

Luís Filipe de Castro Nunes Vicente

Programação
de
Dois Níveis

Departamento de Matemática
Faculdade de Ciências e Tecnologia
Universidade de Coimbra
1992

Trabalho desenvolvido na área de Programação Matemática com vista à realização de Provas de Aptidão Pedagógica e Capacidade Científica.

Agradecimentos

Ao Professor Joaquim João Júdice, mestre e amigo, não só porque sem a sua preciosa orientação este trabalho não se realizaria, mas também por ser responsável por parte da minha actual formação humana e científica.

Aos Professores Paul Calamai e Gilles Savard pela sua importante colaboração.

A todos os meus colegas de grupo, uma palavra de amizade pelo companheirismo e boa disposição que bastante contribuíram.

À memória do meu Pai que ainda bem viva sempre me guiou e ajudou.

À minha Mãe e à Inês, incansáveis a dar todo o seu apoio e amor.

Índice Geral

Introdução.	1
1 Definições e casos particulares.	3
2 Propriedades e condições de optimalidade.	9
2.1 Condições de existência de solução.	9
2.2 Formulações alternativas.	11
2.3 Condições de optimalidade.	14
2.4 Propriedades particulares do caso linear.	17
2.5 Relações com outros programas matemáticos.	19
3 Métodos de resolução para programação de dois níveis linear.	24
3.1 Métodos de enumeração de pontos extremos.	24
3.2 Métodos baseados nas condições de optimalidade.	26
3.3 Métodos Branch and Bound.	32
3.4 Métodos de penalidades.	34
3.5 Extensões ao caso linear-quadrático.	38
4 Problemas teste para programação de dois níveis.	41
4.1 Construção de problemas teste separáveis.	42
4.2 Problemas teste não separáveis.	50
4.3 Experiência computacional com o método sequencial <i>LCP</i>	57
5 Algoritmos de descida para programação de dois níveis quadrática.	61
5.1 Definições e propriedades de um programa de dois níveis quadrático.	62
5.2 Algoritmo de descida em pontos extremos da região induzida.	65
5.3 Algoritmo de descida máxima modificado.	67
5.4 Algoritmo híbrido.	75
5.5 Complexidade da verificação da optimalidade local.	76
Conclusões finais.	80
Bibliografia.	82

Índice de Figuras

1.1	Exemplo de decisões hierarquizadas.	3
1.2	Região Induzida: - - -	6
1.3	Região Induzida: - - -	7
2.1	Região Induzida: - - -	22
3.1	Exemplo de uma árvore binária do método enumerativo.	30
3.2	Região Induzida: - - -	40
4.1	Caso 1 - em que $\rho_k = 3$	44
4.2	Caso 2 - em que $\rho_k = 7$	45
4.3	Caso 3 - em que $\rho_k = 9$	45
5.1	Ponto <i>MLERI</i> com duas direcções <i>EIR</i>	66
5.2	Cálculo exacto dos comprimentos de passo σ_{\max} e σ'_k	73

Índice de Tabelas

4.1	Características dos problemas teste resolvidos.	57
4.2	Legenda das tabelas 4.3, 4.4 e 4.5.	57
4.3	Resultados para o caso estritamente linear (sem START).	59
4.4	Resultados para o caso estritamente linear (com START).	59
4.5	Resultados para o caso linear-quadrático.	60

Lista de Notações

argmin (argmax)	- argumento correspondente ao valor mínimo (máximo)
condições KKT	- condições de <i>Karush-Kuhn-Tucker</i>
PDN	- programa de dois níveis
$PDN2$	- programa de dois níveis sem restrições do primeiro nível
$PDNC$	- programa de dois níveis convexo
$PDNC2$	- programa de dois níveis convexo sem restrições do primeiro nível
$PDNL$	- programa de dois níveis linear
$PDNL2$	- programa de dois níveis linear sem restrições do primeiro nível
$PDNLQ$	- programa de dois níveis linear-quadrático
$PDNLQ2$	- programa de dois níveis linear-quadrático sem restrições do primeiro nível
$PDNQ$	- programa de dois níveis quadrático
$P(x)$	- problema do segundo nível para $x \in X$
PR	- problema relaxado de um programa de dois níveis
$C(x)$	- conjunto admissível do problema $P(x)$ para $x \in X$
$M(x)$	- conjunto de reacção do problema $P(x)$ para $x \in X$
$y(x)$	- elemento de $M(x)$ no caso deste ser um conjunto singular
$v(x)$	- valor óptimo do problema $P(x)$ para $x \in X$
CR	- conjunto admissível do problema relaxado PR
RI	- região induzida (conjunto admissível de um programa de dois níveis)
LCP	- problema linear complementar (<i>linear complementarity problem</i>)
$MLCP$	- programa linear com restrições de complementaridade (<i>minimum linear complementarity problem</i>)
pontos ERI	- pontos extremos da região induzida
direcções RI	- direcções que ligam pontos da região induzida
direcções ERI	- direcções que ligam pontos ERI
algoritmo $ADERI$	- algoritmo de descida em pontos extremos da região induzida
ponto $MLERI$	- mínimo local em estrela da região induzida

Introdução

A programação de dois níveis constitui actualmente uma das áreas mais importantes da optimização global. Os programas de dois níveis apresentam propriedades específicas, algumas relacionadas com o seu elevado grau de não convexidade e não diferenciabilidade, que fazem com que a sua resolução seja particularmente difícil mas em simultâneo um desafio consideravelmente interessante. Além disso, são inúmeros os problemas de aplicação prática que pela sua estrutura hierárquica são formuláveis através de programas de dois níveis.

A primeira formulação de um programa de dois níveis relatada na literatura data do princípio dos anos oitenta, o que mostra que o estudo das suas propriedades e da sua resolução é ainda um tópico de investigação extremamente actual. Talvez por este motivo ainda não exista, infelizmente, nenhum livro clássico de programação não linear que inclua a programação de dois níveis como uma das suas áreas de estudo. Deste modo, julgamos ser bastante útil realizar um trabalho de síntese actualizado sobre programação de dois níveis, que permita ao leitor uma rápida familiarização com esta importante área de investigação corrente.

Esta dissertação inclui ainda duas outras contribuições importantes. Assim é introduzida uma técnica para gerar problemas teste de dois níveis que veio preencher uma das principais lacunas desta área. Esta técnica permite controlar de modo simples diversas características dos problemas gerados e coloca à disposição dos utilizadores dos códigos de resolução de programas de dois níveis um vasto conjunto de problemas teste para comparação dos respectivos resultados.

Como última contribuição desta tese salientamos o desenvolvimento de algoritmos para a obtenção de um mínimo local de um programa de dois níveis em que o programa do segundo nível é quadrático estritamente convexo. Acreditamos que esses processos podem desempenhar no futuro um papel muito importante no desenvolvimento de um método eficiente para a resolução de programas de dois níveis não lineares.

O trabalho de síntese é realizado ao longo de toda esta tese, mas fundamentalmente nos capítulos 1, 2 e 3. No capítulo 1 introduzimos o problema, as suas aplicações, as suas

classes mais importantes e outros tópicos relacionados. As propriedades mais significativas de um programa de dois níveis, tais como condições de existência de solução, formulações alternativas e condições necessárias de optimalidade são abordadas no capítulo 2. Nesse capítulo são ainda apresentadas várias relações entre alguns programas matemáticos e programas de dois níveis, que realçam a importância da programação de dois níveis no contexto da optimização em geral.

A chamada programação de dois níveis linear tem sido a mais estudada na literatura e por este motivo na penúltima secção do capítulo 2 e em todo o capítulo 3 são abordados as suas propriedades particulares e os seus métodos de resolução mais importantes. No capítulo 4 descrevemos a técnica de geração de programas de dois níveis referida anteriormente, enquanto que no capítulo 5 discutimos as propriedades e algoritmos desenvolvidos para programação de dois níveis quadrática.

Para a leitura deste trabalho assume-se como adquiridos conhecimentos básicos de álgebra linear e análise, bem como de programação linear e não linear. Como notas finais salientamos o facto de todas as demonstrações apresentadas neste trabalho serem originais e de só referirmos os nomes dos autores correspondentes a uma dada referência quando for pretendido dar o respectivo destaque.

Capítulo 1

Definições e casos particulares.

Um programa de dois níveis está associado a um modelo envolvendo dois agentes de decisão. O primeiro agente, designado por superior, decide recorrendo ao primeiro conjunto de variáveis (x), enquanto que o segundo agente, conhecido por inferior, controla o segundo conjunto de variáveis (y). O superior toma uma decisão segundo o seu objectivo. Dada esta decisão o inferior reage de acordo com o objectivo que lhe está associado. A decisão do superior pode influenciar quer a gama de possibilidades de escolha quer mesmo o critério de escolha do inferior. Por sua vez esta reacção do inferior faz com que o superior repense a sua estratégia, tomando novas decisões. Assim, as decisões são tomadas de cima para baixo ao longo da cadeia hierárquica (com apenas dois níveis neste caso), mas decisões de níveis inferiores afectam as decisões de níveis superiores. A figura 1.1 pretende retratar uma possível interacção económica que pode suceder entre agentes de decisão superior e inferior.

São várias as aplicações reais descritas na literatura que foram modeladas através de programas de dois níveis. Entre as aplicações mais frequentes encontram-se o problema

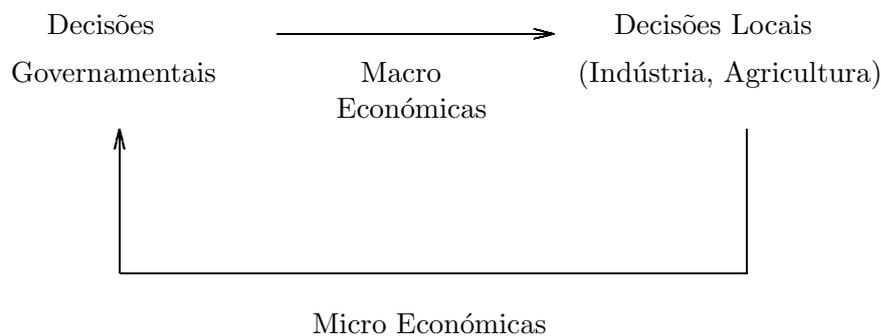


Figura 1.1: Exemplo de decisões hierarquizadas.

de *design* de redes viárias ([BBBL92], [Ma86], [Ma88], [MaMa92] e [SuKi92]), o problema da estimação da procura também em redes viárias [FlCh91] e o problema de localização espacial de facilidades [MiFrTo92]. Alguns problemas de coordenação e controlo de energia eléctrica foram também abordados com o auxílio de programas de dois níveis ([HaLoSa89] e [HoNe92]). Finalmente, é bastante frequente o recurso à programação de dois níveis para modelar problemas de administração e gestão ([Ba83c], [DiJe79], [MeMaTa70] e [On92]).

Um programa de dois níveis (*PDN*), na sua forma mais geral, apresenta a seguinte formulação:

$$\min_{x,y} F(x,y) \quad (1.1)$$

$$\text{sujeito a } x \in X \quad (1.2)$$

$$g(x,y) \leq 0 \quad (1.3)$$

$$y \in \operatorname{argmin}\{f(x,y) : y \in Y, h(x,y) \leq 0\} \quad (1.4)$$

em que X e Y são subconjuntos de \mathbb{R}^n e \mathbb{R}^m respectivamente, F e f são funções reais tais que $F, f : \mathbb{R}^{n+m} \rightarrow \mathbb{R}$, g e h são funções vectoriais reais tais que $g : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^{p_1}$ e $h : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^{p_2}$ com n e $m \in \mathbb{N}$ e p_1 e $p_2 \in \mathbb{N} \cup \{0\}$.

De seguida são apresentadas algumas definições regularmente utilizadas na literatura e que permitem uma mais fácil caracterização das propriedades de um programa de dois níveis. A primeira definição caracteriza o primeiro e o segundo níveis de um programa de dois níveis.

Definição 1.1 *O problema do primeiro nível do problema PDN consiste em minimizar em x e em y a função $F(x,y)$ sujeito às restrições (1.2),(1.3) e (1.4). Este problema coincide com o problema PDN. O problema de minimização em y , parametrizado por x , $P(x)$:*

$$\begin{aligned} & \min_{y \in Y} f(x,y) \\ & \text{sujeito a } h(x,y) \leq 0 \end{aligned}$$

é designado por problema do segundo nível em y .

Deste modo, x é designado por vector das variáveis do primeiro nível e y por vector das variáveis do segundo nível. De igual modo, $g(x,y) \leq 0$ e $h(x,y) \leq 0$ representam respectivamente as restrições do primeiro e do segundo nível. A função $F(x,y)$ é chamada função objectivo do primeiro nível, enquanto que $f(x,y)$ é designada por função objectivo do segundo nível.

A segunda definição pretende classificar conjuntos que desempenham um papel fundamental na teoria da programação de dois níveis. Para o efeito é necessário definir o

problema relaxado associado ao problema PDN . Esta versão relaxada do problema PDN omite a caracterização do objectivo do segundo nível. Assim, o problema relaxado PR reveste a seguinte forma:

$$\begin{aligned} & \min_{x,y \in X \times Y} F(x,y) \\ & \text{sujeito a } g(x,y) \leq 0, h(x,y) \leq 0 \end{aligned}$$

Definição 1.2 *Conjunto admissível do problema relaxado PR ,*

$$CR = \{(x,y) \in X \times Y : g(x,y) \leq 0, h(x,y) \leq 0\}$$

Conjunto admissível do problema do segundo nível $P(x)$ para cada $x \in X$,

$$C(x) = \{y : y \in Y, h(x,y) \leq 0\}$$

Conjunto de reacção do problema do segundo nível para cada $x \in X$,

$$M(x) = \{y : y = \operatorname{argmin}\{f(x,y) : y \in C(x)\}\}$$

Valor óptimo do problema $P(x)$ para cada $x \in X$,

$$v(x) = \min \{f(x,y) : y \in C(x)\}$$

Conjunto admissível do problema PDN , também designado por Região Induzida,

$$RI = \{(x,y) : (x,y) \in CR, y \in M(x)\}$$

Assim, o ponto (x^L, y^L) é uma solução óptima (mínimo) local para o problema PDN se $(x^L, y^L) \in RI$ e se existir uma vizinhança V de (x^L, y^L) tal que $F(x^L, y^L) \leq F(x,y)$, $\forall (x,y) \in V \cap RI$. Do mesmo modo, o ponto (x^G, y^G) é uma solução óptima (mínimo) global do problema PDN se $(x^G, y^G) \in RI$ e $F(x^G, y^G) \leq F(x,y)$, $\forall (x,y) \in RI$. Se apenas mencionarmos o termo solução óptima do problema PDN , referir-nos-emos à solução óptima global.

As diferentes classes do problema PDN diferem entre si consoante as diferentes particularizações das funções F, f, g e h e dos conjuntos X e Y . Em todas as classes é usual considerar programas de dois níveis com ou sem restrições do primeiro nível. O problema PDN descreve a primeira situação. Um programa de dois níveis sem restrições do primeiro nível ($PDN2$) pode ser escrito como:

$$\begin{aligned} & \min_{x,y} F(x,y) \\ & y \in \operatorname{argmin}\{f(x,y) : y \in Y, h(x,y) \leq 0\} \end{aligned}$$

A classe de programação de dois níveis mais frequente da literatura engloba todos os tipos de programas de dois níveis convexos, que são definidos a seguir.

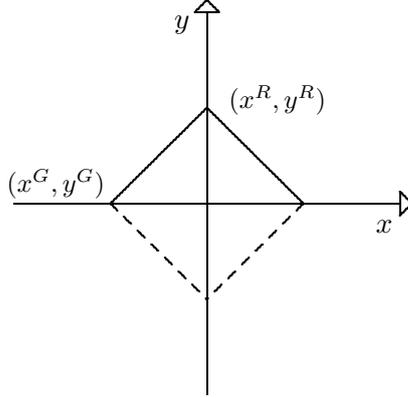


Figura 1.2: Região Induzida: - - -

Definição 1.3 Um programa de dois níveis diz-se convexo (PDNC) quando $X = \mathbb{R}^n$, $Y = \mathbb{R}^m$ e quando são convexas em y as funções $f(x, \cdot)$ e $h(x, \cdot)$ para todo o x em \mathbb{R}^n .

A importância desta classe advém da possibilidade de substituir o problema do segundo nível $P(x)$, desde que verificada uma restrição de qualificação adequada [BaSh79] e que todas as funções envolvidas sejam diferenciáveis, pelas suas condições necessárias e suficientes de Karush-Kuhn-Tucker (KKT). A importância dessa redução será discutida mais adiante.

Os exemplos que são apresentados de seguida ilustram de forma clara os conjuntos e problemas anteriormente definidos. Como primeiro exemplo considere-se o seguinte programa de dois níveis sem restrições do primeiro nível:

$$\begin{aligned} \min_{x,y} \quad & x - 2y \\ \text{sujeito a} \quad & y \in \operatorname{argmin}\{y : -1 \leq x + y \leq 1, -1 \leq x - y \leq 1\} \end{aligned} \quad (1.5)$$

Na figura 1.2 são representados graficamente o conjunto admissível do problema relaxado correspondente $CR = \{(x, y) \in \mathbb{R}^2 : -1 \leq x + y \leq 1, -1 \leq x - y \leq 1\}$ e a tracejado a região induzida $RI = \{(x, y) \in \mathbb{R}^2 : x + y = -1, -1 \leq x \leq 0\} \cup \{(x, y) \in \mathbb{R}^2 : x - y = 1, 0 \leq x \leq 1\}$ do problema (1.5). O ponto $(x^R, y^R) = (0, 1)$ é a solução ótima do problema relaxado ($\min_{(x,y) \in CR} x - 2y$) enquanto que o ponto $(x^G, y^G) = (-1, 0)$ é a solução ótima do programa de dois níveis (1.5) ($\min_{(x,y) \in RI} x - 2y$).

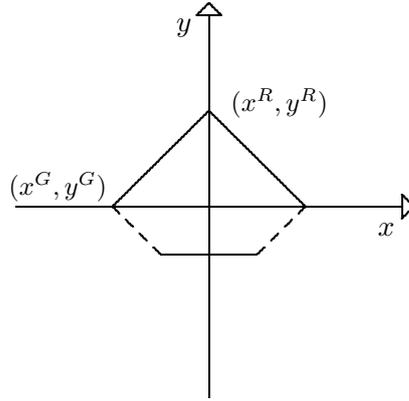


Figura 1.3: Região Induzida: - - -

Se a restrição $y \geq -\frac{1}{2}$ for colocada no primeiro nível do problema anterior obtém-se o seguinte programa de dois níveis:

$$\begin{aligned}
 \min_{x,y} \quad & x - 2y \\
 \text{sujeito a} \quad & y \geq -\frac{1}{2} \\
 & y \in \operatorname{argmin}\{y : -1 \leq x + y \leq 1, -1 \leq x - y \leq 1\}
 \end{aligned} \tag{1.6}$$

Para este exemplo o conjunto CR é $\{(x, y) \in \mathbb{R}^2 : y \geq -\frac{1}{2}, -1 \leq x + y \leq 1, -1 \leq x - y \leq 1\}$ e a região induzida RI é $\{(x, y) \in \mathbb{R}^2 : x + y = -1, -1 \leq x \leq -\frac{1}{2}\} \cup \{(x, y) \in \mathbb{R}^2 : x - y = 1, \frac{1}{2} \leq x \leq 1\}$. Estes dois conjuntos são representados na figura 1.3 onde tal como anteriormente RI é indicada a tracejado. A restrição $y \geq -\frac{1}{2}$, pelo facto de se encontrar no primeiro nível do problema (1.6), separa a região induzida em dois segmentos de recta disjuntos, transformando-a num conjunto desconexo. As soluções óptimas do problema relaxado, (x^R, y^R) , e do próprio problema de dois níveis (1.6), (x^G, y^G) , coincidem com as respectivas soluções óptimas do exemplo anterior.

Os dois exemplos anteriormente apresentados mostram duas características que, regra geral, estão associadas a programas de dois níveis: a região induzida RI é um conjunto não convexo que, na presença de restrições de primeiro nível, pode ser desconexo.

Os casos particulares do problema $PDNC$ mais comuns na literatura são:

- $PDNL$ - programa de dois níveis linear em que todas as funções envolvidas são lineares (caso dos dois exemplos anteriormente discutidos),
- $PDNLQ$ - programa de dois níveis linear-quadrático em que as funções F, g e h são lineares mas em que a função objectivo do segundo nível f é quadrática e estritamente convexa em y ,

- $PDNQ$ - programa de dois níveis quadrático em que a função objectivo do primeiro nível F é também quadrática (convexa ou não).

O grau de dificuldade da resolução de um programa de dois níveis é facilmente aferido pela propriedade NP-Difícil [GaJo79] do caso linear ([BeB190], [HaJaSa92] e [Je85]) e também do caso linear-quadrático [Ba91]. No capítulo 5 provaremos que a simples verificação da optimalidade local de um programa de dois níveis linear é NP-Difícil.

Neste trabalho iremos essencialmente concentrar a nossa atenção nestes três últimos tipos de programas. Contudo outras classes de programas de dois níveis têm vindo a ser discutidos na literatura. Assim, os conjuntos X e Y de um programa de dois níveis PDN podem ser definidos como conjuntos inteiros ou inteiros mistos (isto é, com uma componente inteira e outra contínua), classificando o problema PDN como um programa de dois níveis inteiro ou inteiro misto respectivamente. Existem processos enumerativos para os casos linear binário [BaMo92], inteiro misto linear [MoBa90] e também para o caso inteiro misto não linear [EdBa92].

Outros casos particulares do problema PDN foram já abordados por alguns autores, como o caso da programação de dois níveis geométrica [Se89] ou o caso em que o problema do segundo nível é formulado de um modo diferente, como um problema de desigualdades variacionais [FTCM90]. Também o problema da análise discriminante foi estudado à luz da programação de dois níveis [MaSa91b].

A programação de dois níveis pode ser encarada como caso particular da programação de níveis múltiplos [Sa89]. A complexidade destes programas quando o número de níveis é superior a dois aumenta significativamente [B192]. O caso de três níveis já mereceu alguma atenção na literatura ([Ba84b] e [WeBi86]) enquanto que para um número de níveis genérico é ainda muito reduzido o leque de investigação já realizado ([BaFa82] e [Be89]).

Capítulo 2

Propriedades e condições de optimalidade.

Este capítulo visa resumir e referir importantes propriedades da programação de dois níveis. Em qualquer problema de otimização existe a necessidade de definir condições de existência de solução e a programação de dois níveis não é exceção. Deste modo, na secção 3.1 são abordadas tais condições e é referido o problema de *Stackelberg* cuja relação com um programa de dois níveis está dependente da unicidade de soluções do problema do segundo nível. Em seguida são apresentadas na secção 2.2 diferentes formulações para um programa de dois níveis. Estas reformulações são utilizadas para desenvolver, de modos diferentes, métodos de resolução e condições necessárias de optimalidade e revestem uma papel determinante em programação de dois níveis. Várias condições de optimalidade são abordadas na secção 2.3.

A programação de dois níveis linear apresenta propriedades próprias e importantes que são resumidas na secção 2.4. Finalmente, as relações entre vários programas matemáticos e programas de dois níveis são analisadas na secção 2.5, comprovando a importância deste último problema.

2.1 Condições de existência de solução.

As condições de existência de solução de um programa de dois níveis estão associadas à caracterização da aplicação ponto-conjunto $M(\cdot) : X \subset \mathbb{R}^n \longrightarrow \Omega(\mathbb{R}^m)$ em que $\Omega(\mathbb{R}^m)$ representa o conjunto de todos os subconjuntos de \mathbb{R}^m .

O teorema seguinte indica uma condição suficiente para que a aplicação $M(\cdot)$ seja unívoca (isto é para que $M(\cdot)$ seja uma aplicação ponto-ponto de \mathbb{R}^n em \mathbb{R}^m), contínua e fechada. No caso em que a aplicação $M(\cdot)$ é unívoca, $M(x)$ é um conjunto singular para todo o x e o elemento singular deste conjunto é designado por $y(x)$. Para todo o $x \in X$ é assumido que $M(x) \neq \emptyset$.

Teorema 2.1 *Se para cada $x \in X$, f e h são funções duas vezes continuamente diferenciáveis para todo o $y \in C(x)$, f é estritamente convexa para todo o y em $C(x)$ e o conjunto $C(x)$ é compacto e convexo, então $M(\cdot)$ é uma aplicação unívoca, contínua e fechada.*

A demonstração deste teorema encontra-se em [DaFoSh67] ($M(\cdot)$ unívoca e contínua) e em [Ho73] ($M(\cdot)$ fechada).

Além disso, é possível estabelecer condições suficientes para a existência de solução de um programa de dois níveis:

Teorema 2.2 [EdBa91] *Se além de serem válidas as hipóteses do teorema anterior, F é contínua em x e em y e X é um conjunto compacto, então existe sempre uma solução para o problema PDN2.*

A demonstração deste teorema baseia-se no facto de $F(x, y(x))$ ser uma aplicação contínua em x , uma vez que F e M também o são. Além disso, o facto de $M(\cdot)$ ser fechada e X ser compacto implica que a região induzida RI também seja compacta [EdBa91]. Ao minimizarmos uma função contínua sobre um conjunto compacto estamos a garantir a existência de solução.

Este resultado é extensível ao problema PDN se considerarmos compacto o conjunto definido pelas restrições do primeiro nível $\{(x, y) : g(x, y) \leq 0\}$ e se existir pelo menos um $x \in X$ tal que $g(x, y(x)) \leq 0$. Outras condições alternativas de existência de solução foram abordadas em [HaPa88].

Existe outra classe de programas com dois níveis de resolução, a dos problemas de *Stackelberg* [St52], que é muitas vezes confundida com a programação de dois níveis. No problema de *Stackelberg*, o primeiro agente de decisão também influencia as escolhas do segundo agente de decisão. A diferença está em que as decisões do agente inferior não afectam o critério de escolha do superior e portanto a função objectivo do primeiro nível F é apenas minimizada nas variáveis x . O problema de *Stackelberg* pode então ser formulado na seguinte forma:

$$\begin{array}{ll} \min_x & F(x, y) \\ \text{sujeito a} & x \in X \\ & g(x, y) \leq 0 \\ & y \in \operatorname{argmin}\{f(x, y) : y \in Y, h(x, y) \leq 0\} \end{array}$$

em que $x \in \mathbb{R}^n$ e $y \in \mathbb{R}^m$.

Desta forma, os problemas de *Stackelberg* e os problemas de dois níveis não podem ser considerados como equivalentes e o facto de a aplicação $M(\cdot)$ não ser unívoca ainda acentua mais a diferença entre os dois problemas. Este trabalho versa somente a classe de programas de dois níveis. Um resumo dos principais resultados em problemas de *Stackelberg* pode ser encontrado em [Sa89].

2.2 Formulações alternativas.

As diferentes formulações do problema *PDN* desempenham um papel fundamental no desenvolvimento de algoritmos de resolução para programação de dois níveis. O simples facto de se utilizar formulações distintas pode conduzir a resultados aparentemente não semelhantes e a métodos de resolução totalmente diferentes.

Assim por exemplo, se a aplicação $M(\cdot)$ é unívoca é possível reformular o problema *PDN* como:

$$\begin{aligned} \min_x \quad & F(x, y(x)) \\ \text{sujeito a} \quad & g(x, y(x)) \leq 0 \\ & x \in C_1 \end{aligned} \tag{2.1}$$

em que:

$$C_1 = \{x \in X : \exists y \in Y \text{ tal que } g(x, y) \leq 0\}$$

Por outro lado, se $M(\cdot)$ não é unívoca então podemos escrever:

$$\begin{aligned} \min_{x,y} \quad & F(x, y) \\ \text{sujeito a} \quad & (x, y) \in CR \\ & y \in M(x) \end{aligned}$$

onde CR é o conjunto admissível do problema relaxado.

Estas duas formulações abordam o problema *PDN* como um programa de um só nível mas em que algumas das funções e conjuntos se encontram definidos de modo implícito. A formulação (2.1) foi utilizada por alguns autores com o intuito de desenvolver métodos descendentes para a resolução de programas de dois níveis ([AiSh81] e [KoLa90]).

Outras formulações podem ser consideradas mas para as quais o factor implícito $M(x)$ é substituído pelos conjuntos $C(x)$ ou por $v(x)$ apresentados na definição 2.1. A fim de desenvolver condições necessárias de optimalidade para um programa de dois níveis, Bard [Ba84a] considerou a seguinte formulação equivalente:

$$\begin{aligned}
& \min_{(x,y) \in X \times Y} && F(x, y) \\
& \text{sujeito a} && g(x, y) \leq 0 \\
& && f(x, y) - f(x, z) \leq 0, \text{ para todo } z \in C(x) \\
& && h(x, y) \leq 0
\end{aligned} \tag{2.2}$$

em que $C(x)$ é o conjunto admissível de $P(x)$.

Com o mesmo propósito, Chen e Florian [ChF191] abordaram o problema de dois níveis PDN da seguinte maneira:

$$\begin{aligned}
& \min_{(x,y) \in X \times Y} && F(x, y) \\
& \text{sujeito a} && g(x, y) \leq 0 \\
& && f(x, y) - v(x) = 0 \\
& && h(x, y) \leq 0
\end{aligned} \tag{2.3}$$

onde $v(x)$ é o valor óptimo do problema do segundo nível $P(x)$.

O caso convexo apresenta, por natureza própria, uma formulação alternativa mais simples. Assim, se considerarmos válida uma restrição de qualificação [BaSh79] para o problema $P(x)$ para cada $x \in \mathbb{R}^n$ e se todas as funções envolvidas forem diferenciáveis, é possível rescrever o problema $PDNC$ do seguinte modo:

$$\begin{aligned}
& \min_{x,y,\alpha} && F(x, y) \\
& \text{sujeito a} && g(x, y) \leq 0 \\
& && \nabla_y f(x, y) + \nabla_y h(x, y)^T \alpha = 0 \\
& && \alpha^T h(x, y) = 0 \\
& && h(x, y) \leq 0, \quad \alpha \geq 0
\end{aligned} \tag{2.4}$$

A equivalência entre o problema $PDNC$ e o problema (2.4) resulta da possibilidade de se poder substituir o problema do segundo nível $P(x)$ pelas suas condições necessárias e suficientes de optimalidade. Bi, Calamai e Conn ([BiCaCo89] e [BiCaCo91]) exploraram esta formulação para o caso particular do problema $PDN2$. Com a finalidade de obter várias propriedades para o programa em causa, estes autores escreveram o problema $PDNC2$ na forma não diferenciável:

$$\begin{aligned}
& \min_{x,y,\alpha} && F(x, y) \\
& \text{sujeito a} && \nabla_y f(x, y) + \nabla_y h(x, y)^T \alpha = 0 \\
& && \min (\alpha_i, -h_i(x, y)) = 0, \quad i = 1, \dots, p_2
\end{aligned} \tag{2.5}$$

utilizando para o efeito o operador não diferenciável \min .

Para uma dada formulação de um programa de dois níveis, não é indiferente que um dado conjunto de restrições $a(x, y) \leq 0$ se encontre no primeiro ou no segundo nível. O teorema seguinte caracteriza a relação entre o conjunto admissível do problema $PDNC$ no caso de tais restrições serem introduzidas no primeiro nível ($g(x, y) \leq 0, a(x, y) \leq 0$) e o conjunto admissível do mesmo problema $PDNC$ para o caso de tais restrições serem colocadas no segundo nível ($h(x, y) \leq 0, a(x, y) \leq 0$).

Teorema 2.3 *Sejam C' e C'' dois conjuntos definidos do seguinte modo:*

$$C' = \{(x, y) : g(x, y) \leq 0, a(x, y) \leq 0, y \in \operatorname{argmin}\{f(x, y) : h(x, y) \leq 0\}\}$$

$$C'' = \{(x, y) : g(x, y) \leq 0, y \in \operatorname{argmin}\{f(x, y) : h(x, y) \leq 0, a(x, y) \leq 0\}\}$$

Se $a(x, \cdot)$, $f(x, \cdot)$ e $h(x, \cdot)$ são funções convexas e diferenciáveis em y para todo o x e se verificar uma restrição de qualificação adequada [BaSh79] para o problema do segundo nível, então $C' \subset C''$.

Demonstração: Seja $(\bar{x}, \bar{y}) \in C'$. Então:

$$g(\bar{x}, \bar{y}) \leq 0, a(\bar{x}, \bar{y}) \leq 0 \tag{2.6}$$

$$\bar{y} \in \operatorname{argmin}\{f(\bar{x}, y) : h(\bar{x}, y) \leq 0\} \tag{2.7}$$

Atendendo às hipóteses do teorema a condição (2.7) é equivalente a:

$$\nabla f(\bar{x}, \bar{y}) + \nabla h(\bar{x}, \bar{y})^T \alpha = 0 \tag{2.8}$$

$$h(\bar{x}, \bar{y}) \leq 0, \alpha \geq 0 \tag{2.9}$$

$$\alpha^T h(\bar{x}, \bar{y}) = 0 \tag{2.10}$$

Mas, se (2.8), (2.9) e (2.10) se verificarem também são válidas as seguintes condições:

$$\nabla f(\bar{x}, \bar{y}) + \nabla h(\bar{x}, \bar{y})^T \alpha + \nabla a(\bar{x}, \bar{y})^T \lambda = 0$$

$$h(\bar{x}, \bar{y}) \leq 0, \alpha \geq 0, \lambda \geq 0$$

$$\alpha^T h(\bar{x}, \bar{y}) = \lambda^T a(\bar{x}, \bar{y}) = 0$$

com $\lambda = 0$. Estas últimas condições em conjunto com (2.6) permitem concluir que $(\bar{x}, \bar{y}) \in C''$. Deste modo, C' é subconjunto de C'' . \square

2.3 Condições de optimalidade.

A primeira tentativa de estabelecer condições necessárias de optimalidade para programação de dois níveis foi realizada por Bard [Ba84a]. Bard utilizou a formulação (2.2) que consiste num programa matemático com um número infinito e parametrizado de restrições. Assim mostrou que se $X = \mathbb{R}^n$, $Y = \mathbb{R}^m$, CR é não vazio e compacto, $M(\cdot)$ é uma aplicação unívoca e F, f e h são continuamente diferenciáveis, então (x_0, y_0) é um mínimo local para o problema $PDN2$ se existirem multiplicadores $u \in \mathbb{R}^{p_2}$ e $v \in \mathbb{R}$ tais que:

$$\nabla_x F(x_0, y_0) + \nabla_x h(x_0, y_0)^T u = 0 \quad (2.11)$$

$$\nabla_y F(x_0, y_0) + \nabla_y h(x_0, y_0)^T u + v \nabla_y f(x_0, y_0) = 0 \quad (2.12)$$

$$f(x_0, y_0) - f(x_0, z) \leq 0, \text{ para todo } z \in C(x_0)$$

$$u^T h(x_0, y_0) = 0$$

$$h(x_0, y_0) \leq 0, u \geq 0, v \geq 0$$

No entanto, Clarke e Westerberg [ClWe88] apresentam um contra exemplo para estas condições para o qual o gradiente de F não se encontra no cone gerado pelos gradientes das restrições activas de h e pelo vector $(0, \nabla_y f)$, como seria de esperar de acordo com as condições (2.11) e (2.12). A origem do erro ao estabelecer as falsas condições necessárias residu na forma como o conjunto de restrições:

$$f(x, y) - f(x, z) \leq 0 \text{ para todo } z \in C(x)$$

foi considerado para efeito de aplicação directa das condições KKT. Bard apenas ponderou o facto de o número de restrições ser infinito e ignorou a característica paramétrica do conjunto.

Do decorrido, todos os resultados posteriores de Bard [Ba84a] baseiam-se em pressupostos errados. Também os dois algoritmos descritos em [Ba83a] e [Ba83b] e desenvolvidos para o caso linear e para o caso não linear respectivamente, estão incorrectos.

Chen e Florian [ChFl91] desenvolveram condições necessárias de optimalidade para programação de dois níveis a partir da formulação (2.3), mediante as seguintes hipóteses:

- X e Y conjuntos compactos,
- F e g continuamente diferenciáveis,
- f e h duas vezes continuamente diferenciáveis,
- f localmente *Lipschitz*.

As condições de optimalidade em causa são, sem perda de generalidade, descritas para a versão sem restrições do primeiro nível $PDN2$. É de notar que a formulação (2.3) não utiliza as condições de KKT do problema do segundo nível e portanto nem é necessário assumir qualquer convexidade nem aumentar o número de variáveis do problema. Usando essa formulação, pode-se estabelecer o seguinte resultado:

Teorema 2.4 [ChFl91] *Se (x_0, y_0) é um mínimo local do problema $PDN2$, existe uma restrição de qualificação [BaSh79], $\Pi_0^2(x_0, y_0) = \{0\}$ e o problema do segundo nível $P(x_0)$ tem uma solução óptima única e um único multiplicador óptimo π_0 , então existem $\alpha \in \mathbb{R}$ e $\theta \in \mathbb{R}^{p_2}$ tais que:*

$$\begin{aligned}\nabla_x F(x_0, y_0) + \nabla_x h(x_0, y_0)^T (\theta - \alpha \pi_0) &= 0 \\ \nabla_y F(x_0, y_0) + \alpha \nabla_y f(x_0, y_0) + \nabla_y h(x_0, y_0)^T \theta &= 0 \\ f(x_0, y_0) - v(x_0) &= 0, \\ \theta^T h(x_0, y_0) &= 0 \\ h(x_0, y_0) &\leq 0, \theta \geq 0\end{aligned}$$

O conjunto de multiplicadores de segunda ordem $\Pi_0^2(x_0, y_0)$ está relacionado com o problema do segundo nível $P(x_0)$ e a sua definição encontra-se detalhadamente exposta em [Ro84].

Uma vez estabelecidas as condições necessárias de optimalidade, é possível descrever o cone de direcções de descida admissíveis para um programa de dois níveis. De facto, ao aplicarem directamente o Lema de Farkas às condições do Teorema 2.4 os autores concluíram a seguinte propriedade:

Teorema 2.5 *Sejam válidas as hipóteses do teorema anterior. Se (x_0, y_0) é um mínimo local do problema $PDN2$, então não existe solução para o seguinte sistema de desigualdades em z e w :*

$$\begin{aligned}(z, w)^T \nabla F(x_0, y_0) &< 0 \\ w^T \nabla_y f(x_0, y_0) &\leq 0 \\ z^T \nabla_x h_i(x_0, y_0) &= 0, i \in I(x_0, y_0) \\ w^T \nabla_y h_i(x_0, y_0) &\leq 0, i \in I(x_0, y_0)\end{aligned}$$

com $z \in \mathbb{R}^n$, $w \in \mathbb{R}^m$ e $I(x_0, y_0)$ o conjunto de índices das restrições activas do segundo nível em (x_0, y_0) .

Um programa de dois níveis convexo sem restrições do primeiro nível $PDNC2$ apresenta, no caso de ser sempre válida uma restrição de qualificação para o problema do segundo nível, condições de optimalidade consideravelmente mais simples. Chen e

Florian concluíram, sob as hipóteses do teorema 2.4, que se (x_0, y_0) é um mínimo local do problema $PDNC2$ então existem $\alpha \in \mathbb{R}$, θ e $\pi \in \mathbb{R}^m$ tais que:

$$\begin{aligned}\nabla F(x_0, y_0) + \nabla h(x_0, y_0)^T(\theta - \alpha\pi) &= 0 \\ \theta^T h(x_0, y_0) = \pi^T h(x_0, y_0) &= 0 \\ h(x_0, y_0) &\leq 0, \quad \theta, \pi \geq 0\end{aligned}$$

De igual modo, se (x_0, y_0) é um mínimo local do problema $PDN2$, então o seguinte sistema de desigualdades:

$$\begin{aligned}u^T \nabla F(x_0, y_0) &< 0 \\ u^T \nabla h_i(x_0, y_0) &= 0, \quad i \in I(x_0, y_0)\end{aligned}$$

não tem solução em $u \in \mathbb{R}^{n+m}$.

Dempe [De92] desenvolveu também, através de conceitos de otimização não diferenciável, condições necessárias de optimalidade para programação de dois níveis. Ambas as abordagens falham na tentativa de estabelecer condições necessárias de optimalidade fáceis de aplicar do ponto de vista prático. De facto, as restrições de qualificação são dificilmente verificáveis quando particularizadas às classes mais conhecidas (linear e quadrática) da programação de dois níveis.

Gauvin e Savard [GaSa91] desenvolveram condições necessárias de optimalidade de utilização mais imediata e dirigidas à seguinte versão de um programa de dois níveis:

$$\begin{aligned}\min_{x,y} \quad & F(x, y) \\ \text{sujeito a} \quad & y \in \operatorname{argmin}\{f(x, y) : h(x, y) \leq 0\}\end{aligned}\tag{2.13}$$

Para isso consideraram as seguintes hipóteses:

- (i) $M(\cdot)$ é uma aplicação unívoca,
- (ii) Para cada ponto $(x, y(x)) \in RI$:
 - (a) os gradientes das restrições activas em $(x, y(x))$ são linearmente independentes,
 - (b) é válida uma condição suficiente de optimalidade para o problema $P(x)$ [GaSa91].

O seguinte teorema resume as condições de optimalidade referidas.

Teorema 2.6 [GaSa91] *Sejam válidas as hipóteses (i) e (ii). Se $(x, y(x)) \in RI$ é um ótimo local do problema (2.13) então é não negativo o valor ótimo do programa de dois*

níveis linear-quadrático em z e w , $PDNLQ(x)$:

$$\begin{aligned} \min_{z,w} \quad & \nabla_x F(x, y(x))^T z + \nabla_y F(x, y(x))^T w \\ \text{sujeito a} \quad & w \in \operatorname{argmin}\{(z, w)^T \nabla_{(x,y)}^2 L(x, y(x); \lambda(x))(z, w) : \\ & \nabla_x h_i(x, y(x))^T z + \nabla_y h_i(x, y(x))^T w \leq 0, \quad i \in I(x) \\ & [-\nabla_x L(x, y(x), \lambda(x)) + \nabla_x f(x, y(x))]^T z + \nabla_y f(x, y(x))^T w = 0\} \end{aligned}$$

em que $z \in \mathbb{R}^n$, $w \in \mathbb{R}^m$, $I(x) = \{i \in \{1, \dots, p_2\} : h_i(x, y(x)) = 0\}$ e $L(x, y(x); \lambda(x)) = f(x, y) + \sum_{i \in I(x)} \lambda_i h_i(x, y)$, com λ_i , $i \in I(x)$ os correspondentes multiplicadores óptimos de $P(x)$.

Além disso, se o valor óptimo do programa $PDNLQ(x)$ é negativo, então a correspondente solução óptima (z^*, w^*) representa a direcção de descida máxima em $(x, y(x))$. Assim, da resolução do programa $PDNLQ(x)$ obtém-se ou um ponto estacionário (na maior parte das vezes mínimo local) ou uma direcção de descida. Essa propriedade será explorada devidamente no desenvolvimento de um algoritmo descendente para programação de dois níveis quadrática. Esse assunto só será discutido no capítulo 5.

2.4 Propriedades particulares do caso linear.

Um programa de dois níveis linear $PDNL$ é usualmente formulado do seguinte modo:

$$\begin{aligned} \min_{x,y} \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \end{aligned} \tag{2.14}$$

$$x \geq 0 \tag{2.15}$$

$$y \in \operatorname{argmin}\{d_2^T y : A_2 x + B_2 y \leq b_2, y \geq 0\}$$

em que $c_1 \in \mathbb{R}^n$, $d_1, d_2 \in \mathbb{R}^m$, $A_1 \in \mathbb{R}^{l_1 \times n}$, $A_2 \in \mathbb{R}^{l_2 \times n}$, $B_1 \in \mathbb{R}^{l_1 \times m}$, $B_2 \in \mathbb{R}^{l_2 \times m}$, $b_1 \in \mathbb{R}^{l_1}$ e $b_2 \in \mathbb{R}^{l_2}$, com $l_1, l_2 \geq 0$.

Tal como anteriormente, notaremos por $PDNL2$ o programa de dois níveis linear sem as restrições do primeiro nível (2.14). As restrições de não negatividade (2.15) são consideradas incluídas no problema $PDNL2$ caso nada seja dito em contrário. A presença do termo linear em x na função objectivo do segundo nível não afecta a caracterização da região induzida. Assim, esta formulação padrão inclui somente termos lineares em y na referida função objectivo.

É possível para este caso particular da programação de dois níveis caracterizar a solução óptima sob condições relativamente fracas. O resultado que se apresenta de seguida foi estabelecido em primeiro lugar (para o problema $PDNL2$) por Candler e

Townsley [CaTo82] sob o pressuposto de $M(\cdot)$ ser uma aplicação unívoca e depois por Bialas e Karwan [BiKa84] para o caso de o conjunto CR ser limitado.

Teorema 2.7 [Sa89] *Se existir uma solução óptima finita para o problema PDNL então pelo menos uma solução óptima é atingida num ponto extremo do conjunto admissível do problema relaxado PR.*

O interesse deste resultado reside no facto de ser garantida a obtenção de um mínimo global a todos os algoritmos que apenas percorrerem os pontos extremos do conjunto poliédrico CR .

As demonstrações desta propriedade descritas em [CaTo82] e em [BiKa84] para a versão sem restrições do primeiro nível do Teorema 2.7 incluem algumas propriedades interessantes do problema PDNL2. A primeira baseia-se na definição de uma versão particular do problema relaxado PR. Dada uma base óptima H para o problema do segundo nível $P(x)$, esta versão ($PR(H)$) reveste a seguinte forma:

$$\begin{aligned} \min_{x, y_H} \quad & c_1^T x + d_1^T y_H \\ \text{sujeito a} \quad & A_2 x + H y_H \leq b_2 \end{aligned} \tag{2.16}$$

$$x, y_H \geq 0 \tag{2.17}$$

em que o vector y_H engloba somente as variáveis do vector y referentes à base H . Como $(x, y(x))$ verifica as restrições (2.16) e (2.17) este problema é sempre admissível. Seja (x^*, y_H^*) a solução óptima do problema $PR(H)$. É importante realçar que a base H continua a ser óptima para o problema do segundo nível $P(x^*)$ pois, por um lado ao verificar (2.16) e (2.17) a sua admissibilidade mantém-se e por outro os seus custos reduzidos não sofrem qualquer tipo de alteração.

O teorema seguinte caracteriza a relação entre o problema $PR(\cdot)$ e o problema PDNL2 e permite concluir, sob determinadas condições, que pelo menos uma solução óptima do problema PDNL2 é ponto extremo do conjunto CR .

Teorema 2.8 [CaTo82] *Se o problema PDNL2 tem solução óptima finita e a aplicação $M(\cdot)$ é unívoca, então existe uma base óptima H para o problema do segundo nível tal que a solução óptima do problema $PR(H)$ é solução óptima do problema PDNL2.*

A demonstração de Bialas e Karwan tem por base o seguinte resultado:

Teorema 2.9 [BiKa84] *Seja CR um conjunto limitado e z_1, \dots, z_r quaisquer pontos de CR . Se $z = \sum_{i=1}^r \lambda_i z_i \in RI$ em que $\sum_{i=1}^r \lambda_i = 1$, $\lambda_i > 0$, $i = 1, \dots, r$ então $z_i \in RI$, $i = 1, \dots, r$.*

Este resultado atribui de certo modo um relativo grau de convexidade à região induzida RI . Com efeito, pontos do conjunto CR de admissibilidade do problema relaxado cuja combinação convexa pertença à região induzida, também pertencem à região induzida. Esta propriedade pode ser utilizada para garantir que, de um ponto não extremo do conjunto CR pertencente à região induzida é sempre possível realizar-se um deslocamento directo (unidireccional) para um ponto extremo de CR ainda pertencente à região induzida. Tal assunto será discutido no capítulo 5.

Bard [Ba84a] provou de forma diferente de qualquer dos autores já referidos o mesmo resultado 2.7 para programas de dois níveis sem restrições do primeiro nível. Assim, sob a hipótese de CR ser um conjunto limitado, o autor provou que a região induzida do problema $PDNL2$ é um conjunto construído à custa de uma função linear em módulos (*piecewise linear*), isto é que:

$$RI = \{(x, y) \in CR : v(x) - (c_1^T x + d_1^T y) = 0\}$$

em que $v(x) = \max_{j \in \{1, \dots, p\}} u_j^T (b_2 - A_2 x)$, com u_j , $j = 1, \dots, p$ vértices do programa dual do problema do segundo nível, é uma função linear em módulos.

2.5 Relações com outros programas matemáticos.

Nesta secção mostraremos que a programação de dois níveis tem importantes relações com alguns problemas de optimização conhecidos, nomeadamente a programação quadrática, a programação linear inteira, os problemas min-max e a optimização multi-critério.

Programação quadrática e programação linear inteira.

Qualquer programa bilinear [Ko76a]:

$$\begin{array}{ll} \min_{x,y} & r^T x + x^T Q y + s^T y \\ \text{sujeito a} & Ax \leq b, x \geq 0 \\ & Cy \leq d, y \geq 0 \end{array}$$

é redutível a um caso particular de um programa de dois nível linear [GaU177].

Konno [Ko76b] provou que um programa quadrático côncavo é equivalente a um programa bilinear e Raghavachari [Ra69] estabeleceu que a programação linear inteira é um caso particular da programação côncava. Assim, é possível reduzir programas quadráticos côncavos e programas lineares inteiros a problemas de programação de dois níveis linear.

Este resultado anteriormente mencionado é extensível a qualquer programa quadrático, bilinear ou não bilinear. O teorema seguinte formaliza este resultado.

Teorema 2.10 *Qualquer programa quadrático é formulável como um programa de dois níveis bilinear-bilinear, ou seja um programa de dois níveis em que todas as funções envolvidas são lineares à exceção das funções objectivo que são bilineares.*

Demonstração: Considere-se o seguinte programa quadrático:

$$\begin{aligned} \min_z \quad & z^T Q z \\ \text{sujeito a} \quad & A z \leq b \\ & z \geq 0 \end{aligned}$$

Ao introduzir a mudança de variável $Q^T z = t$, reformula-se o problema do seguinte modo:

$$\begin{aligned} \min_{z,t} \quad & t^T z \\ \text{sujeito a} \quad & A z \leq b \\ & Q^T z = t \\ & z \geq 0 \end{aligned}$$

Usando a teoria da dualidade linear, o programa anterior é equivalente a:

$$\begin{aligned} \min_{t,u,v} \quad & \max_{u,v} \quad b^T u + t^T v \\ & A^T u + Q v \geq t, \quad u \geq 0 \end{aligned}$$

Rescrevendo a anterior formulação obtemos o seguinte programa de dois níveis bilinear-bilinear:

$$\begin{aligned} \min_{t,u,v} \quad & b^T u + t^T v \\ \text{sujeito a} \quad & u, v \in \operatorname{argmax} \{b^T u + t^T v : A^T u + Q v \geq t, u \geq 0\} \quad \square \end{aligned}$$

Note-se que apesar da função objectivo do segundo nível ser não linear, ela é linear nas variáveis do segundo nível, condição suficiente para que neste caso o problema do segundo nível $P(t)$ seja convexo.

Problema min-max.

Uma classe de programas matemáticos redutível à programação de dois níveis é o problema min-max (ou max-min) [Fa72], que normalmente é definido como:

$$\begin{aligned} \min_x \max_y \quad & c^T x + d^T y \\ \text{sujeito a} \quad & A x + B y \leq b \\ & x, y \geq 0 \end{aligned}$$

Como já foi referido por vários autores ([ChF191] e [HaJaSa92]) este programa é equivalente ao seguinte programa de dois níveis linear:

$$\begin{aligned} \min_{x,y} \quad & c^T x + d^T y \\ \text{sujeito a} \quad & y \in \operatorname{argmin}\{-c^T x - d^T y : Ax + By \leq b, x, y \geq 0\} \end{aligned}$$

Deste modo, um programa min-max é um caso particular da programação de dois níveis linear.

Optimização multi-objectivo.

A existência de duas funções objectivo num programa de dois níveis levou vários investigadores a estudar a relação entre um programa de dois níveis e um programa de dois objectivos (ou critérios). O programa de dois critérios *PDC*, associado a um programa de dois níveis linear sem restrições no primeiro nível, é formulável como:

$$\begin{aligned} \min \quad & (c_1^T x + d_1^T y, d_2^T y) \\ \text{sujeito a} \quad & A_2 x + B_2 y \leq b_2 \\ & x, y \geq 0 \end{aligned}$$

O objectivo associado a um programa deste tipo é calcular as soluções ditas *eficientes*. Um ponto (\bar{x}, \bar{y}) diz-se *eficiente* quando é admissível, isto é satisfaz:

$$A_2 x + B_2 y \leq b_2, \quad x, y \geq 0$$

e não existe nenhum outro ponto admissível (x, y) para o qual:

$$c_1^T x + d_1^T y \leq c_1^T \bar{x} + d_1^T \bar{y} \tag{2.18}$$

$$d_2^T y \leq d_2^T \bar{y} \tag{2.19}$$

com pelo menos uma das desigualdades (2.18) e (2.19) a ser verificada de um modo estrito.

O seguinte teorema expõe um resultado clássico de caracterização de soluções eficientes do problema *PDC*.

Teorema 2.11 [Ze82] *Um ponto (\bar{x}, \bar{y}) é eficiente para o problema *PDC* se e só se existir um $\lambda \in (0, 1)$ para o qual (\bar{x}, \bar{y}) é solução óptima do seguinte programa linear:*

$$\begin{aligned} \min_{x,y} \quad & \lambda(c_1^T x + d_1^T y) + (1 - \lambda)d_2^T y \\ \text{sujeito a} \quad & A_2 x + B_2 y \leq b_2 \\ & x, y \geq 0 \end{aligned} \tag{2.20}$$

A partir desta caracterização necessária e suficiente de eficiência e das condições necessárias de optimalidade para programação de dois níveis descritas na secção 2.3, Bard [Ba84a] estabeleceu, erradamente, que toda a solução óptima de um problema *PDNL2* era eficiente. De facto, as condições de optimalidade do problema (2.20) com $\lambda = \frac{1}{1+v}$ são idênticas às condições de optimalidade apresentadas por Bard para o problema *PDNL2*. No entanto, pelo facto de tais condições estarem incorrectas, o resultado proposto por Bard está errado. Únlú [Un87] corrigiu o resultado de Bard e apresentou um algoritmo de resolução baseado em técnicas de multi-critério. Porém, Únlú apenas se apercebeu que o resultado de Bard estava errado para problemas *PDNL2* em que a solução óptima coincide com a solução óptima do problema relaxado *PR*. Assim também o resultado e o algoritmo de Únlú estão incorrectos.

São vários os contra exemplos apresentados na literatura que mostram que a solução óptima de um programa de dois níveis não é eficiente ([Ca88], [CIWe88], [HaSaWh90] e [Ma88]). Wen e Hsu [WeHs89] afirmaram que $d_1^T d_2 \leq 0$ é uma condição suficiente para que a solução óptima do problema *PDNL2* seja eficiente. Marcotte e Savard [MaSa91a] apresentam um contra exemplo para comprovar a incorrecção deste resultado. Este contra exemplo é tri-dimensional mas pode ser reduzido ao caso bi-dimensional. De facto, considere-se o seguinte programa de dois níveis com duas variáveis:

$$\begin{aligned} \min_{x,y} \quad & x + 2y \\ \text{sujeito a} \quad & x \leq 1 \\ & y \in \operatorname{argmin}\{-y : x + y \leq 2, y \geq 0\} \end{aligned} \tag{2.21}$$

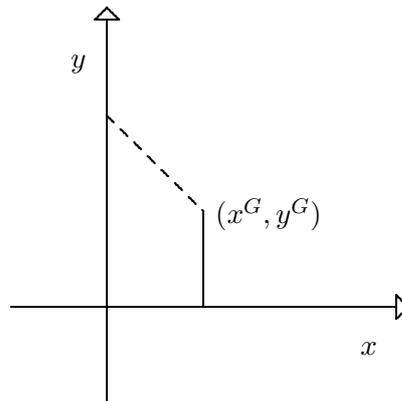


Figura 2.1: Região Induzida: - - -

O conjunto admissível $CR = \{(x, y) \in \mathbb{R}^2 : x + y \leq 2, x \leq 1, y \geq 0\}$ do problema relaxado *PR* e a região induzida $RI = \{(x, y) \in \mathbb{R}^2 : x + y = 2, 0 \leq x \leq 1\}$ do problema (2.21) encontram-se representados na figura 2.1.

A solução óptima do problema (2.21), $(x^G, y^G) = (1, 1)$, não é eficiente para o problema:

$$\begin{aligned} \min \quad & (x + 2y, -y) \\ \text{sujeito a} \quad & x + y \leq 2, x \leq 1, y \geq 0 \end{aligned} \tag{2.22}$$

e no entanto $d_1^T d_2 = 2 \cdot (-1) = -2 < 0$. De facto, o ponto admissível do problema (2.22), $(\bar{x}, \bar{y}) = (0.5, 1.1)$, tem valores objectivos para o primeiro e para o segundo nível, $F(\bar{x}, \bar{y}) = 2.7$ e $f(\bar{x}, \bar{y}) = -1.1$, que são respectivamente inferiores aos valores $F(x^G, y^G) = 3$ e $f(x^G, y^G) = -1$, correspondentes à solução óptima do programa de dois níveis (2.21).

Capítulo 3

Métodos de resolução para programação de dois níveis linear.

O critério de classificação seguido neste capítulo para agrupar, em diferentes categorias, os métodos de resolução para programação de dois níveis linear é semelhante aos propostos por Kolstad [Ko85] e por Savard [Sa89]. Deste modo, abordaremos os métodos de enumeração de pontos extremos, os métodos baseados nas condições de optimalidade, os algoritmos tipo *Branch and Bound* e ainda os métodos de penalidades. Existem porém metodologias que se enquadram em mais do que uma classe. Tais casos são devidamente assinalados. Entre os métodos baseados nas condições de optimalidade damos especial destaque ao método sequencial *LCP* uma vez que é apresentado, no capítulo seguinte, um estudo computacional com o referido processo.

Todos os métodos descritos nas secções 3.1, 3.2 e 3.3 determinam um mínimo global de um programa de dois níveis linear, ao contrário dos considerados na secção 3.4 que apenas garantem a obtenção de um mínimo local.

Na última secção deste capítulo é discutida a extensão dos métodos expostos nas secções 3.1, 3.2, 3.3 e 3.4 para programas de dois níveis lineares-quadráticos. Nessa secção é ainda estudada a validade para programação de dois níveis linear-quadrática das propriedades introduzidas na secção 2.4 para programas de dois níveis lineares.

3.1 Métodos de enumeração de pontos extremos.

Os algoritmos descritos nesta secção exploram de um modo enumerativo os pontos extremos do conjunto admissível do problema relaxado *PR*. Os métodos descritos são devidos a Candler e Townsley [CaTo82] e a Bialas e Karwan [BiKa84] e diferem entre si na ordem pela qual percorrem os pontos extremos de *CR*. Outras possíveis abordagens são consideradas em Dempe [De87], Papavassilopoulos [Pa82] e Tuy [Tu90].

Algoritmo Candler e Townsley.

Este algoritmo é apenas aplicável a programas de dois níveis lineares sem restrições do primeiro nível e que verifiquem as hipóteses do Teorema 2.8. A ideia chave subjacente ao método é construir uma sucessão de bases H_1, \dots, H_k, \dots óptimas para o problema do segundo nível, resolvendo em cada iteração a versão modificada do problema relaxado, $PR(H_k)$. O Teorema 2.7 garante que o algoritmo obtém um mínimo global num número finito de iterações, pois é finito o número possível de bases a considerar.

Com o intuito de reduzir o leque de bases a escolher em cada iteração, os autores estabeleceram o seguinte resultado:

Teorema 3.1 *Se W_i e W_j são bases óptimas dos problemas $PR(H_i)$ e $PR(H_j)$ respectivamente e se o valor óptimo do problema $PR(H_j)$ é menor que o valor óptimo do problema $PR(H_i)$, então a base H_j contém pelo menos uma coluna de custo reduzido positivo do problema $PR(H_i)$, relativamente à sua base óptima W_i .*

O algoritmo explora esta e outras técnicas a fim de reduzir o leque de bases seleccionáveis. Os autores não apresentam qualquer tipo de experiência computacional. No entanto, em [Ba83a] foi realizada uma série de testes comparativos com problemas de pequena-média dimensão ($n + m = 50$ e $l_2 = 25$ no máximo) que revelaram um fraco comportamento do algoritmo. Os elevados tempos de execução obtidos estão directamente relacionados com o grande número de bases que o algoritmo tem necessidade de explorar.

Bialas e Karwan [BiKa84] também descrevem um procedimento de enumeração de bases óptimas do problema do segundo nível. No entanto, este procedimento garante apenas a obtenção de um mínimo local e apenas acede a uma família restrita de bases via pivotações duais degeneradas. A sua aplicabilidade está por isso restringida ao objectivo de obter boas soluções iniciais para outras metodologias.

Algoritmo *k-best* de Bialas e Karwan.

Bialas e Karwan [BiKa84] desenvolveram outro método enumerativo de pontos extremos para o problema $PDNL2$. A aplicação deste método está condicionada aos casos em que CR é limitado e $M(\cdot)$ é uma aplicação unívoca. A principal diferença em relação ao método de Candler e Townsley consiste no tipo de bases a enumerar. Este método enumera as bases do problema relaxado PR em vez das bases do problema do segundo nível. O teorema 2.7 garante que o algoritmo determina um mínimo global do problema num número finito de iterações, uma vez que é finito o número de bases a explorar.

O algoritmo começa por determinar a solução óptima (x_1, y_1) do problema relaxado PR . Em cada iteração i , $i \geq 1$, é determinado um novo ponto extremo (x_{i+1}, y_{i+1}) do problema relaxado tal que $c_1^T x_{i+1} + d_1^T y_{i+1} \leq c_1^T x_i + d_1^T y_i$. O novo ponto extremo (x_{i+1}, y_{i+1}) é determinado a partir de pontos extremos adjacentes aos pontos extremos até então calculados (x_j, y_j) , $j = 1, \dots, i$. O algoritmo termina quando é encontrado o primeiro ponto extremo (x_k, y_k) que pertença à região induzida.

Os testes computacionais realizados pelos autores para problemas de pequena-média dimensão ($n + m = 90$ e $l_2 = 36$ no máximo), revelaram que a eficiência do algoritmo está relacionada com a diferença entre o valor óptimo do problema $PDNL2$ e o valor óptimo do problema relaxado PR . Além disso, problemas de média e grande dimensão criam sérios obstáculos à implementação do algoritmo, pois torna-se computacionalmente insustentável registar todas as bases adjacentes ainda por explorar.

3.2 Métodos baseados nas condições de optimalidade.

Um programa de dois níveis linear é um caso particular de um programa de dois níveis convexo. Deste modo, é possível substituir o problema do segundo nível pelas respectivas condições KKT e escrever o problema $PDNL$ como um programa linear de um só nível mas com restrições de complementaridade. Esse problema é normalmente denotado por $MLCP$ (*minimum linear complementarity problem*) e tem a forma:

$$\begin{aligned} \min \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \\ & -B_2^T \gamma + \beta = d_2 \end{aligned} \tag{3.1}$$

$$A_2 x + B_2 y + \alpha = b_2 \tag{3.2}$$

$$\alpha^T \gamma = \beta^T y = 0 \tag{3.3}$$

$$x, y, \alpha, \beta, \gamma \geq 0$$

Os métodos apresentados nesta secção para a resolução de programas de dois níveis lineares têm por base a formulação $MLCP$. A ideia chave comum a todos esses processos consiste em explorar as restrições de complementaridade $\alpha^T \gamma = \beta^T y = 0$. Ao contrário dos métodos descritos na secção anterior, os algoritmos apresentados nesta secção são aplicáveis a problemas de dois níveis com ou sem restrições do primeiro nível e não requerem, regra geral, quaisquer hipóteses suplementares. Assim abordaremos os métodos de Fortuny e McCarl [FoMc81] e de Bard e Falk [BaFa82] e ainda o método sequencial LCP ([BiKaSh80], [JuFa88] e [JuFa92a]).

Abordagem de Fortuny e McCarl.

Estes autores reformularam o problema linear com restrições de complementaridade *MLCP* no seguinte programa linear inteiro misto:

$$\begin{aligned} \min \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \\ & -B_2^T \gamma + \beta = d_2 \\ & A_2 x + B_2 y + \alpha = b_2 \\ & \alpha \leq M\xi, \quad \gamma \leq M(1 - \xi) \\ & \beta \leq M\eta, \quad y \leq M(1 - \eta) \\ & x, y, \alpha, \beta, \gamma \geq 0 \\ & \xi_i \in \{0, 1\}, \quad i = 1, \dots, l_2 \\ & \eta_j \in \{0, 1\}, \quad j = 1, \dots, m \end{aligned}$$

em que M é um número suficientemente grande e em que as variáveis ξ_i e η_j ao tomarem valores binários simulam as respectivas restrições de complementaridade $\alpha_i \gamma_i = 0$ e $\beta_j y_j = 0$, existentes no problema *MLCP*.

Fortuny e McCarl não apresentam quaisquer resultados computacionais que comprovem a eficiência desta abordagem. É no entanto do conhecimento geral que os métodos de resolução de programas lineares inteiros ou inteiros mistos apresentam, regra geral, tempos de resolução computacional que crescem de forma exponencial com o aumento da dimensão dos problemas.

Ónal [On92] realizou testes computacionais com esta metodologia mas os resultados não são animadores. Com efeito na maior parte dos casos apenas se assegurou a determinação de mínimos locais.

Abordagem de Bard e Falk.

Este processo de resolução consiste em transformar o problema *PDNL2* num programa com variáveis separáveis. Introduzindo as variáveis z e w , podemos substituir as restrições (3.2) e (3.3) pelas seguintes restrições separáveis:

$$\sum_{i=1}^{l_2} (\min(0, z_i) + \gamma_i) = 0 \quad (3.4)$$

$$\sum_{i=1}^m (\min(0, w_i) + \beta_i) = 0 \quad (3.5)$$

$$\begin{aligned}
b_2 - A_2x - B_2y - \gamma &= z \\
y - \beta &= w
\end{aligned}$$

O programa resultante é separável e, apesar da não diferenciabilidade das restrições (3.4) e (3.5), permite a aplicação de um algoritmo para programas separáveis não convexos desenvolvido por Falk [Fa72]. Este algoritmo utiliza um processo tipo *Branch and Bound* e envolve a partição do domínio de admissibilidade. Os autores resolveram pequenos exemplos ($n + m = 5$ e $l_2 = 3$ no máximo) salientando o bom comportamento do método. No entanto, Bard [Ba83a] confirmou, como seria de esperar, a convergência lenta do método para problemas de pequena-média dimensão ($n + m = 50$ e $l_2 = 25$ no máximo). A causa directa desse comportamento é o elevado número de nós a explorar pelo processo *Branch and Bound*.

Método sequencial LCP.

A utilização de esquemas sequenciais para a resolução de programas de dois níveis foi abordada pela primeira vez por Bialas, Karwan e Shaw [BiKaSh80]. O método sequencial proposto resolve em cada iteração k o seguinte problema linear complementar, $LCP(k)$ (*linear complementarity problem*):

$$\begin{aligned}
A_1x + B_1y &\leq b_1 \\
-B_2^T\gamma + \beta &= d_2 \\
A_2x + B_2y + \alpha &= b_2 \\
c_1^T x + d_1^T y &\leq \lambda_k \\
\alpha^T \gamma &= \beta^T y = 0 \\
x, y, \alpha, \beta, \gamma &\geq 0
\end{aligned}$$

A solução deste problema, (x_k, y_k) , pertence à região induzida do problema $PDNL$ e tem valor objectivo para o primeiro nível, $c_1^T x_k + d_1^T y_k$, inferior ou igual a λ_k . O objectivo do método é resolver iterativamente uma sucessão de problemas $LCP(k)$ correspondentes a uma sucessão decrescente de parâmetros λ_k que são actualizados de iteração em iteração. O método termina, ao ser encontrado o primeiro $LCP(k)$ que não tenha solução. Os passos deste algoritmo são descritos do seguinte modo:

- passo 0 - Seja λ_0 um limite superior para $c_1^T x + d_1^T y$ e $k = 0$,
- passo 1 - Resolver o problema $LCP(k)$. Se este problema não tiver solução ir para 3.
Caso contrário seja (x_k, y_k) a solução e ir para 2,
- passo 2 - Fazer $\lambda_{k+1} = c_1^T x_k + d_1^T y_k - \rho(c_1^T x_k + d_1^T y_k)$, $k = k + 1$ e ir para 1,
- passo 3 - (x_{k-1}, y_{k-1}) é solução ϵ -óptima para o problema $PDNL$ com $\epsilon = \rho(c_1^T x_{k-1} + d_1^T y_{k-1})$.

Nesse processo ρ é um parâmetro positivo de valor previamente estabelecido. Além disso,

as restrições $-B_2^T \gamma + \beta = d_2$ podem ser substituídas por:

$$\nu H - B_2^T \gamma + \beta = d_2$$

com H matriz positiva definida e ν escalar positivo de valor reduzido. Esta substituição tem por objectivo aumentar a estabilidade numérica do processo. Escolhas de $H = I$ e de ν entre 10^{-2} e 10^{-4} são sugeridas em [BiKa84].

Ao contrário dos processos anteriormente descritos, o algoritmo apenas garante a obtenção de uma solução ϵ -global. No entanto para a maioria dos problemas práticos tal solução é de facto global [JuFa92a].

Bialas, Karwan e Shaw sugeriram um esquema simplex modificado para resolver o problema $LCP(k)$ para o caso particular em que $d^1 \leq 0$. Este esquema foi originalmente proposto por Wolfe [Wo59] para a resolução de programas quadráticos convexos e consiste em usar uma forma modificada do método simplex em que não é autorizada que variáveis complementares entre si sejam simultaneamente básicas. No entanto ao utilizar este procedimento, o método sequencial anteriormente descrito não converge para a solução óptima do problema $PDNL$. Exemplos demonstrativos deste facto foram já documentados na literatura por Ben-Ayed e Blair [BeBl90] e Júdice e Faustino [JuFa88]. A razão para esta divergência relaciona-se com a existência de variáveis não complementares no problema $LCP(k)$.

O método sequencial LCP tem por base o esquema sequencial anteriormente apresentado, sendo cada $LCP(k)$ resolvido por um processo enumerativo híbrido desenvolvido por Júdice e Faustino ([JuFa88] e [JuFa92a]).

O método enumerativo consiste em resolver o problema $LCP(k)$ através de um esquema enumerativo em árvore binária. Para o efeito o problema $LCP(k)$ é reformulado na seguinte forma:

$$w = q + Mz \tag{3.6}$$

$$w, z \geq 0 \tag{3.7}$$

$$w_i z_i = 0, \quad i = 1, \dots, l_2 + m$$

$$\text{em que } M = \begin{bmatrix} 0 & -B_2 & -A_2 \\ B_2^T & 0 & 0 \\ 0 & -B_1 & -A_1 \\ 0 & -d_1^T & -c_1^T \end{bmatrix}, \quad w = \begin{bmatrix} \alpha \\ \beta \\ \delta \\ v_0 \end{bmatrix}, \quad z = \begin{bmatrix} \gamma \\ y \\ x \end{bmatrix} \quad \text{e } q = \begin{bmatrix} b_2 \\ d_2 \\ b_1 \\ \lambda_k \end{bmatrix}.$$

Em cada ramificação da árvore binária são *fixas* a nível zero pares de variáveis complementares entre si. A figura 3.1 pretende exemplificar uma situação deste tipo.

O nó raiz da árvore está associado a um ponto que satisfaz (3.6) e (3.7). Este ponto inicial é normalmente determinado por uma modificação da fase 1 do método simplex que

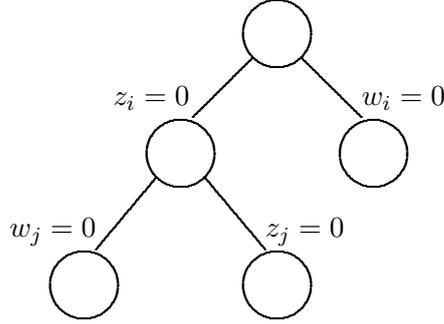


Figura 3.1: Exemplo de uma árvore binária do método enumerativo.

consiste em minimizar uma variável artificial não negativa z_0 no conjunto de restrições:

$$w = q + pz_0 + Mz, \quad w, z \geq 0$$

em que o vector p tem componentes não negativas e $p_i > 0$ para todo o i tal que $q_i < 0$. Este processo de minimização tem por objectivo trazer a variável z_0 a nível zero tentando manter, tanto quanto possível, pares de variáveis entre si complementares não simultaneamente básicas. Eventualmente uma solução complementar para o problema $LCP(k)$ pode ser atingida.

Em [Faus92] é sugerido um processo para a obtenção de um ponto admissível de CR . Nesse processo (designado pelo autor por **START**) considera-se o seguinte problema bilinear que como veremos na secção 3.4 se pode associar ao conjunto de restrições do $MLCP$:

$$\begin{aligned} \min \quad & \begin{bmatrix} c_1 \\ d_1 + d_2 \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} - b_2^T w + w^T \begin{bmatrix} A_2 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\ \text{sujeito a} \quad & A_2 x + B_2 y \leq b_2, \quad x, y \geq 0 \\ & B_2^T w \geq d_2, \quad w \geq 0 \end{aligned}$$

Esse programa é resolvido através de um algoritmo apresentado em [Ko76a] para a resolução de programas bilineares. Assim este método procura minimizar simultaneamente a função do primeiro nível e a *gap* dual do problema do segundo nível. Note-se que tal processo apenas determina um ponto estacionário do programa bilinear anterior. Se esse ponto satisfaz as condições (3.6) e (3.7) e a condição de complementaridade, então é solução inicial do $LCP(0)$. De outro modo o algoritmo fase 1 começa com essa solução.

Em cada nó da árvore é *fixa* a nível zero uma variável complementar escolhida. Para isso a variável complementar escolhida é minimizada no conjunto de restrições (3.6) e (3.7) ao qual se adicionam restrições do tipo $z_i = 0$ ou $w_i = 0$ provenientes de todas as variáveis complementares *fixas* a nível zero ao longo do caminho da árvore até então

percorrido desde a raiz ao nó actual. Se o mínimo obtido é zero a ramificação prossegue com tal variável *fixa* a nível zero para todos os nós descendentes do nó actual. Senão, a ramificação é interrompida e o nó em causa não pode mais vir a ser explorado.

Deste modo, ou é encontrada uma solução complementar para o problema $LCP(k)$ num determinado nó da árvore ou então o processo termina sem mais nós a explorar. Nesse último caso o problema $LCP(k)$ não tem solução.

Algumas regras heurísticas auxiliares para a escolha do par de variáveis a seleccionar e dentro destas da variável a escolher para ramificar foram sugeridas pelos autores no sentido de melhorar a eficiência do processo enumerativo [JuFa88].

Em cada nó e após fixar em zero a variável complementar escolhida é possível utilizar um processo que reduz o esforço para encontrar a solução complementar desejada. Este processo sugerido por Al-Khayyal [Al87] é uma versão modificada do método de gradientes reduzidos e determina um mínimo local em estrela da função $\sum_{i=1}^{l_2+m} z_i w_i$ examinando todos os vértices adjacentes do ponto em causa, (\bar{z}, \bar{w}) . Os autores provaram que tal minimização pode ser realizada recorrendo aos custos reduzidos da seguinte função linear:

$$\sum_{i=1}^{l_2+m} \bar{z}_i w_i + \bar{w}_i z_i$$

A fim de melhorar a eficiência do método sequencial LCP , Júdice e Faustino [JuFa92a] propuseram um esquema de maximização da variável *slack* v_0 após a resolução de cada problema $LCP(k)$. Este esquema tem a finalidade de reduzir o valor do parâmetro λ_{k+1} para a iteração seguinte, aumentando a eficiência do processo iterativo. O esquema sugerido reduz-se à versão proposta por Bialas e Karwan para resolver cada $LCP(k)$ a fim de não perturbar a complementaridade alcançada pelo processo enumerativo.

A experiência computacional realizada pelos autores para problemas sem restrições do primeiro nível ([HaJaSa92] e [JuFa88]), de média-grande dimensão ($n + m = 400$ e $l_2 = 150$ no máximo), revelaram um bom comportamento do método em comparação com o método *Branch and Bound* de Bard e Moore [BaMo90]. Estes testes computacionais revelaram também que em mais de 60% dos casos o óptimo global é atingido. O estudo provou ainda que o esforço computacional do método sequencial é, em grande parte, concentrado na resolução do último $LCP(k)$, isto é, em provar que a solução obtida na iteração anterior é de facto ϵ -óptima. Os autores sugerem ainda escolhas para o parâmetro ρ de 0.001 com possíveis relaxações para 0.01 ou ainda 0.05 a partir da resolução de um determinado número de problemas $LCP(k)$ estimado na ordem de $10(l_2 + m)$.

No capítulo seguinte é realizado um estudo computacional do método sequencial LCP para outros problemas teste [CaVi92]. Como será indicado na altura esse estudo mostra as vantagens e desvantagens do processo indicado nesta secção.

3.3 Métodos Branch and Bound.

Os métodos resumidamente expostos nesta secção utilizam o processo clássico de ramificação em árvore, típico dos algoritmos *Branch and Bound*. Em cada nó da árvore existem condições próprias que uma vez verificadas interrompem a ramificação em curso. O processo é então recomeçado num nó ainda por explorar e termina quando não existem mais nós por explorar. Os algoritmos discutidos nesta secção são devidos a Bard e Moore [BaMo90] e a Hansen, Jaumard e Savard [HaJaSa92] e não requerem quaisquer hipóteses suplementares, sendo aplicáveis a programas de dois níveis lineares com ou sem restrições do primeiro nível.

Algoritmo de Bard e Moore.

Este algoritmo foi desenvolvido para a resolução de programas de dois níveis lineares-quadráticos. No entanto, como o processo não sofre quaisquer alterações quando aplicado a programas de dois níveis lineares, limitar-nos-emos a descrevê-lo nesse caso. É também importante realçar que este algoritmo utiliza directamente a formulação *MLCP* e como tal também se enquadra na classe de algoritmos que se baseiam nas condições de optimalidade.

A ideia chave do método é semelhante à da abordagem de Fortuny e McCarl pelo modo como explora as restrições de complementaridade (3.3). O problema *MLCP* é, em primeiro lugar, resolvido sem as restrições de complementaridade (3.3):

$$\begin{aligned} \min \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \\ & -B_2^T \gamma + \beta = d_2 \\ & A_2 x + B_2 y + \alpha = b_2 \\ & x, y, \alpha, \beta, \gamma \geq 0 \end{aligned}$$

Designaremos este problema por *PL*, mesmo quando a ele se adicionarem mais restrições.

Em cada iteração um teste é realizado para verificar se as restrições de complementaridade (3.3) são ou não satisfeitas. Em caso afirmativo o ponto pertence à região induzida *RI* e é um candidato a óptimo. Se tal não acontecer, um processo tipo *Branch e Bound* é utilizado para implicitamente examinar todas as possíveis combinações complementares, atribuindo alternadamente o valor zero a pares de variáveis complementares.

Para uma breve descrição dos passos do algoritmo seja $u = \begin{bmatrix} \gamma \\ \beta \end{bmatrix}$, $v = \begin{bmatrix} \alpha \\ y \end{bmatrix}$ e \bar{F} limite superior da função objectivo do primeiro nível. Em cada nó k da árvore binária

definem-se os seguintes subconjuntos de índices:

$$\begin{aligned} W_k &\subseteq W \\ S_k^+ &= \{i : i \in W_k \text{ e } u_i = 0\} \\ S_k^- &= \{i : i \in W_k \text{ e } v_i = 0\} \\ S_k^0 &= \{i : i \notin W_k\} \end{aligned}$$

Os passos do algoritmo Bard e Moore são descritos do seguinte modo:

- passo 0 - (Inicialização) Fazer $k = 0$, $S_k^+ = S_k^- = \emptyset$, $S_k^0 = W$ e $\bar{F} = +\infty$
- passo 1 - (Iteração k) Resolver o problema PL com $u_i = 0$ para $i \in S_k^+$ e $v_i = 0$ para $i \in S_k^-$. Se PL não tiver solução fazer $k = k + 1$ e ir para 5. Caso contrário fazer $k = k + 1$ e seja (x_k, y_k) a solução óptima de PL .
- passo 2 - (Teste) Se $F(x_k, y_k) \geq \bar{F}$ ir para 5.
- passo 3 - (Ramificação) Se $u_i v_i = 0$, $i = 1, \dots, l_2 + m$ ir para 4. Caso contrário seleccionar o i para o qual $u_i v_i$ é o maior possível (i_k). Fazer $S_k^+ = S_k^+ \cup \{i_k\}$, $S_k^0 = S_k^0 \setminus \{i_k\}$ e $S_k^- = S_k^-$ e ir para 1.
- passo 4 - (Actualização) $\bar{F} = F(x_k, y_k)$
- passo 5 - (*Backtracking*) Se não existir mais nenhum nó livre (um nó é livre quando ramificado em 3) ir para 6. Caso contrário para um dado nó j livre fazer $S_k^+ = S_j^+ \setminus \{i_j\}$, $S_k^- = S_j^- \cup \{i_j\}$, $S_k^0 = S_j^0$ e ir para 1.
- passo 6 - (Terminação) Se $\bar{F} = +\infty$ o problema $PDNL$ não é admissível. Senão, a sua solução óptima corresponde ao valor final de \bar{F} .

A experiência computacional realizada pelos autores para problemas lineares e lineares-quadráticos de pequena-média dimensão ($n + m = 100$ e $l_2 = 40$ no máximo) revelou um crescimento exponencial dos tempos de execução com o aumento da dimensão dos problemas. Os autores compararam este algoritmo com o de Fortuny e McCarl (utilizando o código ZOOM para programas lineares inteiros) e concluíram que o primeiro processo é 10 a 100 vezes mais rápido. Várias regras de ramificação são sugeridas e testadas pelos autores a fim de melhorarem a eficiência do algoritmo proposto.

Algoritmo de Hansen, Jaumard e Savard.

Para este algoritmo a ramificação é processada igualando a zero uma das variáveis y_j , $j \in \{1, \dots, m\}$ do segundo nível do problema $PDNL$ ou igualando a zero uma das variáveis *slack* das restrições do segundo nível $A_2x + B_2y \leq b_2$. Em qualquer dos casos é sempre possível eliminar uma variável y_j , $j \in \{1, \dots, m\}$, ou seja, fixar o seu valor ao longo da ramificação consequente.

Com o intuito de encontrar condições que reduzam a amplitude do processo de ramificação os autores estabeleceram o seguinte resultado:

Teorema 3.2 [HaJaSa92] *Em qualquer solução óptima do problema PDNL o número de restrições activas (se i é activa (não activa) então $\lambda_i = 1$ ($\lambda_i = 0$)) verifica a seguinte condição:*

$$\sum_{i:(B_2)_{ij}>0} \lambda_i \geq 1 \quad \text{se } (d_2)_j < 0$$

$$\sum_{i:(B_2)_{ij}>0} \lambda_i + \lambda_{l_2+j} \geq 1 \quad \text{se } (d_2)_j > 0$$

para $j = 1, \dots, m$.

Estas condições são actualizadas de nó em nó e permitem uma redução do leque de escolha de variáveis para ramificar, uma vez que se tratam de condições necessárias de optimalidade.

Para cada nó, os autores sugerem um conjunto de testes que permitem inferir alguma informação relativa às características da solução correspondente. Estes testes são classificados de acordo com a nomenclatura proposta em Hansen, Jaumard e Lu [HaJaLu90]. Resumidamente, os testes envolvem a resolução do problema relaxado PR com todas as variáveis y_j , correspondentes aos nós do caminho da árvore até então percorrido, fixas com o valor respectivo e a resolução do problema do segundo nível $P(\bar{x})$ com (\bar{x}, \bar{y}) solução do problema anterior. O problema do segundo nível é resolvido de duas formas diferentes, nomeadamente com as referidas variáveis y_j fixas como em PR e com todas as variáveis y_j , $j = 1, \dots, m$ livres. Os autores sugerem ainda outro tipo de testes relacionados com penalidades associadas ao quadro simplex do problema PR correspondente.

Os autores realizaram testes computacionais para um vasto leque de estratégias que englobam diferentes regras de ramificação e heurísticas para determinar o primeiro ponto admissível (associado ao nó raiz). Os resultados computacionais revelaram um bom comportamento computacional para problemas de média-grande dimensão ($n + m = 400$ e $l_2 = 150$ no máximo) relativamente ao método *Branch and Bound* de Bard e Moore com o qual houve uma comparação directa. Em relação ao método sequencial LCP os autores afirmam que os tempos de execução são relativamente da mesma ordem de grandeza, apesar de não ter havido uma comparação directa entre os dois processos.

3.4 Métodos de penalidades.

Nesta secção abordaremos os métodos de White e Anandaligam [WhAn89] e de Bi, Calamai e Conn [BiCaCo89]. Outras estratégias de resolução envolvendo funções de penalidade e de barreira foram também abordadas na literatura ([AiSh81], [AiSh84] e [IsAi92]). No entanto, estas últimas abordagens são enquadradas nos métodos de resolução para programação de dois níveis não linear e como tal serão discutidas somente no capítulo 5.

Abordagem de White e Anandaligam.

A principal ideia subjacente a esta abordagem consiste em rescrever a função objectivo do primeiro nível do problema *PDNL2* de modo que esta inclua um termo que penalize a *gap* primal-dual do problema do segundo nível. O programa dual do problema do segundo nível $P(x)$ tem a forma:

$$\begin{aligned} & \max_w (b_2 - A_2x)^T w \\ & \text{sujeito a } B_2^T w \geq d_2, \quad w \geq 0 \end{aligned}$$

Desta forma a *gap* primal dual de $P(x)$ é definida como:

$$\pi(x, y, w) = d_2^T y - (b_2 - A_2x)^T w$$

O método de penalidades desenvolvido pelos autores resolve uma sequência de programas bilineares $PP(\delta_k)$:

$$\begin{aligned} & \min_{x,y,w} P_k(x, y, w) = c_1^T x + d_1^T y + \delta_k \pi(x, y, w) \\ & \text{sujeito a } A_2x + B_2y \leq b_2, \quad x, y \geq 0 \end{aligned} \tag{3.8}$$

$$B_2^T w \geq d_2, \quad w \geq 0 \tag{3.9}$$

em que o parâmetro real positivo δ_k é incrementado iterativamente.

Todos os resultados associados ao método de penalidades em questão foram estabelecidos sob as hipóteses de os conjuntos formados respectivamente pelos conjuntos de restrições (3.8) e (3.9):

$$Z = \{(x, y) : A_2x + B_2y \leq b_2, x, y \geq 0\} \quad \text{e} \quad W = \{w : B_2^T w \geq d_2, w \geq 0\}$$

serem limitados e com pontos extremos não degenerados. Além disso é assumido que a aplicação $M(\cdot)$ é unívoca.

Como resultado principal, os autores provaram que a função de penalidade $P_k(x, y, w)$ é exacta. Esta propriedade garante a convergência finita do processo iterativo e é formalizada através do seguinte resultado:

Teorema 3.3 [WhAn89] *Existe $\delta^* > 0$ tal que para $\delta \geq \delta^*$ a solução óptima do problema $PP(\delta)$ coincide com a solução óptima do problema *PDNL2*.*

A teoria dos métodos de penalidades [BaSh79] garante, não só que a sucessão de valores $P_k(x_k, y_k, w_k)$, em que (x_k, y_k, w_k) é a solução óptima de $PP(\delta_k)$, é crescente, como também que a sucessão de valores $\pi(x_k, y_k, w_k)$ é decrescente. Na prática, o facto de a função de penalidade ser exacta implica que existe um inteiro i para o qual $\pi(x_i, y_i, w_i) = 0$, sendo esta condição o verdadeiro critério de paragem do método em questão. O ponto (x_i, y_i, w_i) é solução óptima do problema *PDNL2*.

Como é facilmente compreensível, o comportamento deste método está em estrita dependência da forma como é resolvido cada problema bilinear $PP(\delta_k)$. Os autores propõem um esquema de resolução em que técnicas de pesquisa de pontos extremos são usadas. A utilização deste tipo de procedimento é consequência do facto de existir uma solução óptima (x_k, y_k, w_k) para a qual (x_k, y_k) e w_k são pontos extremos respectivamente dos conjuntos Z e W .

Os autores consideram ainda a possibilidade de o método parar prematuramente, isto é terminar numa iteração i para a qual $\pi(x_i, y_i, w_i) > 0$. Neste caso é possível estabelecer limites inferiores e superiores para medidas de optimalidade referentes aos dois níveis do problema $PDNL2$. Deste modo, os autores concluíram que:

- $\delta_i \pi(x_i, y_i, w_i)$ é um limite inferior para a diferença entre o valor da solução óptima (x^*, y^*) e o valor da função objectivo do primeiro nível em (x_i, y_i) , ou seja:

$$\delta_i \pi(x_i, y_i, w_i) \leq c_1^T x^* + d_1^T y^* - (c_1^T x_i + d_1^T y_i)$$

- $\pi(x_i, y_i, w_i)$ é um limite superior da perda de optimalidade do segundo nível, de y_i em relação a $y(x_i)$:

$$d_2^T y_i - d_2^T y(x_i) \leq \pi(x_i, y_i, w_i)$$

Note-se que y_i não é obrigatoriamente a solução óptima do problema $P(x_i)$, pois se assim fosse $\pi(x_i, y_i, w_i) = 0$ e $(x_i, y_i) = (x^*, y^*)$,

- é também possível limitar inferiormente a perda de optimalidade do primeiro nível, de $(x_i, y(x_i))$ relativamente a (x^*, y^*) , pois tem-se:

$$c_1^T x^* + d_1^T y^* - (c_1^T x_i + d_1^T y(x_i)) \geq \delta_i \pi(x_i, y_i, w_i) + d_2^T (y_i - y(x_i))$$

Como notas finais, saliente-se o facto de, em cada iteração k , ser possível determinar apenas um mínimo local do problema $PP(\delta_k)$. Nesta situação, a solução final obtida é um mínimo local do problema $PDNL2$. É ainda importante realçar que o método proposto é extensível a programas de dois níveis lineares inteiros mistos para os quais as variáveis do primeiro nível x tomam valores discretos.

Abordagem de Bi, Calamai e Conn.

A formulação (2.5) para o caso da programação de dois níveis linear sem restrições do primeiro nível (inclusive de não negatividade) reveste a seguinte forma:

$$\begin{aligned} \min_{x, y, \beta, \gamma} \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & -B_2^T \gamma + \beta = d_2 \\ & \min(\gamma_i, s_i(x, y)) = 0, \quad i = 1, \dots, l_2 \\ & \min(\beta_j, y_j) = 0, \quad j = 1, \dots, m \end{aligned}$$

em que $s_i(x, y)$ é a linha i de $b_2 - A_2x - B_2y$.

A partir desta formulação os autores estabeleceram uma equivalência local entre o problema $PDNL2$ e um programa penalizado sem restrições. Esta equivalência é formalizada no teorema seguinte. Este resultado, bem como todos os restantes estabelecidos pelos autores, pressupõem válidas as seguintes hipóteses:

- $c_1^T x + d_1^T y$ limitada inferiormente em RI ,
- $d_2^T y$ limitada inferiormente em CR ,
- qualquer solução local do problema do segundo nível é tal que os multiplicadores associados às correspondentes restrições activas são estritamente positivos. Esta condição implica que a aplicação $M(\cdot)$ é unívoca.

Teorema 3.4 *Sejam $(\bar{x}, \bar{y}, \bar{\beta}, \bar{\gamma})$ um mínimo local para o problema $PDNL2$ e λ_i , $i = 1, \dots, l_2 + 2m$ os multiplicadores de Lagrange associados ao seguinte programa linear resolvido numa vizinhança local V de (\bar{x}, \bar{y}) [BiCaCo89]:*

$$\begin{aligned} \min \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & \gamma_i = 0 \text{ ou} \\ & s_i(x, y) = 0 \text{ para } \gamma_i > 0 \text{ em } V, \quad i = 1, \dots, l_2 \\ & \beta_j = 0 \text{ ou} \\ & y_j = 0 \text{ para } \beta_j > 0 \text{ em } V, \quad j = 1, \dots, m \\ & t_i(\beta, \gamma) = 0, \quad i = 1, \dots, m \end{aligned}$$

em que $t_i(\beta, \gamma)$ é a linha i de $d_2 + B_2^T \gamma - \beta$. Então para $\lambda^* \geq \lambda = \max_{1 \leq i \leq l_2 + 2m} \lambda_i$, $(\bar{x}, \bar{y}, \bar{\beta}, \bar{\gamma})$ também é um mínimo local da seguinte função:

$$\begin{aligned} P_\lambda(x, y, \beta, \gamma) = \quad & c_1^T x + d_1^T y + \lambda \left(\sum_{i=1}^{l_2} | \min(\gamma_i, s_i(x, y)) | + \right. \\ & \left. \sum_{i=1}^m | \min(\beta_i, y_i) | + \sum_{i=1}^m | t_i(\beta, \gamma) | \right) \end{aligned}$$

Os autores desenvolveram um método descendente para minimizar a função de penalidade $P_\lambda(x, y, \beta, \gamma)$ baseado em técnicas que lidam com a não diferenciabilidade envolvida. O facto de a função de penalidade $P_\lambda(x, y, \beta, \gamma)$ ser exacta [BiCaCo89] permite utilizar um esquema de penalidades que, ao aumentar iterativamente o valor de λ , termina num número finito de passos. No entanto, o método de penalidades exactas proposto tem, ao invés do anteriormente descrito, a contrariedade do ponto final obtido (x^*, y^*) ser apenas um mínimo local do problema $PDNL2$.

3.5 Extensões ao caso linear-quadrático.

Nesta secção iremos investigar a aplicação dos algoritmos e resultados apresentados nas secções anteriores à programação de dois níveis linear-quadrática. Um programa de dois níveis linear-quadrático (*PDNLQ*) é normalmente formulado na seguinte forma:

$$\begin{aligned} \min_{x,y} \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \end{aligned} \tag{3.10}$$

$$x \geq 0 \tag{3.11}$$

$$y \in \operatorname{argmin} \left\{ \frac{1}{2} y^T Q y + y^T S x + d_2^T y : A_2 x + B_2 y \leq b_2, y \geq 0 \right\}$$

em que $c_1 \in \mathbb{R}^n$, $d_1, d_2 \in \mathbb{R}^m$, $A_1 \in \mathbb{R}^{l_1 \times n}$, $A_2 \in \mathbb{R}^{l_2 \times n}$, $Q \in \mathbb{R}^{m \times m}$, $S \in \mathbb{R}^{m \times n}$, $B_1 \in \mathbb{R}^{l_1 \times m}$, $B_2 \in \mathbb{R}^{l_2 \times m}$, $b_1 \in \mathbb{R}^{l_1}$ e $b_2 \in \mathbb{R}^{l_2}$ e Q uma matriz positiva definida.

Tal como anteriormente podemos ainda considerar o problema *PDNLQ2* em que não existem restrições do primeiro nível (3.10). As restrições de não negatividade (3.11) são consideradas incluídas no problema *PDNLQ2* caso nada seja dito em contrário.

Os métodos de enumeração de pontos extremos expostos na secção 3.1 não são aplicáveis ao problema *PDNLQ*. Com efeito iremos mostrar mais adiante que a solução óptima de um programa de dois níveis linear-quadrático pode não ocorrer num ponto extremo do conjunto admissível do problema *PR*.

Relativamente aos métodos baseados nas condições de optimalidade, a sua extensão à programação de dois níveis linear-quadrática é, regra geral, bem sucedida. Com o intuito de clarificar esta questão o problema *PDNLQ* é rescrito na sua versão de um só nível linear com restrições de complementaridade, *MLCP'*, por intermédio da substituição do problema do segundo nível pelas suas condições KKT:

$$\begin{aligned} \min \quad & c_1^T x + d_1^T y \\ \text{sujeito a} \quad & A_1 x + B_1 y \leq b_1 \\ & -Qy - Sx - B_2^T \gamma + \beta = d_2 \\ & A_2 x + B_2 y + \alpha = b_2 \\ & \alpha^T \gamma = \beta^T y = 0 \\ & x, y, \alpha, \beta, \gamma \geq 0 \end{aligned}$$

O problema *MLCP'* apresenta uma estrutura muito semelhante à do problema *MLCP* para programação de dois níveis linear. Deste modo, é imediata as extensões dos métodos de Fortuny e McCarl, Bard e Falk e Bard e Moore e do método sequencial *LCP* a programação de dois níveis linear-quadrática.

Júdice e Faustino [JuFa92b] resolveram com o método sequencial *LCP* programas de dois níveis lineares-quadráticos . A única significativa alteração do método enumerativo proposto por estes autores, residiu na forma como é realizada a procura de um mínimo local em estrela em cada nó da árvore binária. A experiência computacional efectuada pelos autores para problemas *PDNLQ* de média dimensão ($n + m = 100$, $l_1 = 10$ e $l_2 = 30$ no máximo) revelou sensivelmente as mesmas características que a realizada para problemas *PDNL*: melhor comportamento computacional em relação ao método *Branch e Bound* de Bard e Moore e dificuldades em provar que o último problema linear complementar não tem solução. No entanto, a complexidade computacional da resolução de programas de dois níveis lineares-quadráticos é superior, de acordo com os testes realizados, à dos programas de dois níveis lineares. Os autores confirmaram que, na presença de restrições do primeiro nível, o grau de complexidade da resolução de programas de dois níveis é consideravelmente maior. Esta diferença é ainda mais acentuada na presença de problemas de segundo nível de maiores dimensões. De referir ainda que o processo *START* não pode ser utilizado pois neste caso o problema $PP(\rho_k)$ com $\delta_k = 1$ é um programa quadrático indefinido de muito mais difícil resolução.

Os métodos de penalidades descritos na secção 3.4 são extensíveis ao caso linear-quadrático. A abordagem de Bi, Calamai e Conn foi estendida para programas de dois níveis não lineares e é referida no capítulo 5. O método de White e Anandalgam apresenta maiores dificuldades de adaptação ao caso linear-quadrático. Estes obstáculos são de duas ordens:

- dificuldade em encontrar funções de penalidades exactas a fim de garantir a convergência finita do processo iterativo,
- dificuldade em resolver o problema penalizado. Por exemplo, se a função de penalidade for da forma $c_1^T x + d_1^T y + \delta_k \pi(x, y, w)$, em que $\pi(x, y, w)$ é a *gap* primal-dual do problema do segundo nível, o problema penalizado resultante é quadrático indefinido e a sua resolução é consideravelmente mais complexa.

A fim de verificar a validade dos resultados descritos na secção 2.4 para problemas *PDNL*, é apresentado o seguinte programa de dois níveis linear-quadrático definido em \mathbb{R}^2 :

$$\begin{aligned} \min_{x,y} \quad & -y \\ \text{sujeito a} \quad & x \geq 0 \\ & y \in \operatorname{argmin}\{-\frac{1}{2}y^2 + xy : x + y \leq 1, y \geq 0\} \end{aligned} \tag{3.12}$$

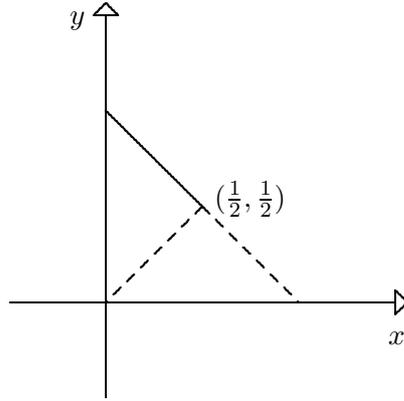


Figura 3.2: Região Induzida: - - -

A Figura 3.2 representa geometricamente o conjunto admissível do problema relaxado $CR = \{(x, y) \in \mathbb{R}^2 : x + y \leq 1, x, y \geq 0\}$ e a região induzida $RI = \{(x, y) \in \mathbb{R}^2 : y = x, 0 \leq x \leq \frac{1}{2}\} \cup \{(x, y) \in \mathbb{R}^2 : x + y = 1, \frac{1}{2} \leq x \leq 1\}$, correspondentes ao problema (3.12). A solução óptima do problema (3.12) é o ponto $(\frac{1}{2}, \frac{1}{2})$ da região induzida RI , para o qual a função objectivo do primeiro nível $-y$ toma o menor valor. Como é facilmente verificável este ponto não é um ponto extremo do conjunto CR . Deste modo, conclui-se que os teoremas 2.7 e 2.8 não são válidos para programação de dois níveis linear-quadrática.

Este exemplo mostra que um ponto não extremo do conjunto CR pode ser ponto extremo da região induzida. Este facto está na origem da maior complexidade do caso linear em relação ao caso linear-quadrático.

O exemplo apresentado também mostra que o teorema 2.9 não é válido para programação de dois níveis linear-quadrática. De facto, os pontos $(0, 1)$ e $(1, 0)$ pertencem ao conjunto CR e a combinação convexa destes dois pontos $(\frac{3}{4}, \frac{1}{4}) = \frac{3}{4}(1, 0) + \frac{1}{4}(0, 1)$ pertence à região induzida RI , mas $(0, 1)$ não é um ponto de RI .

Capítulo 4

Problemas teste para programação de dois níveis.

No capítulo anterior discutimos os principais métodos existentes para a resolução de programas de dois níveis lineares e lineares-quadráticos. Apesar de existirem várias técnicas mais ou menos eficientes para o efeito, não foi ainda apresentado um estudo computacional comparativo desses processos. Uma das razões deste facto reside na ausência de um conjunto de problemas teste de acesso fácil para os utilizadores dos respectivos códigos. Neste capítulo, é descrita uma técnica simples de construção de problemas teste que julgamos poder ser utilizada para o fim referido [CaVi92].

A técnica a apresentar neste capítulo permite ao utilizador ter controlo sobre um vasto leque de características dos problemas teste. Assim, é possível ajustar a dimensão dos problemas, o número e tipo de mínimos locais e globais e a densidade e o número de condição das matrizes utilizadas. Como é perfeitamente conhecido, a dificuldade da determinação de um mínimo global de um programa de dois níveis está relacionada com a diferença entre os valores das soluções óptimas do problema relaxado e do respectivo programa de dois níveis. Essa característica é também considerada na geração dos problemas teste.

A técnica de geração de problemas teste começa por considerar num primeiro passo programas de dois níveis de variáveis separáveis, que são gerados à custa de múltiplos programas de dois níveis com um parâmetro e duas variáveis. Ao estabelecer geometricamente as propriedades de cada programa bi-dimensional atribuem-se as características desejadas ao programa de dois níveis separável.

A fim de gerar programas de dois níveis não separáveis são introduzidas transformações lineares que conduzem a um programa de dois níveis não separável mas com as mesmas propriedades do anterior ([CaVi92] e [CaViJu92]). São discutidas várias características das transformações referidas e alguns processos de as utilizar eficientemente do ponto de vista computacional.

Neste capítulo é ainda apresentada experiência computacional da resolução destes problemas teste lineares e lineares-quadráticos com o método sequencial *LCP*. Os resultados numéricos confirmam as características deste tipo de procedimento nomeadamente, a rapidez na obtenção de uma solução ϵ -ótima (ou mesmo ótima) e a dificuldade no estabelecimento que essa solução foi encontrada.

4.1 Construção de problemas teste separáveis.

Considere-se o seguinte programa de dois níveis linear $PDNL(\rho_k)$ de duas variáveis x_k e y_k e com um parâmetro ρ_k :

$$\begin{aligned} \min_{x_k, y_k} \quad & F_k(x_k, y_k) = (3 - x_k) + y_k \\ \text{sujeito a} \quad & (x_k, y_k) \in \Omega_U^k \\ & y_k \in \operatorname{argmin}\{f_k(x_k, y_k) = -y_k : (x_k, y_k) \in \Omega_L(\rho_k)\} \end{aligned}$$

em que $\Omega_U^k = \{(x_k, y_k) \in \mathbb{R}^2 : 1 \leq x_k \leq 3\}$ abrange as restrições do primeiro nível, $\Omega_L(\rho_k) = \{(x_k, y_k) \in \mathbb{R}^2 : x_k + y_k \leq \rho_k, -2x_k + y_k \leq 0\}$ define as restrições do segundo nível e em que o parâmetro ρ_k toma valores no intervalo fechado $[3, 9]$.

O problema relaxado deste programa de dois níveis consiste em minimizar $F_k(x_k, y_k) = (3 - x_k) + y_k$ no conjunto de restrições:

$$\Omega(\rho_k) = \Omega_U^k \cup \Omega_L(\rho_k) = \{(x_k, y_k) \in \mathbb{R}^2 : 1 \leq x_k \leq 3, x_k + y_k \leq \rho_k, -2x_k + y_k \leq 0\}$$

Para uma avaliação cuidada das propriedades do problema $PDNL(\rho_k)$ é importante caracterizar a sua região induzida.

Proposição 4.1 *A região induzida, RI , do problema $PDNL(\rho_k)$ é a união dos conjuntos:*

$$\begin{aligned} S_1 &= \{(x_k, y_k) \in \Omega(\rho_k) : -2x_k + y_k = 0\} \\ S_2 &= \{(x_k, y_k) \in \Omega(\rho_k) : x_k + y_k = \rho_k\} \end{aligned}$$

Demonstração: Suponhamos que o ponto (x_k, y_k) pertence à região induzida do problema $PDNL(\rho_k)$. As condições KKT para o problema $P(x_k)$ implicam a existência de α_1 e α_2 tais que:

$$\begin{aligned} -1 &= -\alpha_1 - \alpha_2 \\ \alpha_1(\rho_k - x_k - y_k) &= 0 \\ \alpha_2(2x_k - y_k) &= 0 \\ \alpha_1, \alpha_2 &\geq 0 \end{aligned}$$

Então para $(x_k, y_k) \in \Omega(\rho_k)$, com $3 \leq \rho_k \leq 9$, existem quatro possibilidades:

- (i) (x_k, y_k) pertence ao interior de $\Omega(\rho_k)$. Neste caso $\alpha_1 = \alpha_2 = 0$ o que acarreta uma contradição.

- (ii) (x_k, y_k) satisfaz $-2x_k + y_k = 0$ com $3x_k < \rho_k$. Neste caso é descrito todo o segmento de recta S_1 , à excepção do seu ponto extremo do lado direito.
- (iii) (x_k, y_k) satisfaz $x_k + y_k = \rho_k$ com $3x_k > \rho_k$. Esta possibilidade corresponde a todos os pontos do segmento de recta S_2 , à excepção do seu ponto extremo do lado esquerdo.
- (iv) (x_k, y_k) satisfaz ambas as restrições $-2x_k + y_k = 0$ e $x_k + y_k = \rho_k$. Este caso refere-se ao ponto $(\frac{\rho_k}{3}, \frac{2\rho_k}{3})$ que é simultaneamente o ponto extremo do lado direito de S_1 e o ponto extremo do lado esquerdo de S_2 . \square

Dependendo do valor do parâmetro ρ_k , a solução óptima do problema $PDNL(\rho_k)$ ocorre num determinado ponto de S_1 ou de S_2 . Mas como $x_k \leq 3$ e $y_k \geq 0$ para todos os pontos de S_1 e de S_2 , a função do primeiro nível F_k pode ser escrita do seguinte modo:

$$\begin{aligned} F_k(x_k, y_k) &= (3 - x_k) + y_k \\ &= |x_k - 3| + |y_k| \\ &= \left\| \begin{bmatrix} x_k \\ y_k \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \end{bmatrix} \right\|_1 \end{aligned}$$

e o mínimo global, (x_k^G, y_k^G) , do problema $PDNL(\rho_k)$ é o ponto de S_1 ou de S_2 que mais perto, no sentido da norma l_1 , está do ponto $(3, 0)$. De igual modo, um mínimo local para o problema $PDNL(\rho_k)$, (x_k^L, y_k^L) , é um ponto que em determinada vizinhança que o contém, está mais próximo, no sentido de l_1 , do ponto $(3, 0)$.

Seguidamente iremos analisar cinco diferentes instâncias do problema $PDNL(\rho_k)$ para cinco determinados valores do parâmetro ρ_k para as quais obteremos diferente informação sobre a respectiva localização dos mínimos locais e globais. Os cinco casos são os seguintes:

- (i) Caso 1 (Figura 4.1) em que $\rho_k = 3$.
- (ii) Caso 2 (Figura 4.2) em que $\rho_k = 7$.
- (iii) Caso 3 (Figura 4.3) em que $\rho_k = 9$.
- (iv) Caso 4 em que $3 < \rho_k < 7$.
- (v) Caso 5 em que $7 < \rho_k < 9$.

Em cada um dos casos serão invocados argumentos geométricos para justificar as propriedades indicadas.

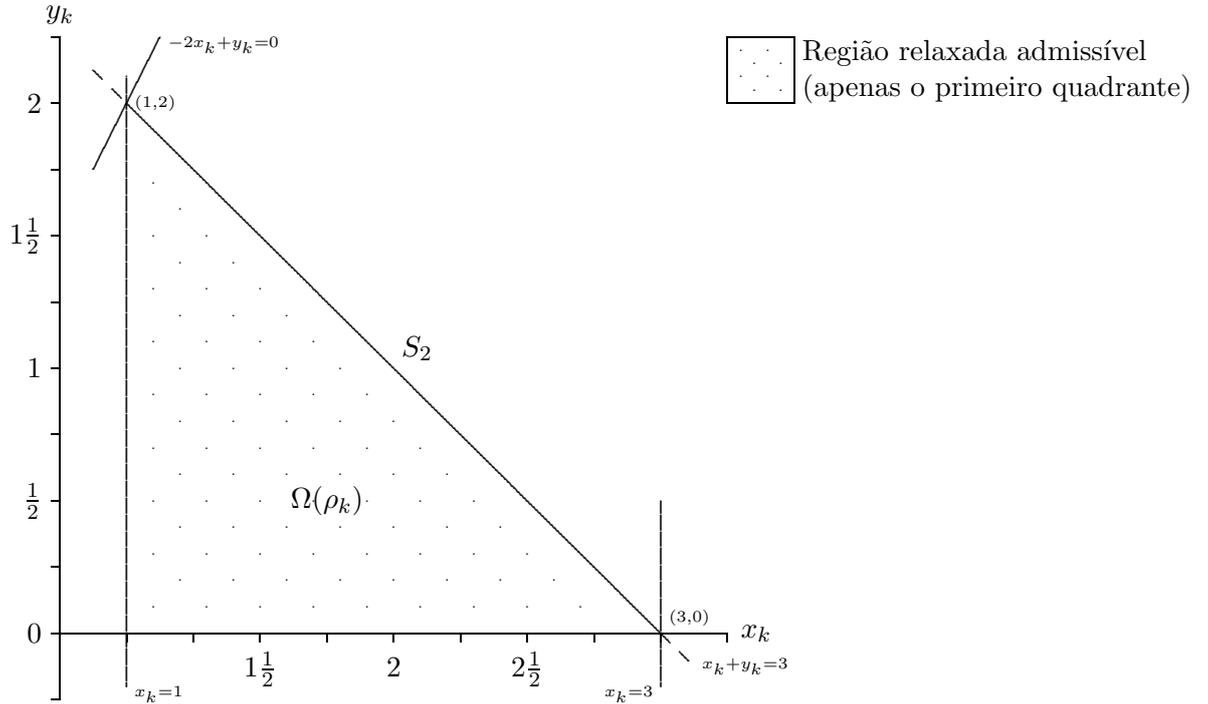


Figura 4.1: Caso 1 - em que $\rho_k = 3$.

Caso 1 ($\rho_k = 3$)

Para este caso, descrito na figura 4.1, a região induzida reduz-se ao segmento de recta S_2 e inclui o ponto $(3, 0)$. Logo existe um só mínimo local (e global), $(x_k^G, y_k^G) = (3, 0)$, com valor objectivo $F_k(x_k^G, y_k^G) = 0$.

Caso 2 ($\rho_k = 7$)

A figura 4.2 descreve este caso em que $RI = S_1 \cup S_2$. Observando as curvas de nível l_1 , centradas em $(3, 0)$, verifica-se que o problema $PDNL(\rho_k)$ possui dois mínimos globais, respectivamente $(x_k^G, y_k^G) = (1, 2)$ e $(x_k^G, y_k^G) = (3, 4)$ com $F_k(x_k^G, y_k^G) = 4$.

Caso 3 ($\rho_k = 9$)

Este caso é ilustrado na figura 4.3 e corresponde à situação em que RI coincide com S_1 . Deste modo $(1, 2)$ é o ponto de RI mais próximo, no sentido de l_1 , de $(3, 0)$. Logo existe um só mínimo global $(x_k^G, y_k^G) = (1, 2)$ com $F_k(x_k^G, y_k^G) = 4$.

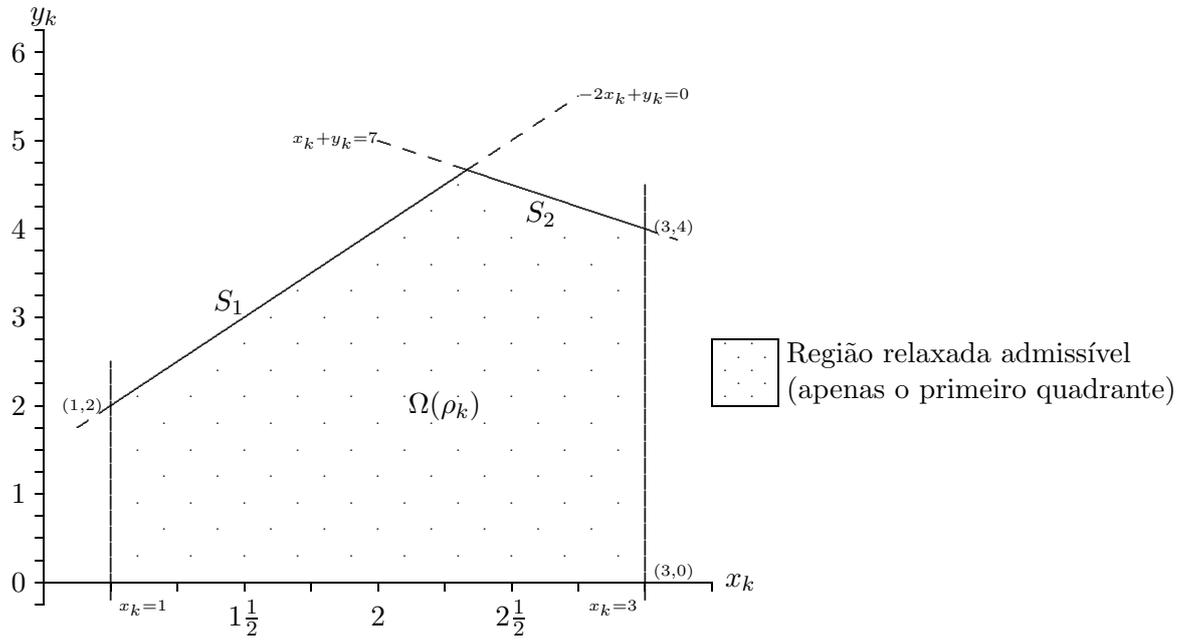


Figura 4.2: Caso 2 - em que $\rho_k = 7$.

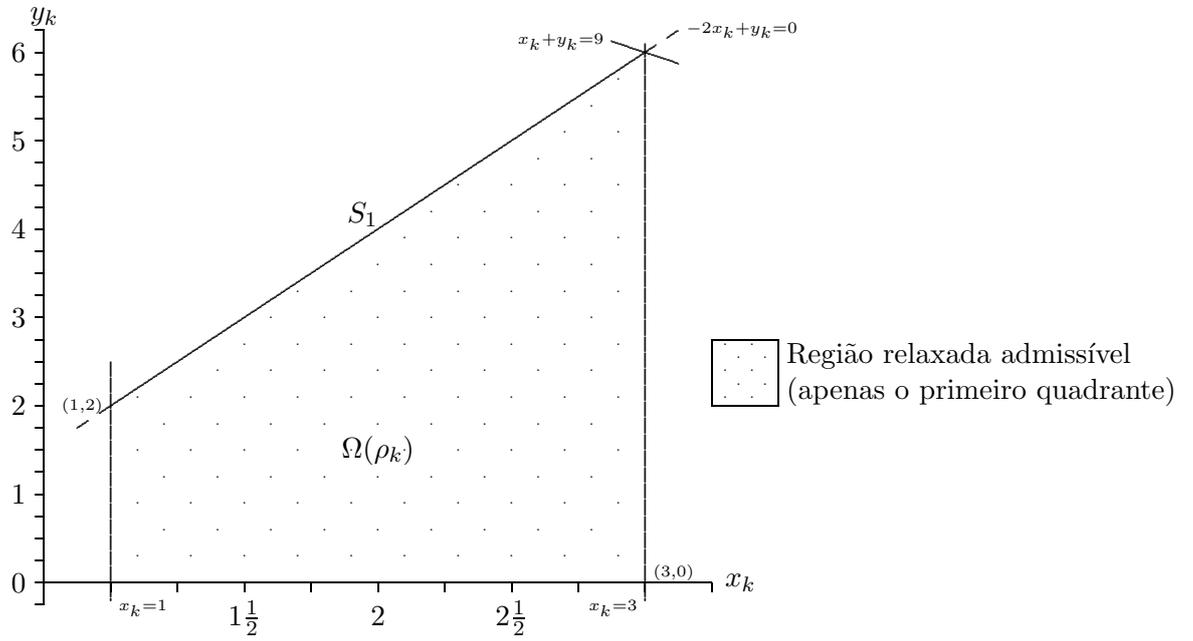


Figura 4.3: Caso 3 - em que $\rho_k = 9$.

Caso 4 ($3 < \rho_k < 7$)

Este caso é algo semelhante ao descrito na figura 4.3. A diferença está em que a linha $x_k + y_k = 7$ é substituída por $x_k + y_k = \rho_k$, ou seja, a recta $x_k + y_k = 7$ é deslocada para baixo (em translacção) $7 - \rho_k$ unidades. O ponto de *RI* mais próximo, no sentido de l_1 , de $(3,0)$, ocorre na intersecção de S_2 com a recta $x_k = 3$. Deste modo $(x_k^G, y_k^G) = (3, 3 - \rho_k)$, com $F_k(x_k^G, y_k^G) = \rho_k - 3$, é o mínimo global de $PDNL(\rho_k)$.

Além disso, $(1,2)$ é o ponto de S_1 mais próximo, no sentido de l_1 , de $(3,0)$. Logo o problema $PDNL(\rho_k)$ possui outro mínimo local (não global), $(x_k^L, y_k^L) = (1, 1)$, com $F_k(x_k^L, y_k^L) = 4$.

Caso 5 ($7 < \rho_k < 9$)

Este caso é também semelhante ao descrito na figura 4.3, pois a linha $x_k + y_k = 7$ é deslocada para cima (em translacção) $\rho_k - 7$ unidades. Neste caso o mínimo global (x_k^G, y_k^G) é o ponto $(1,2)$ com $F_k(x_k^G, y_k^G) = 4$, enquanto que o ponto $(x_k^L, y_k^L) = (3, \rho_k - 3)$ constitui um segundo mínimo local (não global) de valor $F_k(x_k^L, y_k^L) = \rho_k - 3$.

De seguida iremos explicar como um programa de dois níveis separável com múltiplos parâmetros e variáveis pode ser construído à custa de programas de dois níveis bi-dimensionais com um simples parâmetro. Para o efeito, seja $o = \min\{n, m\}$ e $\rho = (\rho_1, \dots, \rho_o)^T$ e considere-se o seguinte programa de dois níveis linear $PDNL(\rho)$:

$$\begin{aligned} \min_{x,y} F(x, y) &= \sum_{k=1}^o F_k(x_k, y_k) + \sum_{o < k \leq n} (3 - x_k) + \sum_{o < k \leq m} y_k \\ &= \sum_{k=1}^n (3 - x_k) + \sum_{k=1}^m y_k \\ \text{sujeito a } (x, y) &\in \Omega_U \\ y &\in \operatorname{argmin}\{f(x, y) = \sum_{k=1}^o f_k(x_k, y_k) = -\sum_{k=1}^o y_k : \\ (x, y) &\in \Omega_L(\rho)\} \end{aligned}$$

com $3 \leq \rho_k \leq 9$, para $k = 1, \dots, o$, e em que:

$$\Omega_U = \bigcap_{k=1}^o \Omega_U^k \bigcap_{o < k \leq n} \{x_k : x_k \leq 3\}$$

resume as restrições do primeiro nível e:

$$\Omega_L(\rho) = \bigcap_{k=1}^o \Omega_L(\rho_k) \bigcap_{o < k \leq m} \{y_k : y_k \geq 0\}$$

define as restrições do segundo nível. O problema relaxado $PR(\rho)$ associado a este problema de dois níveis linear é o seguinte programa linear:

$$\begin{aligned} \min_{x,y} \quad & F(x, y) \\ \text{sujeito a} \quad & (x, y) \in \Omega_U \cap \Omega_L(\rho) \end{aligned}$$

Através desta construção é facilmente verificável que a solução óptima (x^G, y^G) do problema multi-paramétrico $PDNL(\rho)$ é composta pelas soluções óptimas (x_k^G, y_k^G) , para $k = 1, \dots, o$ dos problemas bi-dimensionais $PDNL(\rho_k)$ e ainda pela solução óptima do programa linear:

$$\begin{aligned} \min \quad & \sum_{o < k \leq n} (3 - x_k) + \sum_{o < k \leq m} y_k \\ \text{sujeito a} \quad & x_k \leq 3, \quad o < k \leq n \\ & y_k \geq 0, \quad o < k \leq m \end{aligned}$$

Para explorar as propriedades do problema $PDNL(\rho)$ decompomos o conjunto $O = \{1, \dots, o\}$ nos seguintes subconjuntos:

$$\begin{aligned} O_1 &= \{k \in O : \rho_k = 3\} \\ O_2 &= \{k \in O : \rho_k = 7\} \\ O_3 &= \{k \in O : \rho_k = 9\} \\ O_4 &= \{k \in O : 3 < \rho_k < 7\} \\ O_5 &= \{k \in O : 7 < \rho_k < 9\} \end{aligned}$$

de cardinalidades respectivamente o_1, o_2, o_3, o_4 e o_5 . A propriedade a seguir é uma consequência imediata da forma como o programa de dois níveis $PDNL(\rho)$ foi construído através dos programas de dois níveis lineares $PDNL(\rho_k)$, $k = 1, \dots, o$.

Propriedade 4.1 *O problema $PDNL(\rho)$ tem 2^{o_2} mínimos globais, para os quais $x_k^G = 3$ para $o < k \leq n$, $y_k^G = 0$ para $o < k \leq m$ e*

$$(x_k^G, y_k^G) = \begin{cases} (3, 0) & \text{se } k \in O_1 \\ (1, 2) \text{ ou } (3, 4) & \text{se } k \in O_2 \\ (1, 2) & \text{se } k \in O_3 \cup O_5 \\ (3, \rho_k - 3) & \text{se } k \in O_4 \end{cases}$$

Cada mínimo global tem valor objectivo $F(x^G, y^G) = 4(o_2 + o_3 + o_5) + \sum_{k \in O_4} (\rho_k - 3)$.

Um dos principais atractivos do problema $PDNL(\rho_k)$ é o de poder possuir um número exponencial de mínimos locais. Esta característica torna o problema de difícil resolução computacional. A informação sobre os mínimos locais do problema $PDNL(\rho)$ é resumida na seguinte propriedade:

Propriedade 4.2 *Se $o_4 + o_5 > 0$, o problema $PDNL(\rho)$ possui mínimos locais (não globais). Os mínimos locais, em número $2^{o_2+o_4+o_5} - 2^{o_2}$, ocorrem quando todas as componentes tomam os mesmos valores que na propriedade anterior à exceção de que ou $(x_k, y_k) = (1, 2)$ para algum $k \in O_4$ ou $(x_k, y_k) = (3, \rho_k - 3)$ para algum $k \in O_5$.*

Existem técnicas de otimização global, como por exemplo o método sequencial *LCP*, que utilizam estratégias de cortes sucessivos para, de mínimo local em mínimo local atingirem a solução ótima global desejada. Para testar eficientemente estas técnicas é conveniente gerar problemas teste, não só com um número exponencial de mínimos locais, mas também de valores objectivos diferentes. Em [CaViJu92] são discutidos vários processos de gerar, sem nenhum esforço computacional acrescido, problemas teste com esta propriedade. Apesar de tais técnicas serem propostas para programação quadrática podem também ser aplicadas com sucesso na geração de programas de dois níveis.

O problema $PDNL(\rho)$ partilha ainda de outras propriedades importantes relativas ao problema relaxado (propriedade 4.3) e ao problema do segundo nível (propriedade 4.4).

Propriedade 4.3 *Os mínimos globais do problema $PDNL(\rho)$ diferem da solução ótima do problema relaxado $PR(\rho)$.*

O facto de o problema relaxado $PR(\rho)$ ser ilimitado justifica a validade da propriedade anterior que se mantém correcta, nos casos em que $o_1 < o$, se acrescentarmos as restrições de não negatividade $y_k \geq 0$, $k = 1, \dots, m$. Neste caso a solução do problema relaxado $PR(\rho)$ é $x_k = 3$, $k = 1, \dots, n$ e $y_k = 0$, $k = 1, \dots, m$, diferindo das soluções óptimas do problema $PDNL(\rho)$ em $2(o - o_1)$ variáveis.

A importância desta propriedade provém de ser usual em muitas técnicas de resolução de programas de dois níveis começar por, num passo inicial, resolver o problema relaxado. Deste modo garante-se que tais passos iniciais são insuficientes para a resolução local ou global dos problemas gerados.

Sobre o segundo nível do problema $PDNL(\rho)$ é ainda possível afirmar que:

Propriedade 4.4 *Os gradientes em y das restrições activas são linearmente independentes e as condições de complementaridade são verificadas estritamente em qualquer mínimo local do problema.*

O problema $PDNL(\rho)$ pode ser modificado de diversas formas. Por exemplo, o elevado número de restrições pode ser significativamente reduzido. Quando $k \in O_1$ as restrições:

$$x_k \geq 1 \quad \text{e} \quad -2x_k + y_k \leq 0$$

são redundantes nos problemas $PDNL(\rho_k)$ e, de modo equivalente, quando $k \in O_3$ as restrições:

$$x_k \leq 3 \quad \text{e} \quad x_k + y_k \leq 9$$

podem ser suprimidas no problema $PDNL(\rho_k)$. Deste modo, um número de $2(o_1 + o_3)$ restrições pode ser subtraído ao número total de restrições.

É possível ainda remover mais $2(o - \bar{o})$ restrições ao definir $\bar{o} < o$ e substituir as restrições:

$$\begin{aligned} -x_k &\leq -1, & k = \bar{o} + 1, \dots, o \\ x_k + y_k &\leq \rho_k, & k = \bar{o} + 1, \dots, o \\ -2x_k + y_k &\leq 0, & k = \bar{o} + 1, \dots, o \end{aligned}$$

pelo conjunto de restrições:

$$y_k \geq 0, \quad k = \bar{o} + 1, \dots, o$$

Com estas modificações há que redefinir os conjuntos de índices O_1, \dots, O_5 (e as correspondentes cardinalidades o_1, \dots, o_5) através da alteração $O = \{1, \dots, \bar{o}\}$. Além disso todos os mínimos do problema $PDNL(\rho)$ passam a satisfazer $(x_k^G, y_k^G) = (3, 0)$ para $k \in \{\bar{o} + 1, \dots, o\}$.

Outra modificação importante a ser efectuada no problema $PDNL(\rho)$ consiste em acrescentar restrições de igualdade. De facto, é possível incluir um número considerável de restrições de igualdade sem afectar o número e tipo de mínimos locais e globais do problema $PDNL(\rho)$. Em [CaVi92] é descrita uma forma de o fazer sem perturbar a independência linear dos gradientes em y de todas as restrições de segundo nível, quer de igualdade quer de desigualdade.

Para finalizar esta secção referiremos como se podem gerar problemas teste de dois níveis lineares-quadráticos utilizando toda a informação já apresentada para o caso estritamente linear. De facto, é somente necessário retirar o conjunto de restrições do segundo nível:

$$-2x_k + y_k \leq 0, \quad k \in O$$

e substituir a função objectivo do segundo nível pela seguinte função quadrática estritamente convexa em y :

$$f(x, y) = \frac{1}{2} \sum_{k=1}^o y_k(y_k - 4x_k) + \frac{1}{2} \sum_{o < i \leq m} y_i^2$$

É assim obtido um programa de dois níveis linear-quadrático (designado por $PDNLQ(\rho)$) que, pelo facto da região induzida RI permanecer igual, possui todas as propriedades do caso estritamente linear.

4.2 Problemas teste não separáveis.

Com o intuito de melhorar a notação, rescrevemos o problema $PDNL(\rho)$ na sua forma matricial $PDNL(c_1, d_1, d_2, A_2, B_2, b_2)$:

$$\begin{aligned} \min_{x,y} \quad & F(x, y) = c_1^T x + d_1^T y + \kappa \\ \text{sujeito a} \quad & y \in \operatorname{argmin}\{f(x, y) = d_2^T y : A_2 x + B_2 y \leq b_2\} \end{aligned}$$

onde por uma questão de simplificação as restrições do primeiro nível (uma vez que só figuram variáveis do primeiro nível) aparecem escritas em conjunto com as restrições do segundo nível e usamos as seguintes substituições:

$$c_1 = -1_n, \quad d_1 = 1_m, \quad (d_2)_i = \begin{cases} -1 & \text{se } i \in \{1, \dots, o\} \\ 0 & \text{caso contrário} \end{cases}, \quad \kappa = 3n$$

$$A_2 = \begin{bmatrix} P_x \\ -2P_x \\ \frac{1}{3}P_x \\ -P_x \\ R_x \end{bmatrix}, \quad B_2 = \begin{bmatrix} P_y \\ P_y \\ \mathbf{0} \\ \mathbf{0} \\ R_y \end{bmatrix} \quad \text{e} \quad b_2 = \begin{bmatrix} \rho \\ 0_o \\ 1_o \\ -1_o \\ \epsilon \end{bmatrix}$$

em que:

$\mathbf{0}$ é uma matriz de zeros com o linhas e m colunas,

0_γ e 1_γ são vectores de dimensão γ , respectivamente de zeros e de uns,

$$P_x \in \mathbb{R}^{o \times n} \text{ e } P_y \in \mathbb{R}^{o \times m} \text{ satisfazem } P_{ij} = \begin{cases} 1 & \text{se } 1 \leq i = j \leq o \\ 0 & \text{caso contrário} \end{cases}$$

e, para $\delta = \max\{n, m\} - o$,

$$R_x \in \mathbb{R}^{\delta \times n} \text{ satisfaz } R_{ij} = \begin{cases} 0 & \text{se } m > n \text{ ou } j \neq o + i \\ \frac{1}{3} & \text{caso contrário} \end{cases}$$

$$R_y \in \mathbb{R}^{\delta \times m} \text{ satisfaz } R_{ij} = \begin{cases} 0 & \text{se } n > m \text{ ou } j \neq o + i \\ -1 & \text{caso contrário} \end{cases}, \text{ e}$$

$$\epsilon \in \mathbb{R}^\delta \text{ é tal que } \epsilon_i = \begin{cases} 1 & \text{se } n > m \\ 0 & \text{se } m > n \end{cases}$$

Através de substituições semelhantes é também possível rescrever o problema $PDNLQ(\rho)$ na sua versão matricial $PDNLQ(c_1, d_1, Q, S, d_2, A_2, B_2, b_2)$:

$$\begin{aligned} \min_{x,y} \quad & F(x, y) = c_1^T x + d_1^T y + \kappa \\ \text{sujeito a} \quad & y \in \operatorname{argmin}\{f(x, y) = \frac{1}{2}y^T Q y + y^T S x + d_2^T y : A_2 x + B_2 y \leq b_2\} \end{aligned}$$

em que a matriz Q é a matriz identidade de ordem m .

Considere-se agora a matriz $M = HDH$, em que D é uma matriz diagonal positiva definida e H é uma matriz diagonal por blocos:

$$\begin{bmatrix} H_x & 0 \\ 0 & H_y \end{bmatrix}$$

construída à custa de matrizes de *Householder* ortogonais H_x e H_y , definidas através de:

$$H_x = I_n - 2v_x v_x^T \quad \text{com} \quad v_x^T v_x = 1 \quad \text{e} \quad v_x \in \mathbb{R}^n \quad \text{vector esparso,}$$

$$H_y = I_m - 2v_y v_y^T \quad \text{com} \quad v_y^T v_y = 1 \quad \text{e} \quad v_y \in \mathbb{R}^m \quad \text{vector esparso.}$$

Usando agora a sua inversa $W = HD^{-1}H$, é possível obter programas de dois níveis não separáveis. Para este efeito considere-se $M_x = H_x D_x H_x$ e $M_y = H_y D_y H_y$ em que:

$$D = \begin{bmatrix} D_x & 0 \\ 0 & D_y \end{bmatrix}$$

com $D_x \in \mathbb{R}^{n \times n}$ e $D_y \in \mathbb{R}^{m \times m}$. Seja ainda $W_x = M_x^{-1} = H_x D_x^{-1} H_x$ e $W_y = M_y^{-1} = H_y D_y^{-1} H_y$.

Teorema 4.1 *Através da transformação não singular:*

$$\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix} = W \begin{bmatrix} x \\ y \end{bmatrix}$$

o problema não transformado separável $PDNL(c_1, d_1, d_2, A_2, B_2, b_2)$ nas variáveis x e y é equivalente ao problema transformado não separável $PDNL(M_x c_1, M_y d_1, M_y d_2, A_2 M_x, B_2 M_y, b_2)$ nas variáveis \bar{x} e \bar{y} .

Demonstração: Ao efectuar a mudança de variável:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} M_x & 0 \\ 0 & M_y \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$$

o problema $PDNL(c_1, d_1, d_2, A_2, B_2, b_2)$ é escrito como:

$$\begin{aligned} \min_{\bar{x}, \bar{y}} \quad & \mathcal{F}(\bar{x}, \bar{y}) = (M_x c_1)^T \bar{x} + (M_y d_1)^T \bar{y} + \kappa \\ \text{sujeito a} \quad & \bar{y} \in \operatorname{argmin}\{(M_y d_2)^T \bar{y} : \\ & (A_2 M_x) \bar{x} + (B_2 M_y) \bar{y} \leq b_2\} \end{aligned}$$

que é precisamente o problema $PDNL(M_x c_1, M_y d_1, M_y d_2, A_2 M_x, B_2 M_y, b_2)$ nas variáveis \bar{x} e \bar{y} . \square

Recorrendo à mesma transformação linear é também possível afirmar a equivalência entre os problemas não transformado separável $PDNLQ(c_1, d_1, Q, S, d_2, A_2, B_2, b_2)$ nas variáveis x e y e o problema transformado não separável $PDNLQ(M_x c_1, M_y d_1, M_y Q M_y, M_y S M_x, M_y d_2, A_2 M_x, B_2 M_y, b_2)$ nas variáveis \bar{x} e \bar{y} .

A transformação linear não singular mantém as principais características do problema não transformado. Assim, é fundamental que exista uma correspondência bijectiva entre todos os mínimos locais dos problemas não transformado e transformado, de modo a manter toda a informação já descrita sobre o número e tipo de mínimos locais e globais. Isso é mostrado para programas de dois níveis lineares-quadráticos no teorema a seguir. De igual modo se provava o mesmo resultado para programas de dois níveis lineares.

Teorema 4.2 *Seja $u = (x, y)$ um mínimo local para o problema $PDNLQ(c_1, d_1, Q, S, d_2, A_2, B_2, b_2)$, então $\bar{u} = (\bar{x}, \bar{y}) = Wu$ é também um mínimo local para o problema $PDNLQ(M_x c_1, M_y d_1, M_y Q M_y, M_y S M_x, M_y d_2, A_2 M_x, B_2 M_y, b_2)$.*

Demonstração: Começamos por provar que Wu pertence à região induzida do problema transformado. Como u é admissível para o problema não transformado tem-se:

$$\begin{aligned} Sx + Qy + B_2^T \gamma &= -d_2 \\ A_2 x + B_2 y + \alpha &= b_2 \\ \alpha^T \gamma &= 0, \quad \alpha, \gamma \geq 0 \end{aligned}$$

e substituindo x e y respectivamente por $M_x \bar{x}$ e $M_y \bar{y}$ obtém-se:

$$(SM_x) \bar{x} + (QM_y) \bar{y} + B_2^T \gamma = -d_2 \quad (4.1)$$

$$(A_2 M_x) x + (B_2 M_y) y + \alpha = b_2 \quad (4.2)$$

$$\alpha^T \gamma = 0, \quad \alpha, \gamma \geq 0 \quad (4.3)$$

Se se multiplicar ambos os membros da primeira equação (4.1) pela matriz não singular M_y , então facilmente se conclui que (\bar{x}, \bar{y}) satisfaz as condições (4.2) e (4.3) e:

$$(M_y S M_x) \bar{x} + (M_y Q M_y) \bar{y} + (B_2 M_y)^T \gamma = -M_y d_2$$

Logo (\bar{x}, \bar{y}) pertence à região induzida do problema transformado.

Como por hipótese u é mínimo local para o problema não transformado, então existe $\epsilon > 0$ tal que:

$$F(u) - F(z) < 0, \quad \text{qualquer que seja } z \in V = \{z \in \mathbb{R}^{n+m} : z \in RI, \|u - z\|_2 < \epsilon\}$$

Para provar que $\bar{u} = Wu$ é mínimo local para o problema transformado considere-se a seguinte vizinhança de \bar{u} :

$$V_T = \{\bar{z} \in \mathbb{R}^{n+m} : \bar{z} \in RI_T, \|Wu - \bar{z}\|_2 < \epsilon/\|M\|_2\}$$

em que RI_T designa a região induzida do problema transformado. Desta construção resulta que para todos os pontos $\bar{z} = Wz \in V_T$ se tem:

$$\begin{aligned} \|u - z\|_2 &= \|M(Wu - \bar{z})\|_2 \\ &\leq \|M\|_2 \|Wu - \bar{z}\|_2 \\ &< \epsilon \end{aligned}$$

o que implica que $F(u) - F(z) < 0$, ou seja que, $\mathcal{F}(Wu) - \mathcal{F}(\bar{z}) < 0$. \square

Ao raciocinar-se de forma inversa, do problema transformado para o problema não transformado, fica provada a correspondência bijectiva entre todos os mínimos locais dos problemas referidos.

A transformação proposta possui dois parâmetros que influenciam directamente a estrutura do problema transformado. Em primeiro lugar a esparsidade dos vectores v_x e v_y controla a esparsidade das matrizes M_x e M_y e consequentemente afecta a esparsidade dos vectores e matrizes envolvidos nos problemas transformados. Por outro lado, o número de condição (na norma l_2) das matrizes D_x e D_y é igual ao número de condição das matrizes M_x e M_y (visto H_x e H_y serem matrizes ortogonais) e influencia o número de condição das matrizes presentes nos problemas transformados.

A transformação linear não singular M (e em particular M_x e M_y), possui outras propriedades que a tornam particularmente atractiva do ponto de vista computacional. Assim é possível controlar a esparsidade das matrizes M_x e M_y . Com efeito verifica-se o seguinte resultado para a matrix M_x .

Teorema 4.3 *Seja η o número de elementos não nulos de v_x . A matriz M_x possui um número máximo de $\eta^2 + (n - \eta)$ elementos não nulos. Além disso, o elemento na linha i , coluna j da matriz M_x é dado por:*

$$(M_x)_{ij} = \begin{cases} -2(v_x)_i(v_y)_j[(D_x)_{ii} + (D_x)_{jj} - 2v_x^T D_x v_x] & \text{se } i \neq j \\ (D_x)_{ii} - 4(v_x)_i[(D_x)_{ii} - v_x^T D_x v_x] & \text{se } i = j \end{cases}$$

Demonstração: A primeira parte da demonstração é consequência imediata da reordenação de linhas e colunas da matrix H_x através de permutações principais (ou seja trocas simultâneas de linhas e colunas de igual índice).

O elemento $(M_x)_{ij}$ é dado por $(H_x)_{i*} D_x (H_y)_{*j}$ em que $(H_x)_{i*}$ e $(H_y)_{*j}$ representam respectivamente a linha i e a coluna j da matrix H_x . Mas:

$$(H_x)_{i*} = (e_i - 2(v_x)_i v_x)^T \quad \text{e} \quad (H_y)_{*j} = e_j - 2(v_x)_j v_x$$

com e_i e e_j colunas i e j da matriz identidade de ordem n . Logo:

$$\begin{aligned} (M_x)_{ij} &= (e_i - 2(v_x)_i v_x)^T D_x (e_j - 2(v_x)_j v_x) \\ &= e_i D_x e_j - 2(v_x)_i (v_x)_j (D_x)_{ii} - 2(v_x)_i (v_x)_j (D_x)_{jj} + 4(v_x)_i (v_x)_j v_x^T D_x v_x \\ &= \begin{cases} -2(v_x)_i (v_x)_j [(D_x)_{ii} + (D_x)_{jj} - 2v_x^T D_x v_x] & \text{se } i \neq j \\ (D_x)_{ii} - 4(v_x)_i^2 [(D_x)_{ii} - v_x^T D_x v_x] & \text{se } i = j \end{cases} \end{aligned}$$

o que demonstra a segunda parte do teorema. \square

Como corolário imediato do teorema anterior é possível concluir que:

$$(M_x)_{ij} = \begin{cases} 0 & \text{se } i \neq j \text{ e } (v_x)_i \text{ ou } (v_x)_j \text{ for igual a zero} \\ (D_x)_{ii} & \text{se } i = j \text{ e } (v_x)_i = 0 \end{cases}$$

o que facilita de modo considerável o cálculo computacional da matriz M_x .

Do mesmo modo se poderiam obter resultados semelhantes para a matriz M_y .

Em [CaViJu92] é proposta uma transformação linear não singular da forma $M = DH$ em que H é de igual modo uma matriz do tipo *Householder*. Esta transformação apresenta propriedades mais elegantes nomeadamente quando se pretende uma aproximação exacta do número de condição das matrizes das formas quadráticas.

O exemplo que vamos apresentar de seguida pretende ilustrar todo o processo de geração de problemas teste de dois níveis descrito nas secções 4.1 e 4.2. O exemplo foca o caso estritamente linear, uma vez que foi a esta classe que maior relevo foi dado.

Suponhamos que para o problema não transformado $PDNL(\rho)$ são escolhidos os seguintes valores para os parâmetros controláveis:

- número de variáveis: $n = 5$ (primeiro nível) e $m = 4$ (segundo nível),
- cardinalidades dos conjuntos O_i , $i = 1, \dots, 5$: $o_1 = 0$, $o_2 = 1$, $o_3 = 0$, $o_4 = 2$ e $o_5 = 1$,
- valores dos parâmetros ρ_i , $i = 1, \dots, o = \min\{n, m\} = 4$: $\rho_1 = 7$, $\rho_2 = 4$, $\rho_3 = 6$ e $\rho_4 = 8$.

Com estas atribuições o problema não transformado é o seguinte programa de dois níveis linear separável:

$$\min_{x,y} F(x,y) = \sum_{k=1}^4 (3 - x_k) + y_k + (3 - x_5)$$

sujeito a

$$\begin{array}{rcccccc} x_1 & & & & & \leq & 3 \\ & x_2 & & & & \leq & 3 \\ & & x_3 & & & \leq & 3 \\ & & & x_4 & & \leq & 3 \\ & & & & x_5 & \leq & 3 \\ -x_1 & & & & & \leq & -1 \\ & -x_2 & & & & \leq & -1 \\ & & -x_3 & & & \leq & -1 \\ & & & -x_4 & & \leq & -1 \end{array}$$

$$y \in \operatorname{argmin}\{f(x,y) = -\sum_{k=1}^4 y_k :$$

$$\begin{array}{rcccccccc} x_1 & & & & & + & y_1 & & & \leq & 7 \\ & x_2 & & & & & + & y_2 & & \leq & 4 \\ & & x_3 & & & & & + & y_3 & \leq & 6 \\ & & & x_4 & & & & & + & y_4 & \leq & 8 \\ -2x_1 & & & & & + & y_1 & & & \leq & 0 \\ & -2x_2 & & & & & + & y_2 & & \leq & 0 \\ & & -2x_3 & & & & & + & y_3 & \leq & 0 \\ & & & -2x_4 & & & & & + & y_4 & \leq & 0 \end{array}$$

em que $x = (x_1 \ x_2 \ x_3 \ x_4 \ x_5)^T \in \mathbb{R}^5$ e $y = (y_1 \ y_2 \ y_3 \ y_4)^T \in \mathbb{R}^4$.

Uma vez que $o_2 = 1$, então este programa de dois níveis possui os $2 = 2^{o_2}$ seguintes mínimos globais:

$$(x_1^G, y_1^G) = (1, 3, 3, 1, 3, 2, 1, 3, 2) \quad \text{e} \quad (x_2^G, y_2^G) = (3, 3, 3, 1, 3, 4, 1, 3, 2)$$

de valor objectivo $F(x_1^G, y_1^G) = F(x_2^G, y_2^G) = 4(o_2 + o_3 + o_5) + \sum_{k \in O_4} (\rho_k - 3) = 12$. O restante número de mínimos locais é $2^{o_2+o_4+o_5} - 2^{o_2} = 2^{1+2+1} - 2 = 14$.

Para obter a versão transformada do programa de dois níveis linear consideremos por exemplo os seguintes vectores e matrizes:

$$v_x^T = [-0.5 \ 0 \ 0.7 \ 0.5 \ -0.1], \quad v_y^T = [0.9 \ 0.3 \ 0.3 \ 0.1]$$

e

$$D_x = \operatorname{diag}(10 \ 20 \ 10 \ 20 \ 10), \quad D_y = \operatorname{diag}(10 \ 10 \ 20 \ 20)$$

Então as matrizes de transformação M_x e M_y são dadas por:

$$M_x = \begin{bmatrix} 12.5 & 0 & -3.5 & 2.5 & 0.5 \\ 0 & 20 & 0 & 0 & 0 \\ -3.5 & 0 & 14.9 & -3.5 & -0.7 \\ 2.5 & 0 & -3.5 & 12.5 & 0.5 \\ 0.5 & 0 & -0.7 & 0.5 & 10.1 \end{bmatrix} \quad M_y = \begin{bmatrix} 13.24 & 1.08 & -4.32 & -1.44 \\ 1.08 & 10.36 & -1.44 & -0.48 \\ -4.32 & -1.44 & 16.76 & -1.08 \\ -1.44 & -0.48 & -1.08 & 19.64 \end{bmatrix}$$

Note-se que ambos os vectores v_x e v_y têm norma l_2 unitária e que todos os elementos diagonais das matrizes D_x e D_y são positivos.

Através da mudança de variáveis $x = M_x \bar{x}$ e $y = M_y \bar{y}$ obtém-se a seguinte versão transformada e não separável do programa de dois níveis linear anterior:

$$\min_{\bar{x}, \bar{y}} \mathcal{F}(\bar{x}, \bar{y}) = \begin{array}{cccccc} 12.0\bar{x}_1 & +20.0\bar{x}_2 & +7.2\bar{x}_3 & +12.0\bar{x}_4 & +10.4\bar{x}_5 & \\ -8.56\bar{y}_1 & -9.52\bar{y}_2 & -9.92\bar{y}_3 & -16.64\bar{y}_4 & +15 & \end{array}$$

sujeito a

$$\begin{array}{cccccc} 12.5\bar{x}_1 & & -3.5\bar{x}_3 & +2.5\bar{x}_4 & +0.5\bar{x}_5 & \leq & 3 \\ & 20\bar{x}_2 & & & & \leq & 3 \\ -3.5\bar{x}_1 & & +14.9\bar{x}_3 & -3.5\bar{x}_4 & -0.7\bar{x}_5 & \leq & 3 \\ 2.5\bar{x}_1 & & -3.5\bar{x}_3 & +12.5\bar{x}_4 & +0.5\bar{x}_5 & \leq & 3 \\ 0.5\bar{x}_1 & & -0.7\bar{x}_3 & +0.5\bar{x}_4 & +10.1\bar{x}_5 & \leq & 3 \\ -12.5\bar{x}_1 & & +3.5\bar{x}_3 & -2.5\bar{x}_4 & -0.5\bar{x}_5 & \leq & -1 \\ & -20\bar{x}_2 & & & & \leq & -1 \\ 3.5\bar{x}_1 & & -14.9\bar{x}_3 & +3.5\bar{x}_4 & +0.7\bar{x}_5 & \leq & -1 \\ -2.5\bar{x}_1 & & +3.5\bar{x}_3 & -12.5\bar{x}_4 & -0.5\bar{x}_5 & \leq & -1 \end{array}$$

$$\bar{y} \in \operatorname{argmin}\{f(\bar{x}, \bar{y}) = 8.56\bar{y}_1 + 9.52\bar{y}_2 + 9.92\bar{y}_3 + 16.64\bar{y}_4 : \}$$

$$\begin{array}{cccccccc} 12.5\bar{x}_1 & & -3.5\bar{x}_3 & +2.5\bar{x}_4 & +0.5\bar{x}_5 & +13.24\bar{y}_1 & +1.08\bar{y}_2 & -4.32\bar{y}_3 & -1.44\bar{y}_4 & \leq & 7 \\ & 20\bar{x}_2 & & & & +1.08\bar{y}_1 & +10.36\bar{y}_2 & -1.44\bar{y}_3 & -0.48\bar{y}_4 & \leq & 4 \\ -3.5\bar{x}_1 & & +14.9\bar{x}_3 & -3.5\bar{x}_4 & -0.7\bar{x}_5 & -4.32\bar{y}_1 & -1.44\bar{y}_2 & +16.76\bar{y}_3 & -1.08\bar{y}_4 & \leq & 6 \\ 2.5\bar{x}_1 & & -3.5\bar{x}_3 & +12.5\bar{x}_4 & +0.5\bar{x}_5 & -1.44\bar{y}_1 & -0.48\bar{y}_2 & -1.08\bar{y}_3 & +19.64\bar{y}_4 & \leq & 8 \\ -25.0\bar{x}_1 & & +7.0\bar{x}_3 & -5.0\bar{x}_4 & -1.0\bar{x}_5 & +13.24\bar{y}_1 & +1.08\bar{y}_2 & -4.32\bar{y}_3 & -1.44\bar{y}_4 & \leq & 0 \\ & -40.0\bar{x}_2 & & & & +1.08\bar{y}_1 & +10.36\bar{y}_2 & -1.44\bar{y}_3 & -0.48\bar{y}_4 & \leq & 0 \\ 7.0\bar{x}_1 & & -29.8\bar{x}_3 & +7.0\bar{x}_4 & +1.4\bar{x}_5 & -4.32\bar{y}_1 & -1.44\bar{y}_2 & +16.76\bar{y}_3 & -1.08\bar{y}_4 & \leq & 0 \\ -5.0\bar{x}_1 & & +7.0\bar{x}_3 & -25.0\bar{x}_4 & -1.0\bar{x}_5 & -1.44\bar{y}_1 & -0.48\bar{y}_2 & -1.08\bar{y}_3 & +19.64\bar{y}_4 & \leq & 0 \end{array}$$

em que $\bar{x} = (\bar{x}_1 \ \bar{x}_2 \ \bar{x}_3 \ \bar{x}_4 \ \bar{x}_5)^T \in \mathbb{R}^5$ e $\bar{y} = (\bar{y}_1 \ \bar{y}_2 \ \bar{y}_3 \ \bar{y}_4)^T \in \mathbb{R}^4$.

Este programa de dois níveis possui de igual modo $2 = 2^{o_2}$ mínimos globais $(\bar{x}_1^G, \bar{y}_1^G)$ e $(\bar{x}_2^G, \bar{y}_2^G)$, em que:

$$\bar{x}_1^G = W_x \begin{bmatrix} 1 \\ 3 \\ 3 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.12 \\ 0.15 \\ 0.272 \\ 0.12 \\ 0.304 \end{bmatrix}, \quad \bar{y}_1^G = W_y \begin{bmatrix} 2 \\ 1 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 0.2414 \\ 0.1138 \\ 0.2598 \\ 0.1366 \end{bmatrix}$$

e

$$\bar{x}_2^G = W_x \begin{bmatrix} 3 \\ 3 \\ 3 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 0.295 \\ 0.15 \\ 0.307 \\ 0.095 \\ 0.299 \end{bmatrix}, \quad \bar{y}_2^G = W_y \begin{bmatrix} 4 \\ 1 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 0.409 \\ 0.103 \\ 0.303 \\ 0.151 \end{bmatrix}$$

Além disso o número de restantes mínimos locais (não globais) é também $2^{o_2+o_4+o_5} - 2^{o_2} = 2^{1+2+1} - 2 = 14$.

4.3 Experiência computacional com o método sequencial *LCP*.

Nesta seção apresentamos alguns resultados da utilização do método sequencial *LCP* na resolução de alguns problemas teste referidos neste capítulo. Apenas foram realizados testes com problemas não transformados, por serem estes os problemas em que o código do algoritmo é de mais fácil utilização. Com efeito os programas não separáveis não contêm restrições de não negatividade nos valores das variáveis, pelo que o código necessita de ser transformado para ser aplicável nesse caso.

As características dos problemas teste utilizados encontram-se descritas na tabela 4.1. De notar apenas que todos os problemas gerados tem um número exponencial de mínimos locais.

	Parâmetros o_1, o_2, o_3, o_4 e o_5					Número de mínimos locais
	o_1	o_2	o_3	o_4	o_5	
Tipo A	0	0	$\frac{n}{3}$	$\frac{n}{3}$	$\frac{n}{3}$	$2^{\frac{2n}{3}}$
Tipo B	0	0	0	$\frac{n}{3}$	$\frac{2n}{3}$	2^n
Tipo C	0	0	0	$\frac{2n}{3}$	$\frac{n}{3}$	2^n
Tipo D	0	0	0	0	n	2^n
Tipo E	0	0	0	n	0	2^n

Tabela 4.1: Características dos problemas teste resolvidos.

Os resultados para diferentes dimensões são descritos nas tabelas 4.3 e 4.4 para o caso linear e 4.5 para o caso linear-quadrático. Na tabela 4.2 são apresentados os significados de todos os símbolos apresentados nas tabelas 4.3, 4.4 e 4.5.

Os testes computacionais foram realizados na máquina SUN Spark Station SLC. Os códigos implementados para a geração dos problemas testes foram escritos em FORTRAN 77 e foi utilizado o código do método sequencial *LCP* [JuFa88] também programado na mesma linguagem.

n	n^{O} de variáveis do primeiro nível	NI	n^{O} de iterações
m	n^{O} de variáveis do segundo nível	NP	n^{O} de pivotações
l_1	n^{O} de restrições do primeiro nível	T	tempo CPU
l_2	n^{O} de restrições do segundo nível	*	parâmetro $\rho = 0.001$
U	n^{O} máximo de pivotagens para o último <i>LCP</i>		

Tabela 4.2: Legenda das tabelas 4.3, 4.4 e 4.5.

Dos resultados apresentados podemos retirar as seguintes conclusões:

1. O algoritmo sequencial *LCP* foi capaz de obter a solução óptima global em todos os casos num número relativamente pequeno de pivotações. Em alguns casos houve a necessidade de reduzir o valor do parâmetro $\rho = 0.01$ - casos assinalados nas tabelas com o símbolo *.
2. O algoritmo sequencial *LCP* não foi capaz de mostrar em tempo útil que o último *LCP* não tem solução. Assim por exemplo para problemas do tipo A com $n = m = 6$ e $n = m = 12$, num total de respectivamente 631 e 3819 pivotações o último *LCP* requereu nada mais nada menos que 175 e 3638. Por essa razão estabelecemos um número máximo de pivotações para o último *LCP*, tendo o algoritmo atingido esse número em todos os casos.
3. A técnica START mostrou um comportamento irregular. A incorporação deste processo inicial para começar o método sequencial *LCP* nem sempre deu melhores resultados, como se pode ver nas tabelas 4.3 e 4.4 para o tipo de problemas A, B e D.
4. O número de iterações e pivotações para igual número de variáveis do primeiro nível manteve-se constante. De facto, ao aumentar o número de variáveis do segundo nível o valor do parâmetro $o = \min\{n, m\}$ não é incrementado o que faz com que não haja dificuldades acrescidas para a resolução dos problemas.
5. Os programas lineares-quadráticos foram mais fáceis de resolver que os lineares. Julgamos que este facto se prende com o menor número de restrições de segundo nível existentes nos programas de dois níveis (gerados com a técnica apresentada) lineares-quadráticos em relação aos estritamente lineares.

		Sequential LCP			
		$n = 30, l_1 = 30, l_2 = 60, U = 200$		$n = 60, l_1 = 60, l_2 = 120, U = 500$	
		$m = 30$	$m = 45$	$m = 60$	$m = 90$
Tipo A	<i>NI</i>	12	12	22*	22*
	<i>NP</i>	743	743	2382	2382
	<i>T</i>	17.22	18.98	104.95	116.82
Tipo B	<i>NI</i>	12	12	22	22
	<i>NP</i>	1017	1017	3210	3210
	<i>T</i>	24.09	26.35	140.94	156.33
Tipo C	<i>NI</i>	22	22	43	43
	<i>NP</i>	1381	1381	4786	4786
	<i>T</i>	32.38	35.81	210.20	232.68
Tipo D	<i>NI</i>	2	2	2	2
	<i>NP</i>	292	292	683	683
	<i>T</i>	6.21	7.31	30.10	33.28
Tipo E	<i>NI</i>	32	32	62	62
	<i>NP</i>	1313	1313	4857	4857
	<i>T</i>	26.74	29.85	192.89	213.18

Tabela 4.3: Resultados para o caso estritamente linear (sem START).

		$n = 30, l_1 = 30, l_2 = 60, U = 200$				$n = 60, l_1 = 60, l_2 = 120, U = 500$			
		$m = 30$		$m = 45$		$m = 60$		$m = 90$	
		START	SLCP	START	SLCP	START	SLCP	START	SLCP
Tipo A	<i>NI</i>	3	21	3	21	3	44*	3	44*
	<i>NP</i>	32	1114	32	1114	62	4970	62	4970
	<i>T</i>	0.01	24.45	0.01	27.13	0.03	202.47	0.03	224.64
Tipo B	<i>NI</i>	3	22*	3	22*	3	32	3	32
	<i>NP</i>	32	1294	32	1294	62	4250	62	4250
	<i>T</i>	0.01	28.43	0.01	30.90	0.03	205.40	0.03	228.04
Tipo C	<i>NI</i>	3	12	3	12	3	22	3	22
	<i>NP</i>	32	978	32	978	62	3454	62	3454
	<i>T</i>	0.02	21.23	0.02	23.85	0.03	141.80	0.03	156.65
Tipo D	<i>NI</i>	3	32*	3	32*	3	62*	3	62*
	<i>NP</i>	32	1281	32	1281	62	5347	62	5347
	<i>T</i>	0.02	27.06	0.02	29.51	0.03	209.36	0.03	231.53
Tipo E	<i>NI</i>	3	2	3	2	3	2	3	2
	<i>NP</i>	32	262	32	262	62	623	62	623
	<i>T</i>	0.01	6.44	0.02	6.95	0.03	30.99	0.03	34.11

Tabela 4.4: Resultados para o caso estritamente linear (com START).

		Sequencial LCP			
		$n = 30, l_1 = l_2 = 30, U = 100$		$n = 60, l_1 = l_2 = 60, U = 150$	
		$m = 30$	$m = 45$	$m = 60$	$m = 90$
Tipo A	<i>NI</i>	12	12	22	22
	<i>NP</i>	212	212	371	371
	<i>T</i>	2.92	3.14	12.96	14.46
Tipo B	<i>NI</i>	12	12	22	22
	<i>NP</i>	202	202	352	352
	<i>T</i>	4.02	4.61	19.31	21.59
Tipo C	<i>NI</i>	22	22	42	42
	<i>NP</i>	244	244	434	434
	<i>T</i>	3.58	4.21	14.36	16.59
Tipo D	<i>NI</i>	2	2	2	2
	<i>NP</i>	161	161	271	271
	<i>T</i>	5.14	5.74	22.92	25.91
Tipo E	<i>NI</i>	32	32	62	62
	<i>NP</i>	284	284	512	512
	<i>T</i>	4.65	5.05	20.37	23.19

Tabela 4.5: Resultados para o caso linear-quadrático.

Capítulo 5

Algoritmos de descida para programação de dois níveis quadrática.

O facto dos programas de dois níveis lineares e lineares-quadráticos possuírem funções objectivo lineares no primeiro nível torna mais fácil o desenvolvimento de algoritmos capazes de obter mínimos globais desses programas. Se a função objectivo do primeiro nível é não linear, então os problemas apresentam um maior grau de dificuldade e somente técnicas enumerativas foram até agora desenvolvidas para a obtenção de mínimos globais. Assim Bard [Ba88], Edmunds e Bard [EdBa91] e Jaumard, Savard e Xiong [JaSaXi92] resolveram programas de dois níveis quadráticos com funções objectivo do primeiro nível convexas através de procedimentos do tipo *Branch and Bound* e Al-Khayyal, Horst e Pardalos [AlHoPa92] adaptaram um processo enumerativo para a resolução de *PDNQ*'s com funções do primeiro nível côncavas.

Bi, Calamai e Conn [BiCaCo91] generalizaram a técnica de penalidade exacta referida em 3.4, do caso linear para o não linear. Outra técnica de penalidade foi desenvolvida por Aiyoshi e Shimizu [AiSh81] em que o problema do segundo nível é substituído por um equivalente problema penalizado. Estes autores sugerem em [AiSh84] uma forma diferente de resolver o problema resultante, por intermédio da substituição do problema do segundo nível penalizado pelas suas condições necessárias de optimalidade. Em [IsAi92] é proposto um algoritmo que minimiza um programa de dois níveis transformado, obtido do inicial através de uma aproximação de ambas as funções objectivo por funções la-grangeanas aumentadas apropriadas.

Todas as técnicas de penalidade referidas garantem somente a obtenção de um mínimo local. Métodos descendentes foram ainda desenvolvidos por Florian e Chen [FlCh91] e por Kolstad e Lasdon [KoLa90] para o mesmo fim.

Neste capítulo propomos dois diferentes algoritmos de descida para a resolução de programas de dois níveis quadráticos [ViSaJu92]. O primeiro algoritmo, descrito na secção 5.2, é baseado em operações pivotais modificadas que obrigam a deslocamentos directos ao longo da região induzida. Este algoritmo não converge em geral para um mínimo local à excepção do caso em que a função objectivo do primeiro nível é côncava. O segundo algoritmo é abordado na secção seguinte e consiste de uma modificação do algoritmo da descida máxima de Gauvin e Savard [GaSa91], através do uso do método sequencial *LCP* para o cálculo da direcção de descida e de técnicas exactas para a determinação do comprimento dos passos. É ainda proposto um algoritmo híbrido que engloba os dois algoritmos referidos.

O facto de ser difícil ao método sequencial *LCP* provar a optimalidade local aquando da sua incorporação no método da descida máxima, motivou o estudo da determinação da complexidade de tal tarefa. Assim, demonstramos na última secção deste capítulo que provar que um dado ponto é mínimo local para um programa de dois níveis é um problema NP-Difícil [ViSaJu92].

5.1 Definições e propriedades de um programa de dois níveis quadrático.

Ao longo deste capítulo trabalhamos com o seguinte programa de dois níveis quadrático (*PDNQ*):

$$\begin{aligned} \min_{x,y} \quad & F(x,y) = \frac{1}{2} \begin{bmatrix} x \\ y \end{bmatrix}^T \begin{bmatrix} C_1 & C_3 \\ C_3^T & C_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} \\ \text{sujeito a} \quad & x \geq 0 \\ & y \in \operatorname{argmin}\{f(x,y) = \frac{1}{2}y^T Q y + y^T S x + d_2^T y : \\ & A_2 x + B_2 y \leq b_2, \quad y \geq 0\} \end{aligned}$$

em que $c_1 \in \mathbb{R}^n$, $c_2, d_2 \in \mathbb{R}^m$, $C_1 \in \mathbb{R}^{n \times n}$, $Q, C_2 \in \mathbb{R}^{m \times m}$, $S, C_3^T \in \mathbb{R}^{m \times n}$, $A_2 \in \mathbb{R}^{l_2 \times n}$, $B_2 \in \mathbb{R}^{l_2 \times m}$ e $b_2 \in \mathbb{R}^{l_2}$.

Assume-se que a matriz $C = \begin{bmatrix} C_1 & C_3 \\ C_3^T & C_2 \end{bmatrix}$ e a matriz Q são respectivamente matrizes simétricas positiva semi-definida e positiva definida. Estas hipóteses garantem, que não só o problema relaxado *PR* nas variáveis x e y :

$$\begin{aligned} \min_{x,y} \quad & \frac{1}{2} \begin{bmatrix} x \\ y \end{bmatrix}^T \begin{bmatrix} C_1 & C_3 \\ C_3^T & C_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} \\ \text{sujeito a} \quad & A_2 x + B_2 y \leq b_2 \\ & x, y \geq 0 \end{aligned}$$

mas também o problema do segundo nível $P(x)$ em y :

$$\begin{aligned} \min_y \quad & \frac{1}{2}y^T Qy + (Sx + d_2)^T y \\ \text{sujeito a} \quad & B_2y \leq b_2 - A_2x, \quad y \geq 0 \end{aligned}$$

são programas quadráticos convexos. Note-se no entanto que só este último é estritamente convexo.

Para assegurar a existência de solução para o problema $PDNQ$ assume-se que, para $x \geq 0$, o conjunto $C(x) = \{y \in \mathbb{R}^m : B_2y \leq b_2 - A_2x, y \geq 0\}$ é não vazio. Com efeito, através desta ressalva todas as hipóteses do teorema 2.2 são válidas e o problema $PDNQ$ tem solução óptima. Além disso, o facto de $C(x) \neq \emptyset$ também assegura soluções únicas para o problema do segundo nível $P(x)$, qualquer que seja x de componentes não negativas.

De modo semelhante ao que sucede nas classes estritamente linear e linear-quadrática, a região induzida RI do problema $PDNQ$ é definida pelas seguintes condições de complementaridade:

$$Qy + Sx + d_2 + B_2^T \gamma - \beta = 0 \quad (5.1)$$

$$A_2x + B_2y + \alpha = b_2 \quad (5.2)$$

$$x, y, \alpha, \beta, \gamma \geq 0 \quad (5.3)$$

$$\alpha^T \gamma = y^T \beta = 0 \quad (5.4)$$

em que $\alpha, \gamma \in \mathbb{R}^l$ e $\beta \in \mathbb{R}^m$. Com efeito, o problema $P(x)$ ser convexo em y permite a sua substituição pelas respectivas condições KKT, necessárias e suficientes para caracterizar a optimalidade.

A definição seguinte introduz a noção de *ponto extremo* da região induzida.

Definição 5.1 $u = (x, y)$ diz-se um *ponto extremo da região induzida (ERI)* se existirem α, β e γ tais que $(x, y, \alpha, \beta, \gamma)$ é ponto extremo do conjunto poliédrico definido pelas condições (5.1)-(5.3) e satisfaz as condições de complementaridade (5.4).

À semelhança com o que sucede com a programação linear, um ponto *ERI* diz-se não degenerado se os valores das variáveis básicas forem todos positivos e degenerado caso exista alguma variável básica nula.

Dois pontos *ERI* dizem-se adjacentes se as suas bases diferirem em somente uma coluna.

Do decorrido é possível afirmar que um deslocamento entre dois pontos *ERI* pode ser realizado por intermédio de uma simples operação pivotal que mantenha válida as condições de complementaridade. Para isso basta impedir que duas variáveis complementares do mesmo par possam ser simultaneamente básicas.

A definição 5.2 estabelece uma classe de direcções que estão na base do algoritmo de descida a ser estudado na secção seguinte.

Definição 5.2 *A direcção d diz-se uma direcção de pontos extremos da região induzida (ERI) se liga dois pontos ERI.*

O teorema seguinte assegura que se um ponto *ERI* não for mínimo local, então de entre todas as direcções de descida que emanam desse ponto encontra-se pelo menos uma direcção *ERI*. Este facto motiva o desenvolvimento de um método de resolução para o problema *PDNQ* baseado em direcções da classe *ERI*.

Para simplificar as notações, a todas as direcções que ligam pontos da região induzida chamamos direcções da região induzida, em abreviatura direcções *RI*. Estas direcções são as direcções admissíveis do programa de dois níveis que estamos a considerar.

Teorema 5.1 *Seja u um ponto EIR. Se u não é um mínimo local do problema *PDNQ* então existe pelo menos uma direcção *ERI* de descida em u .*

Demonstração: Se u não é mínimo local então existe pelo menos uma direcção *RI* de descida em u . Pela propriedade da modularidade linear da região induzida [Ba88], esta direcção pode ser decomposta na forma:

$$d = \sum_{i=1}^p \mu_i d_i, \quad \text{em que} \quad \sum_{i=1}^p \mu_i = 1, \quad \mu_i > 0$$

e d_i são direcções *ERI*, $i = 1, \dots, p$.

Suponhamos por absurdo que todas as direcções d_i , $i = 1, \dots, p$, satisfazem:

$$\nabla F(u)^T d_i \geq 0$$

Então:

$$\sum_{i=1}^p \mu_i \nabla F(u)^T d_i \geq 0$$

Mas, através da combinação linear convexa da direcção d é possível escrever:

$$\nabla F(u)^T d \geq 0$$

o que contradiz o facto de d ser uma direcção de descida. Deste modo prova-se que existe pelo menos uma direcção *ERI* (entre as direcções d_i , $i = 1, \dots, p$) descendente. \square

5.2 Algoritmo de descida em pontos extremos da região induzida.

Nesta secção introduzimos um novo algoritmo de descida em pontos extremos da região induzida. Este algoritmo, que por uma questão de abreviatura designaremos por *ADERI*, baseia-se nas noções apresentadas na secção anterior.

Dado um ponto *ERI* não degenerado (\bar{x}, \bar{y}) , uma das três situações ocorre:

- (i) (\bar{x}, \bar{y}) é um mínimo local para o problema *PDNQ*,
- (ii) (\bar{x}, \bar{y}) não é um mínimo local mas existe uma direcção *ERI* para a qual o outro ponto *ERI* adjacente $(\bar{\bar{x}}, \bar{\bar{y}})$, ligado ao primeiro pela direcção referida, satisfaz:

$$F(\bar{\bar{x}}, \bar{\bar{y}}) < F(\bar{x}, \bar{y}),$$

- (iii) (\bar{x}, \bar{y}) não é um mínimo local e:

$$F(\bar{\bar{x}}, \bar{\bar{y}}) \geq F(\bar{x}, \bar{y})$$

para todos os outros pontos *ERI* adjacentes $(\bar{\bar{x}}, \bar{\bar{y}})$. Este tipo de pontos são designados, seguindo a mesma terminologia utilizada em [Al87] e já referida aquando da descrição do método sequencial *LCP*, de mínimos locais em estrela da região induzida (*MLERI*).

Nesta fase de introdução do algoritmo *ADERI* é importante salientar que, se um ponto *MLERI* (\bar{x}, \bar{y}) não é um mínimo local para o problema *PDNQ*, então existe pelo menos uma direcção *ERI* em (\bar{x}, \bar{y}) . A fim de ilustrar esta situação considere-se a seguinte instância de um programa de dois níveis quadrático em \mathbb{R}^3 :

$$\begin{array}{ll} \min_{x_1, x_2, y} & \frac{1}{2}(x_1 - \frac{4}{5})^2 + \frac{1}{2}(x_2 - \frac{1}{5})^2 + \frac{1}{2}(y - 1)^2 \\ \text{sujeito a} & 0 \leq x_1, x_2 \leq 1 \\ & y \in \operatorname{argmin}\{\frac{1}{2}y^2 + y - x_1y + 2x_2y : \\ & 0 \leq y \leq 1\} \end{array}$$

A região induzida deste problema *PDNQ* com três variáveis é formada pela união dos conjuntos:

$$\begin{aligned} & \{(x_1, x_2, y) \in \mathbb{R}^3 : x_1 \leq 1, x_2 \geq 0, -x_1 + 2x_2 \leq 0, y = 1\} \\ & \{(x_1, x_2, y) \in \mathbb{R}^3 : -x_1 + 2x_2 + y = 1, 0 \leq y \leq 1\} \\ & \{(x_1, x_2, y) \in \mathbb{R}^3 : x_1 \geq 0, x_2 \leq 1, -x_1 + 2x_2 \geq 1, y = 0\} \end{aligned}$$

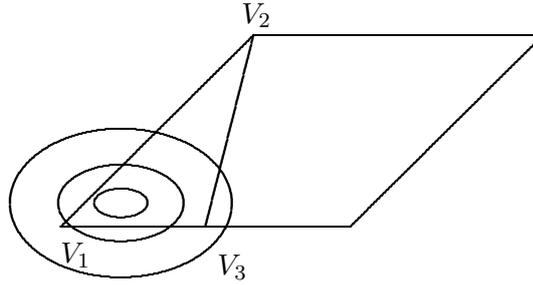


Figura 5.1: Ponto *MLERI* com duas direcções *EIR*.

Estes conjuntos correspondem a três segmentos de plano. O primeiro pertence ao plano horizontal $y = 1$ e é o triângulo de vértices $V_1 = (1, 0, 1)$, $V_2 = (0, 0, 1)$ e $V_3 = (1, 1/2, 1)$, representado na figura 5.1. O terceiro conjunto corresponde a um triângulo assente no plano $y = 0$, enquanto o segundo conjunto é um quadrilátero que une, pelo interior do conjunto CR , os dois triângulos referidos. Apesar de no ponto *EIR*, V_1 , as direcções $\overrightarrow{V_1V_2}$ e $\overrightarrow{V_1V_3}$ serem ambas direcções *EIR* de descida, os valores objectivos de F nos pontos *ERI* adjacentes V_2 e V_3 são maiores ou iguais que o valor de F em V_1 . Do decorrido, conclui-se que V_1 é um mínimo local em estrela da região induzida.

É de certo modo fácil desenvolver um algoritmo que pelo menos consegue determinar um ponto *MLERI* para o problema *PDNQ*. Numa fase inicial o algoritmo tem de achar um ponto *ERI*. Em cada iteração seguinte, ou o ponto *ERI* corrente é um mínimo local ou um ponto *MLERI* e o algoritmo termina, ou um novo ponto *ERI* adjacente é obtido com um valor objectivo para o primeiro nível inferior ao corrente. Os passos do algoritmo são descritos de seguida.

Passo inicial - Calcular um ponto *EIR* inicial, u_0 . Fazer $k = 0$.

Passo geral - Seja D_k o conjunto das direcções *EIR* em u_k :

$$D_k = \{d : \nabla F(u_k)^T d < 0, \text{ com } d \text{ direcção } EIR\}$$

(i) Se $D_k \neq \emptyset$, seleccionar d_k em D_k tal que:

$$F(u_{k+1}) < F(u_k)$$

em que u_{k+1} é o ponto *EIR* adjacente que está ligado a u_k através da direcção d_k . Fazer $k = k + 1$ e repetir este passo. Se não for possível encontrar tal direcção parar: u_k é um ponto *MLERI* para o problema *PDNQ*.

(ii) Se $D_k = \emptyset$, parar: u_k é um mínimo local para o problema $PDNQ$.

Como foi referido na secção anterior, cada iteração do passo geral do algoritmo consiste numa simples operação pivotal que mantém válidas as condições de complementaridade, o que se traduz num baixo custo computacional.

Desde que todos os pontos ERI sejam não degenerados o algoritmo termina sempre num ponto $MLERI$. No entanto, apenas na situação $D_k = \emptyset$ se pode assegurar que u_k é um mínimo local para o problema $PDNQ$. É precisamente esta desvantagem do algoritmo $ADERI$ que motiva o uso de um outro procedimento. Na próxima secção é discutida uma versão modificada do algoritmo de descida máxima para programação de dois níveis quadrática que permite ultrapassar esta desvantagem do algoritmo $ADERI$.

Uma questão importante associada ao algoritmo apresentado é o cálculo do primeiro ponto ERI . Como o problema relaxado PR é um programa quadrático convexo e o seu conjunto de restrições é não vazio, então existe sempre uma solução óptima (x_R, y_R) , que pode ser encontrada em tempo polinomial [KoMiYo91].

Se o ponto (x_R, y_R) pertencer à região induzida então é um mínimo global do problema $PDNQ$. Como já foi visto em alguns exemplos, esta situação raramente ocorre pois um programa de dois níveis é quase sempre caracterizado por uma forte conflitualidade entre os objectivos associados aos dois níveis de resolução. No entanto, um primeiro ponto da região induzida pode ser encontrado ao fixar $x = x_R$ e resolver o programa quadrático $P(x_R)$. Pelo facto de Q ser uma matriz positiva definida e de $C(x_R) \neq \emptyset$, este programa tem uma solução única, $y(x_R)$, que pode ser encontrada também em tempo polinomial.

Deste modo é possível determinar em tempo polinomial um ponto da região induzida. No entanto, tal solução não é em geral um ponto ERI , uma vez que pode não corresponder a uma solução básica do sistema de equações lineares definido por (5.1), (5.2) e (5.3).

Em [Mu83] é descrito um algoritmo para gerar soluções básicas admissíveis a partir de pontos admissíveis. Este algoritmo tem convergência polinomial [Me91] e consiste apenas em reduzir iterativamente o número de variáveis positivas. Uma vez que as restrições de complementaridade nunca são violadas, apenas pontos da região induzida são visitados e o processo termina num ponto ERI .

Como conclusão final é possível afirmar que um ponto inicial ERI pode ser obtido de forma expedita e em tempo polinomial.

5.3 Algoritmo de descida máxima modificado.

Nesta secção discutimos o uso do método da descida máxima de Gauvin e Savard [GaSa91] para a resolução do problema $PDNQ$. Este método já foi parcialmente descrito na secção

2.3 - teorema 2.6 - para o caso não linear. Entre as hipóteses adiantadas na altura verifica-se que é apenas necessário impor ao problema $PDNLQ$ que os gradientes das restrições activas em todos os pontos usados pelo algoritmo sejam linearmente independentes.

Numa dada iteração k do algoritmo, a direcção de descida máxima $d_k = (z_k, w_k)$ no ponto $u_k = (x_k, y_k)$ é dada pela resolução do seguinte programa de dois níveis linear-quadrático $PDNLQ(x_k)$:

$$\begin{aligned} \min_{z,w} \quad & (C_1x_k + C_3y_k + c_1)^T z + (C_3^T x_k + C_2y_k + c_2)^T w \\ \text{sujeito a} \quad & -1 \leq z_i \leq 1, \quad i = 1, \dots, n \\ & w \in \operatorname{argmin}\{w^T Q w + 2w^T S z : \\ & A_2' z + B_2' w \leq 0 \\ & (-\phi_k^T A_2')^T z + (Q y_k + S x_k + d_2)^T w = 0, \quad w' \geq 0\} \end{aligned}$$

em que $z \in \mathbb{R}^n$ e $w \in \mathbb{R}^m$. As matrizes A_2' e B_2' contêm todas as linhas de A_2 e B_2 correspondentes a restrições activas em u_k . De modo equivalente, o vector w' é um subvector de w para o qual apenas são considerados índices i associados a variáveis $(y_k)_i$ iguais a zero. Finalmente, ϕ_k são os multiplicadores associados às restrições activas em u_k .

Se o valor óptimo do problema $PDNLQ(x_k)$ for maior ou igual que zero, então u_k é um mínimo local do problema $PDNLQ(x_k)$ (condição necessária e neste caso suficiente de optimalidade já referida na secção 2.3). Caso contrário, a solução óptima do problema $PDNLQ(x_k)$ é a direcção de descida máxima (z_k, w_k) e um novo ponto na região induzida é calculado utilizando a fórmula:

$$(x_{k+1}, y_{k+1}) = (x_k, y_k) + \sigma_k(z_k, w_k)$$

em que σ_k é um comprimento de passo adequado.

É importante referir que, se apenas se pretender uma direcção de descida, então não é necessário resolver o problema $PDNLQ(x_k)$ até ao fim. De facto, todos os valores compreendidos entre 0 e o valor óptimo (negativo) do problema $PDNLQ(x_k)$ correspondem a direcções de descida [GaSa91]. Esta propriedade resulta do facto da função objectivo do primeiro nível constituir o produto interno do gradiente de F em x e em y pelas variáveis z e w e deve influir na escolha do algoritmo para a resolução do problema $PDNLQ(x_k)$. Nesta secção mostramos como o método sequencial LCP se adequa a este propósito. Além disso iremos propor um processo de calcular comprimentos de passo de forma exacta.

5.3.1 Utilização do método sequencial LCP para resolver o problema $PDNLQ(x_k)$.

Como o segundo nível do problema $PDNLQ(x_k)$ é um programa quadrático convexo em w pode ser substituído pelas condições KKT e ser escrito na forma $MLCP'$ equivalente:

$$\begin{aligned}
\min \quad & (C_1x_k + C_3y_k + c_1)^T z + (C_3^T x_k + C_2y_k + c_2)^T w \\
\text{sujeito a} \quad & 2Qw + 2Sz + B_2'^T \gamma' - \beta' + (Qy_k + Sx_k + d_2)^T \xi = 0 \\
& A_2' z + B_2' w + \alpha' = 0 \\
& (-\phi_k^T A_2')^T z + (Qy_k + Sx_k + d_2)^T w = 0 \\
& \alpha'^T \gamma' = \beta'^T w' = 0 \\
& w', \alpha', \beta', \gamma' \geq 0, \quad -1 \leq z_i \leq 1, \quad i = 1, \dots, n
\end{aligned}$$

em que a dimensão dos vectores α' e γ' é igual ao número de linhas da matriz B_2' e o vector β' tem o mesmo número de componentes que o vector w' .

O método sequencial LCP procura um mínimo global deste problema resolvendo uma sequência de problemas $LCP(i)$ da forma:

$$\begin{aligned}
& (C_1x_k + C_3y_k + c_1)^T z + (C_3^T x_k + C_2y_k + c_2)^T w \leq \lambda_i \\
& 2Qw + 2Sz + B_2'^T \gamma' - \beta' + (Qy_k + Sx_k + d_2)^T \xi = 0 \\
& A_2' z + B_2' w + \alpha' = 0 \\
& (-\phi_k^T A_2')^T z + (Qy_k + Sx_k + d_2)^T w = 0 \\
& \alpha'^T \gamma' = \beta'^T w' = 0 \\
& w', \alpha', \beta', \gamma' \geq 0, \quad -1 \leq z_i \leq 1, \quad i = 1, \dots, n
\end{aligned}$$

em que, como foi visto na secção 3.2, $\{\lambda_i\}$ constitui uma sequência de valores reais decrescentes.

O método termina quando for encontrado um $LCP(j)$ sem solução. Neste caso a solução do $LCP(j - 1)$ anterior é uma solução ϵ -global para o problema $PDNLQ(x_k)$. Como foi referido na secção 3.2 e comprovado aquando dos testes computacionais realizados na secção 4.3, o método sequencial comporta-se bem para encontrar a solução ϵ -global (quase sempre global) mas encontra bastantes dificuldades para comprovar a optimalidade de tal solução. De facto, provar que um problema linear complementar não tem solução é de um grau de dificuldade muito maior que apenas encontrar uma solução para tal problema. Aliás esta questão é, na sua forma geral, um problema actual e ainda em aberto em optimização global.

No entanto, não é exigido que se encontre um mínimo global para o problema $PDNLQ(x_k)$. Com efeito qualquer solução (z, w) de um $LCP(i)$ com $\lambda_i < 0$ é uma direcção de descida. Como o método sequencial LCP resolve uma sequência de $LCP(i)$

com valores decrescentes de λ_i , então o algoritmo pode terminar logo que seja encontrada uma solução de um $LCP(j)$ tal que $\lambda_j < 0$. Deste modo é possível contornar a maior desvantagem do método sequencial LCP .

Suponhamos agora que o método de descida máxima encontra um mínimo local. Como não existem direcções de descida nesse ponto o valor óptimo do problema $PDNLQ(x_k)$ é não negativo. Neste caso, há a obrigatoriedade de o método sequencial LCP realizar o seu último passo para assegurar que de facto o mínimo local foi atingido. No entanto esta tarefa só é realizada uma única vez ao longo do algoritmo da descida máxima.

Para finalizar esta secção, concluímos que é adequado o uso do algoritmo sequencial LCP para encontrar uma direcção descendente para o método da descida máxima, apesar de ser difícil estabelecer a optimalidade local no último passo deste método. É conhecida a dificuldade de tal tarefa em programação quadrática não convexa, tendo sido já provado que verificar a optimalidade local num tal programa é um problema NP-Difícil. A discussão anterior parece indicar que a mesma propriedade também se verifica para programação de dois níveis quadrática. Na secção 5.5 provamos que verificar a optimalidade local para programação de dois níveis linear é NP-Difícil. Como um programa de dois níveis linear é um caso particular de programa de dois níveis quadrático estabelece-se também a referida propriedade para programação de dois níveis quadrática.

5.3.2 Cálculo exacto do comprimento do passo.

A propriedade de modularidade linear da região induzida obriga a que se determinem critérios exactos para o cálculo do comprimento do passo. De facto, dados um ponto $u_k = (x_k, y_k)$ e uma direcção $d_k = (z_k, w_k)$ da região induzida, é conveniente desenvolver um critério eficiente para calcular o maior comprimento de passo admissível. Para conjuntos poliédricos tal critério é facilmente estabelecido através de uma regra de quociente mínimo. No caso de uma região induzida, em que as fronteiras são definidas implicitamente, o problema de encontrar o maior comprimento de passo admissível σ_{\max} reveste uma forma mais complicada.

Seja η o número de restrições activas em $u_k + \sigma d_k$, para pequenos valores positivos de σ . Se $\eta = 0$, então não existem multiplicadores nas restrições duais das condições KKT em $u_k + \sigma d_k$, e estas restrições podem ser escritas como:

$$Q(y_k + \sigma w_k) + S(x_k + \sigma z_k) + d_2 = 0$$

A direcção RI , $d_k = (z_k, w_k)$, deve verificar estas condições e consequentemente o comprimento de passo σ_{\max} é o maior valor de σ que verifica:

$$\begin{aligned} A_2(x_k + \sigma z_k) + B_2(y_k + \sigma w_k) &\leq b_2 \\ x_k + \sigma z_k &\geq 0, \quad y_k + \sigma w_k \geq 0 \end{aligned}$$

Logo para este caso apenas é necessário uma regra de quociente mínimo para calcular σ_{\max} .

Considere-se agora o caso $\eta > 0$ e sejam:

$$r_i^T x + s_i^T y = t_i, \quad i = 1, \dots, \eta \quad (5.5)$$

as η restrições do segundo nível activas em $u_k + \sigma d_k$. As restrições duais das condições KKT em $u_k + \sigma d_k$ apresentam a forma:

$$Q(y_k + \sigma w_k) + S(x_k + \sigma z_k) + d_2 + \delta_1 s_1 + \dots + \delta_\eta s_\eta = 0 \quad (5.6)$$

em que δ_i, i, \dots, η são os correspondentes multiplicadores não negativos. Como Q é uma matriz não singular, esta última condição implica que:

$$y_k + \sigma w_k = -Q^{-1}S(x_k + \sigma z_k) - Q^{-1}d_2 - \delta_1 Q^{-1}s_1 - \dots - \delta_\eta Q^{-1}s_\eta$$

Ao substituir-se esta expressão de $y_k + \sigma w_k$ nas restrições activas (5.5) obtém-se o seguinte sistema linear nas variáveis $\delta_i, i = 1, \dots, \eta$:

$$Z\delta = q' + \sigma q'', \quad (5.7)$$

em que:

$$Z = \begin{bmatrix} -s_1^T Q^{-1} s_1 & \dots & -s_1^T Q^{-1} s_\eta \\ \vdots & \vdots & \vdots \\ -s_\eta^T Q^{-1} s_1 & \dots & -s_\eta^T Q^{-1} s_\eta \end{bmatrix}, \quad \delta = \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_\eta \end{bmatrix}$$

$$q' = \begin{bmatrix} t_1 - r_1^T x_k + s_1^T Q^{-1} S x_k + s_1^T Q^{-1} d_2 \\ \vdots \\ t_\eta - r_\eta^T x_k + s_\eta^T Q^{-1} S x_k + s_\eta^T Q^{-1} d_2 \end{bmatrix} \quad \text{e} \quad q'' = \begin{bmatrix} -r_1^T z_k + s_1 Q^{-1} S z_k \\ \vdots \\ -r_\eta^T z_k + s_\eta Q^{-1} S z_k \end{bmatrix}$$

Através da resolução dos dois sistemas lineares $\eta \times \eta$:

$$Z\nu' = q' \quad \text{e} \quad Z\nu'' = q''$$

é possível obter uma relação linear entre todos os η multiplicadores e o parâmetro σ :

$$\delta_i = \nu'_i + \nu''_i \sigma, \quad i = 1, \dots, \eta$$

Antes de referir como calcular o valor exacto de σ_{\max} é importante medir o esforço computacional necessário para calcular os vectores ν' e ν'' . O teorema seguinte formaliza esta contagem.

Teorema 5.2 *O número total de sistemas necessário para calcular o maior comprimento de passo admissível σ_{\max} é $\eta+2$ (η sistemas com a matriz Q e dois sistemas com a matriz $Z = [z_{ij}]_{\eta \times \eta}$, para a qual $z_{ij} = -s_i Q^{-1} s_j$).*

Demonstração: De facto, após a resolução dos η sistemas $Q\xi = s_i$, $i = 1, \dots, \eta$ apenas são precisos produtos internos para preparar todos os dados necessários à resolução do sistema (5.7).

Além disso é ainda necessário resolver dois sistemas com a matriz Z para calcular os vectores ν' e ν'' . \square

Dado que a matriz Q é simétrica positiva definida, a sua factorização de *Cholesky* $Q = LL^T$ pode ser determinada, em que L é uma matriz triangular com elementos diagonais positivos. Note-se que esta factorização só precisa ser calculada uma vez durante todo o método da descida máxima. Deste modo, em cada iteração só é necessário resolver 2η sistemas triangulares com a matriz L e dois sistemas com a matriz Z .

Depois de calculados os vectores ν' e ν'' , o maior comprimento de passo admissível σ_{\max} é o maior valor de σ que verifica:

$$\begin{aligned} A_2(x_k + \sigma z_k) + B_2(y_k + \sigma w_k) &\leq b_2 \\ x_k + \sigma z_k &\geq 0, \quad y_k + \sigma w_k \geq 0 \\ \nu'_i + \nu''_i \sigma &\geq 0, \quad i = 1, \dots, \eta \end{aligned}$$

Assim, prova-se que também no caso $\eta > 0$ o valor de σ_{\max} é calculado através de uma regra de quociente mínimo. A diferença em relação ao caso $\eta = 0$ consiste no considerável esforço computacional que é necessário para obter as expressões em σ dos multiplicadores δ_i , $i = 1, \dots, \eta$.

Para o cálculo do comprimento do passo σ_k , estudamos o comportamento da função:

$$G(\sigma) = \frac{1}{2}(u_k + \sigma d_k)^T C(u_k + \sigma d_k) + c^T(u_k + \sigma d_k)$$

em $]0, \sigma_{\max}]$ (com $c^T = (c_1^T, c_2^T)$). Uma vez que C é uma matriz positiva semi-definida, a função G é convexa e o seu minimizante σ'_k em \mathbb{R} pode ser calculado através da expressão $G'(\sigma) = 0$. Efectuando alguns cálculos elementares concluímos que:

$$\sigma'_k = \begin{cases} -\frac{c^T d_k + d_k^T C u_k}{d_k^T C d_k} & \text{se } d_k^T C d_k > 0 \\ +\infty & \text{se } d_k^T C d_k = 0 \end{cases} \quad (5.8)$$

O comprimento de passo σ_k é então calculado através da fórmula:

$$\sigma_k = \begin{cases} \sigma'_k & \text{se } 0 < \sigma'_k < \sigma_{\max} \\ \sigma_{\max} & \text{caso contrário} \end{cases} \quad (5.9)$$

traduzindo a possível não admissibilidade do comprimento de passo σ'_k .

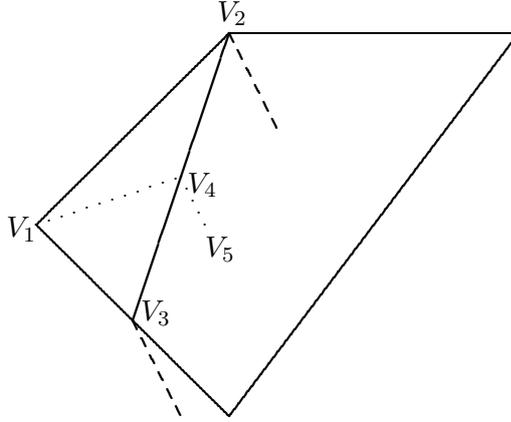


Figura 5.2: Cálculo exacto dos comprimentos de passo σ_{\max} e σ'_k .

Com intuito de exemplificar o cálculo do comprimento de passo exacto σ_k , consideramos o seguinte programa de dois níveis quadrático com três variáveis:

$$\begin{aligned} \min_{x_1, x_2, y} \quad & \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - \frac{2}{5})^2 + \frac{1}{2}(y - \frac{4}{5})^2 \\ \text{sujeito a} \quad & 0 \leq x_1, x_2 \leq 1 \\ & y \in \operatorname{argmin}\{\frac{1}{2}y^2 + y - x_1y + 3x_2y : \\ & 0 \leq y \leq 1\} \end{aligned}$$

para o qual a região induzida é formada pela união dos conjuntos:

$$\begin{aligned} & \{(x_1, x_2, y) \in \mathbb{R}^3 : x_1 \leq 1, x_2 \geq 0, -x_1 + 3x_2 \leq 0, y = 1\} \\ & \{(x_1, x_2, y) \in \mathbb{R}^3 : -x_1 + 3x_2 + y = 1, 0 \leq y \leq 1\} \\ & \{(x_1, x_2, y) \in \mathbb{R}^3 : x_1 \geq 0, x_2 \leq 1, -x_1 + 3x_2 \geq 1, y = 0\} \end{aligned}$$

À semelhança da figura 5.1, a figura 5.2 descreve o primeiro destes conjuntos - o triângulo de vértices $V_1 = (1, 0, 1)$, $V_2 = (0, 0, 1)$ e $V_3 = (3/2, 1/2, 1)$. Se o ponto inicial do algoritmo de descida máxima for $u_0 \equiv V_1$ então $d_0 = (0, 1/3, 0)$ é a primeira direcção de descida máxima e um novo ponto na região induzida é calculado através de $u_1 = u_0 + \sigma_0 d_0$ com $\sigma_0 = \sigma_{\max} = 1/3$. Na segunda iteração, $d_1 = (0, 1/15, -1/5)$ e $u_2 = u_1 + \sigma_1 d_1 = (1, 2/5, 4/5) \equiv V_5$, com $\sigma_1 = \sigma'_1 = \sqrt{10}/15$. O ponto u_2 é um mínimo local e o algoritmo termina. É importante verificar que, enquanto na primeira iteração o comprimento de passo σ'_0 não é admissível e o deslocamento ao longo da direcção d_0 é realizado com o auxílio do maior comprimento de passo admissível σ_{\max} , na segunda

iteração o comprimento de passo σ'_1 já é admissível e é através dele que é feito o deslocamento ao longo de d_1 .

Verifica-se com facilidade que a determinação de σ'_k é uma tarefa menos complicada que achar o valor do maior comprimento de passo admissível σ_{\max} . Este facto motiva a apresentação de um modo alternativo do cálculo do comprimento do passo que tenta evitar a determinação do valor σ_{\max} . Este procedimento alternativo necessita de resolver um programa quadrático convexo e os seus passos são descritos de seguida:

- (i) calcular σ'_k como em (5.8) e fazer $\bar{x} = x_k + \sigma'_k z_k$.
- (ii) resolver o programa quadrático estritamente convexo $P(\bar{x})$. Seja $y(\bar{x})$ a sua solução óptima. Se $y(\bar{x}) = y_k + \sigma'_k w_k$ considere-se o novo ponto:

$$u_{k+1} = (x_{k+1}, y_{k+1}) = (\bar{x}, y(\bar{x}))$$

- (iii) caso contrário σ'_k não é um comprimento de passo admissível e o valor de σ_{\max} tem de ser calculado de modo a fazer-se:

$$u_{k+1} = u_k + \sigma_{\max} d_k$$

Como primeiro comentário, saliente-se que a noção de comprimento de passo não admissível se refere ao espaço de restrições activas em que os pontos correntes do algoritmo se encontram. De facto, se a noção de admissibilidade fosse a usual, desde que $x \geq 0$, todos os pontos ao longo das respectivas direcções seriam considerados admissíveis.

O programa quadrático $P(\bar{x})$ pode ser resolvido em tempo polinomial num número de iterações que não depende do número de restrições activas [CLMS90]. Desta forma, o procedimento apresentado pode constituir de facto uma alternativa válida, particularmente quando é elevado o número de restrições activas η . No entanto, chame-se mais uma vez a atenção para a necessidade de calcular o valor de σ_{\max} quando σ'_k não é admissível.

É ainda de acrescentar que todas as técnicas descritas nesta secção para o cálculo do comprimento do passo podem ser aplicáveis a outros métodos descendentes, desde que utilizem direcções *RI* para resolver o problema *PDNQ*.

Ambos os algoritmos *ADERI* e da descida máxima modificado são generalizáveis a programas de dois níveis quadráticos com restrições do primeiro nível, desde que estas só incorporem variáveis do primeiro nível. Aliás, ambos os exemplos introduzidos atrás e correspondentes às figuras 5.1 e 5.2, incluem restrições do primeiro nível em x .

5.4 Algoritmo híbrido.

Nas secções anteriores descrevemos dois algoritmos para a solução de um programa de dois níveis quadrático. O algoritmo de descida em pontos extremos da região induzida *ADERI* é bastante simples mas nem sempre consegue assegurar um mínimo local para o problema *PDNQ*. Por outro lado, o algoritmo de descida máxima modificado que foi apresentado é computacionalmente mais elaborado mas consegue sempre convergir para um mínimo local.

Combinando ambos os algoritmos elabora-se um método de resolução que tira partido das vantagens de ambas as metodologias. Seguidamente são apresentados os passos desta técnica híbrida.

Passo 1 - Aplicar o algoritmo *ADERI*. Parar no caso de o método terminar com um mínimo local. Caso contrário, seja u_k o ponto *MLERI* obtido no final deste algoritmo.

Passo 2 - Resolver o problema $PDNLQ(x_k)$ para encontrar uma direcção de descida d_k . Se o valor óptimo deste problema for maior ou igual que zero, parar: u_k é um mínimo local para o problema *PDNQ*. Senão, calcular σ_k como em (5.9) e actualizar u_{k+1} do seguinte modo: $u_{k+1} = u_k + \sigma_k d_k$. Fazer $k = k + 1$ e repetir o passo 2.

É ainda de considerar a possibilidade de se passar após algumas iterações do passo 2 para o passo 1. Esta poderá ser uma maneira de se caminhar para a obtenção de um melhor mínimo local para o problema *PDNQ*. Para o cálculo de um novo ponto extremo da região induzida pode ser utilizado o procedimento descrito na secção 5.2.

O caso da função F ser convexa mas não linear em geral, não acarreta alterações significativas nos algoritmos expostos. O método sequencial *LCP* continua a ser aplicável com sucesso ao cálculo da direcção de descida e apenas a determinação do comprimento de passo σ_k necessita de um processo alternativo para resolver a equação não linear $G'(\alpha) = 0$.

No final desta secção vamos estudar o comportamento do algoritmo *ADERI* e consequentemente do algoritmo híbrido para a resolução de programas de dois níveis para os quais a função do primeiro nível F é côncava, ou seja, em que a matriz C é negativa semi-definida. Neste caso, verifica-se para programação de dois níveis uma conhecida propriedade da programação côncava.

Teorema 5.3 *Se F é uma função estritamente côncava então todo o mínimo local para o problema *PDNQ* é um ponto extremo da região induzida.*

Demonstração: Em primeiro lugar provamos que todos os pontos da região induzida pertencentes a uma dada face formam um conjunto poliédrico. Suponhamos que uma dada face é constituída pelo conjunto de η restrições activas na forma (5.5). As respectivas restrições duais das condições KKT formam um conjunto poliédrico em $\mathbb{R}^{n+m+\eta}$ e a projecção deste conjunto sobre \mathbb{R}^{n+m} constitui também um conjunto poliédrico (definido no máximo por m hiperplanos). A intersecção deste conjunto convexo com a face considerada forma um subconjunto poliédrico pertencente à face da região induzida.

Seja agora u um mínimo local para o problema $PDNQ$. Do que foi provado atrás a região induzida RI do problema $PDNQ$ é a união finita de conjuntos poliédricos:

$$RI = \cup_{i=0}^K P_i$$

em que P_i é um conjunto poliédrico, $i = 1, \dots, K$. Logo, existe pelo menos um $k \in \{1, \dots, K\}$ tal que $u \in P_k$. Como u é um mínimo local em P_k a demonstração do teorema é concluída de imediato se atendermos à teoria da minimização de funções côncavas sujeitas a restrições lineares. \square

Se aplicarmos o algoritmo $ADERI$ ao problema $PDNQ$ no caso em que a função do primeiro nível é côncava e na ausência de degenerescência, obtemos sempre um mínimo local para o problema em causa. Este facto é consequência imediata dos teoremas 5.1 e 5.3 e faz com que não seja necessário aplicar o algoritmo da descida máxima. No entanto, se o último ponto ERI obtido for degenerado então não existe garantia que tal ponto seja de facto um mínimo local.

Como nota final refira-se que na presença de funções do primeiro nível côncavas mas não lineares em geral, o teorema 5.3 e as considerações anteriores mantêm-se válidos.

5.5 Complexidade da verificação da optimalidade local.

Nesta secção provamos que estabelecer se um dado ponto é um mínimo local estrito ou não é um problema NP-Difícil. Este tópico está directamente relacionado com a discussão no final da subsecção 5.3.1. De facto, a dificuldade do algoritmo sequencial LCP em comprovar a optimalidade local do último ponto gerado pelo método da descida máxima motivou o estudo da complexidade de tal tarefa. Para provar os resultados mencionados usamos as mesmas ideias descritas em Pardalos e Schnitger [PaSc88] para programação quadrática indefinida. Estes autores estabeleceram uma equivalência entre o problema da verificação da optimalidade local, estrita ou não, e o conhecido problema da tripla satisfação, em abreviatura designado por problema 3SAT. O facto deste problema 3SAT ser NP-Completo [Co71] atribui a característica NP-Difícil à programação quadrática indefinida.

Como referimos anteriormente, o livro de Garey e Johnson [GaJo79] apresenta um estudo detalhado da teoria da complexidade. Sugerimos também [Va91] para a área da complexidade que diz directamente respeito ao estudo dos problemas de optimização.

Os resultados mencionados em [PaSc88] foram estabelecidos utilizando um programa quadrático indefinido com n variáveis e tal que todos os valores próprios da forma quadrática são negativos à excepção de um que é positivo. Este programa tem as seguintes restrições:

$$\begin{aligned} A_S x &\geq \frac{3}{2} + c \\ \frac{1}{2} - x_0 &\leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\ x_i &\geq 0, \quad i = 0, \dots, n \end{aligned} \quad (5.10)$$

associadas a uma instância S do problema 3SAT.

A fim de provar os nossos resultados de complexidade, consideramos programas de dois níveis lineares cujas restrições do primeiro nível são as restrições (5.10) e mostramos que estes programas de dois níveis satisfazem propriedades semelhantes às do programa quadrático indefinido.

Teorema 5.4 *Verificar a optimalidade local estrita em programação de dois níveis linear é NP-Difícil.*

Demonstração: Considere-se o seguinte programa de dois níveis linear:

$$\begin{aligned} \min_{x,l,m,z} \quad & F(x, l, m, z) = \sum_{i=1}^n z_i \\ \text{sujeito a} \quad & A_S x \geq \frac{3}{2} + c \\ & \frac{1}{2} - x_0 \leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\ & x_i \geq 0, \quad i = 0, \dots, n \\ & l, m, z \in \operatorname{argmax} \left\{ \sum_{i=1}^n z_i : \right. \\ & \quad x_i - l_i = \frac{1}{2} - x_0 \\ & \quad x_i + m_i = \frac{1}{2} + x_0 \\ & \quad z_i \leq l_i, \quad z_i \leq m_i, \quad i = 1, \dots, n \\ & \quad \left. z \geq 0 \right\} \end{aligned}$$

Uma vez que todos os valores das variáveis z_i são não negativos tem-se:

$$F(x, l, m, z) \geq 0$$

Se se considerar o ponto x^* definido por:

$$x_0^* = 0, \quad x_i^* = \frac{1}{2}, \quad i = 1, \dots, n \quad (5.11)$$

então x^* verifica as restrições do primeiro nível. Além disso, $l^* = 0$, $m^* = 0$ e $z^* = 0$ é a solução óptima do problema do segundo nível $P(x^*)$. Logo (x^*, l^*, m^*, z^*) pertence à região induzida do $PDNL$ definido atrás. Pelo facto de $F(x^*, l^*, m^*, z^*) = 0$, conclui-se que (x^*, l^*, m^*, z^*) é mínimo global desse mesmo programa.

Tal como em [PaSc88] o teorema fica provado se estabelecermos que $F(x, l, m, z) = 0$ se e só se $x_i \in \{\frac{1}{2} - x_0, \frac{1}{2} + x_0\}$, para todo o $i = 1, \dots, n$. Se esta última condição se verificar então $l_i = 0$ ou $m_i = 0$ e $z_i = 0$ para todo o $i = 1, \dots, n$, o que implica $F(x, l, m, z) = 0$. Para mostrar a implicação inversa suponhamos que $x_i \neq \frac{1}{2} - x_0$ e $x_i \neq \frac{1}{2} + x_0$ para algum i . Uma vez que o problema do segundo nível está definido como um problema de maximização, a variável z_i tem de ser positiva. Deste modo $F(x, l, m, z) > 0$ e esta desigualdade prova por absurdo a segunda implicação. \square

Teorema 5.5 *Verificar a optimalidade local em programação de dois níveis linear é NP-Difícil.*

Demonstração: Considere-se agora um novo programa de dois níveis linear com a introdução das variáveis w :

$$\begin{aligned} \min_{x, l, m, z, w} \quad & F(x, l, m, z, w) = \sum_{i=1}^n z_i - \frac{1}{2n} \sum_{i=1}^n w_i \\ \text{sujeito a} \quad & A_S x \geq \frac{3}{2} + c \\ & \frac{1}{2} - x_0 \leq x_i \leq \frac{1}{2} + x_0, \quad i = 1, \dots, n \\ & x_i \geq 0, \quad i = 0, \dots, n \\ & l, m, z, w \in \operatorname{argmax} \left\{ \sum_{i=1}^n z_i - \sum_{i=1}^n w_i : \right. \\ & x_i - l_i = \frac{1}{2} - x_0 \\ & x_i + m_i = \frac{1}{2} + x_0 \\ & z_i \leq l_i, \quad z_i \leq m_i, \quad i = 1, \dots, n \\ & w_i \geq x_i - \frac{1}{2}, \quad w_i \geq \frac{1}{2} - x_i, \quad i = 1, \dots, n \\ & \left. z, w \geq 0 \right\} \end{aligned}$$

Seja RI a região induzida deste programa e x^* o ponto definido por (5.11). Ao resolvermos o problema do segundo nível $P(x^*)$ obtemos desta vez:

$$l_i^* = m_i^* = z_i^* = w_i^* = 0, \quad \text{para todo } i = 1, \dots, n$$

e deste modo $(x^*, l^*, m^*, z^*, w^*)$ pertence a RI . Além disso:

$$F(x^*, l^*, m^*, z^*, w^*) = 0$$

e este ponto $(x^*, l^*, m^*, z^*, w^*)$ desempenha a mesma função que o ponto x^* no teorema 2 de [PaSc88]. Deste modo provamos o teorema se demonstrarmos que:

$$F(x, l, m, z, w) > 0$$

para todo o $(x, l, m, z, w) \in RI$ satisfazendo $x_1 \geq \frac{1}{2} - \frac{x_0}{3}$ e $x_0 > 0$.

O facto do problema do segundo nível ser de maximização faz com que:

$$z_i = \min\{l_i, m_i\}, \quad i = 1, \dots, n$$

Logo:

$$\begin{aligned} \sum_{i=1}^n z_i &= \sum_{i=1}^n \min\{l_i, m_i\} \\ &= \sum_{i=1}^n \min\{x_i - \frac{1}{2} + x_0, -x_i + \frac{1}{2} + x_0\} \end{aligned}$$

Mas $x_1 \geq \frac{1}{2} - \frac{x_0}{3}$, o que implica:

$$\sum_{i=1}^n z_i \geq \min\{x_1 - \frac{1}{2} + x_0, -x_1 + \frac{1}{2} + x_0\} = \frac{2}{3}x_0 \quad (5.12)$$

Por outro lado:

$$w_i \geq x_i - \frac{1}{2}, \quad w_i \geq \frac{1}{2} - x_i, \quad i = 1, \dots, n$$

e como maximizamos a função objectivo do segundo nível vem:

$$w_i = |x_i - \frac{1}{2}|, \quad i = 1, \dots, n$$

Além disso as restrições do segundo nível forçam a que:

$$|x_i - \frac{1}{2}| \leq x_0, \quad i = 1, \dots, n$$

e daí:

$$\frac{1}{2n} \sum_{i=1}^n w_i \leq \frac{1}{2n} \sum_{i=1}^n x_0 = \frac{x_0}{2} \quad (5.13)$$

Então de (5.12) e de (5.13) podemos concluir que:

$$F(x, l, m, z, w) \geq \frac{2}{3}x_0 - \frac{x_0}{2} = \frac{x_0}{6} > 0$$

o que tal como em [PaSc88] prova o teorema desejado. \square

Conclusões finais.

Neste trabalho é de salientar uma revisão exaustiva dos principais resultados e algoritmos da programação de dois níveis, a geração de problemas teste de dois níveis e a introdução de várias abordagens descendentes para programação de dois níveis quadrática.

Julgamos terem sido contribuições importantes em áreas da programação de dois níveis ainda pouco exploradas. O conjunto de problemas teste encontra-se à disposição de qualquer utilizador e possibilita estudos computacionais comparativos entre as várias técnicas de resolução de programas de dois níveis lineares e quadráticos.

Em programação de dois níveis quadrática foi introduzida uma teoria baseada em definições e propriedades específicas desses programas. Esta foi complementada com algoritmos que determinam soluções locais e que se comportam de modos diferentes consoante o tipo de funções objectivo do primeiro nível.

Como facilmente se conclui da leitura desta tese, a programação de dois níveis é ainda uma área de potencial investigação, onde existem possíveis pistas para trabalho futuro. Nesse sentido são de realçar os seguintes pontos que já pertencem às nossas preocupações actuais:

- desenvolvimento de métodos de pontos interiores (*mpi*) para a resolução de programas de dois níveis lineares e lineares-quadráticos. Uma primeira abordagem será a tentativa de resolução de cada problema linear complementar generalizado do método sequencial *LCP* através de métodos *mpi*. Uma segunda hipótese consiste em procurar funções potenciais próprias que englobem *gaps* duais dos problemas relaxado e do segundo nível, de modo a que iterativamente o método *mpi* force o compromisso entre a minimização da função objectivo do primeiro nível e o deslocamento perto ou mesmo dentro da região induzida.
- o estudo de problemas práticos de aplicação, como por exemplo alguns problemas de localização, que pela sua estrutura de decisão hierarquizada apresentam atractivos para serem formulados com recurso à programação de dois níveis.
- a aplicação de técnicas clássicas de programação inteira (como *decomposição de*

Benders ou dualidade lagrangeana) para a resolução de programas de dois níveis lineares inteiros ou inteiros mistos.

- o desenvolvimento de técnicas para programas de dois níveis não lineares (com informação de segunda ordem não constante), usando os algoritmos da programação de dois níveis quadrática.

Bibliografia

- [AiSh81] Aiyoshi, E. e Shimizu, K., *Hierarchical decentralized systems and its new solution by a barrier method*, IEEE Transactions on Systems, Man, and Cybernetics 11 (1981) 444-449.
- [AiSh84] Aiyoshi, E. e Shimizu, K., *A solution method for the static constrained Stackelberg problem via penalty method*, IEEE Transactions on Automatic Control 29 (1984) 1111-1114.
- [Al87] Al-Khayyal, F., *An implicit enumeration procedure for the general linear complementarity problem*, Mathematical Programming Studies 31 (1987) 1-20.
- [AlHoPa92] Al-Khayyal, F., Horst, R. e Pardalos, P., *Global optimization of concave functions subject to quadratic constraints: an application in nonlinear bilevel programming*, Annals of Operations Research 34 (1992) 125-147.
- [Ba83a] Bard, J., *An efficient point algorithm for a linear two-stage optimization problem*, Operations Research 31 (1983) 670-684.
- [Ba83b] Bard, J., *An algorithm for solving the general bilevel programming*, Mathematics of Operations Research 8 (1983) 260-272.
- [Ba83c] Bard, J., *Coordination of a multidivisional organization through two levels of management*, Omega 11 (1983) 457-468.
- [Ba84a] Bard, J., *Optimality conditions for the bilevel programming problem*, Naval Research Logistics Quarterly 31 (1984) 13-26.
- [Ba84b] Bard, J., *An investigation of the linear three-level programming problem*, IEEE Transactions on Systems, Man and Cybernetics 14 (1984) 711-717.
- [Ba88] Bard, J., *Convex two-level optimization*, Mathematical Programming 40 (1988) 15-27.
- [Ba91] Bard, J., *Some properties of the bilevel programming problem*, Journal of Optimization Theory and Applications 68 (1991) 371-378.

- [BaFa82] Bard, J. e Falk, J., *An explicit solution to the multi-level programming problem*, Computers and Operations Research 9 (1982) 77-100.
- [BaMo90] Bard, J. e Moore, J., *A branch and bound algorithm for the bilevel programming problem*, SIAM Journal on Scientific and Statistical Computing 11 (1990) 281-292.
- [BaMo92] Bard, J. e Moore, J., *An algorithm for the discrete bilevel programming problem*, Naval Research Logistics 39 (1992) 419-435.
- [BaSh79] Bazaraa, M. e Shetty, C., *Nonlinear programming: theory and applications*, John Wiley, New York (1979).
- [BeBl90] Ben-Ayed, O. e Blair, C., *Computational difficulties of bilevel linear programming*, Operations Research 38 (1990) 556-559.
- [BBBL92] Ben-Ayed, O., Blair, C., Boyce, D. e LeBlanc, L., *Construction of a real-world bilevel linear programming model of the highway design problem*, Annals of Operations Research 34 (1992) 219-254.
- [Be89] Benson, H., *On the structure and properties of a linear multi level programming problem*, Journal of Optimization Theory and Applications 60 (1989) 353-373.
- [BiCaCo89] Bi, Z., Calamai, P. e Conn, A., *An exact penalty function approach for the linear bilevel programming problem*, Technical Report #167-O-310789, Department of Systems Design Engineering, University of Waterloo (1989).
- [BiCaCo91] Bi, Z., Calamai, P. e Conn, A., *An exact penalty function approach for the nonlinear bilevel programming problem*, Technical Report #180-O-170591, Department of Systems Design Engineering, University of Waterloo (1991).
- [BiKa84] Bialas, W. e Karwan, M., *Two-level linear programming*, Management Science 30 (1984) 1004-1020.
- [BiKaSh80] Bialas, W., Karwan, H. e Shaw, J., *A parametric complementary pivot approach for two-level linear programming*, Research Report No. 80-2, Operations Research Program, DIE - State University of New York at Buffalo (1980).
- [Bl92] Blair, C., *The computational complexity of multi-level linear programs*, Annals of Operations Research 34 (1992) 13-19.
- [CaVi92] Calamai, P. e Vicente, L., *Generating linear and linear-quadratic bilevel programming problems*, SIAM Journal on Scientific and Statistical Computing (a aparecer).

- [CaViJu92] Calamai, P., Vicente, L. e Júdeice, J., *A new technique for generating quadratic programming problems*, Mathematical Programming (a aparecer).
- [Ca88] Candler, W., *A linear bilevel programming algorithm: a comment*, Computers and Operations Research 15 (1988) 297-298.
- [CaTo82] Candler, W. e Townsley, R., *A linear two-level programming problem*, Computers and Operations Research 9 (1982) 59-76.
- [CLMS90] Carpenter, T., Lustig, I., Mulvey, J. e Shanno, D., *Higher order predictor-corrector interior point methods with application to quadratic objectives*, Rutcor Research Report RRR 67-90, Rutgers University (1990).
- [ChFl91] Chen, Y. e Florian, M., *The nonlinear bilevel programming problem: a general formulation and optimality conditions*, CRT-794, Centre de Recherche sur les Transports (1991).
- [ClWe88] Clarke, P. e Westerberg, A., *A note on the optimality conditions for the bilevel programming problem*, Naval Research Logistics 35 (1988) 413-418.
- [Co71] Cook, S., *The complexity of theorem proving procedures*, Proc. 3rd Ann. ACM Symp. on Theory of Computing (1971) 151-158.
- [DaFoSh67] Dantzig, G., Folkman, J. e Shapiro, N., *On the continuity of the minimum set of a continuous function*, Journal of Mathematical Analysis and Applications 17 (1967) 519-548.
- [De87] Dempe, S., *A simple algorithm for the linear bilevel programming problem*, Optimization 18 (1987) 373-385.
- [De92] Dempe, S., *A necessary and a sufficient optimality condition for bilevel programming problems*, Optimization (a aparecer).
- [DiJe79] Dirickx, Y. e Jennegren, L., *Systems analysis by multilevel methods: with applications to economics and management*, John Wiley, New York (1979).
- [EdBa91] Edmunds, T. e Bard, J., *Algorithms for nonlinear bilevel mathematical programs*, IEEE Transactions on Systems, Man, and Cybernetics 21 (1991) 83-89.
- [EdBa92] Edmunds, T. e Bard, J., *An algorithm for the mixed-integer nonlinear bilevel programming problem*, Annals of Operations Research 34 (1992) 149-162.
- [Fa72] Falk, J., *A linear max-min problem*, Mathematical Programming 5 (1972) 169-188.

- [Faus92] Faustino, A., *Complementaridade linear e aplicação em optimização global*, Tese de Doutouramento, Universidade de Coimbra (1992).
- [FlCh91] Florian, M. e Chen, Y., *A bilevel programming approach to estimating O-D matrix by traffic counts*, CRT-750, Centre de Recherche sur les Transports (1991).
- [FoMc81] Fortuny-Amat, J. e McCarl, B., *A representation and economic interpretation of a two-level programming problem*, Journal of the Operational Research Society 32 (1981) 783-792.
- [FTCM90] Friesz, T., Tobin, R., Cho, H. e Mehta, N., *Sensitivity analysis based heuristic algorithms for mathematical programs with variational inequality constraints*, Mathematical Programming 48 (1990) 265-284.
- [GaUl77] Gallo, G. e Úlkúćü, A., *Bilinear programming: an exact algorithm*, Mathematical Programming 12 (1977) 173-194.
- [GaJo79] Garey, M. e Johnson, D., *Computers and intractability: a guide to the theory of NP-Completeness*, W.H.Freeman and Company (1979).
- [GaSa91] Gauvin, J. e Savard, G., *The steepest descent direction for the nonlinear bilevel programming problem*, École Polytechnique de Montréal, Montréal (1989).
- [HaJaLu90] Hansen, P., Jaumard, B. e Lu, S., *An analytical approach to global optimization*, Mathematical Programming 52 (1991) 227-254.
- [HaJaSa92] Hansen, P., Jaumard, B. e Savard, G., *New branching and bounding rules for linear bilevel programming*, SIAM Journal on Statistical and Scientific Computing (a aparecer).
- [HaPa88] Harker, P. e Pang, J., *Existence of optimal solutions to mathematical programs with equilibrium constraints*, Operations Research Letters 7 (1988) 61-64.
- [HaLoSa89] Haurie, A., Loulou, R. e Savard, G., *A two-level systems analysis model of power cogeneration under asymmetric pricing*, G-89-34, Groupe d'études et de recherche en analyse des décisions (1989).
- [HaSaWh90] Haurie, A., Savard, G. e White, D., *A note on: an efficient point algorithm for a linear two-stage optimization problem*, Operations Research 38 (1990) 553-555.
- [HoNe92] Hobbs, B. e Nelson, S., *A nonlinear bilevel model analysis of electric utility demand-side planning issues*, Annals of Operations Research 34 (1992) 255-274.
- [Ho73] Hogan, W., *Point-to-set maps in mathematical programming*, SIAM Review 15 (1973) 591-603.

- [IsAi92] Ishizuka, Y. e Aiyoshi, E., *Double penalty method for bilevel linear programming*, Annals of Operations Research 34 (1992) 73-88.
- [JaSaXi92] Jaumard, B, Savard, G. e Xiong, J., *An exact algorithm for convex bilevel programming*, Optimization Days, Canada (1992).
- [Je85] Jeroslow, R., *The polynomial hierarchy and a simple model for competitive analysis*, Mathematical Programming 32 (1985) 146-164.
- [JuFa88] Júdice, J. e Faustino, A., *The solution of the linear bilevel programming problem by using the linear complementarity problem*, Investigaç o Operacional 8 (1988) 77-95.
- [JuFa92a] Júdice, J. e Faustino, A., *A sequential LCP method for bilevel linear programming*, Annals of Operations Research 34 (1992) 89-106.
- [JuFa92b] Júdice, J. e Faustino, A., *The linear-quadratic bilevel programming problem*, Infor (a aparecer).
- [KoMiYo91] Kojima, M., Mizuno, S. e Yoshise, A., *A polynomial-time algorithm for a class of linear complementarity problems*, Mathematical Programming 50 (1991) 331-342.
- [Ko85] Kolstad, C., *A review of the literature on bi-level mathematical programming*, Technical Report No. LA-10284-MS, UC-32, Los Alamos National Laboratory, New Mexico (1985).
- [KoLa90] Kolstad, C. e Lasdon, L., *Derivative evaluation and computational experience with large bilevel mathematical programs*, Journal of Optimization Theory and Applications 65 (1990) 485-499.
- [Ko76a] Konno, H., *A cutting plane algorithm for bilinear programs*, Mathematical Programming 11 (1976) 14-27.
- [Ko76b] Konno, H., *Maximization of a convex quadratic function under linear constraints*, Mathematical Programming 11 (1976) 117-127.
- [Ma86] Marcotte, P., *Network design problem with congestion effects: a case of bilevel programming*, Mathematical Programming 34 (1986) 142-162.
- [Ma88] Marcotte, P., *A note on a bilevel programming algorithm by LeBlanc and Boyce*, Transportation Research 22 B (1988) 233-237.

- [MaMa92] Marcotte, P. e Marquis, G., *Efficient implementation of heuristics for the continuous network design problem*, Annals of Operations Research 34 (1992) 163-176.
- [MaSa91a] Marcotte, P. e Savard, G., *A note on the pareto optimality of solutions to the linear bilevel programming problem*, Computers and Operations Research 18 (1991) 355-359.
- [MaSa91b] Marcotte, P. e Savard, G., *Novel approaches to the discrimination problem*, G-91-21, Groupe d'études et de recherche en analyse des décisions (1991).
- [Me91] Megiddo, N., *On finding primal- and dual-optimal bases*, ORSA Journal on Computing 3 (1991) 63-65.
- [MeMaTa70] Mesanovic, M., Macko, D. and Takahara, Y., *Theory of hierarchical, multi-level systems*, Academic Press, New York and London (1970).
- [MiFrTo92] Miller, T., Friesz, T. e Tobin, R., *Heuristic algorithms for delivered price spatially competitive network facility location problems*, Annals of Operations Research 34 (1992) 177-202.
- [MoBa90] Moore, J. e Bard, J., *The mixed integer linear bilevel programming problem*, Operations Research 38 (1990) 911-921.
- [Mu83] Murty, K., *Linear programming*, John Wiley and Sons (1983).
- [On92] Ónal, H., *Computational experience with a mixed solution method for bilevel linear/quadratic programs*, University of Illinois at Urbana-Champaign (1992).
- [Pa82] Papavassilopoulos, G., *Algorithms for static Stackelberg games with linear costs and polyhedra constraints*, Proceedings of the 21st IEEE Conference on Decision Control (1982).
- [PaSc88] Pardalos, P. e Schnitger, G., *Checking local optimality in constrained quadratic programming is NP-Hard*, Operations Research 7 (1988) 33-35.
- [Ra69] Raghavachari, M., *On connections between zero-one integer programming and concave programming under linear constraints*, Operations Research 17 (1969) 680-684.
- [Ro84] Rockafellar, R., *Directional differentiability of the optimal value function in a nonlinear programming problem*, Mathematical Programming Study 21 (1984) 213-226.
- [Sa89] Savard, G., *Contribution a la programmation mathématique a deux niveaux*, Tese de Doutouramento, École Polytechnique, Université de Montréal (1989).

- [Se89] Segall, R., *Bi-level geometric programming: a new optimization model*, DMS - University of Lowell Olsen Hall (1989).
- [St52] Stackelberg Von, H., *The theory of the market theory*, Oxford University Press, Oxford (1952).
- [SuKi92] Suh, S. e Kim, T., *Solving nonlinear bilevel programming models of equilibrium network design problems: a comparative review*, *Annals of Operations Research* 34 (1992) 203-218.
- [Tu90] Tuy, H., *A global optimization approach for the linear two-level program*, LiTH-MAT-R-90-17, Linköping Institute of Technology (1990).
- [Un87] Únlú, G., *A linear bilevel programming algorithm based on bicriteria programming*, *Computers and Operations Research* 14 (1987) 173-179.
- [Va91] Vavasis, S., *Nonlinear optimization, Complexity issues*, Oxford University Press, New York (1991).
- [ViSaJu92] Vicente, L., Savard, G. e Júdice, J., *Descent approaches for quadratic bilevel programming*, submetido a publicação.
- [WeBi86] Wen, U. e Bialas, W., *The hybrid algorithm for solving the three-level linear programming problem*, *Computers and Operations Research* 13 (1986) 367-377.
- [WeHs89] Wen, U. e Hsu, S., *A note on a linear bilevel programming algorithm based on bicriteria programming*, *Computers and Operations Research* 16 (1989) 79-83.
- [WhAn89] White, D. e Anandalingam, G., *A penalty function approach for solving bi-level linear programs*, Department of Systems, University of Pennsylvania (1989).
- [Wo59] Wolfe, P., *Simplex method for quadratic programming*, *Econometria* 27 (1959) 382-398.
- [Ze82] Zeleny, M., *Multiple criteria decision making*, McGraw-Hill, New York (1982).