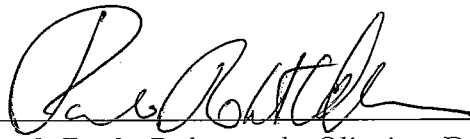


IMPLEMENTAÇÃO DE UM MÉTODO DE PONTOS INTERIORES
USANDO CENTROS ANALÍTICOS PARA RESOLVER PROBLEMAS
DE PROGRAMAÇÃO LINEAR DE GRANDE PORTE

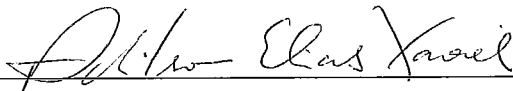
Sérgio Assunção Monteiro

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO
DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE
MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E
COMPUTAÇÃO.

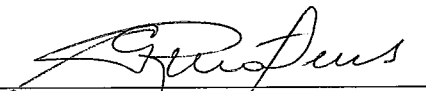
Aprovada por:



Prof. Paulo Roberto de Oliveira, Dr. Ing.



Prof. Adilson Elias Xavier, D.Sc.



Prof. Geraldo Robson Mateus, D.Sc.

RIO DE JANEIRO, RJ - BRASIL
JUNHO DE 2002

MONTEIRO, SÉRGIO ASSUNÇÃO

Implementação de um método de pontos interiores usando centros analíticos para resolver problemas de programação linear de grande porte [Rio de Janeiro] 2002

XI, 72 p. 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e Computação, 2002)

Tese – Universidade Federal do Rio de Janeiro, COPPE

1 - Método de Centros Analíticos

2 - Métodos de Pontos Interiores

3 - Programação Linear

I. COPPE/UFRJ II. Título (série)

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

IMPLEMENTAÇÃO DE UM MÉTODO DE PONTOS INTERIORES
USANDO CENTROS ANALÍTICOS PARA RESOLVER PROBLEMAS
DE PROGRAMAÇÃO LINEAR DE GRANDE PORTE

Sérgio Assunção Monteiro

Junho/2002

Orientador: Paulo Roberto Oliveira

Programa: Engenharia de Sistemas e Computação

Neste trabalho, avaliamos o desempenho dos métodos propostos em [28] para a resolução de problemas com a decomposição de Dantzig-Wolfe, afim de resolver dois problemas de grande porte: o problema de associação generalizada e um modelo de telecomunicações móvel proposto em [20]. A metodologia de decomposição utiliza a estrutura especial destes problemas, produzindo sub-problemas mais simples e gerenciados por um programa principal, chamado de problema mestre. Realizamos algumas modificações no método de planos de corte, baseado em [25], com a respectiva implementação, e testes, que foram analisados.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IMPLEMENTATION OF A METHOD OF THE INTERIOR POINTS
USING ANALYTIC CENTERS TO SOLVE PROBLEMS OF LINEAR
PROGRAMMING OF LARGE-SCALE

Sérgio Assunção Monteiro

June/2002

Advisor: Paulo Roberto Oliveira

Department: Computing and Systems Engineering

In this work, we evaluated the behaviour of the methods proposed in [28] for the resolution of problems by using the Dantzig-Wolfe decomposition principle, aiming the resolution of two large scale problems: the generalized assignment problem and a model of wireless telecommunication system proposed in [20]. The using of the decomposition methodology for the special structure of those problems, turn them into simpler sub-problems which are controlled by a main program, the so called master problem. Also, based in [25], we did some adaptations in the cutting plane. Numerical tests are presented and analyzed.

Agradecimentos

A Deus.

Aos meus queridos pais Antônio e Izabel e aos meus irmãos Tony e Andréia e ao meu sobrinho Israel, pelo amor e incentivo dispensados a mim.

A minha querida esposa que me apoiou em momentos tão difíceis, dando-me atenção e amor. Ao meu filho Sérgio Absalão que me dá, a cada dia que passa, vontade de vencer.

Ao meu orientador Paulo Roberto, por sua atenção.

Aos professores Adilson Elias Xavier e Geraldo Robson Mateus, por participarem da minha banca.

Ao professor Néelson Maculan, por sua atenção e seu exemplo de profissional e de pessoa.

Ao meu co-orientador Rodrigo Leone, por sua atenção e sua amizade.

Aos meus professores de graduação. De maneira bastante especial àqueles que me ajudaram a ingressar nesta Universidade. Aos professores Marcos Negreiros, Plácido Rogério, Wamberto Weber Miranda e Helano gostaria de agradecer não somente pelos conhecimentos transmitidos e pela ajuda que me ofereceram na época de seleção para o mestrado, mas, sobretudo, pela oportunidade de conviver com eles, aprendendo a partir de seus exemplos.

Ao professor Marcos Negreiros, em especial, vai a minha homenagem pelo profissional e amigo sem o qual, certamente, não teria feito o curso de mestrado.

A todos os amigos, próximos e distantes, como Carlos Henrique, Michele, Jorge Bergson, Elder, Jurandir, Léo, Kilbert, Emanuela, Natal, entre outros. Muito obrigado.

A Cláudio Prata Santiago, Tibérius Bonates, Mara Franklin, Talita de Oliveira e Isabela, em especial, tenho que agradecer pela grande amizade e apoio que me deram em momentos tão difíceis.

A todos os colegas de estudos, funcionários e professores do Programa.

Ao CNPq e à CAPES pelas bolsas de estudos concedidas, tanto na época de graduação quanto durante o período de mestrado.

Conteúdo

1	Introdução	1
2	Centros Analíticos	3
2.1	O Método de Centros Analíticos Aplicado à Programação Linear	3
2.1.1	O Algoritmo Conceitual	5
2.1.2	O Método de Newton	5
2.2	O Algoritmo	7
2.2.1	Um exemplo	8
3	Decomposição de Problemas de Programação Linear	11
3.1	Princípio da decomposição de Dantzig-Wolfe	12
3.2	Decomposição de Dantzig-Wolfe com Centros Analíticos	15
3.3	O Algoritmo	16
3.3.1	Direção de Newton	16
3.3.2	Iteração Principal	18
3.3.3	Reinicialização.	19
3.3.4	O Sub-problema Escravo	32
3.4	Aspectos Teóricos.	34
3.5	Problemas Tratados	36

3.5.1	Problema1 (P01)	36
3.5.2	Problema2 (P02)	37
3.5.3	Problema3 (P03)	37
3.5.4	Problema4 (P04)	38
3.5.5	Problema5 (P05)	39
3.5.6	Problema6 (P06)	39
3.5.7	Problema7 (P07)	40
3.5.8	Problema8 (P08)	40
4	Problema de Telecomunicações Móvel	42
4.1	O Modelo	43
4.2	O Modelo Linearizado	45
4.3	Decomposição via Relaxação Lagrangeana	46
4.3.1	Subproblema em z	48
4.3.2	Subproblema em (p, r, x)	48
4.4	Problemas Tratados	49
4.4.1	Problema1 (Pt01)	49
4.4.2	Problema2 (Pt02)	50
4.4.3	Problema3 (Pt03)	51
4.4.4	Problema4 (Pt04)	52
5	Problema de Associação Generalizada	54
5.1	Relaxação Lagrangeana	55
5.2	Algoritmo Adaptado	55
5.2.1	O Algoritmo	57

5.3	Problemas Tratados	58
5.3.1	Problema1 (Pg01)	58
5.3.2	Problema2 (Pg02)	59
5.3.3	Problema3 (Pg03)	59
6	Experiência Numérica	61
6.1	Inicialização	62
6.2	O método branch-bound com centros analíticos	62
7	Conclusões	66

Lista de Figuras

3.1	Iteração k	21
3.2	Gerando um novo corte	22
3.3	Relaxando o corte	23
3.4	Ativando o corte	24

Lista de Tabelas

2.1	algoritmo de centros analíticos	8
3.1	algoritmo de centros analíticos para programação convexa re- solvendo sub-problemas exatamente	26
3.2	algoritmo de centros analíticos para programação convexa re- solvendo sub-problemas aproximadamente	27
3.3	Oráculo	27
3.4	recuperação de um ponto interior	28
3.5	encontrar o centro analítico da nova região.	29
3.6	Resolvendo o sub-problema escravo aproximadamente	30
5.1	algoritmo de centros analíticos para programação convexa	57
6.1	Resultados para problemas gerais com o algoritmo do capítulo 3, com escravos resolvidos exatamente	63
6.2	Resultados para problemas gerais com o algoritmo do capítulo 3, com escravos resolvidos aproximadamente	64
6.3	Resultados para problemas gerais com o algoritmo modificado	64
6.4	Resultados do Problema de Associação Generalizada	65
6.5	Resultados do método branch-bound para o problema de tele- comunicações	65

6.6 Resultados do algoritmo modificado para o problema de tele-
comunicações 65

Capítulo 1

Introdução

Métodos de decomposição têm sido objeto de pesquisa há várias décadas, citando-se, por exemplo, o clássico Dantzig and Wolfe [1][4][6]. Alguns deles se tornaram a melhor opção para resolver problemas de otimização de grande porte. Por outro lado, a partir da descoberta de algoritmos polinomiais de pontos interiores para uma ampla classe de problemas de otimização contínua, as implementações baseadas naqueles métodos têm sido preferencialmente utilizadas no lugar do simplex [15][16].

Nesta dissertação, resolveremos tanto o problema mestre como os sub-problemas escravos, aproximadamente, e faremos uso da desagregação dos problemas escravos. A vantagem em proceder desta forma está em resolver sub-problemas mais simples, diminuindo potencialmente, assim, o número de iterações necessárias para se gerar um novo corte [12] [13].

O principal problema que abordaremos ao longo da dissertação é um modelo de otimização em telecomunicações proposto em [20] que se caracteriza por garantir a instalação mínima de estações de rádio base (ERBs) mantendo

a garantia da qualidade de serviço. Abordaremos também o problema de associação generalizada. Este problema está presente em várias aplicações, veja [2] [11].

Para abordarmos estes problemas, utilizar-nos-emos da decomposição de Dantzig-Wolfe [6]. Nós decompomos o problema original em um problema mestre e sub-problemas escravos e passamos a resolvê-los pelo método proposto em [28], que é uma versão do algoritmo de centros analíticos. Este algoritmo se destaca por utilizar-se da estabilidade do método de centros analíticos e resolver parcialmente tanto o problema mestre como os sub-problemas escravos.

Este trabalho é dividido por mais seis capítulos. O capítulo 2 apresenta os conceitos fundamentais sobre o método de centros analíticos e um exemplo numérico. O capítulo 3 trata da decomposição de Dantzig - Wolfe com centros analíticos. No capítulo 4 apresentaremos o modelo de telecomunicações móvel a ser tratado. O capítulo 5 mostra como aplicamos a decomposição ao problema de associação generalizada. No capítulo 6 são avaliados os desempenhos dos métodos. E, no último capítulo, são apresentadas as conclusões.

Capítulo 2

Centros Analíticos

Para tratar a dificuldade associada com o cálculo de centros em poliedros, o método de centros analíticos tem-se mostrado muito eficiente [12][13][14].

A cada iteração do método é gerada uma aproximação do centro analítico. Esta noção de centro foi introduzida por Huard [17].

Neste capítulo, explicaremos o funcionamento do método de centros analíticos aplicado à programação linear, apresentando os conceitos fundamentais, o algoritmo e, em seguida, um exemplo numérico.

2.1 O Método de Centros Analíticos Aplicado à Programação Linear

O centro analítico [25] é o único máximo de uma determinada função distância $d(x, S)$. Assim, ele não está associado a um conceito geométrico e, por isto, o seu cálculo é computacionalmente mais simples.

Seja o problema:

(P) :

$$\min \quad c^\top x \quad (2.1)$$

s.a :

$$Ax \leq b$$

$$x \geq 0$$

$$x \leq M$$

onde:

$x \in R^n$ são as variáveis de decisão;

$c \in R^n$ é o vetor custo;

$A_{m \times n}$ é a matriz de restrições;

$b \in R^m$ é o vetor de requerimentos

Seja S o conjunto viável definido como:

$$S = \{x \in R^n : 0 \leq x \leq M \text{ e } (a_i)^\top x \leq b_i, i = 1 \dots m\}$$

onde $(a_i)^\top$, $i = 1, \dots, m$ representam as linhas da matriz A e b_i , $i = 1, \dots, m$, os coeficientes do vetor b . Supõe-se que $\text{int}(S) \neq \emptyset$. Para $\rho > c^\top x^*$, onde x^* é o valor ótimo de 2.1 (suposto existente), $x \in \text{int}(S)$ e $q \geq m + 1$, define-se a função barreira logarítmica[10] associada a S como:

$$f_\rho(x) = -q \ln(\rho - c^\top x) - \sum_{i=1}^m \ln(b_i - a_i^\top x) - \sum_{i=1}^n \ln(x_i) - \sum_{i=1}^n \ln(M - x_i) \quad (2.2)$$

O método de centros analíticos associa a 2.1 a seguinte família de problemas não-lineares e convexos:

$$\min f_\rho(x) : x \in \text{int}(S) \quad (2.3)$$

Seja

$$x(\rho) = \arg \min \{f_\rho(x) : x \in \text{int}(S_\rho)\} \quad (2.4)$$

com S_ρ dado por:

$$S_\rho = \{x \in S : \underbrace{c^\top x \leq \rho, \dots, c^\top x \leq \rho}_{q \text{ vezes}}\} \quad (2.5)$$

onde $c^\top x \leq \rho$ é repetido q vezes e S_ρ é diferente de vazio e limitado. A curva contínua que une os pontos $x(\rho)$ é conhecida como trajetória central e cada $x(\rho)$ é o centro analítico de S_ρ . O centro analítico pode também ser interpretado, geometricamente, como o ponto que maximiza o produto das distâncias de todas as faces de S_{ρ^k} , veja, por exemplo, [28] e [31].

2.1.1 O Algoritmo Conceitual

O algoritmo inicia com $x^0 \in \text{int}(S)$ e com $c^\top x^0 \leq \rho^0$. A cada iteração, é encontrado o novo centro analítico da região atual e o limitante superior é decrescido. A q restrição associada à função objetivo, replicada q vezes, é removida e outra é adicionada, definindo, assim, uma nova região. Esse procedimento é repetido até que seja alcançado um ponto próximo à trajetória central e, suficientemente, próximo da solução ótima.

2.1.2 O Método de Newton

Para encontrar um ponto na vizinhança de $x(\rho^k)$, o qual é o centro analítico da região S_{ρ^k} , utilizar-se-á o método de Newton. Seja f_ρ definido como a equação 2.2, tem-se que o vetor gradiente e a matriz hessiana são dados por:

$$g_\rho(x) = \left[\frac{q}{\rho - c^\top x} c + \sum_{i=1}^m \frac{1}{b_i - a_i^\top x} a_i - x^{-1} + (Me - x)^{-1} \right]$$

e

$$H_\rho(x) = \left[\frac{q}{(\rho - c^T x)^2} c c^T + \sum_{i=1}^m \frac{1}{(b_i - a_i^T x)^2} a_i a_i^T + X^{-2} + (MI - X)^{-2} \right]$$

onde as seguintes notações foram utilizadas:

x^{-1} é o vetor cujas componentes são x_i^{-1} ;

$e \in R^n$ é o vetor de "1";

$(Me - x)^{-1}$ é o vetor de componentes $1/(M - x_i)$;

X é a matriz diagonal cujas componentes não nulas são os elementos de x ,

e I é a matriz identidade $n \times n$.

Observe que a primeira parcela de H_ρ é semidefinida positiva (sdp) no interior de S_ρ , a segunda é, no mínimo, sdp no interior de S_ρ e as duas restantes são definidas positivas (dp), logo f_ρ é estritamente convexa, e possui um único ponto de mínimo. Se x^{k+1} é o ponto mínimo em f , então $\frac{\partial f(x^{k+1})}{\partial x} = 0$ (onde 0 é o vetor nulo). Deste modo, derivando 2.2 e igualando a zero, tem-se que

$$g_\rho + H_\rho(x^{k+1} - x^k) = 0$$

Substituindo $(x^{k+1} - x^k) = d^k$, obtém-se:

$$g_\rho + H_\rho d^k = 0,$$

$$H_\rho d^k = -g_\rho$$

Como $H_\rho(x)$ é uma matriz definida positiva, d^k sempre será uma direção de descida.

O método é executado em um processo iterativo, tal que:

$$x^{k+1} = x^k + \lambda d^k,$$

onde $\lambda = \arg \min\{f(x^k + \lambda d^k) : (x^k + \lambda d^k) \in \text{int}(S_\rho)\}$, caso se escolha a busca linear exata.

Um aspecto crítico que separa a metodologia de centros, sob a perspectiva clássica (veja, por exemplo, o livro de Fiacco and McCormick [10]), e os algoritmos polinomiais de centros é o cálculo, no último caso, aproximado do centro analítico. Portanto, é necessário estabelecer um critério de aproximação do ponto x^{k+1} ao centro analítico. Por motivos teóricos (veja, por exemplo, den Hertog [[7]]), se utiliza a norma induzida por $H_\rho(x)$:

$$\|x\|_{H_\rho} = \sqrt{x^t H_\rho x}$$

e a proximidade de x a $x(\rho)$ é dada através da direção de Newton por

$$\delta_\rho(x) = \|H_\rho(x)^{-1} g_\rho(x)\|_{H_\rho} = \sqrt{g_\rho^t(x) H_\rho(x)^{-1} g_\rho(x)}$$

Os critérios de convergência (em tempo polinomial) exigirão apenas que $\delta_\rho(x^{k+1}) \leq \epsilon$, para algum $\epsilon \in (0, 1)$.

Logo, conhecidos ρ^0 , um limitante superior para o valor da função objetivo, e $x^0 \in \text{int}(S_\rho)$ tal que $\delta_\rho(x^0) \leq \epsilon$, o método é iniciado com a subsequente aproximação à trajetória central. Para mais detalhes sobre este método, consultar [27] e [31].

2.2 O Algoritmo

A seguir, apresenta-se um algoritmo de centros analíticos aplicado ao problema 2.1 com as notações anteriores.

Algoritmo de Centros Analíticos

INICIALIZAÇÃO

TOLERANCIA

$0 < \theta < 1$ (atualização da cota)

$0 < \epsilon < 1$ (proximidade)

Obtenha

$x^0 \in \text{int}(S_{\rho^0}) : \delta_{\rho^0} \leq \epsilon, \rho^0 > c^t x^0$

$k = 0$

repita

se $(\rho^k - c^t x^k) < TOLERANCIA$

Então Terminar

fim-se

{Atualizar a cota superior da função custo }

$\rho^{k+1} = \rho^k + \theta(c^t x^k - \rho^k)$

{Cálculo do centro}

$z = x^k$

repita

$h = -H_{\rho^{k+1}}^{-1} g_{\rho^{k+1}}$

$v = \|h\|_{H_{\rho^{k+1}}}$

se $v \leq \epsilon$

então interrompa

fim-se

$\lambda = \arg \min_{\lambda \geq 0} \{f_{\rho^{k+1}}(z + \lambda h) : z + \lambda h \in S_{\rho^{k+1}}\}$

$z = z + \lambda h$

fim_ repita

$x^{k+1} = z$

$k = k + 1$

fim-repita

Tabela 2.1: algoritmo de centros analíticos

2.2.1 Um exemplo

Seja o problema:

(P) :

$$\min \quad -2x_1 - 3x_2 \quad (2.6)$$

$$\text{s.a :} \quad (2.7)$$

$$3x_1 + x_2 \leq 15$$

$$5x_1 + 3x_2 \leq 20$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

$$(2.8)$$

Neste caso, como se deve ter $q \geq 5$, será fixado $q = 5$. A função barreira logarítmica para este problema é:

$$f_\rho(x) = -5\ln(\rho+2x_1+3x_2) - \ln(15-3x_1-x_2) - \ln(20-5x_1-3x_2) - \ln(x_1) - \ln(x_2)$$

Seja $x^0 = (0, 2860; 5, 2916)$, uma solução inicial pertencente ao interior da região viável S^0 , formada pelo conjunto de restrições. Seja, agora, a cota superior referente ao custo inicial $\rho = (-2; -3)^\top x^0 + \epsilon = -16, 4468 + \epsilon$, para algum $\epsilon > 0$ com $\delta(x^0) = 0, 5428$ e $g(x^0) = (2, 1452; 0, 2292)$. A solução ótima está contida no semi-espaço S^1 , definido por

$$S^1 = S^0 \cap \{x \in R^2 : (-2; -3)^\top x \leq \rho^0\}$$

Neste momento, é necessário aplicar o método de Newton para calcular o centro analítico x^1 da nova região. Após determinar x^1 , atualizar-se-á o valor de ρ^0 , através de

$$\rho^1 = \rho^0 + \theta((-2; -3)^\top x^0 - \rho^0)$$

onde $\theta \in (0, 1)$. Na nossa implementação, utilizamos o valor de θ igual a 0.35. O processo continua até uma iteração k , onde $\rho^k - (-2; -3)^\top x^k \leq TOLERANCIA$. Na solução ótima, o valor da função objetivo é -19.999 e $(x_1, x_2) = (0; 6.6666)$.

Capítulo 3

Decomposição de Problemas de Programação Linear

Os métodos de planos de corte foram introduzidos, independentemente, por Kelley [19] e Cheney e Goldstein [5] como uma técnica para problemas de otimização convexa. Embora estes métodos não sejam competitivos em relação a métodos mais recentes para otimização não-suave, eles são fundamentais para a compreensão da otimização não-diferenciável e sua consequente aplicação à decomposição.

Eles são baseados em aproximações poliédricas das funções convexas, e formam com os métodos subgradientes [30] as duas classes tradicionais na resolução de problemas não-diferenciáveis. Os algoritmos de planos-de-corte, que têm sido desenvolvidos nos últimos anos, têm demonstrado promissores resultados computacionais, veja, por exemplo, [8].

Neste capítulo, apresentar-se-á o método proposto em [26]. Este método se destaca por gerar uma sequência monótona decrescente de cotas da função custo. Além disto, o algoritmo resolve, em geral, os sub-problemas escravos apenas parcialmente. Desta forma, serão resolvidos aproximadamente tanto o problema mestre, característica do método de centros, como o sub-problema-

escravo.

3.1 Princípio da decomposição de Dantzig-Wolfe

Seja o seguinte problema de programação linear:

(P_1) :

$$\min \quad c^T x \quad (3.1)$$

s.a :

$$Ax \leq b$$

$$Dx \leq d$$

$$x \geq 0$$

onde $c, x \in R^n$, $b \in R^{m_1}$, $d \in R^{m_2}$, $A \in R^{m_1 \times n}$, e $D \in R^{m_2 \times n}$.

Com fins de decomposição, reescreveremos 3.1 como:

(P'_1) :

$$\min \quad c^T x \quad (3.2)$$

s.a :

$$Ax \leq b \quad (3.3)$$

$$x \in X \quad (3.4)$$

onde $X = \{x : Dx \leq d \text{ e } x \geq 0\}$. Vamos supor que X seja compacto.

A função lagrangeana associada a (3.2), (3.3) e (3.4) com o vetor de multiplicadores não-negativos $u \in R^{m_1}$ é:

$$L(x, u) = c^T x + u^t(Ax - b), x \in X, u \geq 0 \quad (3.5)$$

A função dual $D(u)$ associada é:

$$D(u) = -u^t b + \min\{(c^t + u^t A)x : x \in X\}, u \geq 0 \quad (3.6)$$

Para cada $u \geq 0$ fixo, resolve-se o seguinte sub problema (problema escravo):

$$\min(c^t + u^t A)x : x \in X \quad (3.7)$$

Uma importante propriedade da função dual é o teorema da dualidade fraca.

Teorema 1 *Teorema da dualidade fraca:* $\forall(x, u)$ viável: $x \in X, Ax \leq b, u \geq 0$, tem-se: $c^t x \geq D(u)$

Demonstração: Seja (\bar{x}, \bar{u}) um ponto viável. Então, temos que

$$\begin{aligned} D(\bar{u}) &= -\bar{u}^t b + \min\{(c^t + \bar{u}^t A)x : x \in X\} \\ &\leq -\bar{u}^t b + (c^t + \bar{u}^t A)\bar{x} = c^t \bar{x} + \bar{u}^t (A\bar{x} - b) \leq c^t \bar{x} \quad \blacksquare \end{aligned}$$

Desta forma, $D(u)$ é sempre uma cota inferior para o custo ótimo primal. Além disto, esta propriedade sugere que u seja escolhido de modo a se ter $D(u)$ o maior possível, levando ao problema dual, dado por

$$\max D(u) : u \in R^{m_1} \text{ e } u \geq 0 \quad (3.8)$$

Neste capítulo, a estratégia é resolver diretamente o problema dual. Com este propósito, as seguintes propriedades são importantes:

Seja

$$D(u) = -u^t b + (c^t + u^t A)x^*(u) \quad (3.9)$$

onde $x^*(u)$ é a solução do problema 3.7. Então $D(u)$ é côncava, não necessariamente diferenciável em u e o resíduo

$$s = Ax^*(u) - b \quad (3.10)$$

é um super-gradiente de $D(u)$ no ponto u .

Demonstração: Temos que:

$D(w) = -w^t b + (c^t + w^t A)x^*(w) \leq -w^t b + (c^t + w^t A)x^*(u)$, para todo w , $u \in R^{m_1}, w, u \geq 0$, devido a que $x^*(w)$ é o ponto que minimiza $D(w)$ em w .

Então, temos que:

$$\begin{aligned} D(w) &\leq -w^t b + (c^t + w^t A)x^*(u) = \\ &= -w^t b + (c^t + w^t A)x^*(u) + u^t A[x^*(u) - x^*(u)] + u^t [b - b] \\ &= -u^t b + (c^t + u^t A)x^*(u) - w^t b + w^t Ax^*(u) + u^t b - u^t Ax^*(u) \\ &= D(u) + w^t (Ax^*(u) - b) - u^t (Ax^*(u) - b) \\ &= D(u) + s^t (w - u) \end{aligned} \quad \blacksquare$$

O par utilizado em cada iteração, formado pelo gradiente (ou super-gradiente) e o valor da função dual, é denominado oráculo.

Necessitaremos do seguinte conceito:

Definição: Um ϵ -super-gradiente de uma função côncava $f : R^{m_1} \rightarrow R$ em $u \in R^{m_1}$ é um elemento do conjunto ϵ -subdiferencial definido como:

$\partial_\epsilon D(u) := \{s \in R^{m_1} : D(w) \leq D(u) + s^t (w - u) + \epsilon, \text{ para todo } w \in R^{m_1} \text{ e } \epsilon > 0\}$. Portanto, uma solução $x_k(u) \in X$ do problema 3.7 que atende a um dado critério de aproximação da trajetória central, gerará, para algum $\epsilon > 0$, um ϵ -supergradiente de D em u e produz cotas para o valor de $D(u)$.

3.2 Decomposição de Dantzig-Wolfe com Centros Analíticos

No método de planos secantes de (Kelley,1960) [19], uma função côncava $D(u)$ é sucessivamente aproximada por um conjunto de formas lineares dadas por:

$$D(u) \approx D^\Delta(u) = \min\{D(u_i) + s_i^t(u - u_i), i = 1 \dots k, u \in S\}.$$

onde S é um conjunto compacto do tipo:

$S = \{u : 0 \leq u \leq M, i = 1 \dots m\}$ e que contém $u^* = \arg \max D(u)$. Observe que a compacidade é uma exigência para a convergência do algoritmo de planos secantes, tanto quanto para a definição do método de centros. Em cada iteração, u_{k+1} é computado por:

$$u_{k+1} = \arg \max\{\min\{D(u_i) + s_i^t(u - u_i), i = 1 \dots k, u \in S\}\}$$

que é equivalente a

$$(u_{k+1}, r^{k+1}) = \arg \max\{r : r \leq D(u_i) + s_i^t(u - u_i), i = 1 \dots k, u \in S, r \in R\} \quad (3.11)$$

ou ainda

(P_m) :

$$\max \quad r \quad (3.12)$$

s.a :

$$s_i^t u - r \geq d_i \quad i = 1 \dots k$$

$$u \in S \quad r \in R$$

onde $d_i = s_i^t u_i - D(u_i)$

O problema 3.11 é conhecido como problema mestre e o subproblema 3.7,

como problema escravo.

O método de centros analíticos associa a 3.12 a seguinte função potencial.

$$f_\rho(u, r) = q \ln(r - \rho) + \sum_{i=1}^k \ln(s_i^t u - r - d_i) + \sum_{i=1}^m \ln(u_i) + \sum_{i=1}^m \ln(M - u_i) \quad (3.13)$$

onde $q \geq 2m + k + 1$ e $\rho < r$

Considera-se, então, o problema auxiliar: dado ρ , seja

$$\max f_\rho(u, r) : (u, r) \in \text{int}(S_{\rho^k}) \quad (3.14)$$

onde:

$$S_{\rho^k} = \{(u, r) \in S^k : \underbrace{r \geq \rho^k, \dots, r \geq \rho^k}_{q \text{ vezes}}, q \geq 2m + k + 1\}.$$

O centro analítico de S_{ρ^k} é definido pela solução de 3.14:

$$(\bar{u}, \bar{r})_\rho^k = \arg \max \{f_{\rho^k}(u, r) : (u, r) \in \text{int}(S_{\rho^k}^k)\} \quad (3.15)$$

A curva que une os pontos $(\bar{u}, \bar{r})_{\rho^k}$ é conhecida como trajetória central.

3.3 O Algoritmo

3.3.1 Direção de Newton

O método inicia com um ponto próximo ao centro analítico da região S_{ρ^0} com um limitante inferior $\rho^0 < r$, tal que:

$$\delta_{\rho^k}(u, r) = \|H_{\rho^k}^{-1} \nabla f_{\rho^k}\|_{H_{\rho^k}} = (\nabla f_{\rho^k}^t H_{\rho^k}^{-1} \nabla f_{\rho^k})^{1/2}$$

onde H_{ρ^k} e ∇f_{ρ^k} são, respectivamente, a Hessiana e o gradiente da função

3.13 e são dados por:

$$\nabla f(u, r) = \begin{bmatrix} \sum_{j=1}^k \frac{1}{s_j^t u - r - d_j} s_j & + & u^{-1} - (Me - u)^{-1} \\ q \frac{1}{r - \rho^k} & - & \sum_{j=1}^k \frac{1}{s_j^t u - r - d_j} \end{bmatrix}$$

onde e_i , $i = 1, \dots, m$, é i -ésimo vetor unitário.

$$H_{\rho^k} = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

$$H_{11} = - \sum_{j=1}^k \frac{s_j s_j^t}{(s_j^t u - r - d_j)^2} - U^{-2} - (MI - U)^{-2}$$

$$H_{12} = \sum_{j=1}^k \frac{1}{(s_j^t u - r - d_j)^2} s_j$$

$$H_{21} = \sum_{j=1}^k \frac{1}{(s_j^t u - r - d_j)^2} s_j^t$$

$$H_{22} = -q \frac{1}{(r - \rho^k)^2} - \sum_{j=1}^k \frac{1}{(s_j^t u - r - d_j)^2}$$

onde $U = \text{diag}(u_1, \dots, u_m)$ e I é a matriz identidade $m \times m$.

Portanto, conforme vimos anteriormente, o ponto (u, r) será considerado próximo de (u_{ρ^k}, r_{ρ^k}) , se $\delta = \delta_{\rho^k}(u, r) < \epsilon$ para algum $\epsilon \in (0, 1)$.

O cálculo de um novo ponto próximo ao centro analítico é dado por:

$$(u, r)^{k+1} = (u, r)^k + \lambda d_k$$

onde

$$\lambda = \arg \min \{ f_{\rho^k}((u, r)^k + \lambda d_k) : (u, r)^k + \lambda d_k \in \text{int}(S_{\rho^k}^k) \}$$
 e

$$d = -H_{\rho^k}^{-1} \nabla f_{\rho^k}(u, r)$$

3.3.2 Iteração Principal

O algoritmo inicia com (u^0, r_0) próximo ao centro analítico da região S_{ρ^0} e com um limitante inferior ρ^0 menor que r^0 . A cada iteração é feita uma chamada ao oráculo onde é calculado um super-gradiente. Após isto, o algoritmo verifica se o ponto $(u, r)^k$ ainda pertence ao interior da região S_{ρ^k} . Se pertencer, a nova restrição é adicionada ao conjunto S_{ρ^k} , o limite inferior será aumentado e será calculado um novo ponto (u, r) próximo ao centro analítico da nova região $S_{\rho^{k+1}}$. Caso contrário, ou seja, o ponto (u, r) não pertença ao interior do novo conjunto, será calculado um parâmetro de relaxação γ com o objetivo de deslocar a fronteira do novo corte e mantendo a proximidade ao centro. Após isto, será calculado o novo ponto $(u, r)^{k+1}$ próximo ao centro analítico da nova região.

3.3.3 Reinicialização.

Se o ponto $(u, r)^k$ não pertencer ao novo conjunto $S_{\rho^{k+1}}$ com a inclusão da nova restrição dada pelo subgradiente gerado após a chamada ao oráculo, calcula-se um parâmetro de relaxação γ que será obtido adiante e, em seguida, aplica-se o método de centros para encontrar um novo ponto interior a este conjunto.

Como é viável, a nova restrição é adicionada $p(= k + 2m + 1)$ vezes, para que tenha maior influência no cálculo da função barreira logarítmica associada a esta região. A nova região é dada por:

$$S_\gamma = S_{\rho^k} \cap \underbrace{\{(u, r) : s_{k+1}^t u - r \geq \gamma, \dots, s_{k+1}^t u - r \geq \gamma\}}_{p \text{ vezes}}$$

e a nova função barreira logarítmica é dada por:

$$f_\gamma(u, r) = p \ln(s^t u - r - \gamma) + f_{\rho^k}(u, r)$$

onde f_{ρ^k} é a função barreira logarítmica associada a S_{ρ^k} .

Para ativar o novo corte é necessário encontrar um ponto interior a $S_{\rho^{k+1}}$ e isto pode ser feito aplicando-se o método de centros analíticos ao seguinte problema:

(P_γ) :

$$\max \quad z = s_{k+1}^t u - r \tag{3.16}$$

s.a :

$$(u, r) \in S_{\rho^k}$$

Na verdade, não é, em geral, necessário resolver o problema acima. As iterações serão realizadas até que se encontre $(\bar{u}, \bar{r}) \in \text{int}S_{\rho^{k+1}}$.

No método de centros, associa-se a (3.16) a seguinte família de problemas

não lineares:

$$\min f_{\gamma^k}(u, r) : (u, r) \in \text{int}(S_{\gamma^k}) \quad (3.17)$$

Agora, aplica-se iterações do método de centros analíticos a (3.16) até que seja encontrado o ponto desejado.

O comportamento deste procedimento pode ser visto nas figuras 3.1, 3.2, 3.3 e 3.4.

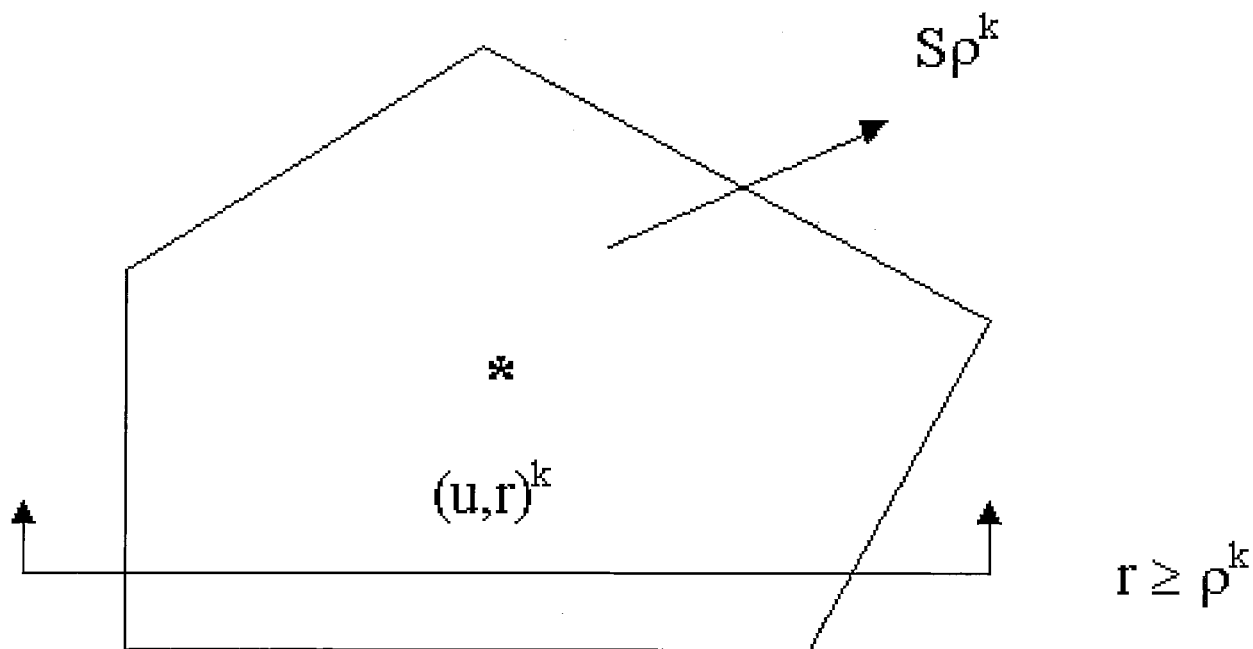


Figura 3.1: Iteração k

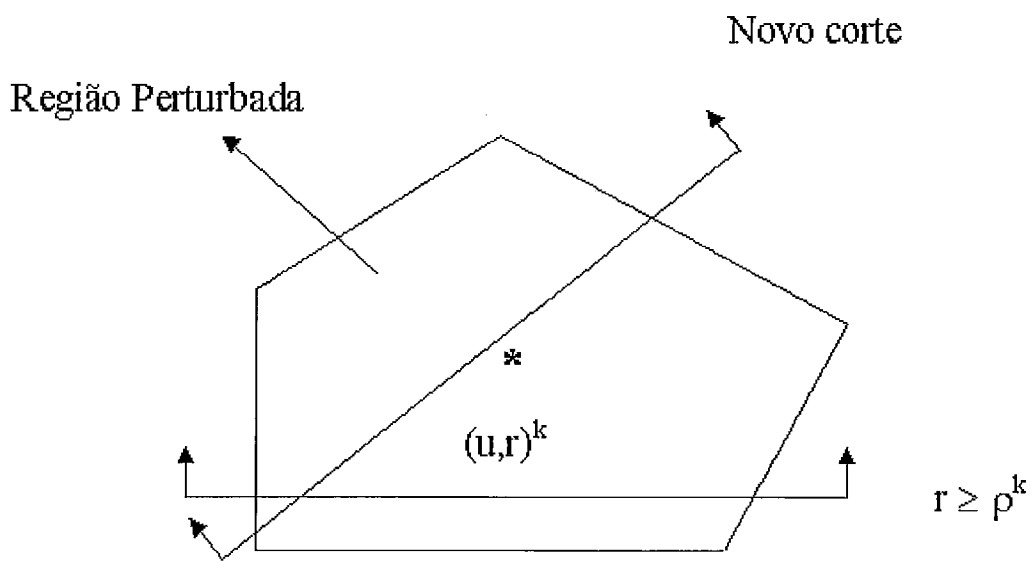


Figura 3.2: Gerando um novo corte

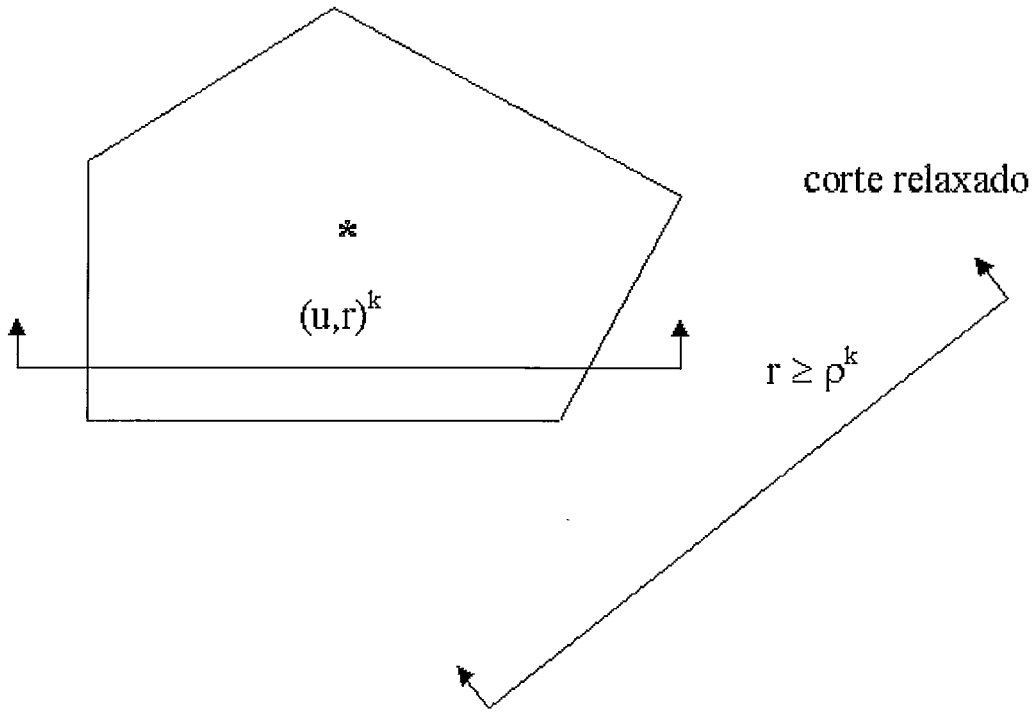


Figura 3.3: Relaxando o corte

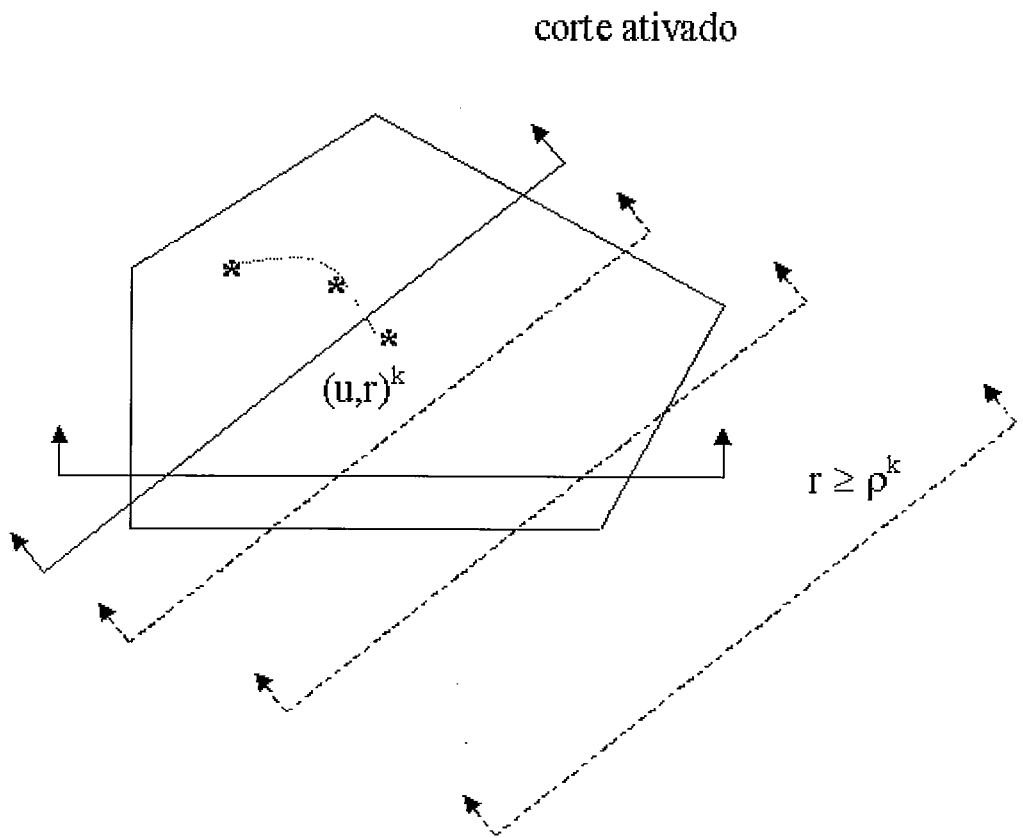


Figura 3.4: Ativando o corte

O teorema a seguir determina o parâmetro γ . Seguimos a demonstração feita em [28], onde os valores obtidos para γ são exatos. (Veja [9] para a mesma técnica de relaxação, porém com cálculo aproximado γ .)

Teorema 2 *Ver [28] Sejam:*

$$w = (s^t, -1), \theta = w^t H^{-1} w, \alpha = w^t H^{-1} w \text{ e } \epsilon^2 < 1 - \delta^2$$

onde $\delta < 1$ é a proximidade de (u, r) ao centro analítico do problema perturbado. Então para

$$\gamma \leq s^t u - r - \frac{p\alpha + \beta}{\epsilon^2}$$

a proximidade de (u, r) ao centro analítico da região perturbada é ϵ .

Demonstração: Seja $P_\gamma(u, r) = p \ln(s^t u - r - \gamma) + P(u, r)$

e defina:

$$w = (s, -1)^t, \eta = s^t u - r - \gamma \text{ e } \mu = \frac{p^{1/2}}{\eta} w \text{ e } y = (u, r). \quad (3.18)$$

Da definição de $P_\gamma(u, r) = P_\gamma(y)$, temos:

$$\nabla P_\gamma(y) = \frac{p}{\eta} w + g = p^{1/2} \mu \quad (3.19)$$

$$\nabla^2 P_\gamma(y) = -\mu \mu^t + H \quad (3.20)$$

onde $g = \nabla P(y)$ e $H = \nabla^2 P(y)$. Interessa-nos calcular a proximidade em relação a P_γ , isto é:

$$[\nabla P_\gamma(y)^t (\nabla^2 P_\gamma(y))^{-1} \nabla P_\gamma(y)]^{1/2}$$

ou

$$[(g + p^{1/2} \mu)^t (H - \mu \mu^t)^{-1} (g + p^{1/2} \mu)]^{1/2}$$

Utilizando-se a identidade de Sherman-Morrisson [23], temos

$$(g + p^{1/2} \mu)^t \left(H^{-1} + \frac{H^{-1} \mu \mu^t H^{-1}}{1 - \mu^t H^{-1} \mu} \right) (g + p^{1/2} \mu) = g^t H^{-1} g + 2p^{1/2} g^t H^{-1} \mu + p \mu^t H^{-1} \mu + \frac{1}{(1 - \mu^t H^{-1} \mu)} [g^t H^{-1} \mu \mu^t H^{-1} g + 2p^{1/2} g^t H^{-1} \mu \mu^t H^{-1} \mu + p \mu^t H^{-1} \mu \mu^t H^{-1} \mu]$$

Algoritmo para Decomposição de Dantzig-Wolfe

Obtenha $(u^0, r^0) \in \text{int}(S_{\rho^0}) : \delta_{\rho^0} \leq 0, 1$

Escolha $D_m^0 = D(x^0)$, $\rho^0 > D(x^0)$, $\theta \in (0, 1)$ e $\epsilon > 0$

$k = 0$

repita

se $(r^k - \rho^k) < \epsilon$

Então Terminar

fim-se

{Resolva o sub-problema escravo}

Compute $D(u^k)$, $s_{k+1}^\top \in \partial D(u^k)$ e $d_{k+1} = s_{k+1}^\top u^k - D(u^k)$

{Compute γ } { Ver Teorema 2 }

{ Ver Tabela 3.4 }

{Reduza γ até d_{k+1} } Aplicação do método de C.A.}

{ Ver Tabela 3.3.3 }

$S_{\rho^{k+1}} = S_{\rho^k} \cap \{(u, r) \in R^{n+1} : s_{k+1}^\top u - r \geq d_{k+1}\}$

$D_m^k = \max\{D(u^k), D_m^k\}$

$\rho^{k+1} = \rho^k + \theta(D_m^k - \rho^k)$

{Incorpora Corte}

{Obtenha (u^{k+1}, r^{k+1}) Aplicação do método de C.A.}

{ Ver Tabela 3.5 }

$k = k + 1$

fim-repita

Tabela 3.1: algoritmo de centros analíticos para programação convexa resolvendo sub-problemas exatamente

Algoritmo para Decomposição de Dantzig-Wolfe

Obtenha $(u^0, r^0) \in \text{int}(S_{\rho^0}) : \delta_{\rho^0} \leq 0, 1$
 Escolha $D_m^0 = D(x^0)$, $\rho^0 > D(x^0)$, $\theta \in (0, 1)$ e $\epsilon > 0$
 $k = 0$

repita

se $(r^k - \rho^k) < \epsilon$
 Então Terminar
 fim-se

{Resolva o sub-problema escravo}
 Compute $D_u(u^k)$, $s_{k+1} \in \partial_\epsilon D(u^k)$ e $d_{k+1} = s_{k+1}u^k - D(u^k)$

{Compute γ } { Ver Teorema 2 }
 {Reduza γ até d_{k+1} Aplicação do método de C.A.}
 { Ver Tabela 3.4 }

$S_{\rho^{k+1}} = S_{\rho^k} \cap \{(u, r) \in R^{n+1} : s_{k+1}u - r \geq d_{k+1}\}$
 Compute $D_l(u^k)$
 $D_m^k = \max\{D_l(u^k), D_m^k\}$
 $\rho^{k+1} = \rho^k + \theta(D_m^k - \rho^k)$

{Incorpora Corte}
 {Obtenha (u^{k+1}, r^{k+1}) Aplicação do método de C.A.}
 { Ver Tabela 3.5 }
 $k = k + 1$

fim-repita

Tabela 3.2: algoritmo de centros analíticos para programação convexa resolvendo sub-problemas aproximadamente

Compute $D(u_k)$, $s_{k+1} \in \partial D(u_k)$ e $d_{k+1} = s_{k+1}^\top u_k - D(u_k)$

Obtenha $x^*(u_k)$ do algoritmo que resolve o sub-problema escravo
 $D(u_k) = -u_k^\top b + (c^\top + u^\top A)x^*(u_k)$
 $s_{k+1} = Ax^*(u_k) - b$
 $d_{k+1} = s_{k+1}^\top u_k - D(u_k)$

Tabela 3.3: Oráculo

Reduzir γ até d

$$i = 0$$

$$\rho_c^i = \gamma$$

$$(u_c^i, r_c^i) = (u^k, r^k)$$

repita

$$\text{se } (s_{k+1}^t u^i - r^i) > d_{k+1}$$

então Terminar

fim-se

$$\text{se } (\rho_c^i > \rho^{k+1}) \text{ então}$$

$$\rho_c^{i+1} = \rho^{k+1}$$

$$p = 1$$

senão

$$\rho_c^{i+1} = \rho_c^i + \theta(s_{k+1}^t u^i - r^i - \rho_c^i)$$

fim-se

$(u_c^{i+1}, r_c^{i+1}) =$ ponto próximo ao centro analítico de

$$S_{\rho_c^{i+1}} = S_{\rho_c^k} \cap \left\{ (u, r) \in R^{n+1} : \underbrace{s_{k+1}^t u^i - r^i \geq \rho_c^{i+1}, \dots, s_{k+1}^t u^i - r^i \geq \rho_c^{i+1}}_{p \text{ vezes}} \right\}$$

$$i = i + 1$$

fim-repita

$$(u_c^k, r_c^k) = (u_c^i, r_c^i)$$

Tabela 3.4: recuperação de um ponto interior

Obter (u, r) próximo de C.A

$i = 0$
 $\rho_c^i = \rho^k$
 $(u_c^i, r_c^i) = (u^k, r^k)$

repita

 se $(\rho_c^i = \rho^{k+1})$
 então Terminar
 fim-se

 se $(r^i > \rho^{k+1})$ então
 $\rho_c^{i+1} = \rho^{k+1}$
 $p = 1$
 senão
 $\rho_c^{i+1} = \rho_c^i + \theta(\rho_c^i - r_c^i)$
 fim-se

$(u_c^{i+1}, r_c^{i+1}) =$ ponto próximo ao centro analítico de
 $S_{\rho_c^{i+1}} = S_{\rho_c^k} \cap \{(u, r) \in R^{n+1} : \underbrace{r \geq \rho_c^{i+1}, \dots, r \geq \rho_c^{i+1}}_{p \text{ vezes}}\}$

$i = i + 1$

fim-repita

$(u_c^k, r_c^k) = (u_c^i, r_c^i)$

Tabela 3.5: encontrar o centro analítico da nova região.

Algoritmo de Centros Analíticos para resolver parcialmente os sub-problemas escravos

Obtenha

$$x^0 \in \text{int}(S_{\rho^0}) : \delta_{\rho^0} \leq 0, 1,$$

$$\rho > (c^\top + u^\top A)x^0$$

$$k = 0$$

repita

$$\text{se } (\rho^k - (c^\top + u^\top A))x^k < \text{TOLERANCIA} \text{ ou } ((r + u^\top b) > (c^\top + u^\top A)x^k)$$

então Terminar

fim-se

$$\{\text{Determinar o plano secante } H^{k+1} = \{x \in R^n : (c + u^\top A)^\top x^k \leq \rho^{k+1}\}\}$$

$$\rho^{k+1} = \rho^k + \theta((c^\top + u^\top A) - \rho^k)$$

{Cálculo do centro}

$$z = x^k$$

repita

$$h = -H_{\rho^{k+1}} g_{\rho^{k+1}}$$

$$v = \|h\|_{H_{\rho^{k+1}}}$$

$$\text{se } v \leq \epsilon$$

então interrompa

fim-se

$$\lambda = \arg \min_{\lambda \geq 0} \{f_{\rho^{k+1}}(z + \lambda h) : z + \lambda h \in S_{\rho^{k+1}}\}$$

$$z = z + \lambda h$$

fim-repita

$$x^{k+1} = z$$

$$k = k + 1$$

fim-repita

Tabela 3.6: Resolvendo o sub-problema escravo aproximadamente

Agora, substituindo-se μ por 3.18, e representando por $\theta := w^t H^{-1} w$ e $\alpha = w^t H^{-1} g$, temos a expressão acima dada por:

$$g^t H^{-1} g + \Delta \quad (3.21)$$

onde

$$\begin{aligned} \Delta &= 2\frac{p}{\eta}\alpha + \frac{p^2}{\eta^2}\theta + \frac{1}{1 - \frac{p\theta}{\eta^2}} \left(\frac{p}{\eta^2}\alpha^2 + \frac{2p^2}{\eta^3}\alpha\theta + \frac{p^3}{\eta^4}\theta^2 \right) \\ &= \frac{2p\alpha\eta^3 + p^2\eta^2\theta + p\alpha^2\eta^2}{\eta^2(\eta^2 - p\theta)} \\ &= \frac{2p\alpha\eta + p^2\theta + p\alpha^2}{(\eta^2 - p\theta)} \end{aligned} \quad (3.22)$$

Supondo $g^t H^{-1} g = \delta^2$, $\delta \in (0, 1)$, podemos escolher $\epsilon \in (0, 1)$, tal que:

$$\delta^2 + \epsilon^2 < 1 \text{ e } \Delta \leq \epsilon^2$$

Deste modo, obteremos um novo sistema com proximidade $\sqrt{\delta^2 + \epsilon^2}$. Portanto, de 3.22 se obtém:

$$\epsilon^2\eta^2 - 2p\alpha\eta - p[\alpha^2 + \theta(\epsilon^2 + p)] \geq 0 \quad (3.23)$$

Resolvamos este trinômio do segundo grau em η . Do discriminante, definimos

$$\beta = \sqrt{p^2\alpha^2 + \epsilon^2p(\alpha^2 + \theta(\epsilon^2 + p))}$$

Portanto, para obter soluções positivas que verificam (3.23), devemos ter:

$$\eta \geq \frac{p\alpha + \beta}{\epsilon^2}$$

Chegamos, enfim, ao resultado do teorema, pela substituição da expressão de η .

$$\gamma \leq s^t u - r - \frac{p\alpha + \beta}{\epsilon^2}$$

■

3.3.4 O Sub-problema Escravo

A resolução do sub-problema escravo é tratada de duas formas neste capítulo: na primeira, o sub-problema é resolvido exatamente, usando o algoritmo apresentado no capítulo 2; na segunda forma, o sub-problema escravo é resolvido parcialmente, tal que seja satisfeito:

$$(c^\top - u_k^\top A)x_k(u_k) < r^k - u_k^\top b \quad (3.24)$$

Desta forma, são realizadas iterações do algoritmo até que seja encontrado um hiperplano de separação. No entanto, este hiperplano pode não existir e, então, o algoritmo, neste caso, resolverá o sub-problema exatamente.

O algoritmo para resolução aproximada é apresentado nas tabelas 3.2 e 3.6. Em seguida, apresentaremos alguns resultados teóricos que justificam o algoritmo.

Para obter esta solução sub-ótima, supõe-se conhecida uma solução inicial x^0 , próxima do centro analítico da região X , que será utilizada em todas as iterações do algoritmo.

Seja $z_k(\pi)$ o valor da função objetivo do problema dual associado ao problema escravo 3.7, avaliado no ponto π que minimiza a função potencial associada ao dual do problema 3.7. Do teorema da dualidade fraca, tem-se que:

$$\begin{aligned} (c^t - u_k^t A)x^* &\geq z_k(\pi) \Rightarrow \\ u_k^t b + (c^t - u_k^t A)x^* &\geq u_k^t b + z_k(\pi) \Rightarrow \\ D(u_k) &\geq u_k^t b + z_k(\pi) \end{aligned}$$

Seja $D_l(u_k) = u_k^t b + z_k(\pi)$, uma cota inferior para D . A fórmula de atualização

de $D_{\text{máx}}^{k+1}$ passa a ser

$$D_{\text{máx}}^{k+1} := \text{máx}\{D_l(u_k), D_{\text{máx}}^{k+1}\}$$

3.4 Aspectos Teóricos.

O algoritmo utiliza a propriedade de $D(u)$ ser côncava. Através de um procedimento que resolve apenas parcialmente o problema escravo 3.7, obtém uma solução sub-ótima, que dá origem a um ϵ -super-gradiente.

O lema 1 mostra que uma solução viável do problema escravo produz um ϵ -super-gradiente para $D(u)$ em u . Com esta solução, obtém-se $D_{upper}(u)$, uma cota superior para $D(u)$ em $(u_k, x_k(u_k))$.

O lema 2 mostra que com $D_{upper}(u_k)$ e, se $s \in \partial_\epsilon D(u_k)$, tem-se um corte válido:

$$(u^*, r^*) \in (u, r) \in \mathbb{R}^{m_1+1} : s^t u - r \geq s^t u_k - D_{upper}(u, k).$$

A seguir os lemas são demonstrados.

Lema 1 *Sejam $x \in X$ e u dados, então $s = Ax - b \in \partial_\epsilon D(u)$, para algum $\epsilon \geq 0$.*

Demonstração:

Para $w \geq 0$

$$\begin{aligned} D(w) &= -w^t b + \min\{(c^t + w^t A)y : y \in X\} \\ D(w) &\leq -w^t b + (c^t + w^t A)x \\ &= -w^t b + (c^t + w^t A)x - u^t b + u^t b - u^t Ax + u^t Ax \\ &= (w - u)^t (Ax - b) + (c^t + u^t A)x - u^t b \end{aligned} \tag{3.25}$$

Por outro lado, temos:

$$D(u) = -u^t b + \min\{(c^t + u^t A)y : y \in X\}$$

$$= -u^t b + (c^t + u^t A)x - \epsilon, \text{ para algum } \epsilon \geq 0.$$

Logo, substituindo a expressão acima em 3.25, obtemos:

$$D(w) \leq (w - u)^t (Ax - b) + D(u) + \epsilon,$$

portanto $s = (Ax - b) \in \partial_\epsilon D(u)$. ■

Lema 2 *Seja $s \in \partial_\epsilon D(u^k)$ e $D_{upper}(u_k) = D(u^k) + \epsilon$, então*

$$(u^*, r^*) \in \{(u, r) \in R^{m+1} : s^t u - r \geq s^t u_k - D_{upper}(u_k)\}.$$

Demonstração: Por definição, se $s \in \partial_\epsilon D(u)$, então

$$D(u) \leq s^t (u - u_k) + D(u_k) + \epsilon, \forall u > 0$$

Agora, usemos o fato de que $r \leq D(u)$ e a hipótese, para obter

$$r \leq s^t (u - u_k) + D_{upper}(u_k)$$

Observação: do Lema 1, $D_{upper}(u_k) = D(u^k) + \epsilon$

para $D_{upper}(u_k) = -u_k^t b + (c^t + u_k^t A)x_k$. ■

Lema 3 *Seja $x_k(u_k)$ uma solução sub-ótima do sub-problema*

$$\min\{(c^\top + u_k^\top A)x : x \in X\}$$

tal que $r^k > (c^\top + u_k^\top A)x_k(u_k) - u_k^\top b$, então para $s_{k+1} \in \partial_\epsilon D(u)$, tem-se

$$s_{k+1}^\top u_k - r^k < d_{k+1} = s_{k+1}^\top u_k - D_{upper}(u_k), \text{ onde } D_{upper}(u_k) = -u_k^t b + (c^t + u_k^t A)x_k(u_k).$$

Demonstração: Por hipótese, $r^k > (c^\top + u_k^\top A)x_k(u_k) - u_k^\top b$

$$\Rightarrow -r^k < -(c^\top + u_k^\top A)x_k(u_k) + u_k^\top b$$

$$\Rightarrow s_{k+1}^\top u_k - r^k < s_{k+1}^\top u_k - \underbrace{(c^\top + u_k^\top A)x_k(u_k) + u_k^\top b}_{-D_{upper}(u_k)}$$

$$\Rightarrow s_{k+1}^\top u_k - r^k < s_{k+1}^\top u_k - D_{upper}(u) = d_{k+1}$$
 ■

3.5 Problemas Tratados

A seguir apresentaremos uma lista de problemas que foram abordados para os algoritmos deste capítulo. O modelo é dado por 3.1. Eles serão apresentados em subseções com um nome associado a cada um. Maiores detalhes de implementações e resultados podem ser obtidos no capítulo 6.

Estamos chamando estes problemas de gerais, porque eles não possuem nenhuma estrutura especial e foram escolhidos de livros. A decomposição de cada um deles também foi feita sem levar em consideração qualquer estrutura especial, ou seja, o objetivo destes testes foi apenas de avaliar a performance do algoritmo e validar as implementações que foram realizadas ao longo deste trabalho. Para todos os exemplos a seguir $x_i \geq 0$.

3.5.1 Problema1 (P01)

$$c = \begin{bmatrix} -2 \\ -3 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 1 \\ 5 & 3 \end{bmatrix}$$

$$b = \begin{bmatrix} 15 \\ 20 \end{bmatrix}$$

$$D = \begin{bmatrix} 3 & -2 \\ 4 & -5 \end{bmatrix}$$

$$d = \begin{bmatrix} 4 \\ 10 \end{bmatrix}$$

3.5.2 Problema2 (P02)

$$c = \begin{bmatrix} -3 \\ -5 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$b = \begin{bmatrix} 4 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 1 \\ 3 & 2 \end{bmatrix}$$

$$d = \begin{bmatrix} 6 \\ 18 \end{bmatrix}$$

3.5.3 Problema3 (P03)

$$c = \begin{bmatrix} -6 \\ -8 \\ 5 \end{bmatrix}$$

$$A = \begin{bmatrix} -4 & -4 & -1 \\ 0 & 2 & -2 \end{bmatrix}$$

$$b = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & -2 & 2 \\ 2 & 2 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$d = \begin{bmatrix} 1 \\ 5 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

3.5.4 Problema4 (P04)

$$c = \begin{bmatrix} -2 \\ -1 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} 5 \end{bmatrix}$$

$$D = \begin{bmatrix} -1 & 1 \\ 6 & 2 \end{bmatrix}$$

$$d = \begin{bmatrix} 0 \\ 21 \end{bmatrix}$$

3.5.5 Problema5 (P05)

$$c = \begin{bmatrix} -4 \\ -5 \end{bmatrix}$$

$$A = [-1 \quad -4]$$

$$b = [-5]$$

$$D = [-3 \quad -2]$$

$$d = [-7]$$

3.5.6 Problema6 (P06)

$$c = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$A = [-1 \quad -2 \quad -3]$$

$$b = [-10]$$

$$D = \begin{bmatrix} -2 & -1 & 1 \\ 0 & 0 & -1 \end{bmatrix}$$

$$d = \begin{bmatrix} -20 \\ -4 \end{bmatrix}$$

3.5.7 Problema7 (P07)

$$c = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} 8 \end{bmatrix}$$

$$D = \begin{bmatrix} 2 & 1 \end{bmatrix}$$

$$d = \begin{bmatrix} 24 \end{bmatrix}$$

3.5.8 Problema8 (P08)

$$c = \begin{bmatrix} -5 \\ -6 \\ -9 \\ -8 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}$$

$$b = [6]$$

$$D = \begin{bmatrix} 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \end{bmatrix}$$

$$d = \begin{bmatrix} 5 \\ 3 \end{bmatrix}$$

Capítulo 4

Problema de Telecomunicações Móvel

O tratamento em conjunto dos problemas de localização de estações de rádio base (ERBs) e de controle de potência requer uma atenção especial na formulação das restrições e na escolha da função objetivo. Além de viabilizar isoladamente cada um dos problemas, o modelo deve conter uma variável que crie um elo entre eles, de modo a impedir falsas soluções, bem como a perda dos investimentos necessários em infra-estrutura. (Para mais detalhes de aplicações em redes de telecomunicações e decomposição ver [21][22].)

O modelo que tratamos baseia-se em um modelo básico de localização de ERBs e no modelo de controle de potência (MCP), desenvolvido em [29].

4.1 O Modelo

(P) :

$$\min \quad \sum_{i=1}^N c_i z_i + C \cdot \sum_{i=1}^N \sum_{j=1}^M p_{ij} \quad (4.1)$$

s.a :

$$\sum_{j=1}^M a_{ij} x_{ij} \leq \Lambda_i z_i \quad \forall i$$

$$p_{ij} \leq P_j x_{ij} \quad \forall i, \forall j$$

$$R_j x_{ij} \leq r_{ij} \quad \forall i, \forall j$$

$$Wh_j p_{ij} \geq (r_{ij} [(\sum_{k \neq j} a_{ik} h_k p_{ik}) + (\sum_{s \neq i} \sum_{t=1}^m a_{sj} h_t p_{st}) + \eta_0 W]) \gamma_j \quad \forall i, \forall j$$

$$\sum_{i=1}^N a_{ij} x_{ij} \geq 1 \quad \forall j$$

$$z_i \in \{0, 1\} \quad \forall i$$

$$x_{ij} \in \{0, 1\} \quad \forall i, \forall j$$

$$p_{ij} \geq 0 \quad \forall i, \forall j$$

$$r_{ij} \geq 0 \quad \forall i, \forall j$$

Primeiramente, dividimos a região em estudo em M quadrículas de tamanho variável, enumerando-as por $j = 1, \dots, M$. Nessa região, espalhamos N ERBs candidatas a instalação, indicando-as por $i = 1, \dots, N$. Cada ERB i tem um custo de instalação c_i e uma capacidade máxima, Λ_i , de quadrículas atendidas. Cada quadrícula j possui um valor máximo, P_j , para a potência de transmissão. Valores superiores a esse máximo podem danificar o equipamento do usuário, além de consumir mais bateria e aumentar a interferência, prejudicando a comunicação. Cada quadrícula define também um valor mínimo requerido, γ_j , para a qualidade de serviço e um valor mínimo de taxa de transmissão, R_j . Valores abaixo desses mínimos inviabilizam a comu-

nicação. Entretanto, valores muito superiores a R_j aumentam a quantidade de potência transmitida necessária para a comunicação com qualidade, tanto para a quadrícula em questão, como para as quadrículas interferentes, aumentando, conseqüentemente, o custo. Consideramos, ainda, as constantes W , largura da banda, η_0 , ruído gaussiano e C , custo da potência transmitida para ERB i .

O objetivo do modelo é minimizar os custos de instalação das estações rádio base e da transmissão de potência entre as ERBs e as quadrículas. Para homogeneizar a função objetivo como minimizadora de custo, o somatório da potência transmitida foi multiplicado pela constante C , que pode ser entendida como o custo de um watt.

O conjunto das variáveis do modelo pode ser separado em quatro conjuntos: z_i , variáveis de instalação, que assumem o valor unitário se a ERB i for instalada, ou nulo, caso contrário; x_{ij} , variáveis de cobertura, que assumem valor unitário, se a quadrícula j for coberta pela ERB i , ou nulo, caso contrário; p_{ij} , variáveis que indicam potência transmitida pelas quadrículas j para ERB i . Essas variáveis são contínuas, limitadas inferiormente por zero e superiormente por P_j . Observemos ainda que, quando a quadrícula j não é coberta pela ERB i , o valor de p_{ij} é igual a zero. Finalmente, o conjunto r_{ij} , variáveis relativas à taxa de transmissão entre as quadrículas e as ERBs. Assim como p_{ij} , essas variáveis são contínuas, porém limitadas apenas inferiormente por R_j , taxa mínima requerida por cada quadrícula j para viabilização da comunicação. Observemos que a taxa mínima, R_j , vem multiplicada por x_{ij} , fazendo com que o limite inferior para a taxa seja nulo, quando a quadrícula j não estiver sendo coberta pela ERB i . O limite superior para a taxa de transmissão está condicionado aos valores da potência de

transmissão, no segundo conjunto de restrições.

Pelo primeiro conjunto de restrições, o somatório dos sinais originários de cada ERB i que chegam nas demais quadrículas não pode ultrapassar a sua capacidade máxima de atendimento. Mais ainda, cada restrição considera a possibilidade da ERB em questão não ser instalada. Nesse caso, o somatório anterior é forçosamente nulo.

O quarto conjunto de restrições garante a qualidade de serviço requerida por cada quadrícula. Além disso, cada restrição leva em conta o fato da ERB i não cobrir a quadrícula j . Nesse caso, r_{ij} seria nulo, assim como a qualidade de serviço requerida. Consequentemente, para esse caso, teríamos imediatamente a potência transmitida igual a zero.

O quinto conjunto de restrições garante que cada quadrícula é coberta pelo menos por uma ERB. Isso significa que o sinal naquela quadrícula, originário de alguma ERB, é suficiente.

4.2 O Modelo Linearizado

Nessa seção, apresentaremos uma simplificação do modelo não linear 4.1, de modo a reescrevê-lo em uma forma linear. A nossa estratégia foi a seguinte: o único conjunto de restrições não-lineares foi substituído por sua aproximação de Taylor de primeira ordem. Dessa forma, obtemos o seguinte modelo:

(P) :

$$\min \quad \sum_{i=1}^N c_i z_i + C \cdot \sum_{i=1}^N \sum_{j=1}^M p_{ij} \quad (4.2)$$

s.a :

$$\sum_{i=1}^N a_{ij} x_{ij} \geq 1 \quad \forall j$$

$$\sum_{j=1}^M a_{ij} x_{ij} \leq \Lambda_i z_i \quad \forall i$$

$$W h_j p_{ij} - \gamma_j r_{ij}^0 (\sum_{k \neq j} a_{ik} h_k p_{ik} + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st})$$

$$- \gamma_j r_{ij} (\sum_{k \neq j} a_{ik} h_k p_{ik}^0 + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st}^0 + \eta_0 W)$$

$$+ \gamma_j r_{ij}^0 (\sum_{k \neq j} a_{ik} h_k p_{ik}^0 + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st}^0) \geq 0 \quad \forall i, \forall j$$

$$p_{ij} \leq P_j x_{ij} \quad \forall i, \forall j$$

$$R_j x_{ij} \leq r_{ij} \quad \forall i, \forall j$$

$$z_i \in \{0, 1\} \quad \forall i$$

$$x_{ij} \in \{0, 1\} \quad \forall i, \forall j$$

$$p_{ij} \geq 0 \quad \forall i, \forall j$$

$$r_{ij} \geq 0 \quad \forall i, \forall j$$

4.3 Decomposição via Relaxação Lagrangeana

Relaxando o segundo conjunto de restrições e associando a este conjunto o vetor de multiplicadores não-negativos $u \in R^N$, temos a função dual dada por:

6.

Nas próximas subseções, apresentaremos os dois sub-problemas escravos. E, na última seção deste capítulo, apresentaremos uma lista de testes que tratamos. Os resultados computacionais podem ser consultados no capítulo

onde $L(z, p, n)$ é a função de Lagrange correspondente.

$$\begin{aligned}
 & \sum_{i=1}^N a_{ij} x_{ij} \geq 1 & A_j \\
 & W h_j p_{ij} - \gamma_j r_{ij}^0 \left(\sum_{k \neq j} a_{ik} h_k p_{ik} + \sum_{t=1}^M a_{sj} h_t p_{st}^0 \right) + \gamma_j r_{ij}^0 \left(\sum_{k \neq j} a_{ik} h_k p_{ik}^0 + \sum_{t=1}^M a_{sj} h_t p_{st}^0 \right) \geq 0 & A_i, A_j \\
 & p_{ij} \leq P_j x_{ij} & A_i, A_j \\
 & R_j x_{ij} \leq r_{ij} & A_i, A_j \\
 & z_i \in \{0, 1\} & A_i \\
 & x_{ij} \in \{0, 1\} & A_i, A_j \\
 & d_{ij} \geq 0 & A_i, A_j \\
 & r_{ij} \geq 0 & A_i, A_j
 \end{aligned}$$

s.o.:

$$\begin{aligned}
 D(n) := \min & L(z, p, n) = \sum_{i=1}^N c_i z_i + C \cdot \sum_{i=1}^N \sum_{j=1}^M d_{ij} p_{ij} \\
 & + \sum_{i=1}^N u_i (-V_i z_i + \sum_{j=1}^M a_{ij} x_{ij}) \\
 & + \sum_{i=1}^N u_i (-V_i z_i + \sum_{j=1}^M a_{ij} x_{ij})
 \end{aligned} \tag{4.3}$$

(D):

4.3.1 Subproblema em z

(SPz) :

$$\min \sum_{i=1}^N (c_i - u_i \Lambda_i) z_i \quad (4.4)$$

s.a :

$$z_i \in \{0, 1\} \quad \forall i$$

4.3.2 Subproblema em (p, r, x)

(P) :

$$\min \sum_{i=1}^N \sum_{j=1}^M u_i a_{ij} x_{ij} + C \cdot \sum_{i=1}^N \sum_{j=1}^M p_{ij} \quad (4.5)$$

s.a :

$$\sum_{i=1}^N a_{ij} x_{ij} \geq 1 \quad \forall j$$

$$W h_j p_{ij} - \gamma_j r_{ij}^0 (\sum_{k \neq j} a_{ik} h_k p_{ik} + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st})$$

$$- \gamma_j r_{ij} (\sum_{k \neq j} a_{ik} h_k p_{ik}^0 + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st}^0 + \eta_0 W)$$

$$+ \gamma_j r_{ij}^0 (\sum_{k \neq j} a_{ik} h_k p_{ik}^0 + \sum_{s \neq i} \sum_{t=1}^M a_{sj} h_t p_{st}^0) \geq 0 \quad \forall i, \forall j$$

$$p_{ij} \leq P_j x_{ij} \quad \forall i, \forall j$$

$$R_j x_{ij} \leq r_{ij} \quad \forall i, \forall j$$

$$x_{ij} \in \{0, 1\} \quad \forall i, \forall j$$

$$p_{ij} \geq 0 \quad \forall i, \forall j$$

$$r_{ij} \geq 0 \quad \forall i, \forall j$$

O subproblema em z pode ser resolvido por inspeção. Quanto ao subproblema em (p, r, x) , baseado no fato de que o conjunto $\{x_{ij} : 0 \leq x_{ij} \leq 1 \ \forall i, j \text{ e } \sum_{i=1}^N a_{ij}x_{ij} \geq 1, \forall j\}$ satisfaz a propriedade da integralidade (veja [20]), resolvemos partir desta hipótese para o caso dado pelas restrições em (4.5), onde apenas as variáveis x_{ij} são inteiras. Nos testes que realizamos, esta hipótese se verificou correta. Pudemos, assim, utilizar o algoritmo de pontos interiores (contínuos) para este sub-problema.

4.4 Problemas Tratados

Cada problema apresentado foi testado com o CPLEX e com os algoritmos que tratamos ao longo deste trabalho. Eles serão apresentados em subseções com um nome associado a cada um. Maiores detalhes de implementações e resultados podem ser obtidos no capítulo 6.

4.4.1 Problema1 (Pt01)

$$\eta_0 = 10^{-6}$$

$$C = 80$$

$$c = \begin{bmatrix} 350 \\ 300 \\ 350 \end{bmatrix}$$

$$\Lambda = [5 \ 5 \ 7]$$

$$P = [5.0 \ 4.8 \ 4.9 \ 5.0 \ 5.2 \ 5.0]$$

$$R = [7.9 \ 8 \ 7.9 \ 8.1 \ 8 \ 8]$$

$$H = [0.25 \ 0.24 \ 0.25 \ 0.26 \ 0.24 \ 0.23]$$

$$\gamma = [0.5 \ 0.6 \ 0.7 \ 0.6 \ 0.7 \ 0.5]$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

4.4.2 Problema2 (Pt02)

$$\eta_0 = 10^{-6}$$

$$C = 80$$

$$c = \begin{bmatrix} 29 \\ 30 \\ 37 \\ 36 \end{bmatrix}$$

$$\Lambda = [4 \ 4 \ 5 \ 6]$$

$$P = [0.5 \ 0.48 \ 0.49 \ 0.50 \ 0.52 \ 0.50 \ 0.48 \ 0.49]$$

$$R = [7.9 \ 8 \ 7.9 \ 8.1 \ 8 \ 8 \ 7.8 \ 7.9]$$

$$H = [0.25 \ 0.24 \ 0.25 \ 0.26 \ 0.24 \ 0.23 \ 0.23 \ 0.26]$$

$$\gamma = [0.5 \ 0.6 \ 0.7 \ 0.6 \ 0.7 \ 0.5 \ 0.7 \ 0.6]$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

4.4.3 Problema3 (Pt03)

$$\eta_0 = 10^{-6}$$

$$C = 80$$

$$c = \begin{bmatrix} 29 \\ 30 \\ 37 \\ 36 \\ 43 \end{bmatrix}$$

$$\Lambda = [4 \ 4 \ 5 \ 6 \ 2]$$

$$P = [0.5 \ 0.48 \ 0.49 \ 0.50 \ 0.52 \ 0.50 \ 0.48 \ 0.49]$$

$$R = [7.9 \ 8 \ 7.9 \ 8.1 \ 8 \ 8 \ 7.8 \ 7.9]$$

$$H = [0.25 \ 0.24 \ 0.25 \ 0.26 \ 0.24 \ 0.23 \ 0.23 \ 0.26]$$

$$\gamma = [0.5 \ 0.6 \ 0.7 \ 0.6 \ 0.7 \ 0.5 \ 0.7 \ 0.6]$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

4.4.4 Problema4 (Pt04)

$$\eta_0 = 10^{-6}$$

$$C = 80$$

$$c = \begin{bmatrix} 29 \\ 30 \\ 37 \\ 36 \\ 43 \end{bmatrix}$$

$$\Lambda = [4 \ 4 \ 5 \ 6 \ 2]$$

$$P = [0.5 \ 0.48 \ 0.49 \ 0.50 \ 0.52 \ 0.50 \ 0.48 \ 0.49 \ 0.53]$$

$$R = [7.9 \ 8 \ 7.9 \ 8.1 \ 8 \ 8 \ 7.8 \ 7.9 \ 7.1]$$

$$H = [0.25 \ 0.24 \ 0.25 \ 0.26 \ 0.24 \ 0.23 \ 0.23 \ 0.26 \ 0.25]$$

$$\gamma = [0.5 \ 0.6 \ 0.7 \ 0.6 \ 0.7 \ 0.5 \ 0.7 \ 0.6 \ 0.7]$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Capítulo 5

Problema de Associação Generalizada

O problema de associação generalizada consiste em associar n tarefas a m agentes, tal que cada tarefa esteja associada apenas a um agente, mas cada agente é limitado apenas pela quantidade de recurso. Seja c_{ij} o lucro de associar a tarefa j ao agente i , seja a_{ij} o recurso necessário pelo agente i para realizar a tarefa j e seja b_i o recurso disponível para o agente i . Suponha que uma variável 0 – 1, x_{ij} , é 1, se a tarefa j é associada ao agente i , 0 caso o contrário. Então, uma formulação de programação inteira para o problema de associação generalizada é:

(P) :

$$\max \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (5.1)$$

s.a :

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i$$

$$\sum_{i=1}^m a_{ij} x_{ij} \leq b_j \quad \forall j$$

$$x_{ij} \in \{0, 1\} \quad \forall i = 1..m \quad \forall j = 1..n$$

5.1 Relaxação Lagrangeana

Estimulados pelos resultados obtidos na resolução de pequenos problemas na formulação geral usando o algoritmo apresentado no capítulo 3 e devido aos resultados obtidos em [18], resolvemos aplicar o algoritmo de centros analíticos para o problema de associação generalizada usando a decomposição proposta em [32] e no artigo [18], ou seja:

(P) :

$$L(x, u) = \max \sum_{i=1}^m \sum_{j=1}^n (c_{ij} - u_i) x_{ij} + \sum_{i=1}^m u_i \quad (5.2)$$

s.a :

$$\sum_{i=1}^m a_{ij} x_{ij} \leq b_j \quad \forall j$$

$$x \in \{0, 1\}$$

Para esta decomposição, é necessário resolver n problemas da mochila para o subproblema escravo, para que, utilizamos a técnica de programação dinâmica conforme apresentado em [32]. Para resolver o problema mestre, tentamos aplicar o algoritmo apresentado no capítulo 3, no entanto tivemos dificuldades com o algoritmo de reinicialização. Fizemos, então, uma adaptação do algoritmo de acordo com o trabalho apresentado em [8]. Em seguida, mostraremos em detalhes o algoritmo adaptado.

5.2 Algoritmo Adaptado

Conforme explicitamos na seção anterior, o algoritmo do capítulo 3 apresentou alguns problemas para obter um novo ponto interior. Nesta seção, apresentaremos o algoritmo adaptado (veja a tabela 5.1).

A adaptação consiste na recuperação de um novo ponto interior. Sub-

stituimos o método computar γ e Reduzir γ até d_{k+1} pelo método *BIG – M*. Computacionalmente, esta substituição foi muito eficiente, pois este método utilizou poucas iterações para obter um ponto interior próximo à trajetória central.

5.2.1 O Algoritmo

Algoritmo para Programação Convexa

Obtenha $u^0 \in \text{int}(S_{\rho^k})$

{Resolver Problema(s) Escravo(s)}

$z_{inferior} = Du$

{Obter um ponto interior para o problema mestre (*BIG-M*).}

{Obter o centro analítico resolvendo o problema mestre.}

$z_{superior} = r$

$k = 0$

repita

se $(z_{superior} - z_{inferior}) < \epsilon$

então Terminar

fim-se

{Resolva o sub-problema escravo}

Compute $D(u^k)$, $s_{k+1} \in \partial D(u^k)$ e $d_{k+1} = s_{k+1}u^k - D(u^k)$

$z_{inferior} = \max\{z_{inferior}, D(u)\}$

{Incorpora Corte e atualiza o conjunto de localização}

{Obter um ponto interior para o novo conjunto de localização (*BIG -M*).}

{Obter o centro analítico resolvendo o problema mestre.}

$z_{superior} = \min\{z_{superior}, r\}$

$k = k + 1$

fim-repita

Tabela 5.1: algoritmo de centros analíticos para programação convexa

5.3 Problemas Tratados

A seguir apresentaremos a lista dos problemas que foram abordados. Os resultados computacionais estão no capítulo 6.

5.3.1 Problema1 (Pg01)

$$c = \begin{bmatrix} 110 & 16 & 25 & 78 & 59 \\ 65 & 69 & 54 & 28 & 71 \\ 19 & 93 & 45 & 45 & 9 \\ 89 & 31 & 72 & 83 & 20 \\ 62 & 17 & 77 & 18 & 39 \\ 37 & 115 & 87 & 59 & 97 \\ 89 & 102 & 98 & 74 & 61 \\ 78 & 96 & 87 & 55 & 77 \\ 74 & 27 & 99 & 91 & 5 \\ 88 & 97 & 99 & 99 & 51 \end{bmatrix}$$

$$A = \begin{bmatrix} 95 & 1 & 21 & 66 & 59 \\ 54 & 53 & 44 & 26 & 60 \\ 3 & 91 & 43 & 42 & 5 \\ 72 & 30 & 56 & 72 & 9 \\ 44 & 1 & 71 & 13 & 27 \\ 20 & 99 & 87 & 52 & 85 \\ 72 & 96 & 97 & 73 & 49 \\ 75 & 82 & 83 & 44 & 59 \\ 68 & 8 & 87 & 74 & 4 \\ 69 & 83 & 98 & 88 & 45 \end{bmatrix}$$

$$b = \begin{bmatrix} 91 \\ 87 \\ 109 \\ 88 \\ 64 \end{bmatrix}$$

5.3.2 Problema2 (Pg02)

$$c = \begin{bmatrix} 6 & 10 & 1 \\ 12 & 12 & 5 \\ 15 & 4 & 3 \\ 10 & 3 & 9 \\ 8 & 9 & 5 \end{bmatrix}$$

$$A = \begin{bmatrix} 5 & 7 & 2 \\ 14 & 8 & 7 \\ 10 & 6 & 12 \\ 8 & 4 & 15 \\ 6 & 12 & 5 \end{bmatrix}$$

$$b = \begin{bmatrix} 15 \\ 15 \\ 15 \end{bmatrix}$$

5.3.3 Problema3 (Pg03)

$$c = \begin{bmatrix} 19 & 23 & 18 & 21 & 12 \\ 14 & 22 & 19 & 23 & 23 \\ 21 & 18 & 21 & 23 & 21 \\ 19 & 21 & 25 & 25 & 17 \\ 10 & 19 & 17 & 10 & 22 \\ 24 & 12 & 10 & 22 & 22 \\ 13 & 13 & 22 & 16 & 18 \\ 11 & 10 & 14 & 13 & 11 \\ 22 & 18 & 25 & 19 & 24 \\ 16 & 19 & 22 & 12 & 24 \end{bmatrix}$$

$$A = \begin{bmatrix} 24 & 5 & 15 & 22 & 19 \\ 18 & 18 & 11 & 24 & 5 \\ 5 & 11 & 15 & 19 & 20 \\ 8 & 13 & 14 & 12 & 9 \\ 24 & 22 & 24 & 13 & 11 \\ 23 & 20 & 18 & 22 & 22 \\ 7 & 14 & 12 & 15 & 14 \\ 24 & 19 & 12 & 16 & 18 \\ 15 & 21 & 8 & 12 & 5 \\ 18 & 9 & 21 & 23 & 16 \end{bmatrix}$$

$$b = \begin{bmatrix} 26 \\ 24 \\ 24 \\ 28 \\ 22 \end{bmatrix}$$

Capítulo 6

Experiência Numérica

As implementações de todos os algoritmos apresentados nesta dissertação foram feitas em MATLAB 5.0, sendo executados em um PC PENTIUM III 850 MHz com 128 MB de memória RAM no ambiente Windows 98.

Primeiramente, foram implementados os algoritmos propostos em [28] resolvendo parcialmente o problema mestre e resolvendo exatamente e aproximadamente os sub-problemas escravos, para resolver problemas gerais conforme definimos no capítulo 3, afim de verificar a confiabilidade da nossa implementação do método. Os resultados obtidos podem ser consultados nas tabelas 6.1 e 6.2. Esses resultados, no entanto, devem ser analisados sob outras perspectivas além dos meros resultados numéricos que estão expostos nas tabelas, pois, como analisaremos no capítulo 7, os ajustes dos diversos parâmetros são essenciais para se atingir uma boa performance computacional. Em seguida, passamos a testar os algoritmos de planos de corte para resolver os problemas dos capítulos 4 e 5. Como o algoritmo de reinicialização, exposto no capítulo 3, não apresentou uma boa performance, realizamos as adaptações, conforme apresentados no capítulo 5 na seção 5.2. E foi com este algoritmo que fizemos os testes com o problema de associação

generalizada e com o problema de telecomunicações móvel. Os resultados obtidos com este método que adaptamos podem ser consultados nas tabelas 6.6. Comparamos, ainda, este método com os mesmos problemas do capítulo 3 que podem ser consultados na tabela 6.3. Observação: todos os resultados relacionados a tempo estão em segundos.

6.1 Inicialização

Para inicializar o método de planos de corte, foi implementado o método Big-M utilizando centros analíticos para gerar uma solução viável para o problema primal; em seguida, foram estabelecidos limites inferiores e superiores para as variáveis duais e dentro destes limites, obteve-se uma solução viável. Com esta solução, então, faz-se uma chamada ao oráculo, gerando um subgradiente, $D(u)$ e d_i . Por fim, chama-se o método de centros analíticos para obter uma solução próxima da trajetória central, ou seja, com $\delta_{\rho^0}(u^0, r^0) \in (0, 1)$ e, então, se realizam as iterações do problema mestre e do sub-problema escravo até que seja atingida a tolerância desejada.

6.2 O método branch-bound com centros analíticos

Para poder fazer comparações de eficiência com o método de planos de corte, foi implementado o método de branch-bound com centros analíticos para resolver apenas o problema de telecomunicações móvel. Os resultados podem ser obtidos na tabela 6.5. A principal característica nesta implementação está na recuperação de um novo ponto interior, onde usamos o mesmo algoritmo que foi utilizado para resolver os problemas de associação

generalizada e de telecomunicações móvel, ou seja, o método Big-M com centros analíticos.

Na parte final deste capítulo, estão listadas várias tabelas com suas respectivas legendas onde podem ser consultados os resultados obtidos utilizando-se dos métodos expostos nesta dissertação.

Problema	C1	C2	C3	C4	C5
<i>P01</i>	1	3	69	$1,8 * 10^{-5}$	62,340
<i>P02</i>	1	3	63	$9,00 * 10^{-5}$	61,730
<i>P03</i>	1	2	57	$9,17 * 10^{-4}$	11,499
<i>P04</i>	1	3	52	$8,91 * 10^{-4}$	37,29
<i>P05</i>	1	3	54	$8,55 * 10^{-4}$	51,349
<i>P06</i>	1	3	58	$8,53 * 10^{-3}$	62,94
<i>P07</i>	1	3	46	$9,16 * 10^{-3}$	39,44
<i>P08</i>	1	3	45	$8,57 * 10^{-3}$	44,66

- C1 ⇒ Número médio de iterações para obter um ponto interior
- C2 ⇒ Número médio de iterações para obter um centro analítico
- C3 ⇒ Número de iterações do programa principal (problema mestre)
- C4 ⇒ gap de dualidade
- C5 ⇒ Tempo de execução do método de centros analíticos

Tabela 6.1: Resultados para problemas gerais com o algoritmo do capítulo 3, com escravos resolvidos exatamente

Problema	C1	C2	C3	C4	C5	C6
P01	1	3	76	$9,80 * 10^{-5}$	466,20	-19,97
P02	1	2	200	$1,24 * 10^{-4}$	834,54	-35,99
P03	1	2	8	$1,1 * 10^{-3}$	32,60	-11,499
P04	1	3	53	$8,09 * 10^{-4}$	35,05	-75,18
P05	1	3	55	$9,156 * 10^{-4}$	31,69	-11,20
P06	1	3	58	$8,36 * 10^{-3}$	44,76	-16,00
P07	1	3	56	$9,89 * 10^{-3}$	60,95	-14,05
P08	1	3	45	$8,78 * 10^{-3}$	324,56	-16,999

- C1 \Rightarrow Número médio de iterações para obter um ponto interior
C2 \Rightarrow Número médio de iterações para obter um centro analítico
C3 \Rightarrow Número de iterações do programa principal (problema mestre)
C4 \Rightarrow gap de dualidade
C5 \Rightarrow Tempo de execução
C6 $\Rightarrow c^T x^*$

Tabela 6.2: Resultados para problemas gerais com o algoritmo do capítulo 3, com escravos resolvidos aproximadamente

Problema	C1	C2	C3	C4	C5	C6
P01	3	7	3	2.58	-19,208	$1.044.10^{-4}$
P02	800	1	1	1.43	-35.999	$1.349.10^{-4}$
P03	166	36	3	1.70	-11.499	$8.736.10^{-5}$
P04	123	17	3	1.05	-6.999	$1.415.10^{-4}$
P05	56	28	3	0.770	-13	$2.80.10^{-4}$
P06	806	1	2	1.449	-16	$6.61.10^{-5}$
P07	64	30	3	0.76	-13,93	$2.26.10^{-4}$
P08	713	1	2	1.38	-16.999	$1.26.10^{-4}$

- C1 \Rightarrow Número médio de iterações para obter um novo ponto interior
C2 \Rightarrow Número médio de iterações de Newton para obter o novo centro analítico
C3 \Rightarrow Número de iterações do programa principal (problema mestre)
C4 \Rightarrow Tempo total de execução do método de centros analíticos
C5 $\Rightarrow c^T x^*$
C6 \Rightarrow gap de dualidade

Tabela 6.3: Resultados para problemas gerais com o algoritmo modificado

Problema	C1	C2	C3	C4	C5	C6	C7	C8
<i>Pg01</i>	50	10	15	14	87	$1.310 * 10^{-6}$	41	44.440
<i>Pg02</i>	15	5	8	69	66	$1.88 * 10^{-6}$	14	5.8990
<i>Pg03</i>	50	10	15	80	99	$8 * 10^{-11}$	32	38.880

C1 \Rightarrow Número de variáveis inteiras

C2 \Rightarrow Número de restrições de igualdade

C3 \Rightarrow Número de total de restrições

C4 \Rightarrow Número médio de iterações de Newton para obter novo ponto interior 1

C5 \Rightarrow Número médio de iterações de Newton para obter novo centro analítico

C6 \Rightarrow gap de dualidade

C7 \Rightarrow Número de iterações do programa principal (problema mestre)

C8 \Rightarrow Tempo de execução do método de Centros Analíticos

Tabela 6.4: Resultados do Problema de Associação Generalizada

Problema	C1	C2	C3
<i>Pt01</i>	650,0094	41	572,90
<i>Pt02</i>	65,0127	15	854,10
<i>Pt03</i>	65,0127	5	647,60
<i>Pt04</i>	65,0143	5	959,60

C1 \Rightarrow Valor ótimo

C2 \Rightarrow Número de nós necessários para encontrar a solução ótima

C3 \Rightarrow Tempo de execução

Tabela 6.5: Resultados do método branch-bound para o problema de telecomunicações

Problema	C1	C2	C3	C4	C5	C6	C7	C8
<i>Pt01</i>	57	21	90	21	59	6,9800	$8,390 * 10^{-5}$	7
<i>Pt02</i>	100	36	160	15	51	42,7500	$2,550.10^{-5}$	12
<i>Pt03</i>	125	45	200	15	54	112,3200	$3 * 10^{-5}$	17
<i>Pt04</i>	140	50	225	15	55	167,910	6.10^{-5}	17

C1 \Rightarrow Número de variáveis

C2 \Rightarrow Número de variáveis inteiras

C3 \Rightarrow Número de restrições de desigualdade

C4 \Rightarrow Número médio de iterações de Newton para obter novo ponto interior

C5 \Rightarrow Número médio de iterações de Newton para obter novo centro analítico

C6 \Rightarrow Tempo de execução do método de Centros Analíticos

C7 \Rightarrow gap de dualidade

C8 \Rightarrow Número de iterações do programa principal (problema mestre)

Tabela 6.6: Resultados do algoritmo modificado para o problema de telecomunicações

Capítulo 7

Conclusões

Existem muitos pontos a serem abordados em um trabalho de implementação. O primeiro aspecto que deve ser analisado é sobre as características do problema a ser tratado, pois a observação de algumas propriedades pode facilitar os cálculos. Por exemplo, o problema do capítulo 4 possui a propriedade da integralidade para as componentes x_{ij} , o que facilitou os cálculos no uso do método de branch-bound, pois as únicas variáveis com as quais tínhamos que nos preocupar eram as variáveis z_i . Outro ponto importante a ser analisado é sobre a medida de tolerância.

No início de nosso trabalho de implementação, trabalhávamos com aproximações da ordem de 10^{-6} . No entanto, à medida que fazíamos os testes, verificamos que não era necessária uma precisão tão rigorosa, pois, os resultados obtidos com uma precisão menor, apresentavam pequenas diferenças com os de uma precisão muito rigorosa, e, portanto, passamos a trabalhar com tolerâncias na ordem de 10^{-3} , obtendo bons resultados e tempos de execução inferiores aos obtidos anteriormente. Outro aspecto importante que deve ser levado em conta é a respeito do procedimento de recuperação de um ponto interior quando um novo corte é gerado através da resolução dos sub-

problemas escravos, pois este foi o trecho do programa onde foi dispendido o maior número de iterações para os problemas dos capítulos 4 e 5, impedidos de testar os algoritmos do capítulo 3 para estes problemas e levando-nos a fazer adaptações dos algoritmos deste capítulo com o trabalho em [25], como explicado no capítulo 5.

Outro fato que deve ser levado em conta está na inversão da hessiana. Implementamos os métodos de Cholesky e de gradientes conjugados. O que teve uma melhor performance foi o método de gradientes conjugados. Nesse momento, vale ressaltar a necessidade da implementação de métodos que ajudem a garantir a estabilidade numérica, como as matrizes pré-condicionadoras.

Vale a pena chamar a atenção, ainda, para os ajustes nos diversos parâmetros, como θ , por exemplo. Para os cortes na função objetivo, temos:

$$\rho^{k+1} = \rho^k + \theta(c^t x^k - \rho^k)$$

Observe que se θ estiver próximo a 1, ρ^{k+1} será próximo do valor atual da função objetivo, o que, por sua vez, vai provocar muitas iterações internas do método de Newton, por outro lado, se θ estiver próximo a zero, ter-se-á ρ^{k+1} próximo ao valor de ρ^k , o que provocará muitas iterações externas do método.

Notemos que os resultados computacionais podem melhorar, se forem utilizadas técnicas que levem em conta as características da hessiana do problema dual, como feito em [24].

Por fim, reafirmamos que, da experiência computacional não se pode garantir que o método do capítulo 3 não seja competitivo. Mais testes são necessários, para que sejam avaliados os efeitos dos diversos parâmetros.

Bibliografia

- [1] BERTSEKAS, D. P., *Nonlinear Programming*. Belmont, Massachusetts, Athena Scientific, 1995.
- [2] BOCANEGRA, S., *Solução do problema não-linear de fluxo multiproducto utilizando pontos interiores*. Tese de mestrado, Departamento de Ciência da Computação/UFMG , Belo Horizonte, MG, Brasil, 2001.
- [3] BEAN, J. C., NOON, C. E., RYAN, S. M., SALTON, G. J., "Selecting Tenants in a Shopping Mall". *Interfaces*, 18, 1-9, 1988.
- [4] CHAVTAL, V., *Linear Programming*. Freeman, York, 1980,.
- [5] CHENEY, W., GOLDSTEIN, A. A., "Newton's method for convex programming and Chebyshev approximation". *Numerische Mathematik*, pp.253-268, 1-5(1959).
- [6] DANTZIG, G. B., WOLFE, P., "The decomposition algorithm for linear programs". *Operations Research*, 8:101-111, 1960.
- [7] den Hertog, D., Roos, C. and Terlaky, T., "A potencial reduction variant of Renegar's short-step path-following method for linear programming". *Linear Algebra and its Application*, 152:43-68, 1991.
- [8] ELHEDHLI, S., GOFFIN, J. -L, "The integration of an interior-point cutting plane method within a branch-and-price algo-

ritm". *GERARD/Faculty of Management*, McGill University, Montreal, CA, march, 2001.

- [9] FELJOO B., SÁNCHEZ, A., GONZAGA C.C., "Mantaining closedness to the analytic center of a polytope by perturbing added hyperplanes". *Journal of Applied Mathematics and Optimization*, 2, pp.22-32, 1997.
- [10] FIACCO, A. V., MCCORMICK, G. P., "Nonlinear Programming: Sequential Unconstrained Minimization Techniques". *Research Analysis Corporation*, McLean Viginia, 1968. Republished in 1990 by SIAM, Philadelphia.
- [11] FISHER M. L., JAIKUMAR, R., "A Generalized Assignment Heuristic for Vehicle Routing", *Networks*, 11, pp.109-124,1981.
- [12] GONDZIO, J., SARKISSIAN, R., VIAL, J. -P., "Using an interior point method for the master problem in a decomposition approach". *Logilab,HEC, Section of management studies*, University of Geneva, Technical report, 1996.
- [13] GOFFIN, J. -L., GONDZIO, J., SARKISSIAN, R., VIAL, J. -P., "Solving Nonlinear multicommodity flow problems by the analytic center cutting plane method". *Math. Programming*, 76:136-154, 1997.
- [14] GOFFIN, J. L., VIAL, J. P., "Shallow, deep and very deep cuts in the analytic center cutting plane method". *Mathematical Programming* 84,pp 89-103 , 1999.
- [15] GONDZIO, J.,MERLE, O.D., SARKISSIAN, R., VIAL, J. -P., "ACCPM - A library for convex optmization based on an analytic center cutting plane method". *Logilab/HEC, section of management studies*, University of geneva, Technical report, 1995.

- [16] GONDZIO, J., MERLE, O.D., "Analytic center cutting plane method, user's guide for the library". *Logilab/HEC, section of management studies*, University of Geneva, Technical report, 1995.
- [17] HUARD, P., "Resolution of Mathematical Programming with Nonlinear Constraints by the Method of Centres". In: *Nonlinear Programming*, J. Naddie ed., North-Holland Publish Company, Amsterdam, Holland, 207-219.
- [18] KARABAKAL, N., Lohmann, J.R., "A Steepest Descent Multiplier Adjustment Method for the Generalized Assignment Problem". Technical Report 92-11, Department of Industrial and Operations Engineering, the University of Michigan, Ann Arbor, 1992.
- [19] KELLEY, J.E., "the cutting plane method for solving convex programs", *Journal of the SIAM* 8, pp. 703-712, (1960).
- [20] LEONE, R. J. G., *Modelagem e otimização de um sistema de telecomunicações sem fio*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2002.
- [21] LISSER, A., SARKISSIAN, R., VIAL, J. P., "Survivability in telecommunication networks". Technical report, 1995.
- [22] LISSER, A., SARKISSIAN, R., VIAL, J. P., *Optimal joint synthesis of base and reserve telecommunication networks*, 1995.
- [23] MARTINEZ, J. M., SANTOS, S.A., *Métodos Computacionais de Otimização*. XX Colóquio brasileiro de matemática, pp. 25-153, 1995.
- [24] MERLE, D. O., *Points intérieurs et plans coupants: mise en oeuvre et développement d'une méthode pour l'optimisation convexe et la pro-*

grammation linéaire structurée de grande taille. Thèse présentée à la Faculté des Sciences Économiques et Sociales l'Université de Genève, Département HEC pour l'obtention du grade de Docteur ès économiques et sociales mention économie commercial et industrielle. Genève, 1995.

- [25] MOKHTARIAN, F. S., GOFFIN, J. L., "A Nonlinear analytic center cutting plane method for a class of convex programming problems". SIAM Journal Optimization, 8, pp. 1108-1131, 1998.
- [26] SANTOS, M.A., OLIVEIRA, P. R., "Interior point algorithms for decomposition of Dantzig-Wolfe". *Nonconvex optimization and its Applications*, series advances in Convex Analysis and Global Optimization, Dordrecht/Boston/London, aceito.
- [27] OLIVEIRA, P. R., SANTOS, M.A., "Using analytic center and cutting planes methods for nonsmooth convex programming". *Lectures Notes in Economics and mathematical Systems*, 481:339-356, 2000.
- [28] SANTOS, M.A., *Métodos de centros para a programação convexa*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 1998.
- [29] SAMPATH, A., KUMAR, P. S., HOLTZMAN, J. M., "power control and resource management for a multimedia CDMA wireless system".
- [30] SHOR, N. Z., "Subgradient methods: A survey of Sovietic research, nonsmooth optimization". *Proceedings of the IIASA workshop* march 28 - April 8, 1997, C. Lemaréchal and R. Mifflin eds., Pergamon Press, 1978.
- [31] VIAL, J. -P., "A generic path-following algorithm with a sliding constraint and its application to linear programming and the computation

of analytic centers”.*Logilab/HEC, section of management studies*, University of Geneva, 1996.

[32] WOLSEY, L. A., *Integer Programming*. Wiley-Interscience publication, cap.10, pgs.167-183, 1998.