de Fluxos em Rede para Multi-produtos

A matriz de restrições do problema linear (5) associada agora à matriz de restrições do problema de fluxos em rede para multi-produtos (1) – (3) é visualizada a seguir:

$$\mathbf{A}' = \begin{bmatrix} A & O & . & O & O \\ O. & A & . & . & O. \\ . & . & . & . & . \\ O & . & . & A & O. \\ I & I & . & I & I \end{bmatrix}$$

(7)

onde cada bloco **A** corresponde para cada produto k, k=1,...,K, e é obtida da matriz de incidência da rede correspondente, depois de remover a última linha da matriz para evitar a deficiência de posto de linha. As matrizes de identidade **I** correspondem às restrições de capacidade.

A direção de busca **dy** , eliminando o índice de iteração j para maior clareza do texto, é obtida resolvendo o sistema linear do passo (4b) do algoritmo afim dual acima descrito. Neste caso, **D** é a matriz diagonal dada a seguir:

$$\mathbf{D} = \begin{bmatrix} D_1 & & & & \\ & . & & & \\ & & . & & \\ & & & D_K & \\ & & & & D_v \end{bmatrix}$$

(8)

onde D_k é uma matriz diagonal para cada produto $k = 1,...,K$. e D_v é a matriz diagonal para a variável de folga $\mathbf{x}^v$.

Fazendo as multiplicações de bloco, a matriz $\mathbf{A}'\,\mathbf{D}^2\,(\mathbf{A}')^t$ tem a seguinte estrutura:

$$\mathbf{A}' \quad \mathbf{D}^2 \quad (\mathbf{A}')^t \quad = \quad \begin{bmatrix} AD_1^2A^t & O & & O & AD_1^2 \\ & . & & & . \\ & & . & & . \\ O & & & AD_K^2A^t & AD_K^2 \\ D_1^2A^t & . & . & D_K^2A^t & D_v^2+\sum_{k=1}^{K} D_k^2 \end{bmatrix}$$

(9)

Seja $\mathbf{B}$ a matriz diagonal de bloco com $\mathbf{B}_k = \mathbf{A}\ \mathbf{D}_k^2\mathbf{A}^t$ para k = 1 ,..., K, e $\mathbf{C}$ uma matriz tal que $\mathbf{C}^t = [\mathbf{D}_1^2\ \mathbf{A}^t, \ldots, \mathbf{D}_K^2\ \mathbf{A}^t]$ e $\mathbf{F} = \mathbf{D}_v^2 + \sum_{k=1}^{K} D_k^2$.

Então, da relação (9) temos:

$$\mathbf{A}'\mathbf{D}^2(\mathbf{A}')^t \quad = \begin{bmatrix} B & C \\ C^t & F \end{bmatrix}$$

(10)

Desde que $\mathbf{D}_v^2$ e $\mathbf{D}_k^2$, k = 1,...,K são matrizes diagonais e definidas positivas, então a matriz $\mathbf{F}$ é diagonal e definida positiva. Usando o (10), e eliminando o índice da iteração j para maior clareza do texto, a relação no passo (4b) pode ser reescrito da seguinte forma:

$$\begin{bmatrix} B & C \\ C^t & F \end{bmatrix} \quad \begin{bmatrix} dy^1 \\ dy^2 \end{bmatrix} \quad = \quad \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

(11)

onde $\mathbf{dy} = [\ \mathbf{dy}^1 , \mathbf{dy}^2\ ]^t$, $\mathbf{b} = [\ \mathbf{b}_1 , \mathbf{b}_2]^t$.

O sistema linear anterior (11) pode ser dado por:

$$[\ \mathbf{F} - \mathbf{C}^t \ \mathbf{B}^{-1} \ \mathbf{C}\ ]\ \mathbf{dy}^2 = \mathbf{b}_2 - \mathbf{C}^t\ \mathbf{B}^{-1}\ \mathbf{h}_1$$

(12)

$$\mathbf{B}\ \mathbf{dy}^1 = \mathbf{b}_1 - \mathbf{C}\ \mathbf{dy}^2.$$

onde a matriz $(\mathbf{F} - \mathbf{C}^t\ \mathbf{B}^{-1}\ \mathbf{C})^{-1}$ é conhecida como o complemento de Schur.

Seja $\mathbf{b}_1 = [\ \mathbf{b}_{11} , \ldots , \mathbf{b}_{1K}\ ]$, então temos de (12):

$$[\ \mathbf{F} - \mathbf{C}^t \ \mathbf{B}^{-1} \ \mathbf{C}\ ]\ \mathbf{dy}^2 = \mathbf{b}_2 - \sum_{k=1}^{K} (D_k^2 A^t\ B_k^{-1})\,\mathbf{b}_{1k}$$

(13)

onde

$$\mathbf{F} - \mathbf{C}^t \ \mathbf{B}^{-1} \ \mathbf{C} = (\mathbf{D}_v^2 + \sum_{k=1}^{K} D_k^2 - \sum_{k=1}^{K} (D_k^2 A^t B_k^{-1} A D_k^2)$$

(14)

Castro [2000] menciona por ser $\mathbf{D}_k^2$ matrizes simétrica e definida positiva e $\mathbf{A}$ uma matriz de rede de posto completo, então as matrizes $\mathbf{B}_k = \mathbf{A}\ \mathbf{D}_k^2\ \mathbf{A}^t$ é também simétrica e definida positiva.

Neste trabalho, usamos uma aproximação da matriz inversa de $\mathbf{B}_k$, ver Benzi *et al*. [2000], de tal forma que:

$$\mathbf{B}_k^{-1} \cong \mathbf{Z}_k \; \mathbf{P}_k^{-1} \; \mathbf{Z}_k^t \qquad (15)$$

onde $\mathbf{Z}_k$ é uma matriz triangular superior com elementos na diagonal de 1´s e $\mathbf{P}_k$ é uma matriz diagonal.

Castro [2000] também observa que a matriz $\mathbf{F} - \mathbf{C}^t \, \mathbf{B}^{-1} \, \mathbf{C}$ é simétrica e definida positiva, portanto o sistema (13) pode ser resolvido, entre outros métodos, usando o método do gradiente conjugado pré-condicionado (GCP), devido a só precisar o produto de matriz - vetor, ver por exemplo Nocedal e Wright [1999].

Uma vez determinada $\mathbf{dy}^2$, o sistema $\mathbf{B} \, \mathbf{dy}^1 = \mathbf{b}_1 - \mathbf{C} \, \mathbf{dy}^2$ de (12) pode ser resolvido da seguinte forma:

$$\mathbf{dy}^1 = \mathbf{B}^{-1} \, (\mathbf{b}_1 - \mathbf{C} \, \mathbf{dy}^2)$$

ou usando (15)

$$\mathbf{dy}_k^{\,1} = (\, \mathbf{Z}_k \, \mathbf{P}_k^{-1} \, \mathbf{Z}_k^t \,) \, [\, \mathbf{b}_{1k} - (\, \mathbf{A} \, \mathbf{D}_k^2 \,) \, \mathbf{dy}^2 \,], \;\; k = 1,...,K$$

onde $\mathbf{dy}^1 = [\, \mathbf{dy}_1^{\,1}, \ldots, \mathbf{dy}_K^{\,1} \,]$.

Determinada **dy**, podemos calcular as outras variáveis **ds** usando o passo (4c) e o novo ponto interior (**x**,**y**,**s**) usando o passo (6b), o passo (6a) e o passo (3) respectivamente do algoritmo afim dual.

Chardaire e Lisser [2002] usam por exemplo a decomposição de Cholesky da matriz $\mathbf{B_k}$ e da matriz $\mathbf{F} - \mathbf{C^t} \, \mathbf{B^{-1}} \, \mathbf{C}$ para o caso do fluxo para multi-produtos em uma rede não orientada.

5. Modelo Dinâmico de Distribuição/Estoque.

O problema dinâmico de distribuição/estoque trata com a alocação de produtos, que são produzidos em várias plantas de produção, e transportados a um conjunto de armazéns para sua posterior distribuição nos centros de procura, em um horizonte de planejamento dividido em T períodos de tempo. Este problema é apresentado por Cheung e Powell [1996] para um único produto e demanda desconhecida. Aqui consideramos para mais de um produto com demanda conhecida.

Seja então:

$x_{ijk}(t)$=quantidade do produto k transportado da planta i ao armazém j no período t.
$c_{ijk}(t)$=custo de transporte de uma unidade do produto k da planta i ao armazém j no período t.
$y_{jsk}(t)$=quantidade do produto k transportado do armazém j ao cliente s no período t.
$q_{jsk}(t)$=custo de transporte de uma unidade do produto k do armazém j ao cliente s no período t.
$A_{ik}(t)$=quantidade do produto k produzido na planta i no período t.
$B_{jk}(t)$=quantidade do produto k disponível no armazém j no período t.
$D_{sk}(t)$=quantidade do produto k solicitado pelo cliente s no período t.
$b_{ij}(t)$=capacidade no arco (i,j) da planta i ao armazém j no período t.
$b_{js}(t)$=capacidade no arco (j,s) do armazém j ao cliente s no período t.

Define-se também:

$x_{iik}(t)$=estoque do produto k na planta i no período t.

$c_{iik}(t)$=custo por manter em estoque uma unidade do produto k na planta i no período t.

$y_{jjk}(t)$=estoque do produto k no armazém j no período t.

$q_{jjk}(t)$=custo por manter em estoque uma unidade do produto k no armazém j no período t.

Supondo que não há estoque ao inicio do horizonte de planejamento, o modelo dinâmico integrado de distribuição/estoque pode ser formulado como o seguinte problema de programação linear:

$$\text{minimizar} \quad \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i\in M}\sum_{j\in N} c_{ijk}(t)x_{ijk}(t) + \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j\in N}\sum_{s\in S} q_{jsk}(t)y_{jsk}(t)$$

$$\text{sujeito a:} \quad \sum_{j\in N} x_{ijk}(t) + x_{iik}(t) = A_{ik}(t) + x_{iik}(t-1), \ \ i \in M$$

$$\sum_{s\in S} y_{jsk}(t) + y_{jjk}(t) = B_{jk}(t), \ \ j \in N.$$

$$\sum_{i\in M} x_{ijk}(t) + y_{jjk}(t-1) = B_{jk}(t), \ \ j \in N.$$

$$\sum_{j\in N} y_{jsk}(t) = D_{sk}(t), \ \ s \in S.$$

$$\sum_{k=1}^{K} x_{ijk}(t) \le b_{ij}(t), \ \ i\in M, j\in N, t=1,...T.$$

$$\sum_{k=1}^{K} y_{jsk}(t) \le b_{js}(t), j\in N, s\in S, t=1,\dots,T.$$

$$x_{ijk}(t) \ge 0, y_{jsk}(t) \ge 0, \ \ i\in M, j\in N, s\in S, t=1,\dots,T$$

onde M representa o conjunto de índice das plantas de produção, $i\in M$; N representa o conjunto de índice dos armazéns, $j\in N$; e S representa o conjunto de índice dos clientes, $s\in S$.

6. Experimentos Computacionais

O método proposto de pontos interiores afim escala dual foi inteiramente codificado na linguagem Fortran, sem fazer chamadas externas de *subroutinas*, e testado na solução do problema dinâmico de distribuição/estoque para diferentes valores de K: número de produtos, de T: número de períodos de tempo no horizonte de planejamento e /M/: número de plantas de produção, /N/: número de armazéns, e /S/: número de clientes. Todos os experimentos foram realizados em um microcomputador arquitetura IBM – PC com processador AMD de 1600 Mhz. Os resultados computacionais mostram a eficiência deste algoritmo afim dual para o problema de fluxos para multi-produtos, dispensando o armazenamento da matriz de restrições $\mathbf{A}'$ e da matriz $\mathbf{A}'\mathbf{D}^2(\mathbf{A}')^t$, sendo realizado com a utilização de dois vetores, origem e destino, onde cada par de componentes correspondentes dos dois vetores representa o arco da rede. As matrizes $\mathbf{Z}_k$ e $\mathbf{P}_k$ que formam a decomposição da matriz inversa de $\mathbf{B}_k$ são armazenadas como vetores. Neste trabalho, a solução do sistema linear é obtida usando o método do gradiente conjugado pré-condicionado (GCP), cuja matriz pré-condicionador é simplesmente a inversa da matriz diagonal F, de tal modo a reduzir o número de iterações do GCP. Existem outras matrizes pré-condicionador, ver por exemplo Portugal *et al.* [2000].

7. Conclusões

Este trabalho apresenta, em cada iteração dos métodos de pontos interiores, uma nova decomposição da matriz do sistema linear e de fácil implementação. A resolução deste sistema linear é obtida aplicando o método do gradiente conjugado, dispensando o armazenamento da matriz original $\mathbf{A}'$ de restrições e da matriz do método de pontos interiores $\mathbf{A}'\ \mathbf{D}\ (\mathbf{A}')^t$. O método aqui proposto é competitivo quando comparado com outros métodos existentes na literatura.

O procedimento proposto pode ser adaptado para o outro modelo de fluxos para multi-produtos, onde as variáveis de decisão são canalizadas. O método pode também ser implementado usando processamento paralelo.

8.Referência Bibliográficas:

Ahuja, A., T. Magnanti e J. Orlin, (1993), Network Flows: Theory, Algorithms and Applications, Prentice - Hall, Inc. New Jersey.

Ali, A., R. Helgason, J. Kennington e H. Lall, (1980), Computational Comparison among Three Multicommodity Network Flow Algorithms, Operations Research, v. 28, p. 995 -1000.

Ali, A., D. Barnett, K. Farhangian, J. Kennington, B. Patty, B. Shetty, B. McCarl e P. Wong, (1984), Multicommodity Network Problems : Applications and Computations, IIE Transactions, v. 16, p. 127 – 134.

Assad, A., (1978), Multicommodity Network Flows: A Survey, Networks, v. 8, p. 37-91.

Benzi, M., J. Cullum e M. Tuma, (2000), Robust Approximate Inverse Preconditioning for the Conjugate Gradient Method, SIAM Journal on Scientific Computing, v. 22, n^0 4, p. 1318 – 1332.

Birge, J. e D. Holmes, (1992), Efficient Solution of Two-Stage Stochastic Linear Programs using Interior Point Methods, Computational Optimization and Applications, v. 1, p. 245 – 276.

Castro, J. (2000), A Specialized Interior Point Algorithm for Multicommodity Network Flows, SIAM Journal on Optimization, v. 10, n^0 3, p. 852 – 877.

Chardaire, P. e A. Lisser, (2002), Simplex and Interior Point Specialized Algorithms for Solving Nonoriented Multicommodity Flow Problems, Operations Research, v. 50, n^0 2, p. 260 – 276.

Cheung, R. e W. Powell, (1996), Models and Algorithms for Distribution Problems with Uncertain Demands, Transportation Science, v. 30, no. 1, p. 43 – 59.

Choi, I. e D. Goldfarb, (1990), Solving Multicommodity Network Flow Problems by an Interior Point Method, em Large-Scale Numerical Optimization, T. Coleman e Y. Li (Eds), SIAM, Philadelphia, PA., p. 58 – 69.

Detlefsen, N. e S. Wallace, (2002). The Simplex Algorithm for Multicommodity Networks, Networks, v. 39, no. 1, p. 15 – 28.

Farvolden, J., W. Powell e J. Lustig, (1993), A Primal Partitioning Solution for the Arc-Chain

Formulation of a Multicommodity Network Flow Problem, Operations Research, v. 41, n^0
41, p. 669 – 693.
Jones, K., I. Lusting, J. Farvolden e W. Powell, (1993), Network Flows : The Impact of Formulation on Decomposition, Mathematical Programming, v. 62, p. 95 – 117.
Karmarkar, N., (1984), A New Polynomial Time Algorithm for Linear Programming, Combinatorica, v. 4, p. 373 – 395.
Kennington, J. (1978), A Survey of Linear Cost Multicommodity Network Flows, Operations Research, v. 26, p. 209 – 236.
Kennington, J. e R. Helgason, (1980). Algorithms for Network Programming, John Wiley and
Sons, Inc., New York.
Lustig, J. e G. Li, (1992), An Implementation of a Parallel Primal-Dual Interior Point Method for
Block-Structured Linear Programs, Computational Optimization and Applications, v. 1, p.
141 – 161.
Nocedal, J. e S. Wright, (1999), Numerical Optimization, Springer.
Portugal, L., M. Resende, G. Veiga e J. Júdice, (2000), A Truncated Primal – Infeasible Dual-
Feasible Network Interior Point Method, Networks, v. 35, no. 2, p. 91 – 108.
Schneur, R. e J. Orlin, (1998), A Scaling Algorithm for Multicommodity Flow Problems, Operations Research, v. 42, n^0 2, p. 231 – 246.