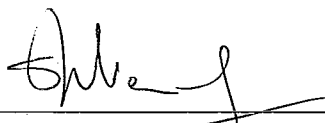


PROGRAMAÇÃO LINEAR EM DOIS NÍVEIS: UMA ABORDAGEM TEÓRICA E COMPUTACIONAL

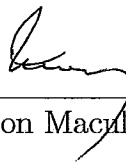
Manoel Bezerra Campêlo Neto

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

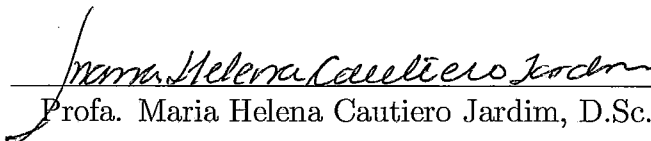
Aprovada por:



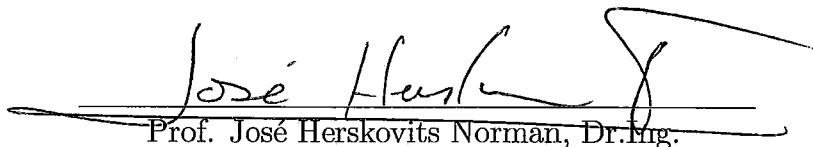
Prof. Susana Scheimberg de Makler, D.Sc.



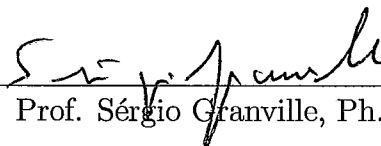
Prof. Nelson Maculan Filho, D.Sc.



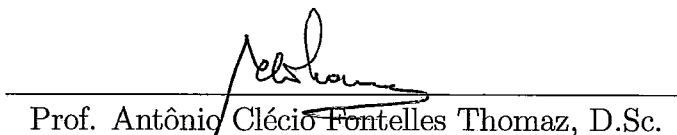
Prof. Maria Helena Cautiero Jardim, D.Sc.



Prof. José Herskovits Norman, Dr.Eng.



Prof. Sérgio Granville, Ph.D.



Prof. Antônio Clécio Fontelles Thomaz, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

DEZEMBRO DE 1999

CAMPÊLO NETO, MANOEL BEZERRA

Programação Linear em Dois Níveis: uma abordagem teórica e computacional [Rio de Janeiro] 1999

X, 169 p. 29,7 cm (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 1999)

Tese -- Universidade Federal do Rio de Janeiro, COPPE

1. Programação Linear em Dois Níveis.
2. Otimização Global e Local.
3. Métodos de Penalidade.

I. COPPE/UFRJ II. Título (série).

À minha avó Hilda.
(*in memoriam*)

Agradecimentos

*O melhor da vitória
é compartilhá-la.*

Nestes últimos quatro anos muito aprendi sobre a vida e, em particular, sobre o fazer acadêmico, sem contar os conhecimentos específicos adquiridos. Um fruto desse aprendizado é o presente trabalho. A todos que contribuíram nesse processo, gostaria de agradecer sinceramente.

Em especial, sou grato à professora Susana Scheimberg, pela dedicação e disponibilidade de sempre, pela orientação nos momentos de dúvidas, pelo estímulo, pelos ensinamentos de vida e, sobretudo, pela amizade. Agradeço ainda pelas valorosas contribuições ao texto final desse trabalho e de todos os outros que fizemos juntos.

Ao professor Nelson Maculan, retribuo com um agradecimento a generosidade, o incentivo, o suporte bibliográfico e os auxílios para congressos.

À professora Sulamita Klein, agradeço pela atenção, pela confiança demonstrada e pela parceria em minha primeira publicação internacional.

Devo agradecer também às seguintes pessoas que me apoiaram em outros aspectos não menos importantes. Afinal, não só com trabalho se faz um trabalho.

À minha família, meu alicerce, fortaleza e referencial, reservo um agradecimento maior, pela compreensão em todos os momentos, pelo apoio às minhas decisões, pela presença à distância, pelo amor que me dedicam. À minha prima Hilda, obrigado, em particular, pelas conversas constantes. À Jaudênia, obrigado por aturar meu estresse e compreender a falta de tempo dos últimos meses principalmente.

Aos meus velhos e novos amigos, sou grato pelo carinho, pelas conversas descontraídas, pela atenção e ... pelos ouvidos.

Às meninas da secretaria e demais funcionários do Programa de Engenharia de Sistemas e Computação, agradeço pelo trabalho competente.

Por fim, devo agradecer a CAPES, pela bolsa concedida, e ao Departamento de Estatística e Matemática Aplicada, pela liberação para o curso.

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

PROGRAMAÇÃO LINEAR EM DOIS NÍVEIS: UMA ABORDAGEM TEÓRICA E COMPUTACIONAL

Manoel Bezerra Campêlo Neto

Dezembro/1999

Orientadora: Susana Scheimberg de Makler

Programa: Engenharia de Sistemas e Computação

A atenção deste trabalho é concentrada no problema de programação linear em dois níveis. Suas características e propriedades básicas são apresentadas. Os principais algoritmos existentes na literatura são descritos. Em dois deles, problemas são detectados e contornados.

Usando-se um método de penalidades, são derivadas caracterizações dos ótimos locais e globais, bem como dos casos de inviabilidade e ilimitação do problema em dois níveis. Em particular, o conceito de ponto de equilíbrio foi introduzido para o desenvolvimento da análise local. Os resultados obtidos são relevantes tanto do ponto de vista teórico quanto numérico.

Um algoritmo local e outro global são propostos. Resultados computacionais são apresentados e algumas comparações realizadas com algoritmos já existentes.

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

LINEAR BILEVEL PROGRAMMING: A THEORETICAL AND COMPUTATIONAL APPROACH

Manoel Bezerra Campêlo Neto

Dezembro/1999

Advisor: Susana Scheimberg de Makler

Department: Computing and Systems Engineering

The focus of this work is the linear bilevel programming problem. Its basic characteristics and properties are presented. The main algorithms from the literature are described. In a couple of them, problems are detected and overcome.

Characterizations of local and global optima of the bilevel problem are derived by using a penalty method. The cases of infeasibility and unboundedness are characterized as well. In particular, the concept of equilibrium point was introduced to develop the local analysis. The results obtained are relevant from the theoretical and numerical points of view.

A local and a global algorithms are proposed. Computational results are presented and comparisons are carried out with existing algorithms.

Índice

Introdução	1
1 Programação Linear em Dois Níveis	3
1.1 O problema	3
1.2 Propriedades	14
1.3 Formulações equivalentes	20
1.4 Problemas relacionados	26
1.5 Relações com outras abordagens	31
1.5.1 Programação bicritério	31
1.5.2 Jogos de Stackelberg	34
1.6 Aplicações	35
2 Algoritmos existentes	37
2.1 Introdução	37
2.2 Enumeração de pontos extremos	38
2.2.1 K -ésimo melhor vértice	38
2.2.2 Pesquisa de bases ótimas do seguidor	40
2.3 Complementaridade linear paramétrica	42
2.4 Branch-and-bound	45
2.4.1 Fixação de variáveis complementares	45
2.4.2 Eliminação de variáveis do seguidor	48
2.5 Penalização de folgas complementares	53
2.5.1 Penalidade implícita	53
2.5.2 Penalidade explícita	55
2.6 Restrição convexa reversa	57
2.6.1 Decomposição primal	57

2.6.2	Decomposição dual	60
2.7	Pontos Interiores	62
3	Análise teórica do problema	66
3.1	Preliminares	66
3.2	Análise global	71
3.3	Análise local	78
3.3.1	Otimalidade local	78
3.3.2	Otimalidade local estrita	88
3.3.3	Otimalidade do ponto de equilíbrio	90
3.4	Extensões	98
4	Algoritmos propostos	104
4.1	Algoritmos para pontos de equilíbrio	104
4.2	Algoritmo local	111
4.3	Algoritmo global	115
4.4	Resultados computacionais	125
4.4.1	O gerador de problemas	125
4.4.2	Testes com os algoritmos propostos	127
5	Erros na literatura: contra-exemplos e modificações	134
5.1	Artigo de Önal	134
5.1.1	Problemas observados	135
5.1.2	Contra-exemplos	138
5.2	Artigo de White e Anandalingam	141
5.2.1	Problemas observados	142
5.2.2	Contra-exemplos	151
	Conclusões e perspectivas	158
	Referências Bibliográficas	161

Lista de Tabelas

4.1	Características de problemas-teste usados em outros trabalhos	128
4.2	Resultados computacionais obtidos pelo algoritmo local	129
4.3	Resultados computacionais obtidos pelo algoritmo global para problemas não transformados	131
4.4	Resultados computacionais obtidos pelo algoritmo global para problemas transformados	131
4.5	Iterações do algoritmo <i>outer approximation</i> adaptado	132
4.6	Resultados computacionais obtidos por Hansen <i>et al.</i>	133

Notação

Geral

argmax	conjunto de pontos que realizam o máximo
argmin	conjunto de pontos que realizam o mínimo
co	envoltória convexa
dim	dimensão de um conjunto
dom	domínio de um aplicação
ir	interior relativo de um conjunto
cl	fecho de um conjunto
$\ \nu\ _\infty$	norma infinito do vetor $\nu \in \mathfrak{R}^p$: $\ \nu\ _\infty = \max\{ \nu_i : i = 1, 2, \dots, p\}$
$B_\varepsilon(\nu)$	ε -vizinhança de um vetor ν na norma infinito
I_k	matriz identidade de ordem k
X_v	conjunto de vértices de um poliedro X
$V[P]$	valor ótimo de um problema de otimização P

Específica

PLDN	problema de programação linear em dois níveis
PLDNP	PLDN sem restrições no primeiro nível
PR	problema relaxado
$P(x)$	problema do segundo nível para x fixo
V	conjunto viável do problema linear em dois níveis
V^*	conjunto solução do problema linear em dois níveis
W	conjunto viável do problema relaxado
W^*	conjunto solução do problema relaxado
$W(x)$	conjunto viável do problema do segundo nível para x fixo
$Y(x)$	conjunto solução do problema do segundo nível para x fixo
f_1, f_2	funções objetivo do primeiro e segundo níveis
m_1, m_2	número de restrições do primeiro e segundo níveis
n_1, n_2	número de variáveis do primeiro e segundo níveis
(P)	PLDNP com $P(x)$ substituindo pelas condições de otimalidade
$P(M)$	(P) com as restrições de complementaridade penalizadas
Z	conjunto viável relaxado incluindo as variáveis de folga
S	conjunto dual do segundo nível imerso no espaço definido por Z
$Z(s)$	pontos em Z complementares a $s \in S$
$S(z)$	pontos em S complementares a $z \in Z$
F, F_M	funções objetivo dos problemas (P) e $P(M)$

Introdução

A programação em dois níveis é um ramo da programação matemática que se ocupa de problemas de otimização, onde a região viável é parcialmente definida pelo conjunto de soluções de um segundo problema, parametrizado pelas variáveis do primeiro. Este tipo de formulação é adequada para descrever sistemas hierárquicos, onde as ações de um nível superior delimitam as possibilidades do nível inferior, mas são também influenciadas pelas reações deste.

O presente trabalho é dedicado ao estudo do problema de programação linear em dois níveis (PLDN). Nos dois primeiros capítulos, descrevemos resultados e algoritmos já conhecidos. Nos dois seguintes, apresentamos a nossa efetiva contribuição. Adicionalmente, no quinto capítulo, identificamos incorreções em artigos publicados e sugerimos formas de contorná-las.

No capítulo 1, destacamos as principais características e propriedades de PLDN, bem como suas relações com outros problemas de programação matemática. Ilustramos as características peculiares através de exemplos e introduzimos, em alguns casos, novas demonstrações e extensões de propriedades conhecidas.

Descrevemos, no capítulo 2, os principais algoritmos já existentes para PLDN. Apontamos, em particular, que tipos de soluções eles encontram: se soluções globais, locais ou apenas pontos estacionários. Identificamos também as hipóteses que são assumidas, no intuito de verificarmos quão restritivos são os algoritmos e, principalmente, de analisarmos que condições poderiam ser relaxadas em nosso trabalho.

O desenvolvimento efetuado nos capítulos 3 e 4 fundamenta-se essencialmente em um método de penalidade. A aplicação de uma única metodologia uniformiza toda a teoria desenvolvida, o que não acontece na literatura. Ao mesmo tempo, ela permite a rigorosidade necessária para a dedução das propriedades. Além disso, o uso de penalidade proporciona um enfoque computacional direto.

Realizamos, no capítulo 3, análises teóricas global e local do problema. Abordamos, a princípio, o caso mais estudado em outros trabalhos, onde só existem restrições no segundo nível (PLDNP). Estendemos para o problema geral, quando possível, as propriedades obtidas. Exemplificamos também os casos onde a extensão não é válida.

Na análise global, mostramos, através do problema penalizado, em que situações o problema original possui solução, é inviável ou é ilimitado. Em geral, a existência de solução global é assumida por hipótese, e os casos de inviabilidade e ilimitação não são abordados.

Na análise local, caracterizamos as soluções locais e soluções locais estritas do problema. Para tal, introduzimos o conceito de ponto de equilíbrio do problema penalizado, que carrega um apelo computacional intrínseco. Observamos que análises locais detalhadas de PLDN são raras na literatura.

A partir dos resultados teóricos do capítulo 3, propomos, no capítulo 4, dois algoritmos que se estruturam sobre a noção de ponto de equilíbrio. O primeiro encontra ótimos locais e o segundo, ε -soluções globais de PLDNP. Apresentamos resultados computacionais para ambos os algoritmos, aplicados a problemas-teste aleatórios. Quando possível, efetuamos comparações com resultados disponíveis na literatura.

No capítulo 5, identificamos alguns erros em duas publicações de 1993 e sugerimos modificações para contornar as dificuldades observadas. É importante destacar que a investigação destes problemas, que aparecem tanto na parte teórica quanto algorítmica, foi o ponto de partida para o desenvolvimento deste trabalho.

Por último, apresentamos nossas conclusões e perspectivas de trabalhos futuros no capítulo final.

Capítulo 1

Programação Linear em Dois Níveis

Este capítulo tem como objetivo apresentar o problema de programação linear em dois níveis (PLDN), suas características, aplicações e relações com outras áreas da programação matemática. A seção 1.1 formula o problema, introduz a notação a ser utilizada e ilustra características peculiares a PLDN através de exemplos. A seção 1.2 formaliza as principais propriedades do problema. A seção 1.3 descreve várias reformulações de PLDN que são usadas na literatura. A seção 1.4 dá uma idéia da abrangência da programação linear em dois níveis, enumerando problemas clássicos de otimização que podem ser formulados como um PLDN. A seção 1.5 apresenta relações entre esse ramo da programação matemática e programação bi-critério e jogos de Stackelberg. Finalmente, a seção 1.6 cita algumas das aplicações existentes na literatura.

1.1 O problema

Muitos processos decisórios envolvem a participação de agentes que estão inseridos em uma estrutura hierárquica de poder ou influência. Frequentemente, cada nível nesta hierarquia tem objetivos próprios (conflitantes ou não) e controla algumas variáveis específicas, mas tem seu espaço de decisão total ou parcialmente determinado por outros níveis.

A programação em dois níveis é uma área da programação matemática cujo objeto de estudo são exatamente problemas de otimização envolvendo dois níveis hierárquicos com as características acima descritas, sendo que o agente do segundo

nível (o seguidor) está subordinado ao do primeiro (o líder).

Ao seguidor, que controla as variáveis y , está associado o seguinte problema de otimização:

$$P(x) \quad \max f_2(x, y) \quad \text{s.a } y \in W_2(x)$$

onde $f_2 : \mathfrak{R}^{n_1} \times \mathfrak{R}^{n_2} \rightarrow \mathfrak{R}$ e $W_2(x)$ é o conjunto viável parametrizado pelas variáveis x controladas pelo líder.

O problema do primeiro nível é definido em função do conjunto de soluções $Y(x)$ do problema paramétrico $P(x)$, do seguinte modo:

$$\max f_1(x, y) \quad \text{s.a } (x, y) \in W_1, y \in \Upsilon(x) \subseteq Y(x) \quad (1.1)$$

onde $f_1 : \mathfrak{R}^{n_1} \times \mathfrak{R}^{n_2} \rightarrow \mathfrak{R}$, $W_1 \subseteq \mathfrak{R}^{n_1} \times \mathfrak{R}^{n_2}$ é um conjunto fechado, e $\Upsilon(x)$, um subconjunto de soluções do problema do seguidor.

A escolha da multifunção Υ leva em conta a expectativa do líder em relação à resposta do seguidor. As duas principais opções são

$$\Upsilon(x) = \arg \max\{f_1(x, y) : y \in Y(x)\} \quad (1.2)$$

e

$$\Upsilon(x) = \arg \min\{f_1(x, y) : y \in Y(x)\}, \quad (1.3)$$

que definem respectivamente os problemas cooperativo e não cooperativo (Zhang [131]). No primeiro caso, supõe-se que o líder é capaz de induzir o seguidor a escolher dentre suas soluções aquela que seja mais conveniente ao primeiro nível. No segundo caso, adota-se a estratégia mais conservadora possível e se considera que o seguidor reage com a pior resposta para o líder.

O caso cooperativo equivale a tomar equivalentemente $\Upsilon \equiv Y$, gerando o problema

$$\begin{aligned} \max \quad & f_1(x, y) \\ \text{s.a} \quad & (x, y) \in W_1 \\ & y \in \arg \max\{f_2(x, y) : y \in W_2(x)\} \end{aligned}$$

Neste trabalho considera-se a formulação acima, onde as funções f_1 e f_2 são lineares e os conjuntos W_1 e $W_2(x)$, poliedros. Assim, define-se o problema de programação linear em dois níveis:

$$(PLDN) \quad \max_{x,y} f_1(x, y) = c_1^T x + c_2^T y \quad (1.4)$$

$$\text{s.a.} \quad B_1 x + B_2 y \leq b \quad (1.5)$$

$$x \geq 0, y \text{ solu\c{c}ão de} \quad (1.6)$$

$$\max_y f_2(x, y) = d^T y \quad (1.7)$$

$$\text{s.a.} \quad A_1 x + A_2 y \leq a \quad (1.8)$$

$$y \geq 0 \quad (1.9)$$

onde $x, c_1 \in \mathfrak{R}^{n_1}$, $c_2, d, y \in \mathfrak{R}^{n_2}$, $a \in \mathfrak{R}^{m_2}$, $b \in \mathfrak{R}^{m_1}$, $A_1 \in \mathfrak{R}^{m_2 \times n_1}$, $A_2 \in \mathfrak{R}^{m_2 \times n_2}$, $B_1 \in \mathfrak{R}^{m_1 \times n_1}$ e $B_2 \in \mathfrak{R}^{m_1 \times n_2}$. Por associa\c{c}ão ao decisor do primeiro n\u00edvel, diz-se que x \u00e9 a vari\u00e1vel do l\u00edder e (1.4) e (1.5) s\u00e3o respectivamente sua fun\c{c}ão objetivo e suas restri\c{c}ões expl\u00edcitas. Analogamente, y \u00e9 denominada a vari\u00e1vel do seguidor e (1.7) e (1.8), sua fun\c{c}ão objetivo e suas restri\c{c}ões.

Vale a pena mencionar que alguns autores adicionam um termo em x \u00e0 fun\c{c}ão objetivo do segundo n\u00edvel. Como este termo \u00e9 constante, preferimos descart\u00e1-lo. Entretanto, a defini\c{c}ão de f_2 como fun\c{c}ão de ambas as vari\u00e1veis \u00e9 conveniente para sua representa\c{c}ão no mesmo espa\c{c}o que f_1 .

O problema PLDN tem sido bastante estudado nas duas \u00faltimas d\u00e9cadas, como mostra a extensa revis\u00e3o bibliogr\u00e1fica realizada por Vicente e Calamai [116]. Resalte-se, por\u00e9m, que a maioria dos trabalhos versa sobre o problema em dois n\u00edveis que n\u00e3o inclui as restri\c{c}ões (1.5). Este modelo particular, a ser referenciado por PLDNP, \u00e9 revisado por Wen e Hsu [120] e Ben-Ayed [19].

Associados ao problema, s\u00e3o definidos os seguintes conjuntos:

- Conjunto vi\u00e1vel relaxado:

$$W = \{(x, y) \in \mathfrak{R}^{n_1} \times \mathfrak{R}^{n_2} : A_1 x + A_2 y \leq a, B_1 x + B_2 y \leq b, x \geq 0, y \geq 0\}.$$

- Conjunto vi\u00e1vel do segundo n\u00edvel para cada $x \in \mathfrak{R}_+^{n_1}$:

$$W(x) = \{y \in \mathfrak{R}^{n_2} : A_2 y \leq a - A_1 x, y \geq 0\}.$$

- Conjunto solu\c{c}ão do segundo n\u00edvel para cada $x \in \mathfrak{R}_+^{n_1}$:

$$Y(x) = \arg \max\{d^T y : y \in W(x)\}.$$

Tamb\u00e9m chamado conjunto rea\c{c}ão do seguidor.

- Conjunto viável de PLDN:

$$V = \{(x, y) \in W : y \in Y(x)\}.$$

Denominado igualmente conjunto de reações racionais ou região induzida.

- Conjunto solução de PLDN:

$$V^* = \arg \max \{c_1^T x + c_2^T y : (x, y) \in V\}.$$

Uma vez que o conjunto viável V está contido no conjunto viável relaxado W , pode-se definir a seguinte relaxação para PLDN:

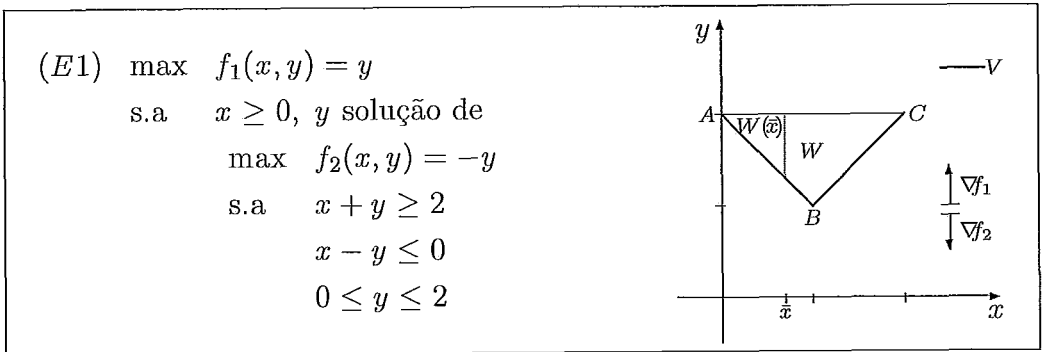
$$(PR) \quad \max c_1^T x + c_2^T y \quad \text{s.a.} \quad (x, y) \in W,$$

cujo conjunto solução será denotado por

$$W^* = \arg \max \{c_1^T x + c_2^T y : (x, y) \in W\}.$$

Os exemplos a seguir ilustram relações entre os conjuntos acima definidos e procuram ressaltar algumas de suas características peculiares.

Exemplo 1.1.1



No exemplo (E1), o conjunto viável relaxado é dado por

$$W = \{(x, y) \in \mathfrak{R} \times \mathfrak{R} : x + y \geq 2, \quad x - y \leq 0, \quad x \geq 0, \quad 0 \leq y \leq 2\},$$

correspondendo no gráfico à região delimitada pelo triângulo ABC . Assim, para cada $x \geq 0$, segue-se que

$$\begin{aligned} W(x) &= \{y \in \mathfrak{R} : y \geq 2 - x, \quad y \geq x, \quad 0 \leq y \leq 2\} \\ &= \{y \in \mathfrak{R} : y \geq 2 - x, \quad y \geq x, \quad y \leq 2\}. \end{aligned}$$

Se $2 - x \geq x$, ou seja, se $x \leq 1$, então $W(x) = \{y \in \Re : 2 - x \leq y \leq 2\}$. Do contrário, tem-se $W(x) = \{y \in \Re : x \leq y \leq 2\}$. Portanto, encontra-se

$$W(x) = \begin{cases} [2 - x, 2], & \text{se } 0 \leq x \leq 1 \\ [x, 2], & \text{se } 1 \leq x \leq 2 \\ \emptyset, & \text{se } x > 2 \end{cases}$$

Graficamente, $W(x)$ corresponde a um segmento vertical no triângulo ABC .

O conjunto solução do seguidor é $Y(x) = \arg \max\{-y : y \in W(x)\}$. Logo, tem-se

$$Y(x) = \begin{cases} \{2 - x\}, & \text{se } 0 \leq x \leq 1 \\ \{x\}, & \text{se } 1 \leq x \leq 2 \\ \emptyset, & \text{se } x > 2 \end{cases}$$

ou seja, $Y(x)$ é a extremidade inferior do segmento que descreve $W(x)$.

Como não existem restrições no primeiro nível além de $x \geq 0$, segue-se que $V = \{(x, y) : x \geq 0, y \in Y(x)\}$, isto é,

$$V = \{(x, 2 - x) : 0 \leq x \leq 1\} \cup \{(x, x) : 1 \leq x \leq 2\}.$$

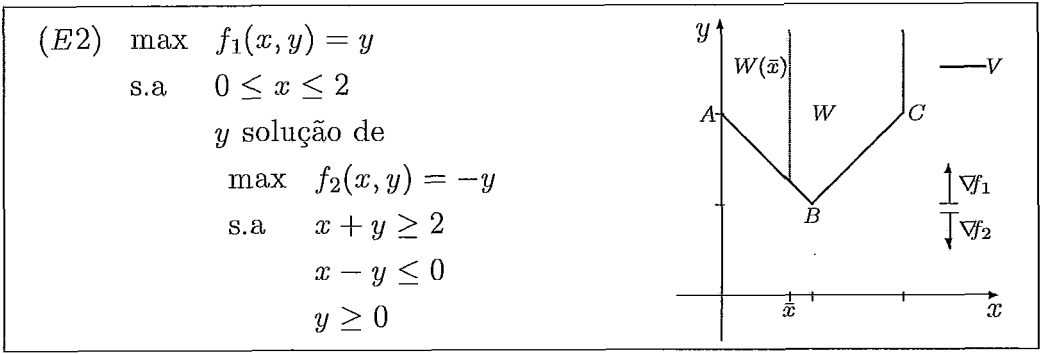
Assim, o conjunto viável de PLDN é definido pelos segmentos em **negrito** no gráfico. Observe-se que este conjunto é não convexo.

A função objetivo do líder $f_1(x, y) = y$ atinge seu máximo sobre V simultaneamente quando $x = 0$ ou $x = 2$, ou seja, nos pontos $A = (0, 2)$ e $C = (2, 2)$. Note-se, pois, que V^* é constituído por pontos extremos do conjunto V . À semelhança de um problema de programação linear, alguma solução de PLDN, se existir, é um ponto extremo do seu conjunto viável, que corresponde a um vértice do conjunto viável relaxado. A diferença é que o conjunto de soluções pode não ser conexo, nem, muito menos, convexo.

Quanto à solução do problema relaxado (PR), vê-se que todos os pontos do segmento AC são máximos de $f_1(x, y)$ sobre W . Sendo assim, tem-se neste caso

$$V^* = \{(0, 2), (2, 2)\} \subset [(0, 2), (2, 2)] = W^*.$$

Exemplo 1.1.2



O exemplo (E2) origina-se de (E1) com a retirada da restrição $y \leq 2$ do segundo nível e inclusão de $x \leq 2$ no primeiro. Assim, obtém-se

$$W = \{(x, y) \in \mathbb{R} \times \mathbb{R} : x + y \geq 2, x - y \leq 0, 0 \leq x \leq 2, y \geq 0\},$$

que corresponde ao conjunto convexo ilimitado indicado na figura. Então, para cada $x \geq 0$, tem-se

$$W(x) = \{y \in \mathbb{R} : y \geq 2 - x, y \geq x, y \geq 0\} = \{y \in \mathbb{R} : y \geq 2 - x, y \geq x\}.$$

Seguindo-se os mesmos cálculos aplicados a (E1), encontra-se

$$W(x) = \begin{cases} [2 - x, +\infty), & \text{se } 0 \leq x \leq 1 \\ [x, +\infty), & \text{se } x \geq 1 \end{cases}$$

Sendo assim, $W(x) \neq \emptyset$ para todo $x \geq 0$ e corresponde agora a uma semi-reta vertical limitada inferiormente.

O conjunto solução do segundo nível, dado por $Y(x) = \arg \max\{-y : y \in W(x)\}$ para cada $x \geq 0$, vale então

$$Y(x) = \begin{cases} \{2 - x\}, & \text{se } 0 \leq x \leq 1 \\ \{x\}, & \text{se } x \geq 1 \end{cases}$$

O conjunto viável de PLDN é formado pelos pontos (x, y) satisfazendo as restrições do primeiro nível e tais que $y \in Y(x)$, ou seja, $V = \{(x, y) : 0 \leq x \leq 2, y \in Y(x)\}$. Logo, segue-se que

$$V = \{(x, 2 - x) : 0 \leq x \leq 1\} \cup \{(x, x) : 1 \leq x \leq 2\}$$

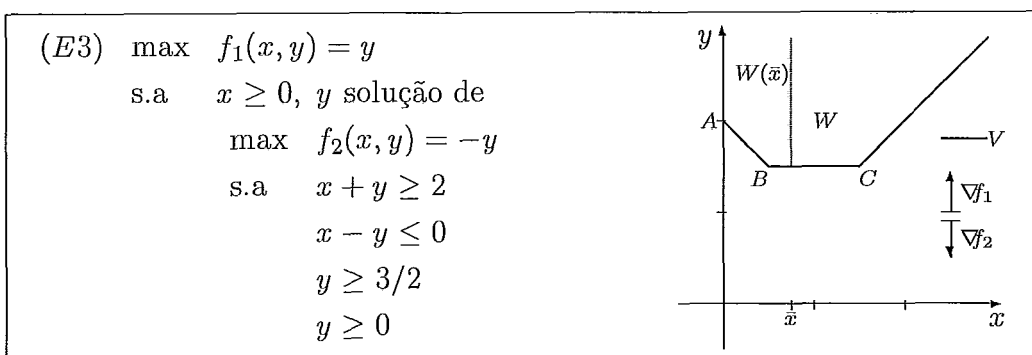
Como V não se altera com relação a (E1), o mesmo acontece com o conjunto solução de PLDN. Consequentemente,

$$V^* = \{(0, 2), (2, 2)\}.$$

Entretanto, a função do líder $f_1(x, y) = y$ é ilimitada sobre W , pois $(0, y) \in W$ para todo $y \geq 2$. Portanto, $W^* = \emptyset$. Perceba-se, assim, que $V^* \not\subset W^*$, embora $V \subset W$. Mais ainda, note-se que PLDN pode ter solução, mesmo quando o problema relaxado (PR) não tem.

Vale a pena mencionar que se a restrição $x \leq 2$ fosse transferida do primeiro para o segundo nível, o conjunto viável V e o conjunto viável relaxado W permaneceriam os mesmos. Consequentemente, ficariam também inalterados os conjuntos de soluções V^* e W^* . Na verdade, esta propriedade é válida para qualquer restrição que só envolva a variável do líder.

Exemplo 1.1.3



O exemplo (E3) é obtido de (E1) com a troca da restrição $y \leq 2$ por $y \geq 3/2$.

Neste caso, tem-se

$$W = \{(x, y) \in \mathbb{R} \times \mathbb{R} : x + y \geq 2, x - y \leq 0, x \geq 0, y \geq 3/2\}.$$

Como se observa no gráfico, o conjunto W descreve uma região convexa ilimitada.

Para cada $x \geq 0$, tem-se agora

$$W(x) = \{y \in \mathbb{R} : y \geq 2 - x, y \geq x, y \geq 3/2\},$$

resultando em

$$W(x) = \begin{cases} [2 - x, +\infty), & \text{se } 0 \leq x \leq 1/2 \\ [3/2, +\infty), & \text{se } 1/2 \leq x \leq 3/2 \\ [x, +\infty), & \text{se } x \geq 3/2 \end{cases}$$

Observe-se que $W(x)$ corresponde a uma semi-reta vertical em W .

A partir da expressão acima, determina-se que o conjunto solução do seguidor

$Y(x) = \arg \max\{-y : y \in W(x)\}$ é dado por

$$Y(x) = \begin{cases} \{2 - x\}, & \text{se } 0 \leq x \leq 1/2 \\ \{3/2\}, & \text{se } 1/2 \leq x \leq 3/2 \\ \{x\}, & \text{se } x \geq 3/2 \end{cases}$$

Como o conjunto viável de PLDN é $V = \{(x, y) : x \geq 0, y \in Y(x)\}$, segue-se que

$$V = \{(x, 2 - x) : 0 \leq x \leq 1/2\} \cup \{(x, 3/2) : 1/2 \leq x \leq 3/2\} \cup \{(x, x) : x \geq 3/2\},$$

correspondendo à região em negrito do gráfico. Neste caso, o conjunto V é ilimitado. Mais ainda, como $(x, x) \in V$ para todo $x \geq 3/2$, tem-se que a função do líder $f_1(x, y) = y$ é ilimitada sobre V , resultando em $V^* = \emptyset$. Evidentemente, o problema relaxado (PR) também não tem solução por motivo similar, de modo que $W^* = \emptyset$.

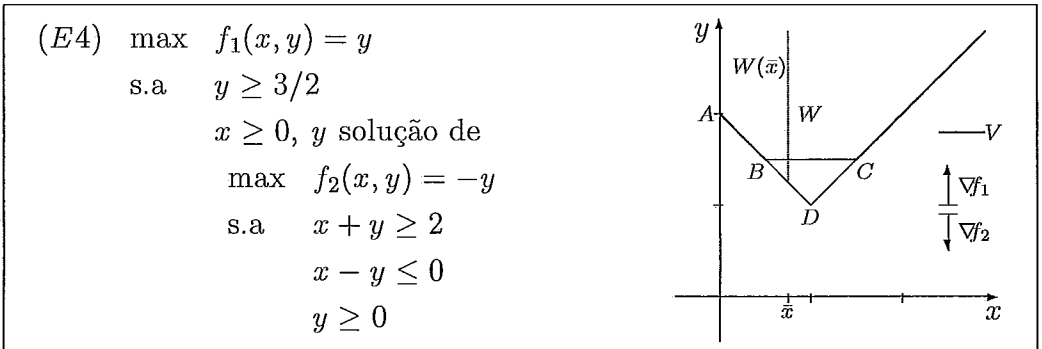
Embora PLDN não tenha solução global nesta situação, é fácil notar que o ponto $A = (0, 2)$ é um ótimo local. De fato, em uma certa vizinhança de A , todo ponto em V encontra-se sobre o segmento AB , sendo da forma $(x, y) = (x, 2 - x)$. Logo,

$$f_1(x, y) = y = 2 - x \leq 2 = f_1(0, 2)$$

e, por conseguinte, A é uma solução local de PLDN.

Pode-se perceber nos exemplos anteriores que as regiões viáveis de PLDN são conexas. Esta situação deve-se ao fato de que as restrições do primeiro nível, quando consideradas, dependiam unicamente da variável x . Todavia, a conexidade de V pode não se manter quando as restrições do líder envolvem a variável y do seguidor, como mostra o seguinte exemplo.

Exemplo 1.1.4



O exemplo (E4) constitui-se em uma modificação de (E3), produzida pela transferência da restrição $y \geq 3/2$ do segundo para o primeiro nível. Assim, o conjunto viável relaxado não muda, sendo novamente igual a

$$W = \{(x, y) \in \mathfrak{R} \times \mathfrak{R} : x + y \geq 2, x - y \leq 0, x \geq 0, y \geq 3/2\}.$$

Por outro lado, o conjunto viável do seguidor torna-se menos restrito. Para cada $x \geq 0$, tem-se

$$W(x) = \{y \in \mathfrak{R} : y \geq 2 - x, y \geq x, y \geq 0\} = \{y \in \mathfrak{R} : y \geq 2 - x, y \geq x\},$$

implicando

$$W(x) = \begin{cases} [2 - x, +\infty), & \text{se } 0 \leq x \leq 1 \\ [x, +\infty), & \text{se } x \geq 1 \end{cases}$$

Destaque-se que o conjunto viável do segundo nível é maior que aquele associado a (E3) quando $1/2 < x < 3/2$, como se observa na figura. Esta expansão de $W(x)$ faz com que o conjunto solução do seguidor seja diferente para x no intervalo especificado. De fato, considerando-se que $Y(x) = \arg \max\{-y : y \in W(x)\}$, chega-se a

$$Y(x) = \begin{cases} \{2 - x\}, & \text{se } 0 \leq x \leq 1 \\ \{x\}, & \text{se } x \geq 1 \end{cases}$$

O conjunto viável de PLDN é definido pelas restrições do primeiro nível juntamente com $y \in Y(x)$. Logo, tem-se $V = \{(x, y) : x \geq 0, y \geq 3/2, y \in Y(x)\}$, ou ainda,

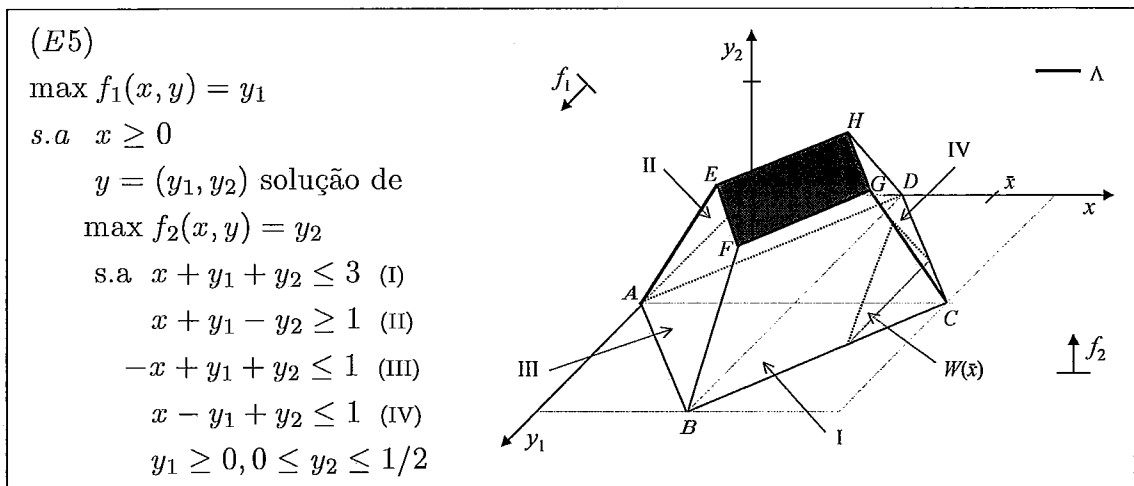
$$V = \{(x, 2 - x) : 0 \leq x \leq 1/2\} \cup \{(x, x) : x \geq 3/2\},$$

representado no gráfico pela união do segmento AB e da semi-reta com origem em C . Finalmente, como $(x, x) \in V$ para todo $x \geq 3/2$, conclui-se que $f_1(x, y) = y$ é ilimitada sobre V . Conseqüentemente, $V^* = W^* = \emptyset$.

É importante notar que a transferência da restrição do segundo para o primeiro nível reduziu o conjunto viável de PLDN. Verifique-se, assim, que uma restrição contendo a variável y afeta diferentemente o conjunto V , dependendo do nível em que ela é considerada. Observe-se também que este tipo de restrição, se posicionada no líder, pode gerar uma região viável de PLDN desconexa.

Além de desconexo, o conjunto viável V também pode ter pontos isolados, como acontece com o conjunto solução V^* . Por exemplo, substituindo-se em (E4) a restrição $y \geq 3/2$ por $y \geq 2$, isto é, elevando-se tal restrição até o ponto A , este ponto passa a ser uma solução viável isolada de PLDN.

Exemplo 1.1.5



No exemplo (E5), adaptado de Bialas e Karwan [26], deseja-se dar uma idéia da possível geometria do conjunto viável V em espaços de dimensões maiores que dois. Neste exemplo, tem-se que a região viável relaxada é definida por

$$W = \{(x, y) \in \mathbb{R} \times \mathbb{R}^2 : x + y_1 + y_2 \leq 3, x + y_1 - y_2 \geq 1, -x + y_1 + y_2 \leq 1, \\ x - y_1 + y_2 \leq 1, x \geq 0, y_1 \geq 0, 0 \leq y_2 \leq 1/2\},$$

que representa o sólido na figura. Assim, para cada $x \geq 0$, segue-se que o conjunto viável do seguidor é

$$W(x) = \{y \in \mathbb{R}^2 : y_1 + y_2 \leq 3 - x, -y_1 + y_2 \leq -1 + x, y_1 + y_2 \leq 1 + x, \\ -y_1 + y_2 \leq 1 - x, y_1 \geq 0, 0 \leq y_2 \leq 1/2\}.$$

Analisando-se os termos em x na expressão de $W(x)$, vê-se que dois casos devem ser considerados: $0 \leq x \leq 1$ e $x \geq 1$. No primeiro caso, tem-se $1 + x \leq 3 - x$ e $-1 + x \leq 1 - x$. Por conseguinte,

$$W(x) = \{y \in \mathbb{R}^2 : -y_1 + y_2 \leq -1 + x, y_1 + y_2 \leq 1 + x, 0 \leq y_2 \leq 1/2\}. \quad (1.10)$$

No segundo caso, segue-se que

$$W(x) = \{y \in \mathbb{R}^2 : y_1 + y_2 \leq 3 - x, -y_1 + y_2 \leq 1 - x, 0 \leq y_2 \leq 1/2\}. \quad (1.11)$$

Em particular, se $x > 2$ na expressão acima, conclui-se que $W(x) = \emptyset$.

Considerando-se as duas primeiras inequações em (1.10) e em (1.11), a expressão de $W(x)$ pode ser simplificada em alguns casos, obtendo-se

$$W(x) = \begin{cases} \{(1, 0)\}, & \text{se } x = 0 \\ \{y \in \mathbb{R}^2 : -y_1 + y_2 \leq -1 + x, y_1 + y_2 \leq 1 + x, y_2 \geq 0\}, & \text{se } 0 < x \leq 1/2 \\ \{y \in \mathbb{R}^2 : -y_1 + y_2 \leq -1 + x, y_1 + y_2 \leq 1 + x, 0 \leq y_2 \leq 1/2\}, & \text{se } 1/2 < x \leq 1 \\ \{y \in \mathbb{R}^2 : y_1 + y_2 \leq 3 - x, -y_1 + y_2 \leq 1 - x, 0 \leq y_2 \leq 1/2\}, & \text{se } 1 \leq x < 3/2 \\ \{y \in \mathbb{R}^2 : y_1 + y_2 \leq 3 - x, -y_1 + y_2 \leq 1 - x, y_2 \geq 0\}, & \text{se } 3/2 \leq x < 2 \\ \{(1, 0)\}, & \text{se } x = 2 \\ \emptyset, & \text{se } x > 2 \end{cases}$$

No gráfico, $W(x)$ está associado aos pontos A (se $x = 0$) ou C (se $x = 2$) ou corresponde a figuras paralelas ao plano y_1y_2 (se $0 < x < 2$). Neste último caso, $W(x)$ é delimitado ora por triângulos ora por trapézios, conforme se observa no desenho.

Para cada $x \geq 0$, o conjunto solução do seguidor é $Y(x) = \arg \max\{y_2 : (y_1, y_2) \in W(x)\}$. Quando $0 \leq x \leq 1/2$ ou $3/2 \leq x \leq 2$, $Y(x)$ é definido por um ponto da reta determinada pelos planos associados às restrições (II) e (III) ou (I) e (IV) respectivamente. Quando $1/2 < x < 3/2$, também a restrição $y_2 \leq 1/2$ torna-se efetiva e $Y(x)$ é dado por um segmento do plano $y_2 = 1/2$ perpendicular ao eixo x . Na verdade, encontra-se

$$Y(x) = \begin{cases} \{(1, x)\}, & \text{se } 0 \leq x \leq 1/2 \\ \{(y_1, 1/2) : 3/2 - x \leq y_1 \leq 1/2 + x\}, & \text{se } 1/2 \leq x \leq 1 \\ \{(y_1, 1/2) : -1/2 + x \leq y_1 \leq 5/2 - x\}, & \text{se } 1 \leq x \leq 3/2 \\ \{(1, 2 - x)\}, & \text{se } 3/2 \leq x \leq 2 \\ \emptyset & \text{se } x > 2 \end{cases}$$

Sendo assim, o conjunto viável de PLDN é definido por

$$\begin{aligned} V = & \{(x, 1, x) : 0 \leq x \leq 1/2\} \cup \\ & \{(x, y_1, 1/2) : 1/2 \leq x \leq 1, 3/2 - x \leq y_1 \leq 1/2 + x\} \cup \\ & \{(x, y_1, 1/2) : 1 \leq x \leq 3/2, -1/2 + x \leq y_1 \leq 5/2 - x\} \cup \\ & \{(x, 1, 2 - x) : 3/2 \leq x \leq 2\} \end{aligned}$$

e representado no gráfico pelos segmentos AE e GC e pela região delimitada pelo retângulo $EFGH$. Note-se que o conjunto viável V é constituído por faces de diferentes dimensões do conjunto viável relaxado W .

Diretamente da figura, constata-se que o máximo da função do líder $f_1(x, y) = y_1$ sobre V é atingido no ponto F e sobre W é encontrado no ponto B . Portanto, tem-se

$$V^* = \{(1, 3/2, 1/2)\} \quad \text{e} \quad W^* = \{(1, 2, 0)\}.$$

Finalmente, vale a pena observar que os intervalos $[A, E)$ e $(G, C]$ compõem-se de ótimos locais para o problema. De fato, dado $(x, y_1, y_2) \in [A, E) \cup (G, C]$, é sempre possível construir uma vizinhança onde todos os pontos viáveis de PLDN estejam no mesmo segmento que (x, y_1, y_2) . Além disso, pela expressão de V , constata-se que todo ponto em $[A, E)$ é da forma $(x, 1, x)$, com $0 \leq x < 1/2$, e que qualquer ponto em $(G, C]$ expressa-se como $(x, 1, 2 - x)$, para $3/2 < x \leq 2$. Logo, $f_1(x, y, y_2) = y_1 = 1$ para todo ponto (x, y_1, y_2) nestes intervalos, o que leva à conclusão desejada.

1.2 Propriedades

Como os próprios exemplos ilustram, muitas características de PLDN não são comumente encontradas na maioria dos problemas tradicionais de programação matemática. Nesta seção são formalizadas as principais propriedades de PLDN.

O conjunto viável V de PLDN está contido no conjunto viável relaxado W . Esta relação pode ser explorada para se estudar a geometria da região viável de PLDN. Na verdade, o conjunto V está ligado a faces do poliedro W .

Definição 1.2.1 *Seja C um conjunto convexo. Uma face de C é um subconjunto convexo C' de C que satisfaz a seguinte condição: se $x, y \in C$ e $(1 - \lambda)x + \lambda y \in C'$ para algum $\lambda \in (0, 1)$, então $x, y \in C'$. As faces de dimensão zero são chamadas pontos extremos de C (Rockafellar [97, pág. 162]).*

Se o conjunto convexo é um poliedro, suas faces também são poliedrais e, portanto, conjuntos fechados. Neste caso, o número de faces é finito, e as faces de dimensão zero são os vértices do poliedro. Ressalte-se que os vértices de uma face são vértices do conjunto original.

Quando não existem restrições no primeiro nível ou elas só dependem da variável x , a caracterização do conjunto viável V como faces do conjunto viável relaxado W pode ser inferida de resultados de Gallo e Ülkücü [59, lema 1.1.2] e Benson [23, teorema 3.2].

Seguindo a mesma argumentação apresentada em [59] mostramos no teorema 1.2.1 abaixo que esta propriedade continua válida mesmo se $B_2 \neq \emptyset$, isto é, se existem restrições no primeiro nível envolvendo a variável y . Para melhor entendimento,

estabelecemos primeiro o seguinte resultado auxiliar, onde se denota interior relativo de um conjunto C por $\text{ir } C$.

Lema 1.2.1 *Seja C um subconjunto convexo e não vazio do conjunto viável relaxado W . Se $\text{ir } C \cap V \neq \emptyset$, onde V é o conjunto viável de PLDN, então $C \subseteq V$.*

Prova: Sejam $C \subseteq W$ um conjunto convexo e $\bar{\omega} = (\bar{x}, \bar{y}) \in \text{ir } C \cap V$. Se $\dim C = 0$, então $C = \text{ir } C = \{\bar{\omega}\}$. E como $\bar{\omega} \in V$, tem-se $C \subseteq V$. Assuma-se agora que $\dim C > 0$ e seja $\omega' = (x', y') \in C$, $\omega' \neq \bar{\omega}$. Deseja-se mostrar que $\omega' \in V$. Como $(x', y') \in W$, basta então verificar que y' resolve o problema $P(x')$ do seguidor, ou seja, que $d^T y' \geq d^T y$ para todo $y \in W(x')$.

Posto que C é convexo e $\bar{\omega} \in \text{ir } C$, existem $\omega'' = (x'', y'') \in C$ e $0 < \alpha < 1$ verificando $\bar{\omega} = \alpha\omega' + (1 - \alpha)\omega''$ (Rockafellar [97, teorema 6.4]). Assim,

$$\bar{x} = \alpha x' + (1 - \alpha)x'' \quad \text{e} \quad \bar{y} = \alpha y' + (1 - \alpha)y''.$$

Seja $y \in W(x') \neq \emptyset$ e defina-se

$$\hat{y} = \alpha y + (1 - \alpha)y''.$$

Então, tem-se

$$(\bar{x}, \hat{y}) = \alpha(x', y) + (1 - \alpha)(x'', y'').$$

Considere-se o conjunto

$$\bar{W} = \{(x, y) : x \geq 0, y \in W(x)\} = \{(x, y) \geq 0 : A_1 x + A_2 y \leq a\}.$$

Como $(x', y) \in \bar{W}$, $(x'', y'') \in C \subseteq W \subseteq \bar{W}$ e \bar{W} é convexo, segue-se que $(\bar{x}, \hat{y}) \in \bar{W}$. Logo, $\hat{y} \in W(\bar{x})$. E como $\bar{\omega} = (\bar{x}, \bar{y}) \in V$, tem-se $\bar{y} \in Y(\bar{x})$. Portanto, $d^T \bar{y} \geq d^T \hat{y}$. Substituindo-se as expressões de \bar{y} e \hat{y} , obtém-se que

$$\alpha d^T y' + (1 - \alpha)d^T y'' \geq \alpha d^T y + (1 - \alpha)d^T y''.$$

Sendo $\alpha > 0$, deve-se ter $d^T y' \geq d^T y$. Portanto, $d^T y' \geq d^T y$ para todo $y \in W(x')$. ■

Teorema 1.2.1 *Sejam W_i , $i \in I$, as faces de W . Então existe $I' \subseteq I$ tal que*

$$V = \bigcup_{i \in I'} W_i,$$

ou seja, o conjunto viável de PLDN é união de faces do conjunto viável relaxado.

Prova: Seja $I' = \{i \in I : W_i \subseteq V\}$. Tem-se que $I' \neq \emptyset$, pois pelo menos a face vazia de W satisfaz a definição de I' . Além disso, $I' \subseteq I$ é finito, pois o número de faces é finito. Então, claramente,

$$\bigcup_{i \in I'} W_i \subseteq V.$$

Se $V = \emptyset$, a inclusão inversa se verifica trivialmente. Do contrário, seja $\omega \in V \subseteq W$. Como W é convexo, tem-se que $\omega \in \text{ir } W_{i'}$, para algum $i' \in I$ (Rockafellar [97, teorema 18.2]). Assim, $\omega \in \text{ir } W_{i'} \cap V$. Como $W_{i'} \subseteq W$ e $W_{i'}$ é convexo, segue-se do lema 1.2.1 que $W_{i'} \subseteq V$. Logo, $i' \in I'$ e, conseqüentemente, $\omega \in W_{i'} \subseteq \bigcup_{i \in I'} W_i$. Portanto,

$$V \subseteq \bigcup_{i \in I'} W_i. \quad \blacksquare$$

Corolário 1.2.1 *O conjunto viável V de PLDN é fechado.*

Prova: Pelo teorema 1.2.1, V é a união de conjuntos fechados. Logo, V é um conjunto fechado. ■

O fato do conjunto viável ser fechado assegura a boa definição de PLDN como um problema de maximização, pois o supremo da função do líder sobre V , se finito, é atingido em um ponto do conjunto. Sendo assim, PLDN é inviável, ilimitado ou tem soluções globais. Ressalte-se, por outro lado, que a não convexidade do conjunto viável possibilita a existência de ótimos locais, conforme ilustram os exemplos.

Embora não convexo, o conjunto viável V pode ser substituído por sua envoltória convexa (co) na definição de PLDN. Este resultado não aparece demonstrado na literatura, o que fazemos a seguir.

Teorema 1.2.2 *Associado a PLDN, seja definido o seguinte problema convexo*

$$PLDN_c \quad \max f_1(x, y) \quad \text{s.a.} \quad (x, y) \in \text{co } V.$$

Os problemas PLDN e $PLDN_c$ são simultaneamente inviáveis, ilimitados ou têm solução. Neste último caso, tem-se ainda que:

(i) *Toda solução de PLDN é solução de $PLDN_c$.*

(ii) Toda solução extrema de PLDN é solução extrema de $PLDN_c$ e vice-versa.

Prova: Se PLDN é inviável, então $V = \text{co } V = \emptyset$ e, conseqüentemente, $PLDN_c$ é inviável. Se PLDN é ilimitado, como $\text{co } V \supseteq V$, segue-se que $PLDN_c$ também é ilimitado. Finalmente, assumamos que PLDN tem solução (x^*, y^*) . Então, $PLDN_c$ é viável. Seja $(x, y) \in \text{co } V$. Logo,

$$(x, y) = \sum_{i \in I} \alpha_i (x^i, y^i),$$

onde I é um conjunto finito, $(x^i, y^i) \in V$ e $\alpha_i \geq 0$ para $i \in I$ e $\sum_{i \in I} \alpha_i = 1$. Assim,

$$c_1^T x + c_2^T y = \sum_{i \in I} \alpha_i (c_1^T x^i + c_2^T y^i) \leq \sum_{i \in I} \alpha_i (c_1^T x^* + c_2^T y^*) = c_1^T x^* + c_2^T y^*.$$

E como $(x^*, y^*) \in \text{co } V$, tem-se que (x^*, y^*) é solução de $PLDN_c$. Usando-se as três implicações obtidas, chega-se, por contraposição, às implicações inversas, completando-se a prova da primeira parte do teorema. Então, a partir do item (i), que já está demonstrado, obtém-se o item (ii), usando-se o fato de que todo ponto extremo V é ponto extremo de $\text{co } V$ e vice-versa. ■

É importante observar que a equivalência entre PLDN e $PLDN_c$ não é total, pois nem toda solução deste problema é solução daquele. De fato, no exemplo (E1), tem-se $\text{co } V = W$. Conseqüentemente, o conjunto solução de $PLDN_c$ é W^* . E, como visto nesse exemplo, $V^* \subset W^*$, mas $W^* \not\subset V^*$.

Os próximos resultados se referem à existência de soluções extremas de PLDN. Eles foram estabelecidos inicialmente para PLDNP por Candler e Townsley [42], Bialas e Karwan [25, 26] e Bard [10], sob hipóteses de compacidade do conjunto W e de solução única do seguidor para cada x fixo do líder. Depois, foram estendidos para PLDN por Savard [98], sem condições particulares. Embora estas propriedades já sejam conhecidas, as demonstrações aqui apresentadas diferem das anteriores.

Como conseqüência do teorema 1.2.2, mostra-se que a programação linear em dois níveis tem em comum com a programação linear clássica a seguinte característica:

Teorema 1.2.3 *Se PLDN tem solução, pelo menos uma delas é atingida em um ponto extremo do conjunto viável V .*

Prova: Se PLDN tem solução, o teorema 1.2.2 garante que $PLDN_c$ também tem. E como $PLDN_c$ é dado pela maximização de uma função linear sobre um conjunto convexo, pelo menos uma de suas soluções é atingida em um ponto extremo de $co V$. Novamente pelo teorema 1.2.2, segue-se que esta é uma solução de PLDN e um ponto extremo de V . ■

Considerando-se o resultado acima, é importante caracterizar os pontos extremos de V .

Teorema 1.2.4 *Todo ponto extremo do conjunto viável V de PLDN é um vértice do conjunto viável relaxado W .*

Prova: Seja ω um ponto extremo de V . Pelo teorema 1.2.1, $\omega \in W_i \subseteq V$, onde W_i é uma face de W . Sejam $\omega', \omega'' \in W$ e $0 < \alpha < 1$, tais que $\omega = \alpha\omega' + (1 - \alpha)\omega''$. Como ω pertence a face W_i , segue-se da definição 1.2.1 que $\omega', \omega'' \in W_i \subseteq V$. Então, sendo ω um ponto extremo de V , deve-se ter $\omega = \omega' = \omega''$. Portanto, ω é vértice de W . ■

Diretamente dos teoremas 1.2.3 e 1.2.4 decorre que:

Corolário 1.2.2 *Se PLDN tem solução, pelo menos uma delas é atingida em um vértice do conjunto viável relaxado W .*

Esta propriedade tem motivado o desenvolvimento de muitos algoritmos que procuram resolver PLDN através da pesquisa de vértices do conjunto viável relaxado. Mais ainda, ela fundamenta a aplicação e/ou adequação de ferramentas típicas de programação linear para PLDN.

Outra característica importante do conjunto viável V é a sua conexidade quando as restrições do líder só dependem da variável x .

Definição 1.2.2 *Um conjunto C é dito conexo quando não pode ser escrito como a união de dois conjuntos A e B não vazios e abertos e tais que*

$$A \cap (cl B) = \emptyset \quad e \quad (cl A) \cap B = \emptyset,$$

onde cl denota o fecho do conjunto (Benson [23]).

O conceito de conexidade de conjuntos está ligado à idéia de continuidade. Na verdade, um conjunto C é conexo quando existe sempre um “caminho contínuo” de pontos de C entre qualquer par de pontos deste conjunto.

A seguinte propriedade é demonstrada por Benson [23].

Teorema 1.2.5 *Se as restrições do primeiro nível independem da variável do segundo nível, isto é, $B_2 = 0$, então o conjunto viável V de PLDN é conexo.*

Verifique-se que o resultado acima não se mantém válido quando $B_2 \neq 0$. O exemplo (E4) ilustra a desconexidade de V quando existem restrições no primeiro nível dependentes da variável y .

Um último aspecto ilustrado pelos exemplos da seção anterior diz respeito à influência, sobre o conjunto V , do posicionamento de uma restrição no primeiro ou segundo nível. Para se estudar este ponto, considerem-se os seguintes problemas de programação linear em dois níveis, com as restrições $Q_1x + Q_2y \leq q$ pertencendo ora ao líder ora ao seguidor.

(PLDN ₁)	(PLDN ₂)
$\max_{x,y} c_1^T x + c_2^T y$	$\max_{x,y} c_1^T x + c_2^T y$
s.a. $B_1x + B_2y \leq b$	s.a. $B_1x + B_2y \leq b$
$Q_1x + Q_2y \leq q$	$x \geq 0, y$ solução de
$x \geq 0, y$ solução de	$\max_y d^T y$
$\max_y d^T y$	s.a. $A_1x + A_2y \leq a$
s.a. $A_1x + A_2y \leq a$	$Q_1x + Q_2y \leq q$
$y \geq 0$	$y \geq 0$

onde Q_1, Q_2 e q possuem dimensões apropriadas.

Vicente [115, teorema 1] mostra que a transferência de uma restrição do primeiro para o segundo nível pode expandir a região viável do problema. Na verdade, o efeito é o mesmo quando ela só depende de x . Estas propriedades são demonstradas a seguir.

Teorema 1.2.6 *O conjunto viável de PLDN₁ está contido no conjunto viável de PLDN₂. Em particular, se $Q_2 = 0$, estes dois conjuntos são iguais.*

Prova: Os conjuntos de soluções para o segundo nível de PLDN₁ e PLDN₂ são respectivamente

$$Y_1(x) = \arg \max \{d^T y : A_2y \leq a - A_1x, y \geq 0\}$$

e

$$Y_2(x) = \arg \max\{d^T y : A_2 y \leq a - A_1 x, Q_2 y \leq q - Q_1 x, y \geq 0\}.$$

Seja (x, y) viável para $PLDN_1$. Então, $Q_1 x + Q_2 y \leq q$ e $y \in Y_1(x)$. Logo, $y \in Y_2(x)$. Além disso, como $B_1 x + B_2 y \leq b$ e $x \geq 0$, tem-se que (x, y) é viável para $PLDN_2$. Assuma-se agora que $Q_2 = 0$ e seja (x, y) viável para $PLDN_2$. Então, $Q_1 x \leq q$ e, por conseguinte, $Y_2(x) = Y_1(x)$. Como $y \in Y_2(x) = Y_1(x)$ e (x, y) satisfaz as restrições do primeiro nível de $PLDN_1$, segue-se que (x, y) é viável para este problema. ■

O caso de igualdade entre os conjuntos viáveis resulta nas seguintes equivalências.

Corolário 1.2.3 *Se $Q_2 = 0$, então os problemas $PLDN_1$ e $PLDN_2$ são equivalentes.*

Corolário 1.2.4 *Se as restrições do primeiro nível em $PLDN$ só envolvem a variável x , ou seja, $B_2 = 0$, então $PLDN$ é equivalente ao $PLDNP$ obtido pela transferência destas restrições para o segundo nível.*

1.3 Formulações equivalentes

Uma prática natural para se estudar e resolver um problema consiste em fazer associações com outros problemas conhecidos. No caso de $PLDN$, diversas abordagens que o tratam indiretamente têm sido propostas. Nesse sentido, são usadas reformulações do problema que, além de suscitarem diferentes análises teóricas, também permitem o desenvolvimento de métodos de resolução alternativos.

Os modelos matemáticos equivalentes a $PLDN$ mais usados na literatura são apresentados a seguir.

Modelo com restrições de complementaridade linear

A grande dificuldade em se lidar com $PLDN$ encontra-se no fato de que uma parte de suas restrições é dada implicitamente pelo conjunto solução de um segundo problema de otimização. Uma maneira de se escapar a esta estrutura multinível consiste em substituir o problema do seguidor por suas condições de Karush-Kuhn-Tucker (KKT). Esta estratégia é usada, entre outros, por Bialas e Karwan [26], Bard [6, 9], Júdice e Faustino [73], Önal [94] e White e Anandalingam [127].

A formulação obtida com as condições de otimalidade do seguidor é:

$$\begin{aligned}
(P) \quad & \max \quad c_1^T x + c_2^T y \\
& \text{s.a} \quad B_1 x + B_2 y \leq b \\
& \quad \quad A_1 x + A_2 y + w = a \\
& \quad \quad x \geq 0, y \geq 0, w \geq 0 \\
& \quad \quad A_2^T u - v = d \\
& \quad \quad u \geq 0, v \geq 0 \\
& \quad \quad v^T y = u^T w = 0
\end{aligned} \tag{1.12}$$

onde $w \in \mathfrak{R}^{m_2}$ é uma variável de folga, e $u \in \mathfrak{R}^{m_2}$ e $v \in \mathfrak{R}^{n_2}$ são variáveis duais associadas às restrições do segundo nível.

O problema (P) é equivalente a PLDN, no sentido de que um ponto (x, y) é solução global deste se, e somente se, (x, y, w, u, v) é solução global daquele para algum (w, u, v) (Audet *et al.* [4, proposição 3.3]). Esta equivalência deve-se ao fato de que as condições de otimalidade de KKT são necessárias e suficientes para o problema linear do segundo nível.

Embora seja (P) um problema tradicional (em um único nível) de programação não linear, os métodos de solução usuais não são efetivos. A não convexidade do conjunto viável, gerada pelas restrições de complementaridade (1.12), torna difícil a determinação de ótimos globais. Duas alternativas para se descrever estas restrições são apresentadas a seguir.

Modelo separável

Uma função $f : \mathfrak{R}^{p_1} \times \mathfrak{R}^{p_2} \times \dots \times \mathfrak{R}^{p_k} \rightarrow \mathfrak{R}$ é dita separável quando

$$f(z^1, z^2, \dots, z^k) = \sum_{i=1}^k f_i(z^i),$$

para um conjunto de funções $f_i : \mathfrak{R}^{p_i} \rightarrow \mathfrak{R}$, $i = 1, 2, \dots, k$. Em decorrência, um problema de otimização é separável se tanto o objetivo quando as restrições são expressos por funções separáveis.

Um modelo separável equivalente a (P) é obtido por Bard e Falk [12] com a reformulação das restrições (1.12). Visto que em (P) as variáveis são todas não negativas, estas restrições podem ser substituídas por

$$\sum_{j=1}^{n_2} \min\{v_j, y_j\} = \sum_{i=1}^{m_2} \min\{u_i, w_i\} = 0$$

ou, equivalentemente, por

$$\sum_{j=1}^{n_2} (\min\{0, y_j - v_j\} + v_j) = \sum_{i=1}^{m_2} (\min\{0, u_i - w_i\} + w_i) = 0. \quad (1.13)$$

Então, definindo-se

$$\nu_j = y_j - v_j, \quad j = 1, \dots, n_2 \quad \text{e} \quad \omega_i = u_i - w_i, \quad i = 1, \dots, m_2,$$

a expressão (1.13) transforma-se em

$$\sum_{j=1}^{n_2} (\min\{0, \nu_j\} + v_j) = \sum_{i=1}^{m_2} (\min\{0, \omega_i\} + w_i) = 0.$$

Desta forma, o problema (P) e, conseqüentemente PLDN, podem ser reescritos como:

$$\begin{aligned} \text{(PS)} \quad & \max \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y \leq b \\ & \quad A_1 x + A_2 y + w = a \\ & \quad A_2^T u - v = d \\ & \quad x \geq 0, y \geq 0, w \geq 0, u \geq 0, v \geq 0 \\ & \quad \nu + v - y = 0 \\ & \quad \omega + w - u = 0 \\ & \quad \sum_{j=1}^{n_2} (\min\{0, \nu_j\} + v_j) = \sum_{i=1}^{m_2} (\min\{0, \omega_i\} + w_i) = 0 \end{aligned}$$

Nesta formulação, mantém-se a não convexidade do conjunto viável e perde-se a diferenciabilidade do problema, devido ao uso da função $\min\{\}$ para redefinir as condições de folgas complementares. Por outro lado, obtém-se a sua separabilidade, pois tanto a função objetivo quanto as restrições são agora expressas por funções separáveis em x , y , w , u , v , ν e ω . Esta estrutura do problema (PS) é explorada por Bard e Falk [12], que usam um algoritmo tipo *branch-and-bound* para resolvê-lo.

Modelo de programação linear 0-1 mista

Quando o problema (P) tem solução, outra possibilidade para se representar as restrições de complementaridade (1.12) encontra-se na introdução de variáveis binárias 0-1, conforme sugerido por Fortuny-Amat e McCarl [54].

Seja $\|z\|_\infty = \max\{|z_j| : j = 1, 2, \dots, p\}$ a norma infinito do vetor $z \in \mathbb{R}^p$. Então, considerando-se $z^* = (x^*, y^*, w^*, u^*, v^*)$ uma solução de (P) e tomando-se

$M \geq \|z^*\|_\infty$, uma formulação equivalente ao problema é dada pelo seguinte modelo de programação linear 0-1 mista:

$$\begin{aligned}
 \text{(PI)} \quad & \max \quad c_1^T x + c_2^T y \\
 & \text{s.a} \quad B_1 x + B_2 y \leq b \\
 & \quad A_1 x + A_2 y + w = a \\
 & \quad A_2^T u - v = d \\
 & \quad x \geq 0 \\
 & \quad 0 \leq y \leq M\nu \tag{1.14}
 \end{aligned}$$

$$0 \leq v \leq M(e_1 - \nu) \tag{1.15}$$

$$0 \leq w \leq M\omega \tag{1.16}$$

$$0 \leq u \leq M(e_2 - \omega) \tag{1.17}$$

$$\nu \in \{0, 1\}^{n_2}, \omega \in \{0, 1\}^{m_2} \tag{1.18}$$

onde $e_1^T = (1, 1, \dots, 1) \in \mathfrak{R}^{n_2}$, $e_2^T = (1, 1, \dots, 1) \in \mathfrak{R}^{m_2}$ e $\{0, 1\}^p$ representa o conjunto dos vetores p -dimensionais com componentes 0-1.

Observe-se que as restrições (1.14)-(1.15) forçam $y_j = 0$ ou $v_j = 0$, dependendo se $\nu_j = 0$ ou $\nu_j = 1$ respectivamente. Da mesma forma, as restrições (1.16)-(1.17) obrigam a se ter $w_i = 0$ ou $u_i = 0$ respectivamente quando $\omega_i = 0$ ou $\omega_i = 1$. Por outro lado, a escolha de M , que limita o valor das variáveis contínuas, garante que uma solução global de (P) seja viável para (PI). Em consequência, este modelo é equivalente a (P) e, por conseguinte, a PLDN. Deve-se ressaltar que esta correspondência parte da hipótese de que o problema original tem solução.

A representação das restrições de complementaridade como em (PI) é usada por Fortuny-Amat e McCarl [54] para reformular problemas em dois níveis com funções objetivo quadráticas. A solução do modelo é obtida através de um método enumerativo.

Modelo de penalidade

Uma forma indireta de se tratar as restrições de folgas complementares em (P) é introduzi-las na função objetivo do líder como um termo de penalidade. Esta idéia é empregada por Bard [9], Anandalingam e White [3, 127], Önal [94], Amouzegar e Moshirvazari [2], Campêlo *et al.* [35] e Campêlo e Scheimberg [36, 37].

Para um parâmetro de penalidade $M \geq 0$, define-se o seguinte problema:

$$\begin{aligned}
P(M) \quad & \max \quad c_1^T x + c_2^T y - M(u^T w + v^T y) \\
\text{s.a} \quad & B_1 x + B_2 y \leq b \\
& A_1 x + A_2 y + w = a, \\
& x \geq 0, y \geq 0, w \geq 0, \\
& A_2^T u - v = d, \\
& u \geq 0, v \geq 0
\end{aligned}$$

Observe-se que a não convexidade do conjunto viável de (P) foi transferida agora para a função objetivo e que $P(M)$ é um problema bilinear para cada M fixo.

Quando não existem restrições no primeiro nível e sob condições de existência de solução e compacidade do conjunto viável relaxado W , Bard [9, teorema 1] mostra que as soluções extremas de PLDN e $P(M)$ são correspondentes para um certo valor do parâmetro de penalidade M . Campêlo *et al.* [35, teorema 7] enfraquecem a última hipótese e mostram que é suficiente supor a função do líder limitada superiormente em W para se obter a mesma correspondência. Na verdade, estes resultados podem ser estendidos para o caso onde existem restrições lineares no líder.

Vale a pena mencionar que Anandalingam e White [3, teorema 3] apresentam outra demonstração de que soluções de $P(M)$, para um certo valor de M , fornecem soluções de PLDN. Esta propriedade é usada em seguida por White e Anandalingam [127]. Entretanto, as hipóteses consideradas pelos autores são inválidas (Campêlo *et al.* [35]).

Modelo convexo reverso

Um problema linear acrescido de uma restrição $g(z) \geq 0$, onde $g : \mathbb{R}^p \rightarrow \mathbb{R}$ é uma função convexa, é dito um problema convexo reverso (Tuy *et al.* [111, pág. 3]). Esta definição decorre justamente do fato da restrição $g(z) \geq 0$ ser chamada de convexa reversa (Horst e Tuy [69, definição I.4]).

Um modelo convexo reverso equivalente a PLDN é usado por Tuy *et al.* [111, 112], Tuy [109] e Tuy e Ghannadam [110]. Este modelo é obtido a partir da descrição de PLDN como:

$$(PLDN) \quad \max_{x,y} \quad c_1^T x + c_2^T y \tag{1.19}$$

$$\text{s.a} \quad B_1 x + B_2 y \leq b \tag{1.20}$$

$$x \geq 0, y \in Y(x) \tag{1.21}$$

Posto que $Y(x) = \arg \max\{d^T y : y \in W(x)\}$, a restrição $y \in Y(x)$ é igual a

$$d^T y = \phi(x), \quad y \in W(x), \quad (1.22)$$

onde $\phi : \mathfrak{R}^{n_1} \rightarrow \mathfrak{R} \cup \{-\infty, +\infty\}$ é a função marginal do segundo nível, dada por

$$\phi(x) = \sup\{d^T y : y \in W(x)\}.$$

Além disso, como $d^T y \leq \phi(x)$ para todo $y \in W(x)$, a equação (1.22) torna-se

$$d^T y \geq \phi(x), \quad y \in W(x) = \{y \geq 0 : A_1 x + A_2 y \leq a\}.$$

Assim, reescreve-se (1.19)-(1.21) como:

$$\begin{aligned} \text{(PCR)} \quad & \max \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y \leq b \\ & \quad A_1 x + A_2 y \leq a \\ & \quad x \geq 0, y \geq 0 \\ & \quad d^T y - \phi(x) \geq 0 \end{aligned} \quad (1.23)$$

A função marginal ϕ é côncava (Tuy *et al.* [111, proposição 1]). Consequentemente, $g(x, y) = d^T y - \phi(x)$ é convexa, ou seja, tem-se uma restrição convexa reversa em (1.23). Portanto, (PCR) é um problema convexo reverso e pode ser tratado com técnicas específicas (ver Horst e Tuy [69]).

Modelo de programação semi-infinita

Um modelo de programação semi-infinita é um modelo de otimização onde existe um número infinito de restrições descritas parametricamente (Bazaraa *et al.* [17, pág. 183]).

Bard [6] observa que PLDN pode ser descrito através de um modelo de programação semi-infinita. De fato, o conjunto solução do segundo nível pode ser representado por um número infinito de desigualdades paramétricas, ou seja,

$$Y(x) = \{y \in \mathfrak{R}^{n_2} : d^T y \geq d^T \omega \quad \forall \omega \in W(x)\}.$$

Sendo assim, a formulação (1.19)-(1.21) é reescrita como:

$$\begin{aligned} \text{(PSI)} \quad & \max \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y \leq b \\ & \quad A_1 x + A_2 y \leq a \\ & \quad x \geq 0, y \geq 0 \\ & \quad d^T y \geq d^T \omega \quad \forall \omega \in W(x) \end{aligned}$$

Usando este modelo, Bard [6] propõe condições necessárias de otimalidade para PLDN. Entretanto, elas são incorretas, conforme verificado por Savard [98, seção 2.3]. Como consequência, os algoritmos sugeridos por Bard [6], LeBlanc e Boyce [78] e Ünlü [113], que se baseiam nestes resultados, são inexatos e fornecem apenas uma solução viável para PLDN. Estas constatações são feitas respectivamente por Haurie *et al.* [62], Marcotte [82] e Candler [38]. Destaque-se ainda que extensões destas condições de otimalidade, apresentadas por Bard [7, 10] para o caso não linear, também não são válidas pelo mesmo motivo (Clark e Westerberg [47]).

1.4 Problemas relacionados

Para se ter uma idéia da abrangência da programação linear em dois níveis, esta seção apresenta importantes problemas de programação matemática que podem ser vistos como casos particulares de PLDN.

Programação max-min linear

A formulação geral de um problema max-min linear, apresentada por Falk [51], é dada por

$$(PMM) \quad \max_{x \geq 0} \min_{y \geq 0} \{c_1^T x + c_2^T y : A_1 x + A_2 y \leq a\}$$

onde c_1, c_2, a, x, y são vetores e A_1, A_2 são matrizes. Um ponto (x, y) é considerado viável para PMM quando satisfaz as restrições e, fixado x , o problema de minimização interno tem solução.

Definido desta forma, o problema acima constitui um caso especial de PLDN onde as funções objetivo do primeiro e segundo níveis são simétricas em relação à adição. De fato, PMM equivale a:

$$(PLDN1) \quad \begin{array}{ll} \max_{x,y} & c_1^T x + c_2^T y \\ \text{s.a} & x \geq 0, y \text{ solução de} \\ & \max_y -c_2^T y \\ \text{s.a} & A_1 x + A_2 y \leq a \\ & y \geq 0 \end{array}$$

As seguintes constatações confirmam a equivalência propalada. Primeiro, note-se que o termo $-c_1^T x$ pode ser adicionado à função objetivo do seguidor, uma vez que a variável x está fixa no segundo nível. Segundo, verifique-se que é indiferente

neste caso se considerar a maximização da função objetivo do líder apenas sobre x ou sobre ambas as variáveis. Realmente, dada a simetria entre os objetivos, a parcela $c_2^T y$ na função do líder é a mesma para qualquer reação y do seguidor com respeito a um determinado x .

Relações entre problemas max-min e PLDN são estudadas por Bard e Falk [12] e Audet *et al.* [5].

Programação bilinear

Um problema de programação bilinear é definido pelo seguinte modelo quadrático:

$$\begin{aligned} \text{(PBL)} \quad & \max \quad c^T x + x^T Q y + d^T y \\ & \text{s.a} \quad Bx \leq b, \quad x \geq 0 \\ & \quad \quad Ay \leq a, \quad y \geq 0 \end{aligned}$$

onde a, b, c, d, x, y são vetores e A, B, Q são matrizes.

Este problema tem sido bastante estudado na literatura, entre outros, por Konno [75], Vaish e Shetty [114], Gallo e Ülkücü [59], Sherali e Shetty [99], Thieu [106], Júdice e Faustino [72], Horst e Tuy [69], White [126].

O problema PBL pode ser reescrito como

$$\max \quad c^T x + f(x) \quad \text{s.a} \quad Bx \leq b, \quad x \geq 0,$$

onde

$$f(x) = \sup\{(Qx + d)^T y : Ay \leq a, y \geq 0\}. \quad (1.24)$$

Assuma-se, a princípio, que o conjunto $\mathcal{Y} = \{y \geq 0 : Ay \leq a\}$ seja não vazio. Então, pela teoria da dualidade, tem-se

$$f(x) = \inf\{a^T \nu : A^T \nu \geq Qx + d, \nu \geq 0\}.$$

Portanto, sob a hipótese considerada, PBL é equivalente ao seguinte PLDN:

$$\begin{aligned} \text{(PLDN2)} \quad & \max_{x, \nu} \quad c^T x + a^T \nu \\ & \text{s.a} \quad Bx \leq b, \quad x \geq 0 \\ & \quad \quad \nu \text{ solução de} \\ & \quad \quad \max_{\nu} \quad -a^T \nu \\ & \quad \quad \text{s.a} \quad A^T \nu - Qx \geq d \\ & \quad \quad \nu \geq 0 \end{aligned}$$

Quando o conjunto \mathcal{Y} é vazio, o problema do segundo nível em PLDN2 pode ser inviável ou ilimitado, pois ele corresponde ao dual do problema linear em (1.24). Em ambos os casos, segue-se que PLDN2 é inviável, mantendo-se assim a equivalência com PBL.

Na verdade, PLDN2 pode ser convertido em um problema em dois níveis sem restrições no líder, conforme o corolário 1.2.4. De fato, as restrições do líder não envolvem a variável ν do seguidor e podem ser transferidas para o segundo nível.

Uma relação inversa entre programação linear em dois níveis e programação bilinear pode ser estabelecida a partir do problema penalizado $P(M)$, apresentado na seção 1.3. Como mencionado, $P(M)$ é um problema bilinear para cada valor do parâmetro de penalidade. Além disso, sob certas hipóteses, existe um valor finito deste parâmetro para o qual PLDN e $P(M)$ possuem soluções correspondentes.

Programação linear 0-1 mista

O problema de programação linear 0-1 mista é formulado, de acordo com Audet *et al.* [4], como:

$$\begin{aligned} \text{(PLIM)} \quad & \max \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y \leq b \\ & \quad x \geq 0, y \in \{0, 1\}^p \end{aligned}$$

onde $\{0, 1\}^p$ representa o conjunto dos vetores p -dimensionais com componentes 0-1, c_1, c_2, b, x, y são vetores e B_1, B_2 são matrizes.

Introduzindo-se uma variável auxiliar $\nu \in \mathfrak{R}^p$, as restrições de integralidade de PLIM podem ser substituídas por um problema de programação linear, gerando o seguinte modelo em dois níveis:

$$\begin{aligned} \text{(PLDN3)} \quad & \max_{x, y, \nu} \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y \leq b \\ & \quad x \geq 0, 0 \leq y \leq e \\ & \quad \nu = 0, \nu \text{ solução de} \\ & \quad \max_{\nu} \quad e^T \nu \\ & \quad \text{s.a} \quad \nu \leq y \\ & \quad \quad \nu \leq e - y \\ & \quad \quad \nu \geq 0 \end{aligned}$$

onde $e^T = (1, 1, \dots, 1) \in \mathfrak{R}^p$.

A equivalência entre os problemas acima decorre da correspondência entre seus conjuntos viáveis. De fato, verifique-se que a solução do seguidor em PLDN3 é sempre

$$\nu_j = \min\{y_j, 1 - y_j\}, \quad j = 1, 2, \dots, p.$$

Assim, o conjunto viável de PLDN3 é

$$\{(x, y, 0) \geq 0 : B_1x + B_2y \leq b, \min\{y_j, 1 - y_j\} = 0 \quad j = 1, 2, \dots, p\}.$$

Logo, tendo em vista que cada restrição $y_j \in \{0, 1\}$ é equivalente a $\min\{y_j, 1 - y_j\} = 0$, segue-se que um ponto (x, y) é viável para PLIM se, e somente se, $(x, y, 0)$ é viável para PLDN3. Finalmente, como os dois problemas possuem a mesma função objetivo, conclui-se que terão soluções ótimas correspondentes.

Embora possa parecer artificial, a reformulação de PLIM como um PLDN sugere que um problema linear em dois níveis seja pelo menos tão difícil de ser resolvido quanto um problema de programação inteira (Audet *et al.* [4, seção 5]).

Vicente *et al.* [118] apresentam outro modelo em dois níveis equivalente a PLIM. Em vez de forçar $\nu = 0$, como em PLDN3, esta restrição é penalizada na função objetivo do líder, obtendo-se o seguinte problema:

$$\begin{aligned} \text{(PLDN4)} \quad & \max_{x, y, \nu} \quad c_1^T x + c_2^T y - M e^T \nu \\ & \text{s.a.} \quad B_1 x + B_2 y \leq b \\ & \quad x \geq 0, 0 \leq y \leq e \\ & \quad \nu \text{ solução de} \\ & \quad \max_{\nu} \quad e^T \nu \\ & \quad \text{s.a.} \quad \nu \leq y \\ & \quad \quad \nu \leq e - y \\ & \quad \quad \nu \geq 0 \end{aligned}$$

onde $e^T = (1, 1, \dots, 1) \in \mathbb{R}^p$. De acordo com o corolário 1.2.4, note-se que as restrições do primeiro nível podem ser transferidas para o segundo, gerando-se assim um PLDNP.

Considerando que o problema PLDN4 tem solução única para todo $M \geq 0$, Vicente *et al.* [118] mostram que este problema e PLIM tem as mesmas soluções para algum M finito. Esta é uma consequência do teorema 3.2 em [118]. Por outro lado, relembre-se, da seção 1.3, que PLDN pode ser transformado em um problema linear 0-1, que também depende de uma constante desconhecida M .

Finalmente, vale a pena mencionar que a idéia usada na obtenção de PLDN3 também é empregada por Audet *et al.* [4] para substituir variáveis binárias no primeiro nível por variáveis contínuas. Se a variável binária é $v \in \{0, 1\}^p$, então define-se uma variável contínua $\nu \in \mathfrak{R}^p$, acrescentam-se as restrições $\nu = 0$ ao primeiro nível e $\nu \leq v$ e $\nu \leq e - v$ ao segundo e adiciona-se o termo $e^T \nu$ à função do seguidor. Este tipo de problema linear-misto em dois níveis é considerado por Bard e Moore [14, 88], Wen e Yang [125] e Vicente *et al.* [118].

Complementaridade linear generalizada

Seja o problema de otimização linear com restrições de complementaridade, considerado por Júdice e Mitra [74]:

$$(PLRC) \quad \max \quad c_1^T x_1 + c_2^T x_2 + c_3^T x_3 \quad (1.25)$$

$$\text{s.a.} \quad Q_1 x_1 + Q_2 x_2 + Q_3 x_3 = q \quad (1.26)$$

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0 \quad (1.27)$$

$$x_2^T x_3 = 0 \quad (1.28)$$

onde $c_1, c_2, c_3, q, x_1, x_2, x_3$ são vetores e Q_1, Q_2, Q_3 são matrizes.

Este modelo generaliza o clássico problema de complementaridade linear, que é dado pelas restrições (1.26)-(1.28), quando Q_1 é a matriz nula e Q_3 , a identidade. Além disso, ele pode ser usado para descrever vários problemas de otimização, entre os quais problemas de programação quadrática e de inequações variacionais lineares (Júdice e Mitra [74]).

O problema PLRC pode ser reformulado através do seguinte PLDN:

$$(PLDN5) \quad \max_{x, \nu} \quad c_1^T x_1 + c_2^T x_2 + c_3^T x_3$$

$$\text{s.a.} \quad Q_1 x_1 + Q_2 x_2 + Q_3 x_3 = q$$

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0$$

$$\nu = 0, \quad \nu \text{ solução de}$$

$$\max_{\nu} \quad e^T \nu$$

$$\text{s.a.} \quad \nu \leq x_2$$

$$\nu \leq x_3$$

$$\nu \geq 0$$

onde $e^T = (1, 1, \dots, 1) \in \mathfrak{R}^p$ tem a mesma dimensão de x_2, x_3 e ν .

De fato, a solução do seguidor em PLDN5 é dado por

$$\nu_j = \min\{x_{2j}, x_{3j}\}, \quad j = 1, 2, \dots, p.$$

Assim, o conjunto viável de PLDN5 é

$$\{(x_1, x_2, x_3, 0) \geq 0 : Q_1 x_1 + Q_2 x_2 + Q_3 x_3 = q, \min\{x_{2j}, x_{3j}\} = 0 \quad j = 1, 2, \dots, p\}.$$

Além disso, como $(x_2, x_3) \geq 0$ em ambos os problemas, a restrição $\min\{x_{2j}, x_{3j}\} = 0$ é equivalente a $x_{2j}x_{3j} = 0$. Portanto, tem-se que um ponto (x_1, x_2, x_3) é viável para PLRC se, e somente se, $(x_1, x_2, x_3, 0)$ é viável para PLDN5. Esta correspondência entre os conjuntos viáveis e o fato das funções objetivo serem iguais resultam na equivalência entre os dois problemas.

Reciprocamente, PLDN pode ser expresso como um PLRC (Júdice e Mitra [74]). Considere-se a seguinte reformulação de PLDN, apresentada na seção 1.3.

$$\begin{aligned} (P) \quad & \max \quad c_1^T x + c_2^T y \\ & \text{s.a} \quad B_1 x + B_2 y + \omega = b \\ & \quad \quad A_1 x + A_2 y + w = a \\ & \quad \quad A_2^T u - v = d \\ & \quad \quad x \geq 0, y \geq 0, w \geq 0, \omega \geq 0 \\ & \quad \quad u \geq 0, v \geq 0 \\ & \quad \quad v^T y = u^T w = 0 \end{aligned}$$

Então, o PLRC equivalente é obtido definindo-se:

$$\begin{aligned} x_1 &= \begin{bmatrix} x \\ \omega \end{bmatrix}, \quad x_2 = \begin{bmatrix} u \\ y \end{bmatrix}, \quad x_3 = \begin{bmatrix} w \\ v \end{bmatrix}, \\ c_1^T &= [c_1^T \quad 0], \quad c_2^T = [0 \quad c_2^T], \quad c_3^T = 0, \quad q^T = [b^T \quad a^T \quad -d^T] \\ Q_1 &= \begin{bmatrix} B_1 & I \\ A_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 0 & B_2 \\ 0 & A_2 \\ -A_2^T & 0 \end{bmatrix}, \quad Q_3 = \begin{bmatrix} 0 & 0 \\ I & 0 \\ 0 & I \end{bmatrix}. \end{aligned}$$

1.5 Relações com outras abordagens

Esta seção correlaciona o problema de programação linear em dois níveis aos problemas de otimização bicritério e de Stackelberg. Estas duas outras formulações também consideram múltiplos objetivos e são frequentemente associadas a PLDN.

1.5.1 Programação bicritério

A programação bicritério procura determinar um compromisso simultâneo entre dois objetivos, considerados sob um mesmo conjunto de restrições. No caso linear,

o problema de programação bicritério é formulado como:

$$\begin{aligned}
 \text{(PLBC)} \quad & \max \quad (c^T z, d^T z) \\
 \text{s.a} \quad & Az \leq a \\
 & z \geq 0
 \end{aligned}$$

onde a, c, d, z são vetores e A é uma matriz.

Associado a PLBC não existe o conceito de solução ótima, mas de solução eficiente (ou Pareto-ótima), entendida como uma solução viável que não pode ser simultaneamente melhorada pelos dois objetivos. Mais precisamente, define-se:

Definição 1.5.1 *Uma solução eficiente (ou Pareto-ótima) para PLBC é uma solução viável z^* tal que, para cada outro ponto z viável, acontece um dos seguintes casos:*

$$c^T z < c^T z^*, \text{ ou } d^T z < d^T z^*, \text{ ou } c^T z = c^T z^* \text{ e } d^T z = d^T z^*.$$

Ao contrário, se z^* e z viáveis não satisfazem um destes casos, diz-se que z^* é uma **solução dominada por z** .

O próprio conceito de solução eficiente mostra que não se considera em PLBC uma hierarquia entre os objetivos. Na verdade, supõe-se que eles sejam de um único decisor ou grupo harmônico de decisores.

Uma conhecida caracterização para soluções eficientes (ver Zeleny [129]) é dada a seguir:

Teorema 1.5.1 *Uma solução z^* é eficiente para PLBC se, e somente se, existe $\lambda \in (0, 1)$ tal que z^* é solução ótima do problema*

$$\max \lambda c^T z + (1 - \lambda) d^T z \quad \text{s.a} \quad Az \leq a, z \geq 0$$

Vários trabalhos procuram estabelecer relações entre PLDNP e o PLBC definido pelas funções objetivo do líder e do seguidor.

Em 1983, Bard [6, seção 2] afirma equivocadamente que toda solução (x^*, y^*) de PLDNP é solução do problema paramétrico

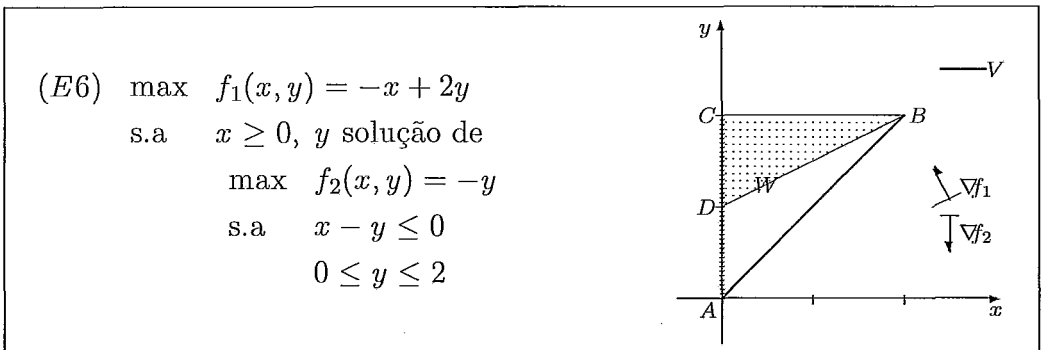
$$\begin{aligned}
 \max \quad & \lambda(c_1^T x + c_2^T y) + (1 - \lambda)d^T y \\
 \text{s.a} \quad & A_1 x + A_2 y \leq a \\
 & x \geq 0, y \geq 0
 \end{aligned}$$

para um certo $\lambda \in (0, 1]$. Tal constatação o leva a concluir que uma solução de PLDNP é eficiente com respeito às funções objetivo do primeiro e segundo níveis. Esta conclusão é reafirmada por Bard [10, corolário 4] em um trabalho do ano seguinte.

Ünlü [113] mostra, em 1987, que o resultado de Bard não é correto em geral e aponta como única exceção o caso onde $\lambda = 1$ e PLDNP tem múltiplas soluções (ver [113, corolário 2]). Entretanto, em 1988, Candler [38] apresenta outro contra-exemplo numa situação diferente. Em 1989, Wen e Tsu [122, teorema 3] supõem provar que uma solução de PLDNP é eficiente quando $c_2^T d \geq 0$, isto é, quando os vetores de coeficientes da variável y nas funções objetivo do primeiro e segundo níveis definem um ângulo agudo. Esta condição também não é válida.

Na verdade, Marcotte e Savard [84] verificam que uma solução de PLDNP só é garantidamente Pareto-ótima para as funções do líder e do seguidor, se $c_2 = kd$ para algum $k > 0$. Este resultado esclarece que não existe uma correspondência direta entre PLDN e o PLBC associado. O exemplo a seguir ilustra esta observação.

Exemplo 1.5.1



O conjunto viável de (E6) e o conjunto de soluções eficientes são dados respectivamente pelos segmentos AB e AC na figura. Verifique-se, então, que a única solução simultaneamente viável para (E6) e eficiente com respeito às funções do líder e do seguidor encontra-se no ponto A , que corresponde a pior solução possível para o problema de dois níveis. Por outro lado, a solução ótima de (E6), dada pelo ponto B , não é eficiente. De fato, ela é dominada por todos os pontos da região pontilhada.

Em particular, a solução de (E6) é dominada por todas as soluções eficientes do intervalo $[C, D]$. Em um extremo (ponto C) o valor da função objetivo do primeiro

nível é incrementado e o valor da função objetivo do segundo nível se mantém. No outro extremo (ponto D) acontece o inverso. Nos pontos intermediários, ambas as funções são melhoradas. De qualquer modo, se o líder e o seguidor acordarem em escolher uma solução em $[C, D]$, nenhum dos dois estará perdendo.

Considerando estas hipóteses de cooperação entre os dois níveis, Wen e Hsu [121] e Wen e Lin [124] propõem algoritmos para encontrar soluções eficientes “melhores” que a solução de PLDNP.

1.5.2 Jogos de Stackelberg

O problema estático de Stackelberg [103] é um problema da teoria de jogos, onde dois agentes procuram otimizar seus objetivos individuais, sujeitos a conjuntos interdependentes de restrições. O jogo é sequencial e a ordem é especificada *a priori*. O primeiro jogador, o líder, escolhe sua estratégia e espera a reação do segundo jogador, o seguidor, que tem seus movimentos influenciados pela escolha do primeiro. Maiores detalhes podem ser encontrados em Simaan e Cruz [101, 102].

No caso linear, o problema é formulado como:

$$\begin{aligned}
 (\text{PES}) \quad & \max_x \quad c_1^T x + c_2^T y(x) \\
 & \text{s.a} \quad B_1 x + B_2 y(x) \leq b \\
 & \quad x \geq 0, \quad y(x) \text{ solução de} \\
 & \quad \max_y \quad d^T y \\
 & \quad \text{s.a} \quad A_1 x + A_2 y \leq a \\
 & \quad y \geq 0
 \end{aligned}$$

onde $y(\cdot)$ é uma aplicação que mapeia x em um ponto

$$y(x) \in Y(x) = \arg \max \{d^T y : A_2 y \leq a - A_1 x, y \geq 0\}.$$

Dado que a solução de PES depende da aplicação $y(\cdot)$, alguns autores (p.e. Shimizu *et al.* [100]) afirmam que este problema é mal-posto. Para contornar tal dificuldade, costuma-se assumir que o problema do seguidor tem solução única para todo x .

Alternativamente, duas variantes de PES são definidas quando se considera que o líder conhece todo o conjunto $Y(x)$ de soluções do seguidor e não apenas um elemento $y(x)$. Estas formulações originam respectivamente o problema forte (PESF) e o

problema fraco (PESf) de Stackelberg, descritos a seguir (ver Loridan e Morgan [80]).

$$\begin{array}{ll}
 \text{(PESF)} & \max_x \max_y c_1^T x + c_2^T y \\
 & \text{s.a. } B_1 x + B_2 y \leq b \\
 & x \geq 0, y \text{ solu\c{c}ao de} \\
 & \max_y d^T y \\
 & \text{s.a. } A_1 x + A_2 y \leq a \\
 & y \geq 0
 \end{array}
 \qquad
 \begin{array}{ll}
 \text{(PESf)} & \max_x \min_y c_1^T x + c_2^T y \\
 & \text{s.a. } B_1 x + B_2 y \leq b \\
 & x \geq 0, y \text{ solu\c{c}ao de} \\
 & \max_y d^T y \\
 & \text{s.a. } A_1 x + A_2 y \leq a \\
 & y \geq 0
 \end{array}$$

Note-se que PESF corresponde exatamente a PLDN. Por isso mesmo, PLDN é muitas vezes descrito como um jogo de soma não zero com as características mencionadas no início desta subseção.

Na verdade, PES, PESF e PESf enquadram-se dentro do modelo geral de problemas em dois níveis (1.1), apresentado no início do capítulo. O primeiro problema é definido por $\Upsilon(x) = \{y(x)\}$. Já os outros dois correspondem exatamente aos casos cooperativo (1.2) e não cooperativo (1.3). Quando a solução do seguidor é única para todo x , ou seja, $Y(x)$ é um conjunto unitário, os três problemas coincidem.

1.6 Aplicações

Modelos de programação linear em dois níveis têm sido usados para descrever diversas aplicações práticas. A estrutura multinível se adequa bem à formulação de problemas que envolvem um processo de decisão hierárquico. Entre as várias aplicações existentes na literatura destacam-se as seguintes.

O problema de projeto de redes (*network design problem*) está relacionado à modificação de um sistema de transporte, pela adição de novos caminhos ou a melhoria de ligações já existentes. Uma formulação em dois níveis para este problema é proposta por LeBlanc e Boyce [78] e depois generalizada por Ben-Ayed *et al.* [22]. Uma nova formulação é apresentada por Ben-Ayed *et al.* [21] para resolver o problema em uma rede real específica. Marcotte [81], Friesz *et al.* [57], Marcotte e Marquis [83] e Suh e Kim [105] também usam modelos em dois níveis para abordar problemas de projeto de redes com restrições de equilíbrio. Migdalas [86] analisa ambos os problemas mencionados e descreve várias possibilidades de solução.

Ainda na área de redes, Florian e Chen [52, 53, 46] estudam o problema de estimação de matrizes de demanda de fluxo entre pontos origem e destino. Tobin e Friesz [107] e Friesz *et al.* [56, 58] consideram problemas de localização de facilidades. Zhang e Zhu [130] e Wolf e Smeers [128] aplicam modelos de dois níveis para dimensionar redes de escoamento de fluidos, onde a topologia já está definida, mas é preciso determinar os diâmetros dos canos a serem usados.

A programação em dois níveis também tem sido empregada no planejamento de políticas ótimas nos setores elétrico e agrícola. No primeiro caso, citem-se os trabalhos de Savard [98], Haurie *et al.* [61] e Hobbs e Nelson [67] e, no segundo, os de Candler e Norton [40, 41], Candler *et al.* [39], Fortuny-Amat e McCarl [54], Parraga [96], Bisschop *et al.* [28], Önal [93] e Önal *et al.* [95].

Problemas de gerência, administração e planejamento na área governamental são considerados por Cassidy *et al.* [43], Kyland [76], deSilva [50] e Bard *et al.* [15]. No setor privado, este tipo de problema é abordado por Burton e Obel [33] e Bard [8], na coordenação de firmas multisetoriais; por Miller *et al.* [87], na localização de pontos de entrega em domicílio; por Parraga [96] e Wen e Jiang [123], no sistema financeiro.

No campo econômico aparecem diversas aplicações, em particular aquelas associadas ao problema estático de Stackelberg. Entre as referências estão Fortuny-Amat e McCarl [54], Friedman [55], Wen [119] e Dempe [49].

Finalmente, podem-se mencionar trabalhos na área militar (Bracken *et al.* [29] e Bracken e McGill [30, 31, 32]), em engenharia (Stavroulakis e Günzel [104] e Herskovits *et al.* [65]) e na indústria (Nicholls [91, 92]).

Embora alguns dos trabalhos citados não considerem apenas formulações lineares, uma alternativa para a obtenção de soluções aproximadas encontra-se muitas vezes na linearização do modelo original.

Capítulo 2

Algoritmos existentes

Este capítulo destina-se a apresentar os principais algoritmos que existem na literatura para resolver PLDN local ou globalmente. Cada seção descreve um grupo de algoritmos construídos sobre uma idéia ou estratégia comum, relacionada a: (i) enumeração de vértices, (ii) complementaridade linear, (iii) *branch-and-bound*, (iv) funções de penalidade, (v) convexidade reversa e (vi) métodos de pontos interiores. Antes dos algoritmos, a primeira seção estuda a complexidade de PLDN.

2.1 Introdução

A dificuldade em se resolver PLDN pode ser deduzida da constatação de que todo problema de programação inteira 0-1 pode ser formulado como um problema linear em dois níveis, conforme seção 1.4. Assim, problemas de programação 0-1 reconhecidamente difíceis, como o problema da mochila, são casos particulares de PLDN.

Alguns trabalhos têm se dedicado a estudar a complexidade teórica de PLDN. Um dos primeiros esforços nesta direção foi empreendido por Jeroslow [71], que mostra ser PLDNP um problema NP-difícil. Demonstrações mais concisas deste resultado são apresentadas em seguida por Ben-Ayed e Blair [20] e Bard [11]. Em linhas gerais, isto significa que é pouco provável a existência de um algoritmo polinomial para resolver o problema. Mais ainda, não se espera que a obtenção de ε -soluções para PLDN possa ser feita em tempo polinomial no tamanho do problema e em ε . Esta é a conclusão decorrente de um resultado estabelecido por Hansen *et al.* [60, corolário 3.2].

Mais recentemente, Vicente *et al.* [117] mostram que mesmo a verificação de otimalidade local em PLDN é NP-Difícil. Além disso, Calamai e Vicente [34] apre-

sentam uma técnica para gerar problemas lineares em dois níveis com um número exponencial de ótimos locais.

Toda esta dificuldade em se resolver PLDN tem motivado a proposição de uma variedade de métodos numéricos de solução. Seguindo-se Hansen *et al.* [60] e Vicente e Calamai [116], nas seções seguintes procura-se agrupar estes procedimentos, de acordo com suas características centrais. Esta classificação, porém, não é rígida, já que existe sempre alguma interseção de idéias entre grupos distintos.

Para cada classe de algoritmos, sua característica básica é apresentada e um ou dois representantes são descritos com maiores detalhes. A escolha destes algoritmos teve como base os seguintes critérios: a frequência com que são citados na literatura e a sua utilização como referência no decorrer deste trabalho.

É importante observar em cada caso quais são as hipóteses consideradas. Adiante-se que a maioria dos algoritmos supõe alguma condição de compacidade e que a totalidade deles não trata o caso onde PLDN é ilimitado.

2.2 Enumeração de pontos extremos

De acordo com os teoremas 1.2.3 e 1.2.4, uma solução de PLDN, se existir, está em um ponto extremo de seu conjunto viável V , que é um vértice do conjunto relaxado W . Sendo assim, uma estratégia para se resolver PLDN é pesquisar os vértices de W . Dois procedimentos que empregam esta estratégia são descritos abaixo.

2.2.1 K -ésimo melhor vértice

Bialas e Karwan [24, 26] propõem um algoritmo para resolver PLDNP globalmente, sob as hipóteses de compacidade do conjunto viável relaxado $W = \{(x, y) \geq 0 : A_1x + A_2y \leq a\}$ e de solução única do problema do seguidor

$$P(x) \quad \max_y \{d^T y : A_2y \leq a - A_1x, y \geq 0\}$$

para cada $x \geq 0$ ([26, pág. 1013]). O procedimento, designado *K -ésimo melhor*, consiste em enumerar os vértices de W , conforme a ordenação dada pela função do líder, até obter um ponto extremo do conjunto viável V .

Seja $(x^1, y^1), (x^2, y^2), \dots, (x^p, y^p)$ uma ordenação dos vértices de W , onde

$$f_1(x^1, y^1) \geq f_1(x^2, y^2) \geq \dots \geq f_1(x^p, y^p).$$

À luz do corolário 1.2.2, resolver PLDNP é equivalente a determinar o menor índice k tal que (x^k, y^k) seja viável para PLDNP, ou seja,

$$k = \min \{i \in \{1, 2, \dots, p\} : (x^i, y^i) \in V\}.$$

Assim, deve-se encontrar a k -ésima melhor solução extrema do problema relaxado (PR), que é definido pelo máximo da função do líder f_1 sobre o conjunto W .

É possível mostrar que a k -ésima melhor solução extrema de (PR) é adjacente a algum vértice (x^i, y^i) para $i \in \{1, 2, \dots, k-1\}$ (Shimizu *et al.* [100, proposição 5.3.4]). Usando este resultado, Bialas e Karwan [26] apresentam o seguinte algoritmo, onde T representa o conjunto de vértices a avaliar e \bar{T} , os vértices já avaliados.

Algoritmo (Bialas e Karwan [26])

Passo 1 Resolva o problema relaxado (PR) via método simplex, obtendo uma solução extrema (x^1, y^1) . Faça $i = 1$, $T = \{(x^i, y^i)\}$ e $\bar{T} = \emptyset$. Vá para o passo 2.

Passo 2 Resolva o problema do seguidor $P(x^i)$ via método simplex, obtendo \hat{y}^i . Se $\hat{y}^i = y^i$, então $(x^i, y^i) \in V$ e é solução ótima de PLDN; pare com $k = i$. Senão, vá para o passo 3.

Passo 3 Seja T^i o subconjunto de vértices (x, y) de W , adjacentes a (x^i, y^i) , tais que $f_1(x, y) \leq f_1(x^i, y^i)$. Faça $\bar{T} = \bar{T} \cup \{(x^i, y^i)\}$ e $T = (T \cup T^i) \setminus \bar{T}$. Vá para o passo 4.

Passo 4 Faça $i = i + 1$ e escolha $(x^i, y^i) \in \arg \max\{f_1(x, y) : (x, y) \in T\}$. Volte para o passo 2.

A convergência do algoritmo é clara, pois o número de vértices de W é finito e cada vértice é considerado no máximo uma vez. Sua otimalidade decorre do corolário 1.2.2.

Observe-se que o procedimento acima não considera o caso inviável, embora os autores não assumam uma hipótese garantindo a existência de solução. A inviabilidade de PLDNP seria detectada após a investigação de todos os vértices de W .

Este algoritmo pode ser também aplicado ao problema com restrições no primeiro nível PLDN (Shimizu *et al.* [100, seção 16.3.1]). Mais ainda, a hipótese de solução única do seguidor pode ser descartada. Neste caso, a condição $\hat{y}^i = y^i$ do passo 2 deve ser substituída por $d^T \hat{y}^i = d^T y^i$. Também a hipótese de compacidade de W pode ser relaxada, pedindo-se apenas que (PR) não seja ilimitado.

Poucas experiências computacionais com problemas gerados aleatoriamente são relatadas por Bialas e Karwan [26]. Mesmo assim, os autores observam que o desempenho do algoritmo é fortemente influenciado pela posição relativa da solução de (PR) com respeito à solução de PLDN.

2.2.2 Pesquisa de bases ótimas do seguidor

Candler e Townsley [42] sugerem um algoritmo para encontrar soluções globais de PLDNP, realizando uma enumeração implícita das bases ótimas do problema do seguidor $P(x)$. A hipótese considerada é de que $P(x)$ possua solução única para cada $x \geq 0$ ([42, pág. 61]). Na verdade, nota-se, no desenvolvimento do trabalho, que os autores também assumem PLDNP não ilimitado.

Como Candler e Townsley [42] trabalham com restrições de igualdade, vai-se supor que o problema do segundo nível é

$$P(x) \quad \max_y \{d^T y : A_2 y = a - A_1 x, y \geq 0\}.$$

Sem perda de generalidade, considera-se que a variável y incorpora as variáveis de folga por ventura existentes. Candler e Townsley [42, pág. 62] assumem que A_2 tenha posto cheio.

Por ser um problema linear, $P(x)$ tem solução ótima em um vértice de $W(x) = \{y \geq 0 : A_2 y = a - A_1 x\}$. Assim, algum ótimo de PLDNP, se existir, está associado a uma submatriz básica de A_2 .

Sejam, então, B uma base da matriz A_2 e d_B, y_B as partes de d, y correspondentes a B . Se B é uma base ótima de $P(x)$ para algum x , então

$$d^T - d_B^T B^{-1} A_2 \leq 0. \tag{2.1}$$

Além disso, sendo B uma base viável, deve-se ter

$$y_B = B^{-1}(a - A_1 x) \geq 0. \tag{2.2}$$

Observe-se que apenas a condição de viabilidade (2.2) depende do valor de x , diferentemente da condição de otimalidade (2.1). Assim, dada uma base B satisfazendo (2.1), toda solução do seguinte problema é viável para PLDNP.

$$\begin{aligned} \text{PR}(B) \quad & \max_{x, y_B} \quad c_1^T x + c_{2B}^T y_B \\ & \text{s.a.} \quad A_1 x + B y_B = a \\ & \quad \quad x \geq 0, y_B \geq 0 \end{aligned}$$

Além disso, como já mencionado, pelo menos uma solução ótima de PLDNP, se existir, está associada a uma base B que verifique (2.1). Logo, tal solução é dada por um ótimo de $\text{PR}(B)$.

A partir destas constatações, Candler e Townsley [42] propõem um método que enumera implicitamente todas as bases de A_2 satisfazendo a condição de otimalidade (2.1). Na descrição formal do algoritmo, os autores assumem que PLDNP não seja ilimitado ([42, pág. 71]). Isto garante a mesma condição para $\text{PR}(B)$ para todo B que satisfaça (2.1).

O algoritmo começa com uma base B de $A_2 = [B \ N]$ que satisfaça (2.1). Então, aplica-se o método simplex ao problema linear $\text{PR}(B)$. Na verdade, o quadro simplex é formado com todas as variáveis (x, y) , inclusive as variáveis associadas à matriz N , que são mantidas como não básicas.

Caso $\text{PR}(B)$ tenha solução, a partir dos custos reduzidos do quadro ótimo é possível determinar o conjunto T_1 , de colunas não básicas de A_2 , que podem incrementar o valor corrente da função objetivo ao entrar na base. De forma similar, se $\text{PR}(B)$ é inviável, os custos reduzidos do quadro final da fase I do simplex são usados para encontrar o conjunto T_2 , de colunas não básicas de A_2 , que podem reduzir a inviabilidade. Por outro lado, considerando-se o processo enumerativo, a definição dos novos conjuntos T_1 e T_2 deve levar em conta seus valores anteriores. Desta forma, a enumeração a partir de B fica restrita às bases satisfazendo $T_1 \cap T_2$.

Para verificar se um subconjunto $C \subseteq T_1 \cap T_2$ de colunas não pode fazer parte de uma base satisfazendo (2.1), o algoritmo resolve o dual do problema do seguidor com as restrições relativas às colunas em C forçadas a serem igualdades. Se este problema é inviável, não existe base formada a partir de C .

A escolha do subconjunto $C \subseteq T_1 \cap T_2$ a ser testado é realizada através de um problema inteiro 0-1. Caso C não verifique o teste descrito acima, uma restrição

proibindo a escolha deste conjunto é incluída no problema inteiro 0-1. Uma solução para o novo problema é gerada ou se verifica sua inviabilidade. Este segundo caso implica a não existência de um subconjunto de $T_1 \cap T_2$ que possa gerar uma base, finalizando o processo enumerativo.

Candler e Townsley [42] não reportam experiências computacionais com o algoritmo. Estretanto, Bard [6] verifica que seu desempenho é pior, quando comparado ao procedimento da subseção anterior.

2.3 Complementaridade linear paramétrica

Júdice e Faustino [73] apresentam um procedimento que fornece ε -soluções globais de PLDNP, resolvendo uma sequência de problemas paramétricos de complementaridade linear. Os autores assumem que PLDNP seja não ilimitado, como se percebe adiante na descrição do algoritmo.

Os problemas de complementaridade linear se originam da reformulação (P) , vista na seção 1.3 e reescrita a seguir para o problema sem restrições no primeiro nível.

$$\begin{aligned}
 (P) \quad & \max \quad f_1(x, y) = c_1^T x + c_2^T y \\
 & \text{s.a} \quad w = a - A_1 x - A_2 y \\
 & \quad \quad v = -d + A_2^T u \\
 & \quad \quad x \geq 0, y \geq 0, w \geq 0, u \geq 0, v \geq 0 \\
 & \quad \quad v^T y = u^T w = 0
 \end{aligned}$$

A idéia é substituir a função objetivo de (P) pela restrição

$$c_1^T x + c_2^T y \geq \lambda, \quad (2.3)$$

onde λ é um parâmetro a ser incrementado iterativamente. Assim, gera-se o seguinte problema de complementaridade linear paramétrico:

$$\begin{aligned}
 \text{PCL}(\lambda) \quad & \begin{pmatrix} w \\ v \\ w_0 \end{pmatrix} = \begin{pmatrix} a \\ -d \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \lambda + \begin{pmatrix} 0 & -A_2 & -A_1 \\ A_2^T & 0 & 0 \\ 0 & c_2^T & c_1^T \end{pmatrix} \begin{pmatrix} u \\ y \\ x \end{pmatrix} \\
 & x \geq 0, y \geq 0, w \geq 0, w_0 \geq 0, u \geq 0, v \geq 0, \quad v^T y = u^T w = 0
 \end{aligned}$$

Na verdade, observe-se que este problema é mais geral do que o problema de complementaridade linear padrão, pois não há restrição de complementaridade para algumas variáveis.

Um ótimo global (x^*, y^*) de PLDNP, se existir, é dado pela solução de $\text{PCL}(\lambda^*)$, onde λ^* é o maior valor de λ tal que $\text{PCL}(\lambda)$ tem solução. Para encontrar uma ε -solução global, ou seja, uma solução viável (x, y) satisfazendo

$$0 \leq f_1(x^*, y^*) - f_1(x, y) \leq \varepsilon,$$

o método proposto começa determinando uma solução (x^0, y^0) do problema obtido de $\text{PCL}(\lambda)$ com a retirada da restrição $w_0 = c_1^T x + c_2^T y - \lambda$. Por conveniência, este problema inicial é denotado por $\text{PCL}(\lambda_0)$, onde $\lambda_0 = c_1^T x^0 + c_2^T y^0$. Ele corresponde a encontrar uma solução (x^0, y^0) viável para PLDNP.

A partir de $\text{PCL}(\lambda_0)$, resolve-se uma série de problemas $\text{PCL}(\lambda_k)$, $k \geq 1$, onde $\{\lambda_k\}$ é uma sequência crescente, definida por

$$\lambda_k = f_1(x^{k-1}, y^{k-1}) + \gamma |f_1(x^{k-1}, y^{k-1})|, \quad (2.4)$$

sendo (x^{k-1}, y^{k-1}) uma solução de $\text{PCL}(\lambda_{k-1})$ e γ um escalar positivo.

O método termina quando, em uma iteração k , o problema $\text{PCL}(\lambda_k)$ não tem solução. Quando isto acontece, por (2.3), tem-se que

$$f_1(x^*, y^*) = c_1^T x^* + c_2^T y^* < \lambda_k.$$

Assim, da expressão (2.4), segue-se que

$$0 > f_1(x^*, y^*) - \lambda_k = f_1(x^*, y^*) - f_1(x^{k-1}, y^{k-1}) - \gamma |f_1(x^{k-1}, y^{k-1})|.$$

Portanto,

$$0 \leq f_1(x^*, y^*) - f_1(x^{k-1}, y^{k-1}) < \gamma |f_1(x^{k-1}, y^{k-1})|,$$

implicando que (x^{k-1}, y^{k-1}) é uma ε -solução de PLDNP para

$$\varepsilon = \gamma |f_1(x^{k-1}, y^{k-1})| \geq 0$$

Na prática, tomando-se $\gamma > 0$ bem pequeno, esta solução acaba sendo um ótimo global. Por outro lado, esta opção pode fazer o algoritmo avançar mais lentamente.

Algoritmo (Júdice e Fautino [73])

Passo 0 Faça $k=0$.

Passo 1 Resolva $\text{PCL}(\lambda_k)$. Se não existe solução, vá para o passo 2. Senão, seja

(x^k, y^k) a solução obtida. Faça $\lambda_{k+1} = f_1(x^k, y^k) + \gamma|f_1(x^k, y^k)|$, para $\gamma > 0$ fixo. Faça $k = k + 1$ e repita o passo 1.

Passo 2 Se $k = 0$, PLDNP é inviável. Senão, (x^{k-1}, y^{k-1}) é uma ε -solução de PLDNP, com $\varepsilon = \gamma|f_1(x^{k-1}, y^{k-1})|$.

Como se vê, este algoritmo não considera o caso ilimitado, conforme mencionado no início da seção. Neste caso, o problema $PCL(\lambda)$ teria solução para todo $\lambda \geq \lambda_0$.

A eficiência computacional deste procedimento depende fortemente de como os problemas $PCL(\lambda_k)$ são solucionados. Embora existam algoritmos para resolver o problema de complementaridade linear tradicional (ver Murty [90]), eles não se aplicam diretamente a $PCL(\lambda_k)$. Júdice e Faustino [73] desenvolvem, então, um método enumerativo híbrido para este fim.

Tal processo enumerativo pode ser visualizado através de uma árvore binária, onde cada ramo é obtido fixando-se em zero uma variável de $\nu = (u, v)$ ou de $\omega = (w, y)$. O nó inicial corresponde ao sistema linear obtido do problema $PCL(\lambda_k)$ sem as restrições de complementaridade. Cada nó subsequente está associado ao sistema relativo ao seu nó antecessor, acrescido da restrição $\nu_i = 0$ ou $\omega_i = 0$, conforme seja a variável fixada que lhe originou.

Para se encontrar uma solução desse sistema, considera-se, na verdade, o problema $\min \nu_i$ ou $\min \omega_i$ sujeito ao sistema de equações lineares do nó precedente. Caso o valor ótimo do problema seja positivo, o sistema não tem solução e, por conseguinte, não é preciso ramificar o nó. O processo termina quando esta situação acontece para todos os nós da árvore. Heurísticas baseadas em Al-Khayyal [1] são empregadas em cada nó para reduzir o número de subproblemas gerados.

Júdice e Faustino [73] apresentam testes computacionais usando problemas de pequeno e médio porte com estrutura esparsa. Para os problemas testados, seu procedimento mostra-se mais eficiente que o algoritmo de Bard e Moore [13], descrito na próxima seção.

Vale a pena mencionar que o algoritmo de Júdice e Faustino [73] é uma modificação do algoritmo PCP (*Parametric Complementary Pivot*), proposto por Bialas *et al.* [27, 26] para resolver PLDNP. Na verdade, Ben-Ayed e Blair [20] mostram que o algoritmo PCP não encontra sempre um ótimo global do problema.

2.4 Branch-and-bound

A possibilidade de PLDN possuir ótimos locais sugere a aplicação de algoritmos enumerativos, como visto nas seções anteriores. Alguns desses processos se assemelham a procedimentos tipo *branch-and-bound*, que é um método geral de otimização global (Horst *et al.* [68, seção 3.7]).

Nesse método, o conjunto viável é dividido subsequencialmente em partes cada vez mais refinadas (*branching*). Em cada subconjunto, é calculado um limite superior para o máximo da função objetivo (*bounding*). Caso o limite superior seja menor que um limite inferior disponível, o subconjunto não precisa ser particionado, pois não contém o ótimo do problema (*prunning*).

Um processo *branch-and-bound* pode ser visualizado através de uma árvore de busca. O nó inicial dessa árvore está associado ao problema-alvo, e os descendentes de um nó correspondem a subproblemas definidos pela partição do conjunto viável relativo a esse nó.

Nesta seção, dois algoritmos tipo *branch-and-bound* são descritos. O primeiro deles resolve a reformulação (P), e o segundo trata PLDN diretamente.

2.4.1 Fixação de variáveis complementares

Bard e Moore [13] sugerem um método para resolver globalmente um problema linear-quadrático em dois níveis onde as restrições do líder só dependem da variável x . O algoritmo proposto pode ser aplicado a PLDN (Shimizu *et al.* [100, seção 16.3.2]), sob as mesmas hipóteses originais, quais sejam: compacidade do conjunto viável relaxado e reposta única do seguidor para cada $x \geq 0$ (Bard e Moore [13, pág. 282], Shimizu *et al.* [100, pág. 392]).

O método considera a formulação equivalente a PLDN dada por:

$$(P) \quad \max \quad c_1^T x + c_2^T y \quad (2.5)$$

$$\text{s.a.} \quad B_1 x + B_2 y \leq b \quad (2.6)$$

$$A_1 x + A_2 y + w = a \quad (2.7)$$

$$x \geq 0, y \geq 0, w \geq 0 \quad (2.8)$$

$$A_2^T u - v = d \quad (2.9)$$

$$u \geq 0, v \geq 0 \quad (2.10)$$

$$v^T y = u^T w = 0 \quad (2.11)$$

A idéia básica é tratar as restrições não lineares (2.11) através de um procedimento tipo *branch-and-bound* que enumera implicitamente todas as soluções viáveis complementares.

A estratégia de ramificação (*branching*), bastante simples, consiste em fixar em zero uma das variáveis (u, v) ou sua complementar em (w, y) . Assim, cada subproblema é definido por (P) com algumas variáveis identicamente nulas. Um limite superior para cada subproblema é dado pelo valor ótimo da relaxação linear que desconsidera as restrições de complementaridade (2.11). Se a solução obtida também satisfaz (2.11), ela é viável para PLDN e fornece um limite inferior para o problema.

Na descrição do algoritmo, a seguinte notação é empregada. Por conveniência, define-se $\nu = (u, v)$ e $\omega = (w, y)$, de modo que a variável ν_i seja complementar a ω_i para todo $i \in I = \{1, 2, \dots, m_2 + n_2\}$. O subproblema corrente, onde algumas variáveis ν_i ou ω_i são nulas, é identificado pelos conjuntos

$$I^+ = \{i \in I : \nu_i = 0\}, \quad I^- = \{i \in I : \omega_i = 0\}, \quad I^0 = I \setminus (I^+ \cup I^-).$$

Usa-se também uma lista L dos índices em $I^+ \cup I^-$, ordenada conforme a sequência em que as variáveis foram fixadas. Finalmente, considera-se $F^* = f_1(x^*, y^*)$, onde (x^*, y^*) é a melhor solução encontrada.

Algoritmo (Bard e Moore [13])

Passo 0 (Inicialização) Faça $I^+ = I^- = \emptyset$, $I^0 = I$, $L = \emptyset$ e $F^* = -\infty$.

Passo 1 (Iteração) Considere o problema (2.5)-(2.10) com $\nu_i = 0$ para todo $i \in I^+$ e $\omega_i = 0$ para todo $i \in I^-$. Se este problema é inviável, vá para o passo 5. Senão, seja $(\bar{x}, \bar{\omega}, \bar{\nu})$ uma solução.

Passo 2 (Teste de otimalidade) Se $f_1(\bar{x}, \bar{y}) \leq F^*$, vá para o passo 5.

Passo 3 (Ramificação) Se $\nu_i \omega_i = 0$ para todo $i \in I$, $(\bar{x}, \bar{\omega}, \bar{\nu})$ é viável para (P) : vá para o passo 4. Senão, seja $i_0 \in \arg \max\{\nu_i \omega_i : i \in I^0\}$; faça $I^+ = I^+ \cup \{i_0\}$, $I^0 = I^0 \setminus \{i_0\}$ e acrescente i_0 ao final da lista L ; volte para o passo 1.

Passo 4 (Atualização) Faça $(x^*, y^*) = (\bar{x}, \bar{y})$ e $F^* = f_1(x^*, y^*)$.

Passo 5 (*Backtracking*) Se $L \subseteq I^-$, então todos os subproblemas já foram considerados; vá para o passo 6. Senão, volte ao subproblema mais recente que

ainda deve ser pesquisado, determinando o último elemento i_1 de L tal que $i_1 \in I^+$; seja \bar{L} os elementos de L depois de i_1 ; faça $I^+ = I^+ \setminus \{i_1\}$, $I^- = I^- \cup \{i_1\} \setminus \bar{L}$, $I^0 = I^0 \cup \bar{L}$; volte para o passo 1.

Passo 6 (Término) Se $F^* = -\infty$, então PLDN é inviável. Senão, (x^*, y^*) é solução ótima de PLDN.

Observe-se no passo 3 que a regra usada para ramificação consiste na escolha da máxima folga complementar $\nu_i \omega_i > 0$ e na fixação de $\nu_i = 1$. Note-se também que os elementos são acrescentados no final da lista L . Desta forma, ao se chegar ao passo 5, o último elemento em L que pertence a I^+ está associado ao subproblema mais recente que é viável (passo 1), cujo limite superior é maior que F^* (passo 2) e que ainda viola uma condição de complementaridade (passo 3). Estas observações esclarecem a escolha de i_1 no passo 5.

A compacidade do conjunto viável relaxado, definido por (2.6)-(2.8), é usada para assegurar a boa definição do passo 1. Assim, garante-se também que PLDN não seja ilimitado. Já a hipótese de solução única do problema do seguidor para cada x é necessária na demonstração de que o algoritmo encontra um ótimo global de PLDN (Shimizu *et al.* [100, proposição 16.3.1]).

Outros algoritmos que também utilizam procedimentos tipo *branch-and-bound* para tratar as restrições de complementaridade são apresentados por Fortuny-Amat e McCarl [54] e Bard e Falk [12]. No primeiro trabalho, os autores usam a reformulação de (P) dada pelo problema de programação linear 0-1 mista (PI), descrito na seção 1.3. Neste caso, um procedimento *branch-and-bound* padrão para este tipo de problema é usado.

Já Bard e Falk [12] utilizam o modelo (PS), também visto na seção 1.3, que define um problema de programação não convexa separável. Então, determinam um poliedro contendo a região viável do problema e aplicam um procedimento *branch-and-bound* que subdivide este poliedro em subconjuntos disjuntos. Um limite superior para o problema é dado pela maximização da função do líder sobre cada subconjunto e a escolha do maior valor obtido. Um teste de otimalidade verifica se a solução gerada é um ótimo global para a aproximação linear por partes do problema separável. Se o teste falha, o subconjunto correspondendo ao maior limite superior é subdividido em dois ou três poliedros e o processo continua.

2.4.2 Eliminação de variáveis do seguidor

Hansen *et al.* [60] desenvolvem um algoritmo tipo *branch-and-bound* para resolver globalmente o problema PLDN, cuja formulação é reescrita abaixo.

$$(PLDN) \max_{x,y} f_1(x, y) = c_1^T x + c_2^T y \quad (2.12)$$

$$\text{s.a. } B_1 x + B_2 y \leq b \quad (2.13)$$

$$x \geq 0, y \text{ solução de} \quad (2.14)$$

$$\max_y f_2(x, y) = d^T y \quad (2.15)$$

$$\text{s.a. } A_1 x + A_2 y \leq a \quad (2.16)$$

$$y \geq 0 \quad (2.17)$$

Hansen *et al.* [60, pág. 1195] assumem que o conjunto definido pelas desigualdades em (2.14), (2.16) e (2.17) seja compacto. Esta condição também garante a compacidade do conjunto viável relaxado W , que é definido pelas mesmas restrições juntamente com (2.13). Embora afirmem que esta hipótese é considerada por simplicidade, os autores não apresentam nenhuma indicação de como tratar os casos ilimitados.

A estratégia básica de ramificação (*branching*) consiste na escolha de restrições do seguidor que devem ser tomadas como igualdades. Assim, geram-se subproblemas de PLDN, onde algumas variáveis y_j foram eliminadas. Um limite superior para o valor ótimo de cada subproblema é obtido através da solução do problema relaxado (PR) definindo pelas variáveis restantes.

Para fundamentar sua estratégia de ramificação, Hansen *et al.* derivam condições necessárias de otimalidade, expressas em termos das restrições que devem ser ativas na solução do problema do seguidor para cada x fixo. Neste sentido, para cada uma das $m_2 + n_2$ restrições do seguidor (incluindo as de não negatividade) eles associam uma variável binária α_i , tal que $\alpha_i = 1$, se a i -ésima restrição deve ser ativa, e $\alpha_i = 0$, caso contrário. Então, estabelecem a seguinte propriedade, que é básica para o desenvolvimento do algoritmo.

Teorema 2.4.1 *Toda solução viável para PLDN verifica o seguinte conjunto \mathcal{R} de relações para $j = 1, 2, \dots, n_2$.*

$$(\mathcal{R}) \quad \sum_{i:(A_2)_{ij}>0} \alpha_i \geq 1 \quad \text{se } d_j > 0 \quad (2.18)$$

$$\alpha_{m_2+j} + \sum_{i:(A_2)_{ij}<0} \alpha_i \geq 1 \quad \text{se } d_j < 0 \quad (2.19)$$

No procedimento proposto, a ramificação é feita fixando-se uma ou mais variáveis α_i em 0 ou 1. No caso mais simples, uma única variável é escolhida. Alternativamente, seleciona-se uma relação de (2.18)-(2.19), por exemplo $\alpha_{i_1} + \alpha_{i_2} + \dots + \alpha_{i_3} + \dots \geq 1$. Neste caso, a ramificação é efetuada de acordo com a regra ($\alpha_{i_1} = 1$) ou ($\alpha_{i_1} = 0$ e $\alpha_{i_2} = 1$) ou ($\alpha_{i_1} = \alpha_{i_2} = 0$ e $\alpha_{i_3} = 1$) e assim por diante.

Quando se fixa $\alpha_i = 1$, a i -ésima restrição do seguidor torna-se uma igualdade. Com isto pode-se eliminar uma variável y_j da igualdade obtida. De fato, explicita-se y_j em função das outras variáveis e substitui-se sua expressão nas restrições. Assim, gera-se um subproblema com menos variáveis.

Quando se faz $\alpha_i = 0$, a i -ésima restrição do seguidor no subproblema obtido deve ser uma desigualdade estrita. Como é difícil trabalhar com variáveis estritamente positivas (de folga, no caso), vai-se substituir o problema do seguidor pelo seu dual com a i -ésima variável fixa em zero (isto deve acontecer pelo teorema de folgas complementares).

O conjunto \mathcal{R} de relações também muda, à medida que variáveis α_i são fixadas. Se o subproblema é obtido pela ramificação $\alpha_i = 0$, exclui-se esta variável de todas as expressões (2.18)-(2.19). Caso o subproblema seja gerado por $\alpha_i = 1$, as relações em \mathcal{R} que incluem esta variável são eliminadas, pois as desigualdades são trivialmente satisfeitas. Por outro lado, novas relações podem ser obtidas com a eliminação de variáveis y_j , visto que os coeficientes das restrições se alteram. Finalmente, são descartadas as relações que se tornam redundantes.

Alguns testes são usados no algoritmo para detectar se o subproblema corrente não precisa ser ramificado. Sejam $\overline{\text{PR}}$ e $\overline{P}(x)$, respectivamente, os problemas relaxado e do seguidor, definidos pelas variáveis que restaram no subproblema. Considere-se ainda o conjunto de relações correntes \mathcal{R} . Os testes e as conclusões decorrentes são:

- (i) uma relação $r_k \in \mathcal{R}$ não pode ser satisfeita, o dual de $\overline{P}(x)$ é inviável ou $\overline{\text{PR}}$ é inviável: o subproblema também não possui soluções viáveis. Observe-se, no segundo caso, que a inviabilidade do dual implica a mesma situação no primal, pela hipótese de compacidade considerada.
- (ii) $\overline{\text{PR}}$ não possui variáveis y ou sua solução é viável para PLDN: a solução do

subproblema é igual a de $\overline{\text{PR}}$. Em particular, se esta solução é viável para PLDN, ela fornece um limite inferior para o problema.

- (iii) O valor ótimo \bar{F} da solução de $\overline{\text{PR}}$ é menor que um limite inferior disponível: a solução de $\overline{\text{PR}}$ não pode ser ótima para PLDN.

Na verdade, no teste (iii), outro limite superior para o subproblema, mais forte que \bar{F} , é usado no algoritmo. Este limite é determinado, como se descreve a seguir, através do cálculo de penalidades usuais em programação linear-inteira mista.

Considere-se o problema linear $\overline{\text{PR}}$ com as restrições transformadas em igualdades. Seja z o vetor de variáveis deste problema, ou seja, z é composto por x , as variáveis y_j remanescentes e as variáveis de folga introduzidas. As expressões da função objetivo e das restrições de $\overline{\text{PR}}$, com respeito a sua solução básica ótima \bar{z} , são dadas por:

$$\begin{aligned} f_1(z) &= \bar{F} - \sum_{j \in J_N} \bar{c}_j z_j, \\ z_i &= \bar{z}_i - \sum_{j \in J_N} \bar{A}_{ij} z_j, \quad i \in J_B \end{aligned}$$

onde J_B, J_N são os índices das variáveis básicas e não básicas, $\bar{c} \geq 0$ é o vetor de custos reduzidos e \bar{A} é a matriz de restrições expressa na base atual. Assim, a penalidade para se ter $z_i = 0$ é

$$p_i = z_i^* \min \left\{ \frac{\bar{c}_j}{\bar{A}_{ij}} : j \in J_N, \bar{A}_{ij} > 0 \right\},$$

isto é, $p_i = 0$ (se $i \in J_N$) ou p_i corresponde ao decréscimo em \bar{F} com a retirada de z_i da base (se $i \in J_B$).

Seja, então, $r_k \in \mathcal{R}$ uma relação que deve ser satisfeita pelo subproblema corrente, onde $r_k = \sum_{i \in J_k} \alpha_i \geq 1$. Logo, pelo menos uma variável de folga z_i ($i \in J_k$) deve ser zero. Sendo assim, um limite superior para o valor ótimo do subproblema é dado por

$$\bar{F}' = \bar{F} - \min_{i \in J_k} p_i.$$

Mais ainda, levando-se em consideração todo o conjunto \mathcal{R} de relações que devem ser satisfeitas, obtém-se um limite ainda melhor

$$\bar{F}'' = \bar{F} - \max_{k: r_k \in \mathcal{R}} \min_{i \in J_k} p_i.$$

Após estas explicações, pode-se finalmente descrever o algoritmo de Hansen *et al.* [60]. Destaque-se que, no subproblema corrente, $\overline{\text{PR}}$ e $\bar{P}(x)$ denotam respectivamente o problema relaxado e o problema do seguidor dados pelas variáveis remanescentes.

Algoritmo (Hansen *et al.* [60])

- Passo 1 (Inicialização) Obtenha uma solução inicial (x_h, y_h) viável para PLDN com uma heurística. Inicialize a melhor solução encontrada (x^*, y^*) com (x_h, y_h) e faça $F^* = f_1(x^*, y^*)$. Se nenhuma solução heurística pode ser encontrada, inicialize (x^*, y^*) com um ponto arbitrário e faça $F^* = -\infty$. Considere todas as variáveis α_i livres ($i = 1, 2, \dots, m_2 + n_2$). Faça $\mathcal{R} = \emptyset$ e considere o problema corrente igual a PLDN.
- Passo 2 (Primeiro teste de otimalidade) Resolva $\overline{\text{PR}}$, obtendo (\bar{x}, \bar{y}) . Se $F^* \geq f_1(\bar{x}, \bar{y})$, vá para o passo 13 (*backtracking*).
- Passo 3 (Primeiro teste de viabilidade) Resolva o dual de $\bar{P}(\bar{x})$. Se não existe solução viável, vá para o passo 13.
- Passo 4 (Teste de solução, parte I) Verifique se (\bar{x}, \bar{y}) é viável para o subproblema corrente: resolva $\bar{P}(\bar{x})$, obtendo \tilde{y} ; se $d^T \bar{y} = d^T \tilde{y}$, então (\bar{x}, \bar{y}) é viável; senão vá para o passo 6.
- Passo 5 (Teste de solução, parte II) Verifique se (\bar{x}, \bar{y}) é viável para PLDN: resolva $P(\bar{x})$, obtendo \hat{y} ; se $d^T \bar{y} = d^T \hat{y}$ então (\bar{x}, \bar{y}) é viável; senão vá para o passo 6. Atualize F^* e (x^*, y^*) se $f_1(\bar{x}, \bar{y}) > F^*$; vá para o passo 13.
- Passo 6 (Segundo teste de otimalidade) Calcule todas as penalidades p_i associadas a variáveis de folga estritamente positivas na solução ótima de $\overline{\text{PR}}$. Faça as outras penalidades p_i iguais a zero. Então, para todo k tal que $r_k \in \mathcal{R}$, calcule

$$\pi_k = \min_{i \in J_k} p_i$$

e determine

$$\Pi = \max_k \pi_k.$$

Se $F^* \geq f_1(\bar{x}, \bar{y}) - \Pi$, vá para o passo 13.

- Passo 7 (Segundo teste de viabilidade) Se $\overline{\text{PR}}$ é inviável, vá para o passo 13.
- Passo 8 (Primeiro teste de otimalidade condicional) Para cada i tal que $F^* \geq f_1(\bar{x}, \bar{y}) - p_i$, fixe $\alpha_i = 0$ e atualize \mathcal{R} .
- Passo 9 (Terceiro teste de otimalidade) Se \mathcal{R} contém uma relação r_k tal que $\alpha_j = 0$ para todo $j \in J_k$, vá para o passo 13.
- Passo 10 (Teste de otimalidade relacional) Para cada y_j que ainda aparece em $f_2(x, y)$, inclua em \mathcal{R} relações tipo (2.18)-(2.19), caso sejam não redundantes. Retire de \mathcal{R} todas as relações que se tornarem redundantes.
- Passo 11 (Segundo teste de otimalidade condicional) Se \mathcal{R} contém uma relação r_k tal que $\alpha_j = 0$ para todo $j \in J_k - \{i\}$, faça $\alpha_i = 1$. Elimine uma variável y_j , modificando o subproblema corrente e retorne ao passo 2.
- Passo 12 (Ramificação) Aplique a regra de ramificação selecionada de modo a escolher ou uma variável α_i livre ou uma relação $r_k \in \mathcal{R}$ onde todas as variáveis α_i ($i \in J_k$) estão livres. No primeiro caso, ramifique com $\alpha_i = 1$. No segundo, ramifique fazendo a primeira variável α_i em r_k igual a 1. Elimine uma variável y_j , obtendo um novo subproblema e retorne ao passo 2.
- Passo 13 (*Backtracking*) Se a ramificação ocorreu em uma variável, determine a última variável $\alpha_i = 1$ escolhida para ramificação; faça $\alpha_i = 0$ e torne livres todas as variáveis α_j fixadas em 0 depois que α_i foi fixada em 1. Caso contrário, considere a última relação r_k para a qual menos de $|J_k|$ ramos tenham sido explorados; considere o próximo ramo. Se não existe tal variável ou relação, pare. Senão, atualize o subproblema corrente e retorne ao passo 2.

Como se vê no passo 12, a regra de ramificação não está especificada. Hansen *et al.* [60] testam sete diferentes opções e apresentam resultados computacionais comparativos entre elas.

Vale a pena notar que a hipótese considerada pelos autores é importante para garantir a existência de solução dos problemas $\overline{\text{PR}}$, $\bar{P}(x)$ e $P(x)$ e, conseqüentemente, a boa definição do algoritmo.

2.5 Penalização de folgas complementares

Esta seção apresenta dois algoritmos que pretendem resolver PLDNP globalmente, através da reformulação que penaliza as restrições de folgas complementares. O primeiro algoritmo usa o parâmetro de penalidade implicitamente, e o segundo, de uma forma explícita, como nos métodos usuais de penalidade.

Ambos os procedimentos apresentam problemas, que serão detectados e/ou contornados a partir dos resultados obtidos no capítulo 3. Por este motivo, descrevem-se agora apenas as idéias gerais destes algoritmos e volta-se a analisá-los oportunamente no capítulo 5.

2.5.1 Penalidade implícita

Önal [94] propõe um procedimento para encontrar ε -soluções globais de PLDNP, gerando ótimos locais do problema

$$\begin{aligned}
 (P) \quad & \max \quad c_1^T x + c_2^T y \\
 & \text{s.a} \quad A_1 x + A_2 y + w = a \\
 & \quad \quad x \geq 0, y \geq 0, w \geq 0 \\
 & \quad \quad A_2^T u - v = d \\
 & \quad \quad u \geq 0, v \geq 0 \\
 & \quad \quad v^T y = u^T w = 0
 \end{aligned}$$

Embora o autor não explicita qualquer hipótese para evitar que PLDNP seja ilimitado, a descrição do algoritmo sugere que esta condição é assumida.

Os ótimos locais de (P) são obtidos a partir do problema penalizado

$$P(M) \quad \max \quad F_M(x, y, w, u, v) = c_1^T x + c_2^T y - M(v^T y + u^T w) \quad (2.20)$$

$$\text{s.a} \quad A_1 x + A_2 y + w = a \quad (2.21)$$

$$x \geq 0, y \geq 0, w \geq 0 \quad (2.22)$$

$$A_2^T u - v = d \quad (2.23)$$

$$u \geq 0, v \geq 0 \quad (2.24)$$

que penaliza as condições de complementaridade através de um parâmetro $M \geq 0$.

A seguinte definição é introduzida por Önal [94]:

Definição 2.5.1 *Uma solução local estável do problema penalizado $P(M)$ é uma solução viável tal que (i) maximiza localmente (2.20), e (ii) tanto o ponto solução quanto o valor da função objetivo não se alteram para qualquer $M > M_0$, para algum valor suficientemente grande M_0 .*

Önal [94, proposição 1] mostra que o termo de penalidade se anula em uma solução local estável. Assim, uma solução local estável é um ponto viável para (P) . Mais ainda, trata-se de uma solução local deste problema, como será demonstrado na seção 3.3.

O algoritmo proposto por Önal [94] compõe-se de duas fases. A primeira encontra uma solução local do problema penalizado, considerando o parâmetro M implicitamente como um valor dominante. Isto é feito através de uma adaptação do algoritmo de Beale [18], que determina ótimos locais de problemas quadráticos. A solução local obtida $(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$ é estável e, por conseguinte, um ótimo local de (P) .

A segunda fase aplica o mesmo procedimento ao problema $P'(M)$, definido por $P(M)$ com o corte linear

$$c_1^T x_1 + c_2^T x_2 \geq f_1(\bar{x}, \bar{y}) + \varepsilon, \quad (2.25)$$

onde $\varepsilon > 0$ é uma tolerância aceitável. Se a solução local obtida é estável, o processo é repetido com o corte atualizado. Caso $P'(M)$ seja inviável ou a solução local não anule o termo de penalidade, Önal deduz que não mais existe solução satisfazendo (2.21)-(2.25) e as folgas complementares. Isto o leva a concluir que a última solução viável obtida é ε -ótima para PLDNP. Entretanto, como será mostrado no capítulo 5, esta argumentação não é correta.

O algoritmo de Önal [93] é descrito a seguir.

Algoritmo (Önal [94])

Passo 0 Inicialize $k = 0$ e $F_0^* = -\infty$ (F_k^* é o valor da solução local encontrada na iteração k).

Passo 1 Se (2.21)-(2.22) ou (2.23)-(2.24) são inviáveis, vá para o passo 5. Senão, encontre soluções básicas viáveis $(\bar{x}, \bar{y}, \bar{w})$ e (\bar{u}, \bar{v}) .

Passo 2 Substitua os valores das variáveis básicas na função (2.20), expressando-a em termos do valor corrente $F^0 = F_M(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$ e das variáveis não

básicas (x_N, y_N, w_N) e (u_N, v_N) . Calcule as derivadas parciais da função em relação às variáveis não básicas e depois iguale estas variáveis a zero, obtendo

$$\left(\frac{\partial F_M}{\partial x_N}, \frac{\partial F_M}{\partial y_N}, \frac{\partial F_M}{\partial w_N}\right)\Big|_0 = (P_x, P_y, P_w) \quad \text{e} \quad \left(\frac{\partial F_M}{\partial u_N}, \frac{\partial F_M}{\partial v_N}\right)\Big|_0 = (-M P_u, -M P_v).$$

Se $(P_x, P_y, P_w) \leq 0$ e $(P_u, P_v) \geq 0$, vá para o passo 4. Senão, vá para o passo 3.

Passo 3 Escolha para entrar na base a variável (primal ou dual) com o maior coeficiente positivo em $(P_x, P_y, P_w, -M P_u, -M P_v)$. Determine a variável a sair pelo critério usual do método simplex. Pivoteie e volte ao passo 1.

Passo 4 A solução corrente é localmente ótima. Se F^0 envolve M (solução instável), vá para o passo 5. Senão, atualize $k = k + 1$ e $F_k^* = F^0$. Escolha $\epsilon > 0$ e forme o corte (2.25) com $f_1(\bar{x}, \bar{y}) = F^0$. Defina uma nova variável de folga e acrescente a nova equação ao sistema primal. Volte ao passo 1.

Passo 5 Pare. Se $k = 0$, o problema é inviável. Se $k \geq 1$, a solução obtida na iteração k é ótima.

No capítulo 5 será visto que a conclusão do passo 5 não é correta em geral. Na ocasião, também se analisará o comportamento do algoritmo quando PLDNP for ilimitado.

2.5.2 Penalidade explícita

White e Anandalingam [127] também usam o problema penalizado (2.20)-(2.24) para resolver PLDNP, sem considerar explicitamente as variáveis de folga w e v . Os autores adotam a seguinte formulação:

$$P(M) \quad \max_{(x,y,u)} \quad \hat{F}_M(x, y, u, M) = c_1^T x + c_2^T y - M\pi(x, y, u) \\ \text{s.a.} \quad (x, y) \in W, \quad u \in U$$

onde

$$W = \{(x, y) \geq 0 : A_1 x + A_2 y \leq a\}, \quad U = \{u \geq 0 : A_2^T u \geq d\}$$

e

$$\pi(x, y, u) = (A_2^T u - d)^T y + u^T (a - A_1 x - A_2 y) = u^T (a - A_1 x) - d^T y.$$

Na verdade, $P(M)$ é substituído pelo problema equivalente

$$\max_{u \in U} \Theta(u, M), \quad (2.26)$$

onde

$$\Theta(u, M) = \max_{x, y} \{ \hat{F}(x, y, u, M) : (x, y) \in W \}$$

é uma função convexa em $u \in \mathfrak{R}^{m_2}$, para M fixo.

A boa definição de Θ é garantida pelas seguintes hipóteses consideradas por White e Anandalingam [127]:

[A1] Se x^* é uma solução ótima do líder então o conjunto solução do seguidor $Y(x^*)$ é um único ponto.

[A2] Os poliedros W e U são não vazios e compactos.

Estas hipóteses também são usadas para demonstrar que a penalidade acima é exata. A partir deste resultado, White e Anandalingam [127] propõem um algoritmo para determinar soluções globais de PLDNP, resolvendo uma sequência de problemas $P(M)$, onde o parâmetro M é incrementado iterativamente, até obter uma solução que anule o termo de penalidade.

A convergência e a boa definição deste procedimento são comprometidas pela invalidade da suposição [A2], conforme será visto no capítulo 5. Adiante-se, porém, que tais propriedades serão reestabelecidas a partir dos resultados obtidos no próximo capítulo.

De modo a resolver o problema (2.26), para cada valor fixo de $M \geq 0$, White e Anandalingam [127] usam uma modificação do algoritmo de Tuy [108] para encontrar máximos de funções convexas sobre poliedros. O procedimento utilizado consiste em determinar um ótimo local (um vértice de U melhor que os vértices adjacentes), aplicar cortes que excluam este ótimo e procurar uma nova solução local. Se os cortes tornaram o problema inviável, uma solução de (2.26) para o valor corrente de M foi então localizada. Caso anule o termo de penalidade, esta é uma solução de PLDN. Caso contrário, M é incrementado, e o processo, reiniciado.

Também no capítulo 5, serão identificadas imprecisões na definição dos cortes e na verificação de que a solução de $P(M)$, para M fixo, já foi obtida. Na ocasião, o algoritmo será descrito detalhadamente.

2.6 Restrição convexa reversa

Como visto na seção 1.3, o problema PLDN pode ser reescrito através do seguinte modelo reverso convexo:

$$\begin{aligned}
 (\text{PCR}) \quad & \max \quad f_1(x, y) = c_1^T x + c_2^T y \\
 \text{s.a} \quad & (x, y) \in W \\
 & d^T y - \phi(x) \geq 0
 \end{aligned} \tag{2.27}$$

onde W é o conjunto viável relaxado,

$$W = \{(x, y) \in \mathfrak{R}^{n_1} \times \mathfrak{R}^{n_2} : A_1 x + A_2 y \leq a, \quad B_1 x + B_2 y \leq b, \quad x \geq 0, \quad y \geq 0\},$$

e ϕ é a função marginal dada por

$$\phi(x) = \sup\{d^T y : A_2 y \leq a - A_1 x, \quad y \geq 0\}.$$

Se $\phi(x)$ é finito, pela teoria da dualidade, segue-se que

$$\phi(x) = \min\{(a - A_1 x)^T u : A_2^T u \geq d, \quad u \geq 0\}. \tag{2.28}$$

Então, ϕ é uma função côncava, pois é o mínimo de uma família de funções lineares (Rockafellar [97, teorema 5.5]).

Tuy [109] apresenta duas reformulações deste modelo convexo reverso, que reduz o problema à dimensão $k+1$, onde k é o posto da matriz A_1 . Estas novas formulações são obtidas via projeção ou dualização do modelo original. Em cada caso, é sugerida uma estratégia de decomposição do modelo gerado.

2.6.1 Decomposição primal

Assuma-se que $\text{posto}(A_1) = k$ e seja $E \in \mathfrak{R}^{k \times n_1}$ uma submatriz formada por k vetores linha de A_1 , linearmente independentes. Particione-se $E = [B \quad N]$, onde $B \in \mathfrak{R}^{k \times k}$ é uma base de E , e defina-se $\bar{E}^T = [(B^{-1})^T \quad 0] \in \mathfrak{R}^{k \times n_1}$. Tuy e Ghannadam [110] mostram que (2.27) é equivalente a:

$$\begin{aligned}
 \max_{x, y, t} \quad & f_1(x, y) \\
 \text{s.a} \quad & (x, y) \in W \\
 & Ex = t \\
 & d^T y - \phi_K(\bar{E}t) \geq 0
 \end{aligned} \tag{2.29}$$

onde

$$\phi_K(x) = \min\{(a - A_1x)^T u : A_2^T u \geq d, 0 \leq u_i \leq K, i = 1, 2, \dots, m_2\}.$$

Observe-se que $\phi_K(x)$ é dado pelo problema dual (2.28) com as variáveis limitadas por K . Na verdade, está implícito no desenvolvimento de Tuy e Ghannadam [110] que $\phi(x)$ é finito para todo x viável para (2.29). Isto garante a boa definição de $\phi_K(x)$ e a existência de K tal que $\phi_K(x) = \phi(x)$ [110, proposição 1].

Um algoritmo tipo *branch-and-bound* para resolver (2.29) globalmente é proposto por Tuy e Ghannadam [110]. Na descrição do algoritmo, assume-se que o problema relaxado tenha solução e que o conjunto $\{Ex : (x, y) \in W\}$ seja compacto.

O método começa construindo um simplex $S_0 \subset \mathbb{R}^k$ com dimensão k , tal que $\{Ex \in \mathbb{R}^k : (x, y) \in W\} \subseteq S_0$. Então, o processo de busca segue, particionando S_0 em subsimplex cada vez menores.

Para cada subsimplex S gerado, calcula-se um limite superior $\beta(S)$ para o sub-problema obtido de (2.29) com $t \in S$. Se $S = [s^1, s^2, \dots, s^{k+1}]$, este limite é dado pelo valor ótimo do seguinte problema linear:

$$\begin{aligned} \max_{x, y, \lambda} \quad & f_1(x, y) \\ \text{s.a} \quad & (x, y) \in W \\ & Ex = \sum_{i=1}^{k+1} \lambda_i s^i \\ & \sum_{i=1}^{k+1} \lambda_i = 1, \quad \lambda_i \geq 0 \quad (i = 1, 2, \dots, k+1) \\ & d^T y - \sum_{i=1}^{k+1} \lambda_i \phi_K(\bar{E} s^i) \geq 0 \end{aligned} \tag{2.30}$$

De fato, como ϕ_K é côncava, tem-se que

$$d^T y - \phi_K(\bar{E}t) \leq d^T y - \sum_{i=1}^{k+1} \lambda_i \phi_K(\bar{E} s^i)$$

para todo $t \in S$. Logo, (2.30) é uma relaxação de (2.29) para $t \in S$.

A estratégia de ramificação consiste em particionar o subsimplex S com maior limite superior $\beta(S)$. O modo como se procede a esta subdivisão é detalhado por Tuy e Ghannadam [110]. A seguir descreve-se sucintamente o algoritmo.

Algoritmo (Tuy e Ghannadam [110])

Passo 0 Resolva o problema relaxado (PR), obtendo (x^0, y^0) . Se $d^T y^0 = \phi(x^0)$, pare: (x^0, y^0) é uma solução ótima de PLDN. Senão, construa um simplex S em R^k , contendo a imagem de W no mapeamento $(x, y) \mapsto Ex \in \mathfrak{R}^k$. Seja $\mathcal{P}_1 = \mathcal{R}_1 = \{S\}$. Se existe solução viável disponível $(x^1, y^1) \in V$, faça $F^1 = f_1(x^1, y^1)$; senão $F^1 = -\infty$. Faça $i = 1$.

Passo 1 Para cada simplex $S = [s^1, s^2, \dots, s^{k+1}] \in \mathcal{P}_i$, calcule os valores $\phi_K(\bar{E}s^j)$, $j = 1, 2, \dots, k + 1$, e resolva o problema (2.30), obtendo uma solução básica $(x(S), y(S), \lambda(S))$ com valor ótimo $\beta(S)$. Seja $t(S) = \bar{E}x(S) = \sum_{j=1}^{k+1} \lambda_j(S)s^j$.

Passo 2 Atualize (x^i, y^i) e F^i , comparando a melhor solução disponível com as novas soluções viáveis obtidas para PLDN. Se nenhuma solução viável foi gerada, os valores são conservados.

Passo 3 Retire de \mathcal{R}_i todo simplex S com $\beta(S) \leq F^i$, obtendo \mathcal{R}'_i . Caso $\mathcal{R}'_i = \emptyset$, pare: se $F^i > -\infty$, (x^i, y^i) é solução de PLDN; senão, PLDN é inviável.

Passo 4 Seja $S_i \in \arg \max\{\beta(S) : S \in \mathcal{R}'_i\}$. Subdivide S_i , obtendo a partição \mathcal{P}_{i+1} . Faça $\mathcal{R}_{i+1} = \mathcal{R}'_i \setminus \{S_i\} \cup \mathcal{P}_{i+1}$, $i = i + 1$ e retorne ao passo 1.

Tuy e Ghannadam [110] observam que o procedimento acima pode não terminar em um número finito de passos. Neste caso, mostram que todo ponto de acumulação da sequência gerada $\{(x^i, y^i)\}$ é uma solução de PLDN ([110, teorema 1]). Por outro lado, os autores sugerem um critério de parada que resultaria na obtenção de um ótimo para a ε -aproximação de PLDN gerada quando y não precisa ser uma solução exata, mas uma ε -solução do seguidor.

Aspectos de implementação relacionados à construção do simplex inicial, às regras de subdivisão e à escolha da constante K podem ser encontrados na referência original. Experiências computacionais são reportadas para problemas-teste gerados aleatoriamente, com tamanhos variando de $(n_1, n_2) = (50, 30)$ e $(m_1, m_2) = (10, 2)$ até $(n_1, n_2) = (400, 100)$ e $(m_1, m_2) = (55, 20)$. Em todos os problemas, $n_2 \leq n_1$ e $k = m_2 \ll n_1 + n_2$. Estas características são apropriadas para o bom desempenho do algoritmo.

2.6.2 Decomposição dual

Seja o conjunto convexo

$$C = \{(x, y) : d^T y - \phi(x) < 0\},$$

onde

$$\phi(x) = \sup\{d^T y : A_2 y \leq a - A_1 x, y \geq 0\}.$$

Então, o modelo (2.27) pode ser reescrito como:

$$\text{(PCR)} \quad \max f_1(x, y) \quad \text{s.a.} \quad (x, y) \in W \setminus C. \quad (2.31)$$

Note-se que o conjunto viável relaxado

$$W = \{(x, y) \in \Re^{n_1} \times \Re^{n_2} : A_1 x + A_2 y \leq a, B_1 x + B_2 y \leq b, x \geq 0, y \geq 0\},$$

está contido no fecho de C , ou seja,

$$W \subseteq \text{cl } C = \{(x, y) : d^T y - \phi(x) \leq 0\}. \quad (2.32)$$

A partir da equivalência entre (PCR) e PLDN, conclui-se que (x^*, y^*) é uma ε -solução de PLDN se, e somente se,

$$W_\varepsilon = \{(x, y) \in W : f_1(x, y) \geq f_1(x^*, y^*) + \varepsilon\} \subseteq C. \quad (2.33)$$

A conclusão de que um ponto (x^*, y^*) , viável para (2.31), é uma ε -solução de PLDN pode ser feita através do seguinte problema

$$\max \eta(x, y) = d^T y - \phi(x) \quad \text{s.a.} \quad (x, y) \in W_\varepsilon. \quad (2.34)$$

Quando $W_\varepsilon = \emptyset$, (x^*, y^*) é claramente uma ε -solução de PLDN. Do contrário, o problema (2.34) tem sempre solução, pois $\eta(x, y) \leq 0$ para todo $(x, y) \in W$, conforme (2.32). Em particular, uma solução (\bar{x}, \bar{y}) é um vértice de W , uma vez que η é uma função convexa e W_ε um poliedro. Caso $\eta(\bar{x}, \bar{y}) < 0$, tem-se que $W_\varepsilon \subseteq C$ e, por conseguinte, (x^*, y^*) é uma ε -solução. Senão, $(\bar{x}, \bar{y}) \in W_\varepsilon \setminus C$ é um ponto melhor que (x^*, y^*) .

Explorando estas idéias, Tuy *et al.* [111] sugerem o seguinte algoritmo para encontrar ε -soluções de PLDN, a partir da solução do problema relaxado (PR).

Embora o procedimento possa começar com qualquer ponto de W , a existência de solução para (PR) é condição suficiente para evitar que PLDN seja ilimitado.

Algoritmo (Tuy *et al.* [111])

Passo 0 Resolva o problema relaxado (PR), obtendo (x^0, y^0) . Se $d^T y^0 = \phi(x^0)$, pare: (x^0, y^0) é uma solução ótima de PLDN. Senão, faça $W_1 = W - (x^0, y^0)$, $C = C - (x^0, y^0)$ e $k = 1$.

Passo 1 Através do problema (2.34), encontre $(x^k, y^k) \in W_k \setminus C$ ou mostre que $W_k \subseteq C$. No primeiro caso, defina $W_{k+1} = W_1 \cap \{(x, y) : f_1(x, y) \geq f_1(x^k, y^k) + \varepsilon\}$, para $\varepsilon > 0$; faça $k = k + 1$ e repita o passo 1. No segundo caso, vá para o passo 2.

Passo 2 Se $k = 1$, PLDN é inviável. Senão, (x^{k-1}, y^{k-1}) é ε -solução de PLDN, pois $W_k = W_1 \cap \{(x, y) : f_1(x, y) \geq f_1(x^{k-1}, y^{k-1}) + \varepsilon\} \subseteq C$.

A translação dos conjuntos W e C no passo 0 é importante para iniciar o método usado para resolver o problema posto no passo 1. Note-se que PLDN é igualmente equivalente a (2.31) definido pelos conjuntos transladados. Assim, a condição de otimalidade (2.33) e as conclusões a partir do problema (2.34) continuam válidas após a translação.

A resolução do problema (2.34) no passo 1 é efetuada com um método de otimização global chamado anexação poliédrica (ver Horst e Tuy [69]). Este método consiste em construir uma sequência de poliedros

$$P_1 \subset P_2 \subset P_3 \subset \dots \subset C,$$

que vão se expandindo no interior de C . Para cada poliedro P_i é resolvido um conjunto de problemas lineares (um para cada faceta de P_i), que permitem descobrir se $W_k \subseteq P_i$ ou encontrar um vértice $(x^i, y^i) \in W_k \setminus P_i$. No primeiro caso, o processo pára, pois $W_k \subseteq P_i \subseteq C$. O mesmo acontece se $(x^i, y^i) \in W_k \setminus C$. Finalmente, se $(x^i, y^i) \in W_k \cap C$, forma-se o poliedro $P_{i+1} = \text{co}(P_i \cup \{(x^i, y^i)\})$, e o processo segue ($\text{co}()$ denota a envoltória convexa). Como o número de vértices de W_k é finito, eventualmente um dos dois critérios de parada é satisfeito por um poliedro P_k .

Na verdade, o algoritmo não lida diretamente com o poliedro P_i , mas com seu polar

$$P_i^* = \{(\nu, \omega) : x^T \nu + y^T \omega \leq 1 \quad \forall (x, y) \in P_i\}.$$

Isto porque as facetas de P_i são definidas pelos vértices de P_i^* . Assim, o algoritmo gera de fato uma sequência

$$P_1^* \supset P_2^* \supset P_3^* \supset \cdots \supset C^*,$$

onde $C^* = \{(\nu, \omega) : x^T \nu + y^T \omega \leq 1 \quad \forall (x, y) \in C\}$ é o conjunto polar de C .

Neste sentido, Tuy [109] observa que o algoritmo trabalha no espaço dual do problema (2.31) e realiza uma aproximação externa (*outer approximation*) do conjunto polar C^* . Além disso, Tuy *et al.* [111] demonstram que C^* está contido em um cone de dimensão igual a $\text{posto}(A_1) + 1$. Logo, o algoritmo fica restrito a este espaço de dimensão menor que a original $n_1 + n_2$.

Outro procedimento, que combina *branch-and-bound* e *outer approximation*, é proposto por Tuy *et al.* [112] para resolver PLDN através de uma reformulação de (2.31). O novo modelo corresponde a maximização de uma função quasi-convexa (ver [112, proposição 27]) sobre o conjunto polar C^* de C . A equivalência com PLDN é obtida sob as hipóteses de que o conjunto W é compacto e não vazio e de que a função ϕ é contínua.

Mencione-se que não são apresentados resultados computacionais nem para este segundo algoritmo nem para o primeiro.

2.7 Pontos Interiores

Herskovits e Leontiev [64] consideram problemas de programação não linear em dois níveis, onde o problema do seguidor é convexo para cada x . É sugerido um algoritmo para encontrar soluções viáveis satisfazendo condições de otimalidade de primeira ordem. A abordagem ao problema de dois níveis baseia-se na formulação que substitui o problema do seguidor por suas condições de Karush-Kuhn-Tucker.

Particularmente, no caso linear, tal formulação corresponde ao problema (P) , reescrito a seguir de modo coerente com a notação de Herskovits e Leontiev [64].

$$\begin{aligned}
(P) \quad & \max f_1(x, y) = c_1^T x + c_2^T y \\
& \text{s.a. } B_1 x + B_2 y \leq b \\
& A_1 x + A_2 y \leq a \\
& A_2^T u - v = d \\
& x \geq 0, y \geq 0, u \geq 0, v \geq 0 \\
& u^T(A_1 x + A_2 y - a) = 0 \\
& v^T(-y) = 0
\end{aligned}$$

Notando a não regularidade do problema (P) , os autores apresentam a seguinte formulação alternativa:

$$\begin{aligned}
(P') \quad & \text{Encontre } \nu = (x, y, u, v) \text{ satisfazendo} \\
& \nu \geq 0, \quad A_1 x + A_2 y \leq a, \tag{2.35}
\end{aligned}$$

tal que resolva

$$\begin{aligned}
(R) \quad & \max_{\nu} f(\nu) = f_1(x, y) \\
& \text{s.a. } B_1 x + B_2 y \leq b \\
& A_2^T u - v = d \tag{2.36} \\
& u^T(A_1 x + A_2 y - a) = 0 \\
& v^T(-y) = 0
\end{aligned}$$

onde (R) é agora um problema regular.

Sob a hipótese de que toda solução local de (P) é estritamente complementar, ou seja,

$$v + y > 0, \quad u + (a - A_1 x - A_2 y) > 0 \quad \forall (x, y, v, u) \text{ solução local de } (P),$$

Herskovits *et al.* [65] mostram que (P) e (P') têm os mesmos ótimos locais. Assim, uma solução de PLDN é dada por uma solução de (R) satisfazendo (2.35).

Herkovits e Leontiev [64] sugerem um algoritmo para determinar pontos que verifiquem as restrições (2.35) e as condições de primeira ordem de Karush-Kuhn-Tucker (KKT) do problema (R) . O algoritmo é baseado em um método de pontos interiores proposto por Herskovits [63], descrito resumidamente a seguir.

Seguindo-se a notação adotada por Herskovits [63], denote-se por $g(\nu) \leq 0$ e $h(\nu) = 0$ as restrições de desigualdade e igualdade em (R) e sejam $\lambda \geq 0$ e μ os multiplicadores de Lagrange associados a estas restrições respectivamente. Defina-se

o conjunto

$$\Delta = \{\nu : g(\nu) < 0, h(\nu) < 0\}.$$

Dados um ponto inicial $\nu^0 \in \Delta$ e multiplicadores $\lambda^0 > 0$ e μ^0 , o algoritmo de Herkovits [63, pág. 140] gera uma sequência de pontos $\nu^k \in \Delta$ (as igualdades são satisfeitas apenas no limite) e multiplicadores $\lambda^k > 0$, μ^k que convergem para pontos verificando as condições de KKT.

A partir do ponto corrente $(\nu^k, \lambda^k, \mu^k)$, determina-se uma direção de descida primal d^k , viável para Δ , e realiza-se uma busca linear inexata para se obter o novo ponto. A direção d^k é obtida através de uma deflexão da direção de descida d_1 , que é determinada pela solução do sistema

$$\begin{aligned} B d_1 + \nabla g(\nu^k) \lambda_1 + \nabla h(\nu^k) \mu_1 &= -\nabla f(\nu^k) \\ \Lambda \nabla g^T(\nu^k) d_1 + G(\nu^k) \lambda_1 &= 0 \\ \nabla h^T(\nu^k) d_1 &= -h(\nu^k) \end{aligned}$$

onde B é uma matriz simétrica e definida positiva e Λ e $G(\nu^k)$, matrizes diagonais com $\Lambda_{ii} = \lambda_i^k$ e $G_{ii}(\nu^k) = g_i(x)$. Em particular, B pode ser a hessiana da função lagrangeana em $(\nu^k, \lambda^k, \mu^k)$ ou uma aproximação dessa matriz ou mesmo a matriz identidade. No primeiro caso, o sistema acima corresponde a uma iteração de Newton para o sistema KKT associado a (R) .

A deflexão de d_1 é gerada a partir da solução d_2 de um segundo sistema, dado por

$$\begin{aligned} B d_2 + \nabla g(\nu^k) \lambda_2 + \nabla h(\nu^k) \mu_2 &= 0 \\ \Lambda \nabla g^T(\nu^k) d_2 + G(\nu^k) \lambda_2 &= -\Lambda \omega^i \\ \nabla h^T(\nu^k) d_2 &= -\omega^e \end{aligned}$$

onde $\omega^i > 0$ e $\omega^e > 0$ são parâmetros. A direção d^k é então calculada como

$$d^k = d_1 + \rho d_2,$$

sendo o escalar $\rho > 0$ escolhido de forma a se ter uma direção de descida.

Uma busca linear sobre d^k fornece o novo ponto $\nu^{k+1} = \nu^k + t d^k$, onde a determinação de $t > 0$ garante que $\nu^{k+1} \in \Delta$. A atualização dos parâmetros $\omega^i > 0$ e $\omega^e > 0$, dos multiplicadores λ e μ e da matriz B pode ser feita de diversas formas.

Em particular, se B não é a hessiana da função lagrangeana, os multiplicadores μ nem são usados explicitamente e, por isso, não precisam ser atualizados.

A modificação introduzida no algoritmo para tratar o problema (P') ocorre na busca linear. Ela é realizada de modo a assegurar que toda a sequência satisfaça (2.35). Para tal, o ponto ν^0 deve verificar essas restrições.

Além de condições técnicas (Herskovits [63, pág. 129]), as seguintes hipóteses são assumidas para mostrar a convergência do algoritmo:

1. Existe $k \in \Re$ tal que o conjunto $\{\nu : g(\nu) \leq 0, h(\nu) \leq 0, f(\nu) \leq k\}$ é compacto e tem interior não vazio.
2. Os pontos estacionários de (R) são isolados ou constituem conjuntos compactos com as mesmas restrições ativas.
3. A sequência gerada pertence a um conjunto compacto.

Sob as condições acima, Herskovits [63] mostra que todo ponto de acumulação da sequência gerada pelo algoritmo é um ponto KKT para o problema (R) . Recorde-se ainda que as restrições (2.35) são verificadas pela sequência.

Experiências computacionais com o algoritmo aplicado a problemas em dois níveis são mencionadas por Herskovits e Leontiev [64].

Capítulo 3

Análise teórica do problema

Neste capítulo concentramos nossa atenção no problema linear em dois níveis sem restrições no líder (PLDNP). A seção 3.1 apresenta nossa abordagem ao problema e introduz a notação a ser utilizada. Ainda nesta primeira seção, derivamos uma propriedade básica que será usada no desenvolvimento teórico das seções seguintes e na definição dos algoritmos propostos no próximo capítulo.

Na seção 3.2 caracterizamos as situações de existência e inexistência de soluções globais para PLDNP, através de um problema penalizado auxiliar. Introduzimos o conceito de ponto de equilíbrio do problema penalizado na seção 3.3, de modo a estudar condições de otimalidade local para PLDNP. Na subseção 3.3.1 derivamos caracterizações de soluções locais e, na subseção 3.3.2, de soluções locais estritas. Na subseção 3.3.3, obtemos ainda condições necessárias e/ou suficientes de otimalidade para um ponto de equilíbrio, que são mais atrativas do ponto de vista computacional.

Finalmente, na seção 3.4 estendemos para PLDN, quando possível, os resultados obtidos para PLDNP. Os casos onde a generalização não acontece são ilustrados através de exemplos.

3.1 Preliminares

Recorde-se que o problema PLDNP é formulado como

$$\begin{aligned} \text{(PLDNP)} \quad & \max_{x,y} f_1(x,y) = c_1^T x + c_2^T y \\ \text{s.a} \quad & x \geq 0, y \text{ solução de} \\ & \max_y f_2(x,y) = d^T y \\ \text{s.a} \quad & A_1 x + A_2 y \leq a \\ & y \geq 0 \end{aligned}$$

onde $c_1, x \in \mathfrak{R}^{n_1}$, $c_2, d, y \in \mathfrak{R}^{n_2}$, $a \in \mathfrak{R}^{m_2}$, $A_1 \in \mathfrak{R}^{m_2 \times n_1}$ e $A_2 \in \mathfrak{R}^{m_2 \times n_2}$.

Este problema têm sido extensivamente estudado na literatura, como mostram as revisões bibliográficas de Wen e Hsu [120] e Vicente e Calamai [116]. Entre os vários trabalhos podem-se citar Candler e Townsley [42], Bard [6, 10], Bialas e Karwan [26], Dempe [48], Júdice e Faustino [73], Ben-Ayed [19], White e Anandalingam [127], Önal [94], Liu e Hart [79], Amouzegar e Moshirvaziri [2].

Embora existam muitos trabalhos sobre este problema, a quase totalidade deles considera hipóteses de compacidade ou de, pelo menos, existência de solução. Aqui, conseguimos derivar resultados teóricos que permitem caracterizar tanto as situações de existência de solução quanto as de inviabilidade e ilimitação de PLDNP. Estes resultados são também importantes do ponto de vista prático, na medida em que podem ser usados para desenvolver métodos de solução para o problema.

Nossa abordagem consiste em examinar PLDNP através da análise de dois problemas relacionados, que aparecem na literatura. O primeiro deles é obtido pela substituição do problema do segundo nível por suas condições de otimalidade de Karush-Kuhn-Tucker, resultando na seguinte formulação apresentada na seção 1.3:

$$(P) \quad \max \quad c_1^T x + c_2^T y \quad (3.1)$$

$$\text{s.a} \quad A_1 x + A_2 y + w = a, \quad (3.2)$$

$$x \geq 0, y \geq 0, w \geq 0, \quad (3.3)$$

$$A_2^T u - v = b, \quad (3.4)$$

$$u \geq 0, v \geq 0, \quad (3.5)$$

$$u^T w = 0, v^T y = 0 \quad (3.6)$$

onde $u \in \mathfrak{R}^{m_2}$ é a variável dual e $w \in \mathfrak{R}^{m_2}$ e $v \in \mathfrak{R}^{n_2}$ são as variáveis de folgas primais e duais respectivamente.

A utilização do problema (P) em lugar de PLDNP aparece, por exemplo, em Bard [10], Júdice e Faustino [73], Önal [94] e White e Anandalingam [127]. Como foi visto na seção 1.3, uma solução global de (P) , se existir, está ligada a uma solução global de PLDNP, e vice-versa. Mais ainda, notamos que os casos de inviabilidade e ilimitação também correspondem. Neste sentido, dizemos que (P) e PLDNP são problemas globalmente equivalentes.

Ressalvamos, todavia, que nem todas as soluções locais de (P) estão associadas a soluções locais de PLDNP, conforme mostraremos adiante. Esta observação é

importante, dado que muitos algoritmos propostos encontram ótimos locais de (P) e não verificam se a mesma propriedade é válida em relação ao problema original.

A dificuldade em se tratar o problema (P) resulta da não convexidade do seu conjunto viável, gerada pelas restrições de folgas complementares (3.6). Como visto na seção 1.3, uma abordagem alternativa para (P) consiste na penalização de tais restrições não lineares e sua introdução na função objetivo (3.1). Desta forma obtém-se o segundo problema auxiliar:

$$P(M) \quad \max \quad c_1^T x + c_2^T y - M(u^T w + v^T y) \quad (3.7)$$

$$\text{s.a} \quad A_1 x + A_2 y + w = a, \quad (3.8)$$

$$x \geq 0, y \geq 0, w \geq 0, \quad (3.9)$$

$$A_2^T u - v = b, \quad (3.10)$$

$$u \geq 0, v \geq 0 \quad (3.11)$$

onde $M \geq 0$ é o parâmetro de penalidade. Observe-se que $P(M)$ é um problema bilinear para cada M fixo.

Esta função de penalidade têm sido considerada na literatura sob hipóteses de compacidade que garantem, entre outras propriedades, a existência de solução de $P(M)$ para todo $M \geq 0$. Por exemplo, Bard [9], Önal [94] e Amouzegar e Moshirvaziri [2] consideram, direta ou indiretamente, que as restrições (3.8)-(3.9) definem um conjunto não vazio e compacto. Já Anandalingam e White [3, 127] assumem esta mesma condição também para o conjunto definido por (3.10)-(3.11).

Sob a primeira hipótese, Bard [9] mostra que uma solução extrema de $P(M^*)$ é um vértice ótimo para (P) e vice-versa, para algum $M^* \geq 0$. Em [35] Campêlo *et al.* relaxam a condição de compacidade e provam que é suficiente considerar (3.1) limitada superiormente sobre (3.8)-(3.9).

Aqui, desconsideramos qualquer suposição sobre o conjunto viável de $P(M)$ e mostramos que esta penalidade é também útil em situações de inviabilidade e ilimitação. Mais ainda, usamos este problema penalizado para caracterizar as soluções locais de PLDNP. Indicamos também que o parâmetro de penalidade M pode ser considerado de forma implícita, conforme sugerido por Önal [94]. Esta característica é atrativa do ponto de vista computacional.

No intuito de simplificar a notação utilizada, considerem-se as matrizes e vetor

em blocos

$$A = [A_1 \ A_2 \ I_{m_2}] \in \mathfrak{R}^{m_2 \times n}, \quad D = [0 \ -I_{n_2} \ A_2^T] \in \mathfrak{R}^{n_2 \times n} \quad \text{e} \quad c^T = (c_1^T, c_2^T, 0) \in \mathfrak{R}^n,$$

onde $n = n_1 + n_2 + m_2$, I_k é a matriz identidade de ordem $(k \times k)$ e 0 é uma matriz nula de dimensão apropriada para cada caso. Definam-se também os seguintes conjuntos:

$$\begin{aligned} Z &= \{z \in \mathfrak{R}^n : Az = a, z^T = (x^T, y^T, w^T) \geq 0\} \\ S &= \{s \in \mathfrak{R}^n : Ds = d, s^T = (0, v^T, u^T) \geq 0\}. \end{aligned}$$

Note-se que Z corresponde ao conjunto viável relaxado com a introdução das variáveis de folga e que S é o conjunto dual do seguidor imerso em \mathfrak{R}^n .

Com esta notação, os problemas auxiliares são reescritos como:

$$\begin{aligned} (P) \quad \max \quad & F(z, s) = c^T z \\ \text{s.a.} \quad & z \in Z, s \in S \\ & s^T z = 0 \end{aligned} \qquad \begin{aligned} P(M) \quad \max \quad & F_M(z, s) = c^T z - M s^T z \\ \text{s.a.} \quad & z \in Z, s \in S \end{aligned}$$

Denote-se por X_v o conjunto de vértices de um poliedro X e por $V[\mathcal{P}]$ o valor ótimo de um problema de maximização (\mathcal{P}) , ressalvando-se que se considera $V[\mathcal{P}]$ igual a $-\infty$ ou $+\infty$ quando (\mathcal{P}) é inviável ou ilimitado respectivamente.

Por último, recorde-se que a norma infinito de um vetor $\nu = (\nu_1, \nu_2, \dots, \nu_p)$ é $\|\nu\|_\infty = \max\{|\nu_i| : 1 \leq i \leq p\}$ e denote-se por $B_\varepsilon(\bar{\nu}) = \{\nu \in \mathfrak{R}^p : \|\nu - \bar{\nu}\|_\infty \leq \varepsilon\}$ a ε -vizinhança de $\bar{\nu} \in \mathfrak{R}^p$ nesta norma.

Estabelecemos agora um resultado que mostra um caminho de como encontrar soluções viáveis de (P) , a partir de pontos viáveis do problema relaxado ou do problema dual do seguidor.

Lema 3.1.1 *Se os conjuntos Z e S são não vazios, as seguintes propriedades análogas se verificam:*

(i) *Se $z \in Z$ então existe $(\hat{z}, \hat{s}) \in Z \times S_v$ viável para o problema (P) satisfazendo:*

(i.1) $\hat{s} \in \arg \min\{z^T s : s \in S\},$

(i.2) $\hat{s} \in S_v$, associado a uma base E de D , e

(i.3) $\hat{z}^T = z^T - z_E^T E^{-1} D$, onde z_E é parte de z relativa a E .

(ii) *Se $s \in S$ então existe $(\hat{z}, \hat{s}) \in Z_v \times S$ viável para o problema (P) satisfazendo:*

$$(ii.1) \hat{z} \in \arg \min\{s^T z : z \in Z\},$$

(ii.2) $\hat{z} \in Z_v$, associado a uma base B de A , e

(ii.3) $\hat{s}^T = (0, \hat{v}^T, \hat{u}^T)$, onde $(\hat{r}^T, \hat{v}^T, \hat{u}^T) = s^T - s_B^T B^{-1} A$ e s_B é parte de s relativa a B .

Prova:

(i) Seja $z \in Z$. Posto que $S \neq \emptyset$ e $z^T s \geq 0$ para todo $s \in S$, o problema linear $\min\{z^T s : s \in S\}$ tem solução. Mais ainda, existe

$$\hat{s} \in \arg \min\{z^T s : s \in S_v\} \subseteq \arg \min\{z^T s : s \in S\}. \quad (3.12)$$

Seja E uma base relativa a $\hat{s}^T = ((E^{-1}d)^T, 0)$. Particionem-se $D = [E \ R]$ e $z^T = (z_E^T, z_R^T)$. Considere-se o vetor de custos reduzidos não negativos associados a (3.12), ou seja,

$$\hat{z}^T = (0, z_R^T - z_E^T E^{-1} R) = z^T - z_E^T E^{-1} D \geq 0.$$

Como $DA^T = [0 \ -I_{n_2} \ A_2^T][A_1 \ A_2 \ I_{m_2}]^T = 0$, obtém-se que $A\hat{z} = Az = a$. Logo, $\hat{z} \in Z$. Finalmente, pelas expressões particionadas de \hat{z} e \hat{s} , conclui-se que $\hat{s}^T \hat{z} = 0$. Portanto, (\hat{z}, \hat{s}) é viável para (P) .

(ii) Seja $s \in S$. Por raciocínio análogo ao aplicado ao item (i), conclui-se que existe $\hat{z} \in \arg \min\{s^T z : z \in Z\}$, $\hat{z} \in Z_v$ e associado a uma base B . Considerem-se $A = [B \ N]$, $s^T = (s_B^T, s_N^T)$ e o vetor de custos reduzidos

$$\hat{s}_o^T = (0, s_N^T - s_B^T B^{-1} N) = s^T - s_B^T B^{-1} A \geq 0.$$

Como $AD^T = 0$ e $\hat{z}^T = ((B^{-1}a)^T, 0)$, deduz-se da expressão acima que

$$D\hat{s}_o = Ds = d \quad \text{e} \quad \hat{s}_o^T \hat{z} = 0. \quad (3.13)$$

Particione-se $\hat{s}_o^T = (\hat{r}^T, \hat{v}^T, \hat{u}^T)$, de acordo com a estrutura de D , e seja $\hat{s}^T = (0, \hat{v}^T, \hat{u}^T)$. Então, por (3.13) e levando-se em conta que $\hat{s}_o \geq 0$, $D = [0 \ -I_{n_2} \ A_2^T]$ e $\hat{z} \geq 0$, segue-se que

$$\hat{s} \geq 0, \quad D\hat{s} = D\hat{s}_o = d \quad \text{e} \quad \hat{s}^T \hat{z} = s_o^T \hat{z} = 0.$$

Por conseguinte, $\hat{s} \in S$ e (\hat{z}, \hat{s}) é viável para (P) . ■

A propriedade acima será usada no desenvolvimento teórico que se segue: a parte (ii) na análise global e a (i), na análise local. Mais ainda, o enfoque algorítmico que ela enseja será empregado no capítulo 4. Neste caso, usa-se, na verdade, o seguinte corolário.

Corolário 3.1.1 *Se os conjuntos Z e S são não vazios, as seguintes propriedades análogas se verificam:*

(i) *Se $z \in Z$ e $\hat{s} \in \arg \min\{s^T z : s \in S_v\}$ então existe $\hat{z} \in Z$ tal que (\hat{z}, \hat{s}) é viável para o problema (P) .*

(ii) *Se $s \in S$ e $\hat{z} \in \arg \min\{z^T s : z \in Z_v\}$ então existe $\hat{s} \in S$ tal que (\hat{z}, \hat{s}) é viável para o problema (P) .*

Prova: Na demonstração do lema 3.1.1, nota-se que \hat{s} é um elemento arbitrário de $\arg \min\{z^T s : s \in S_v\}$ na expressão (3.12). Logo, segue-se o item (i) do corolário. Analogamente, conclui-se que o item (ii) se verifica. ■

3.2 Análise global

Nesta seção caracterizamos os casos de existência e inexistência de solução global para o problema PLDNP, através da análise do problema penalizado. Mais precisamente, demonstramos a ocorrência das seguintes possibilidades: ou ambos os problemas (P) e $P(M)$ são simultaneamente inviáveis ou ilimitados para todo $M \geq 0$, ou existe $M^* \geq 0$ tal que eles têm o mesmo conjunto de soluções globais para todo $M > M^*$. O resultado análogo, com respeito a PLDNP, é obtido diretamente da equivalência global entre este problema e (P) .

Para examinar as várias situações, mostramos primeiramente que a penalidade que estamos usando é exata para uma formulação mais geral que (P) . Consideramos os problemas abaixo:

$$\begin{array}{ll}
 (\mathcal{P}) \quad \max & \mathcal{F}(z, s) \\
 \text{s.a} & z \in \mathcal{Z}, s \in \mathcal{S} \\
 & s^T z = 0
 \end{array}
 \qquad
 \begin{array}{ll}
 \mathcal{P}(M) \quad \max & \mathcal{F}_M(z, s) = \mathcal{F}(z, s) - M s^T z \\
 \text{s.a} & z \in \mathcal{Z}, s \in \mathcal{S}
 \end{array}$$

onde $\mathcal{Z}, \mathcal{S} \subseteq \mathbb{R}_+^n$ são poliedros quaisquer, $\mathcal{F} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função linear e $M \geq 0$ é o parâmetro de penalidade. A introdução deste problema mais geral (\mathcal{P}) e

de seu correspondente problema penalizado é conveniente para o nosso desenvolvimento teórico.

Obtemos a seguinte propriedade para estes problemas:

Lema 3.2.1 *Se o problema (\mathcal{P}) é viável e o problema $\mathcal{P}(M_0)$ tem solução para algum $M_0 \geq 0$, então existe $M^* \geq M_0$ tal que:*

(1) *Existe $(z^*, s^*) \in \mathcal{Z}_v \times \mathcal{S}_v$ satisfazendo*

$$(s^*)^T z^* = 0 \quad e \quad V[\mathcal{P}] = V[\mathcal{P}(M)] = \mathcal{F}(z^*, s^*) \quad \forall M \geq M^*.$$

(2) *Mais ainda, (\mathcal{P}) e $\mathcal{P}(M)$ têm o mesmo conjunto (não vazio) de soluções globais para todo $M > M^*$.*

Prova:

(1) Pelas hipóteses consideradas, $\mathcal{P}(M)$ é viável para todo $M \in \mathfrak{R}$. Por outro lado, dado que $V[\mathcal{P}(M_0)] < +\infty$ para algum $M_0 \geq 0$ e $s^T z \geq 0$ para todo $(z, s) \in \mathcal{Z} \times \mathcal{S}$, segue-se que

$$-\infty < V[\mathcal{P}(M)] \leq V[\mathcal{P}(M_0)] < +\infty \quad \forall M \geq M_0.$$

Logo, $\mathcal{P}(M)$ tem solução para todo $M \geq M_0$. Mais ainda, pelo menos uma solução é atingida em um vértice, uma vez que $\mathcal{P}(M)$ é um problema bilinear para cada M fixo (ver Horst e Tuy [69, proposição IX.1]). Então, obtém-se que

$$V[\mathcal{P}(M)] = \max\{\mathcal{F}(z, s) - M s^T z : z \in \mathcal{Z}_v, s \in \mathcal{S}_v\} \quad \forall M \geq M_0. \quad (3.14)$$

Seja o conjunto $\mathcal{Z}_v \times \mathcal{S}_v$ particionado nos subconjuntos finitos

$$C_0 = \{(z, s) \in \mathcal{Z}_v \times \mathcal{S}_v : z^T s = 0\} \quad e \quad C_1 = (\mathcal{Z}_v \times \mathcal{S}_v) \setminus C_0. \quad (3.15)$$

Como (\mathcal{P}) é viável, deve-se ter $C_0 \neq \emptyset$. De fato, suponha, por contradição, que $C_0 = \emptyset$ e seja $(\bar{z}, \bar{s}) \in \mathcal{Z} \times \mathcal{S}$ com $\bar{s}^T \bar{z} = 0$. Então, para todo $M > \max\{(\mathcal{F}(z, s) - \mathcal{F}(\bar{z}, \bar{s}))/s^T z : z \in \mathcal{Z}_v, s \in \mathcal{S}_v\}$, com $M \geq M_0$, tem-se

$$\max\{\mathcal{F}(z, s) - M s^T z : z \in \mathcal{Z}_v, s \in \mathcal{S}_v\} < \mathcal{F}(\bar{z}, \bar{s}) \leq V[\mathcal{P}(M)],$$

o que contradiz (3.14). Assim, $C_0 \neq \emptyset$. Seja, então, $(z^*, s^*) \in C_0$ satisfazendo

$$\mathcal{F}(z^*, s^*) = \max\{\mathcal{F}(z, s) : (z, s) \in C_0\}. \quad (3.16)$$

Se $C_1 \neq \emptyset$, tome-se

$$M^* = \max \left\{ M_0, \max \{ (\mathcal{F}(z, s) - \mathcal{F}(z^*, s^*)) / s^T z : (z, s) \in C_1 \} \right\}. \quad (3.17)$$

Portanto, para cada $M \geq M^*$, segue-se que

$$M \geq (\mathcal{F}(z, s) - \mathcal{F}(z^*, s^*)) / s^T z \quad \forall (z, s) \in C_1.$$

Assim,

$$L(M) = \max \{ \mathcal{F}(z, s) - M s^T z : (z, s) \in C_1 \} \leq \mathcal{F}(z^*, s^*) \quad \forall M \geq M^*. \quad (3.18)$$

Usando-se (3.14)-(3.18) e levando-se em conta que $M^* \geq M_0$, obtém-se que

$$V[\mathcal{P}(M)] = \max \{ \mathcal{F}(z^*, s^*), L(M) \} = \mathcal{F}(z^*, s^*) \quad \forall M \geq M^*. \quad (3.19)$$

Se $C_1 = \emptyset$, trivialmente tem-se $V[\mathcal{P}(M)] = \mathcal{F}(z^*, s^*)$ para todo $M \geq M^* = M_0$.

Finalmente, como $(s^*)^T z^* = 0$ por (3.16), conclui-se que

$$\mathcal{F}(z^*, s^*) \leq V[\mathcal{P}] \leq V[\mathcal{P}(M)] = \mathcal{F}(z^*, s^*) \quad \forall M \geq M^*, \quad (3.20)$$

o que prova a primeira parte.

(2) Seja $\bar{M} > M^*$. Pelo resultado acima, tem-se

$$-\infty < V[\mathcal{P}] = V[\mathcal{P}(\bar{M})] = V[\mathcal{P}(M^*)] < +\infty. \quad (3.21)$$

Denotem-se por $O[\mathcal{P}] = \arg \max(\mathcal{P})$ e $O[\mathcal{P}(\bar{M})] = \arg \max(\mathcal{P}(\bar{M}))$ os conjuntos soluções de (\mathcal{P}) e $\mathcal{P}(\bar{M})$. Por (3.21), verifica-se que

$$O[\mathcal{P}] \neq \emptyset \quad \text{e} \quad O[\mathcal{P}(\bar{M})] \neq \emptyset. \quad (3.22)$$

Além disso, o fato de toda solução viável para (\mathcal{P}) ser viável para $\mathcal{P}(\bar{M})$ juntamente com (3.21) garantem que

$$O[\mathcal{P}] \subseteq O[\mathcal{P}(\bar{M})]. \quad (3.23)$$

Seja, agora, $(z, s) \in O[\mathcal{P}(\bar{M})]$. Como $\bar{M} > M^*$, segue-se novamente de (3.21) que

$$V[\mathcal{P}] = V[\mathcal{P}(\bar{M})] = \mathcal{F}(z, s) - \bar{M} s^T z \leq \mathcal{F}(z, s) - M^* s^T z \leq V[\mathcal{P}(M^*)] = V[\mathcal{P}].$$

Assim, as relações acima são satisfeitas na igualdade. Isto implica $(\bar{M} - M^*)s^T z = 0$ e, conseqüentemente, $s^T z = 0$. Portanto, (z, s) é viável para (\mathcal{P}) e $V[\mathcal{P}] = \mathcal{F}(z, s)$, isto é, $(z, s) \in O[\mathcal{P}]$. Logo,

$$O[\mathcal{P}(\bar{M})] \subseteq O[\mathcal{P}]. \quad (3.24)$$

Como $\bar{M} > M^*$ é arbitrário, a segunda parte segue de (3.22)-(3.24). ■

Note-se que a definição de M^* coincide com a expressão dada por Bard [10], quando $C_1 \neq \emptyset$ e $\mathcal{P}(M)$ tem solução para todo M . Adicionalmente, observe-se que o segundo item do lema não se verifica para $M = M^*$. Realmente, quando M^* é dado pelo segundo termo de (3.17), segue-se de (3.18)-(3.19) que $V[\mathcal{P}(M^*)] = L(M^*) = \mathcal{F}(z, s)$ para algum $(z, s) \in C_1$. Então, (z, s) é ótimo para $\mathcal{P}(M^*)$, mas inviável para (\mathcal{P}) .

Está claro que o lema 3.2.1 abrange os problemas (P) e $P(M)$, uma vez que são instâncias especiais de (\mathcal{P}) e $\mathcal{P}(M)$, onde $Z = Z$, $S = S$ e $\mathcal{F}(z, s) = c^T z$. Assim, obtemos diretamente a primeira propriedade relacionando os problemas auxiliares.

Proposição 3.2.1 *Se o problema (P) é viável e o problema $P(M_0)$ tem solução para algum $M_0 \geq 0$, então existe $M^* \geq M_0$ tal que (P) e $P(M)$ têm o mesmo conjunto (não vazio) de soluções globais para todo $M > M^*$.*

Vale a pena ressaltar que as hipóteses da proposição acima são mais fracas que aquelas consideradas por Bard [9] e Campêlo *et al.* [35]. Observe-se também que a implicação inversa a esta proposição se verifica trivialmente.

Agora, direcionamos nossa atenção às situações que a proposição 3.2.1 não abrange, ou seja, ilimitação e inviabilidade. Começamos estudando o caso ilimitado, a partir da introdução do seguinte problema que envolve direções de recessão do conjunto primal definido por (3.2)-(3.3):

$$\begin{aligned} P_r(M) \quad \max \quad & F_M(h, s) = c^T h - M s^T h \\ \text{s.a} \quad & h \in Z_r, s \in S \end{aligned}$$

onde $Z_r = \{h \in \mathbb{R}^n : Ah = 0, h \geq 0, \|h\|_\infty \leq 1\}$ é o cone de recessão de Z (ver Rockafellar [97, pág. 61]) restrito ao hipercubo unitário.

Estabelecemos, preliminarmente, duas propriedades que relacionam $P(M)$ e $P_r(M)$.

Lema 3.2.2 *Seja $\bar{M} \geq 0$. O problema $P(\bar{M})$ é ilimitado se, e somente se, o conjunto Z é não vazio e $V[P_r(\bar{M})] > 0$.*

Prova: Seja $\bar{M} \geq 0$. Assuma-se que $P(\bar{M})$ é ilimitado. Então, $Z \neq \emptyset$ e $S \neq \emptyset$. Mais ainda, do teorema de representação poliédrica (Bazaraa *at al.* [17, teorema 2.6.7]), deduz-se que $Z = \{z + \alpha h : z \in \text{co}(Z_v), h \in Z_r, \alpha \geq 0\}$, onde $\text{co}(\cdot)$ denota a envoltória convexa. Assim, segue-se que

$$\begin{aligned} +\infty &= V[P(\bar{M})] = \sup\{c^T \tilde{z} - \bar{M} s^T \tilde{z} : \tilde{z} \in Z, s \in S\} \\ &= \sup\{c^T z - \bar{M} s^T z + \alpha(c - \bar{M} s)^T h : z \in \text{co}(Z_v), h \in Z_r, \alpha \geq 0, s \in S\}. \end{aligned} \quad (3.25)$$

Por contradição, suponha-se que $V[P_r(\bar{M})] \leq 0$. Logo, $(c - \bar{M} s)^T h \leq 0$ para todo $(h, s) \in Z_r \times S$. E como $s^T z \geq 0$ para todo $(z, s) \in \text{co}(Z_v) \times S$, a expressão (3.25) resulta em

$$+\infty = V[P(\bar{M})] \leq \sup\{c^T z : z \in \text{co}(Z_v)\},$$

o que é uma contradição, pois $\text{co}(Z_v)$ é um conjunto compacto. Portanto, deve-se ter $V[P_r(\bar{M})] > 0$.

Reciprocamente, assumamos que $Z \neq \emptyset$ e $V[P_r(\bar{M})] > 0$. Então, existem $\bar{z} \in Z$ e $(\bar{h}, \bar{s}) \in Z_r \times S$ com $(c - \bar{M} \bar{s})^T \bar{h} > 0$. Além disso, como $\bar{z} + \alpha \bar{h} \in Z$ para todo $\alpha \geq 0$, obtém-se que

$$V[P(\bar{M})] \geq \sup\{c^T \bar{z} - \bar{M} \bar{s}^T \bar{z} + \alpha(c - \bar{M} \bar{s})^T \bar{h} : \alpha \geq 0\} = +\infty.$$

Assim, $P(\bar{M})$ é ilimitado. ■

Lema 3.2.3 *O problema $P(M)$ é ilimitado para todo $M \geq 0$ se, e somente se, o conjunto Z é não vazio e existe $(h^*, s^*) \in (Z_r)_v \times S_v$ tal que $(s^*)^T h^* = 0$ e $c^T h^* > 0$.*

Prova: Seja $P(M)$ ilimitado para todo $M \geq 0$. Então, tem-se $Z \neq \emptyset$ e $S \neq \emptyset$. Além disso, como $0 \in Z_r$, o problema (P_r) dado por $\max\{c^T h : h \in Z_r, s \in S, s^T h = 0\}$ é viável. Por outro lado, sendo Z_r um conjunto compacto, verifica-se que $V[P_r(0)] = \max\{c^T h : h \in Z_r\} < +\infty$. Portanto, aplicando-se o lema 3.2.1 a (P_r) e $P_r(M)$, assegura-se que existem $M^* \geq 0$ e $(h^*, s^*) \in (Z_r)_v \times S_v$ tais que

$$(s^*)^T h^* = 0 \quad \text{e} \quad V[P_r(M^*)] = c^T h^*.$$

Como $P(M^*)$ é ilimitado por hipótese, segue-se do lema 3.2.2 que

$$V[P_r(M^*)] = c^T h^* > 0.$$

Reciprocamente, assumamos que $Z \neq \emptyset$ e seja $(h^*, s^*) \in (Z_r)_v \times S_v$ com $(s^*)^T h^* = 0$ e $c^T h^* > 0$. Então,

$$V[P_r(M)] \geq c^T h^* - M(s^*)^T h^* = c^T h^* > 0$$

para todo $M \geq 0$. Do lema 3.2.2 resulta que $P(M)$ é ilimitado para todo $M \geq 0$. ■

Agora podemos apresentar a seguinte equivalência:

Proposição 3.2.2 *O problema (P) é ilimitado se, e somente se, o problema $P(M)$ é ilimitado para todo $M \geq 0$.*

Prova: Se (P) é ilimitado, $P(M)$ também é ilimitado para todo $M \geq 0$, pois $V[P(M)] \geq V[P] = +\infty$ para todo $M \geq 0$. Por outro lado, seja $P(M)$ ilimitado para todo $M \geq 0$. Pelo lema 3.2.3, $Z \neq \emptyset$ e existe $(h^*, s^*) \in (Z_r)_v \times S_v$ tal que

$$(s^*)^T h^* = 0 \quad \text{e} \quad c^T h^* > 0. \quad (3.26)$$

Dado que $Z \neq \emptyset$ e $s^* \in S$, o lema 3.1.1 garante a existência de $(\hat{z}, \hat{s}) \in Z_v \times S$ com

$$\hat{z}^T \hat{s} = 0, \quad (3.27)$$

onde

$$\hat{s}^T = (0, \hat{v}^T, \hat{u}^T) \quad \text{e} \quad \hat{s}_o^T = (\hat{r}^T, \hat{v}^T, \hat{u}^T) = (s^*)^T - (s_B^*)^T B^{-1} A \geq 0, \quad (3.28)$$

sendo B uma base relativa a \hat{z} . Por outro lado, como $h \in (Z_r)_v$, tem-se

$$h^* \geq 0 \quad \text{e} \quad Ah^* = 0. \quad (3.29)$$

Então,

$$\hat{z} + \alpha h^* \in Z \quad \forall \alpha \geq 0. \quad (3.30)$$

Mais ainda, por (3.28), (3.29) e (3.26), segue-se que

$$0 \leq \hat{s}^T h^* \leq \hat{s}_o^T h^* = (s^*)^T h^* = 0. \quad (3.31)$$

Logo, por (3.27), (3.30), (3.31) e dado que $\hat{s} \in S$ deduz-se que $(\hat{z} + \alpha h^*, \hat{s})$ é viável para (P) para todo $\alpha \geq 0$. Então, a partir de (3.26), resulta que

$$V[P] \geq \sup\{c^T \hat{z} + \alpha c^T h^* : \alpha \geq 0\} = +\infty,$$

ou seja, (P) é ilimitado. ■

Para o caso inviável, obtemos a seguinte propriedade:

Proposição 3.2.3 *Seja $\bar{M} \geq 0$. O problema (P) é inviável se, e somente se, o problema $P(\bar{M})$ é inviável.*

Prova: Seja $\bar{M} \geq 0$. Se $P(\bar{M})$ é inviável, claramente (P) é inviável, pois a região viável deste segundo problema está contida no conjunto viável do primeiro. Assuma-se agora que (P) é inviável. Suponha-se, por absurdo, que $P(\bar{M})$ é viável. Então, $Z \neq \emptyset$ e $S \neq \emptyset$. Pelo lema 3.1.1, existe (\hat{z}, \hat{s}) viável para (P) , contradizendo a hipótese. ■

Note-se que o problema $P(\bar{M})$ é inviável para algum \bar{M} se, e somente se, $P(M)$ é inviável para todo M . Então, sumarizando as relações entre os problemas auxiliares dadas pelas proposições 3.2.1-3.2.3, temos que:

Teorema 3.2.1 *Exatamente um dos seguintes casos acontece:*

- (1) os problemas (P) e $P(M)$ são inviáveis para todo $M \geq 0$;
- (2) os problemas (P) e $P(M)$ são ilimitados para todo $M \geq 0$;
- (3) os problemas (P) e $P(M)$ têm o mesmo conjunto (não vazio) de soluções globais para todo $M > M^*$, para algum $M^* \geq 0$.

Corolário 3.2.1 *Existe $M^* \geq 0$ tal que os problemas (P) e $P(M)$ têm o mesmo conjunto de soluções globais para todo $M > M^*$.*

Finalmente, expressamos o resultado do teorema 3.2.1 em termos do problema original.

Teorema 3.2.2 *Os problemas PLDNP e $P(M)$ são ambos inviáveis ou ilimitados para todo $M \geq 0$, ou então eles têm solução para algum $M_0 \geq 0$. Neste último caso, z^* é uma solução global de PLDNP se, e somente se, existe s^* tal que (z^*, s^*) é solução global de $P(M)$ para todo $M > M^*$, para algum $M^* \geq M_0$.*

Na subseção 3.3.2 mostraremos que um ponto z^* é uma solução global estrita de PLDNP se, e somente se, existe s^* tal que (z^*, s^*) é solução global estrita de $P(M)$ para todo $M > M^*$, para algum $M^* \geq 0$.

3.3 Análise local

Recorde-se que as regiões viáveis de PLDNP e (P) e a função objetivo de $P(M)$ não são convexas. Logo, estes problemas podem ter ótimos locais. Na próxima subseção apresentamos relações teóricas entre as soluções locais destes problemas, que são também interessantes do ponto de vista numérico. Os resultados obtidos são reexaminados na subseção 3.3.2, de modo a derivar condições de otimalidade estrita. Visando mais ainda o aspecto computacional, derivamos outras condições de otimalidade na subseção 3.3.3.

3.3.1 Otimalidade local

Para nosso desenvolvimento, será útil considerar as funções ponto-conjunto

$$S(z) = \{s \in S : z^T s = 0\} \quad \text{e} \quad Z(s) = \{z \in Z : s^T z = 0\},$$

que levam um ponto $z \in Z \subset \mathbb{R}^n$ ($s \in S \subset \mathbb{R}^n$) em um poliedro $S(z) \subset S$ ($Z(s) \subset Z$).

Note-se que a região viável do problema (P) é dada igualmente pelo gráfico das aplicações $S(\cdot)$ e $Z(\cdot)$. Com efeito,

$$\{(z, s) \in Z \times S : s^T z = 0\} = \{(z, s) : z \in Z, s \in S(z)\} = \{(z, s) : s \in S, z \in Z(s)\}.$$

Já o conjunto viável de PLDNP é o domínio da aplicação $S(\cdot)$, ou seja,

$$\text{dom } S(\cdot) = \{z \in Z : S(z) \neq \emptyset\}.$$

Observe-se também que $\text{dom } S(\cdot) \neq \emptyset$ se, e somente se, $\text{dom } Z(\cdot) \neq \emptyset$.

Particularmente, quando z é um ponto viável para PLDNP, obtemos a seguinte caracterização do conjunto $S(z)$.

Lema 3.3.1 *Se z é viável para o problema PLDNP então $S(z)$ é uma face não vazia de S e seu conjunto de vértices, também não vazio, é $S_v(z) = S(z) \cap S_v$.*

Prova: Seja z um ponto viável para PLDNP. Então, $S(z) \neq \emptyset$. Como $z \geq 0$ e $s \geq 0$ para todo $s \in S$, $S(z)$ é o conjunto solução do problema linear $\min\{z^T s : s \in S\}$ e, por conseguinte, uma face não vazia de S . Assim, o conjunto de vértices de $S(z)$ é também não vazio e composto por vértices de S . Esta constatação implica a segunda parte do enunciado. ■

A seguir estabelecemos dois resultados auxiliares que serão usados para derivar relações locais entre os problemas PLDNP e (P) .

Lema 3.3.2 *Verificam-se as seguintes propriedades:*

- (1) *Se $\bar{z} \in Z$ então existe $\varepsilon > 0$ tal que $S(z) \subseteq S(\bar{z})$ para todo $z \in Z \cap B_\varepsilon(\bar{z})$.*
- (2) *Se $\bar{s} \in S$ então existe $\varepsilon > 0$ tal que $Z(s) \subseteq Z(\bar{s})$ para todo $s \in S \cap B_\varepsilon(\bar{s})$.*
- (3) *Se $(\bar{z}, \bar{s}) \in Z \times S$ então existe $\varepsilon > 0$ tal que $S(z) \subseteq S(\bar{z})$ e $Z(s) \subseteq Z(\bar{s})$ para todo $(z, s) \in (Z \times S) \cap B_\varepsilon(\bar{z}, \bar{s})$, onde $B_\varepsilon(\bar{z}, \bar{s}) = B_\varepsilon(\bar{z}) \times B_\varepsilon(\bar{s})$.*

Prova:

(1) Seja $\bar{z} \in Z$. Considerem-se dois casos: $\bar{z} = 0$ e $\bar{z} \neq 0$. Se $\bar{z} = 0$, então $S(\bar{z}) = S$. Portanto, como $S(z) \subseteq S$ para todo $z \in Z$, tem-se que $S(z) \subseteq S(\bar{z})$ para todo $z \in Z \cap B_{\varepsilon_1}(\bar{z})$ e todo $\varepsilon_1 > 0$. Se $\bar{z} \neq 0$, particione-se o conjunto de índices $J = \{1, 2, \dots, p\}$ nos subconjuntos

$$J_0 = \{j \in J : \bar{z}_j = 0\} \quad \text{e} \quad J_1 = J \setminus J_0 = \{j \in J : \bar{z}_j > 0\}.$$

Primeiro vai-se provar que existe uma vizinhança $B_{\varepsilon_1}(\bar{z})$ tal que, se $z \in Z \cap B_{\varepsilon_1}(\bar{z})$, então $z_j > 0$ para todo $j \in J_1$. Com efeito, posto que $\bar{z} \neq 0$, deve-se ter $J_1 \neq \emptyset$. Para cada $j \in J_1$, defina-se $C_j = \{z \in \mathfrak{R}^n : z_j = 0\}$. Assim, o conjunto C_j é fechado, convexo e não-vazio ($0 \in C_j$). Como $\bar{z} \notin C_j$, de acordo com Rockafellar [97, corolário 11.4.2 e teorema 11.4], existe um hiperplano separando \bar{z} e C_j fortemente e, portanto,

$$d(\bar{z}, C_j) = \min\{\|z - \bar{z}\|_\infty : z \in C_j\} > 0.$$

Consequentemente, existe $\varepsilon_1 \in \mathfrak{R}$ tal que

$$0 < \varepsilon_1 < \min\{d(\bar{z}, C_j) : j \in J_1\}.$$

Seja $z \in Z \cap B_{\varepsilon_1}(\bar{z}) \neq \emptyset$. Então, $z \notin C_j$ para todo $j \in J_1$. Assim, $z_j > 0$ para todo $j \in J_1$, como desejado. Mostra-se agora a inclusão $S(z) \subseteq S(\bar{z})$. Novamente, dois casos são considerados. Se $S(z) = \emptyset$, trivialmente $S(z) \subseteq S(\bar{z})$ (pode acontecer $S(\bar{z}) = \emptyset$). Do contrário, seja $s \in S(z)$. Como $s^T z = 0$, deve-se ter $s_j = 0$ para todo $j \in J_1$. Por conseguinte,

$$\bar{z}^T s = \sum_{j \in J_1} \bar{z}_j s_j + \sum_{j \in J_0} \bar{z}_j s_j = 0.$$

Logo, $s \in S(\bar{z})$, implicando $S(z) \subseteq S(\bar{z})$. Assim, existe $\varepsilon_1 > 0$ tal que

$$S(z) \subseteq S(\bar{z}) \quad \forall z \in Z \cap B_{\varepsilon_1}(\bar{z}). \quad (3.32)$$

(2) Permutando-se \bar{z} com \bar{s} e Z com S na argumentação acima, mostra-se analogamente que existe $\varepsilon_2 > 0$ tal que

$$Z(s) \subseteq Z(\bar{s}) \quad \forall s \in S \cap B_{\varepsilon_2}(\bar{s}). \quad (3.33)$$

(3) Para demonstrar este último item, basta tomar $\varepsilon \in \mathfrak{R}$ tal que $0 < \varepsilon < \min\{\varepsilon_1, \varepsilon_2\}$. O resultado segue-se de (3.32)-(3.33). ■

Corolário 3.3.1 *Seja $(\bar{z}, \bar{s}) \in Z \times S$. Existe $\varepsilon > 0$ tal que todo ponto $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$, viável para o problema (P), satisfaz $s^T z = \bar{s}^T z = \bar{z}^T s = 0$, isto é, (z, \bar{s}) e (\bar{z}, s) também são viáveis para (P).*

Prova: Seja $(\bar{z}, \bar{s}) \in Z \times S$. Pelo lema 3.3.2, existe $\varepsilon > 0$ tal que

$$S(z) \subseteq S(\bar{z}) \quad \text{e} \quad Z(s) \subseteq Z(\bar{s}) \quad \forall (z, s) \in (Z \times S) \cap B_\varepsilon(\bar{z}, \bar{s}). \quad (3.34)$$

Seja $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$, viável para (P). Então, $s \in S(z)$ e $z \in Z(s)$. Além disso, como $(z, s) \in (Z \times S) \cap B_\varepsilon(\bar{z}, \bar{s})$, segue-se de (3.34) que $s \in S(\bar{z})$ e $z \in Z(\bar{s})$. Logo, $\bar{z}^T s = \bar{s}^T z = 0$. ■

Podemos agora caracterizar as soluções locais do problema PLDNP em função das soluções locais de (P).

Teorema 3.3.1 *Um ponto \bar{z} é uma solução local do problema PLDNP se, e somente se, \bar{z} é viável para PLDNP e (\bar{z}, s) é uma solução local do problema (P) para todo vértice $s \in S_v(\bar{z})$.*

Prova: Seja $\bar{z} \in Z$ uma solução local de PLDNP e $B_\varepsilon(\bar{z})$ uma ε -vizinhança onde \bar{z} é globalmente ótima. Então,

$$c^T z \leq c^T \bar{z} \quad \forall z \in B_\varepsilon(\bar{z}) \text{ com } S(z) \neq \emptyset. \quad (3.35)$$

Como \bar{z} é viável para PLDNP, pelo lema 3.3.1, tem-se $S_v(\bar{z}) \neq \emptyset$. Seja, então, $\bar{s} \in S_v(\bar{z})$. Logo, (\bar{z}, \bar{s}) é viável para (P) . Considere-se $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$, viável para (P) . Assim, $z \in B_\varepsilon(\bar{z})$ e $S(z) \neq \emptyset$. Logo, por (3.35), deve-se ter $c^T z \leq c^T \bar{z}$. Portanto, (\bar{z}, \bar{s}) é uma solução local de (P) . Como \bar{s} é um elemento arbitrário em $S_v(\bar{z})$, conclui-se que (\bar{z}, s) é uma solução local de (P) para todo $s \in S_v(\bar{z})$.

Reciprocamente, seja $\bar{z} \in Z$ com $S_v(\bar{z}) \neq \emptyset$ e (\bar{z}, s) solução local de (P) para todo $s \in S_v(\bar{z})$. Pelo lema 3.3.2, existe $\bar{\varepsilon} > 0$ tal que $S(z) \subseteq S(\bar{z})$ para todo $z \in Z \cap B_{\bar{\varepsilon}}(\bar{z})$. Assim,

$$S_v(z) = S(z) \cap S_v \subseteq S(\bar{z}) \cap S_v = S_v(\bar{z}) \quad \forall z \in Z \cap B_{\bar{\varepsilon}}(\bar{z}). \quad (3.36)$$

Suponha-se, por contradição, que \bar{z} não é solução local de PLDNP. Dado que \bar{z} é viável para PLDNP, existe outro ponto \hat{z} , viável para o problema, tal que

$$\hat{z} \in Z \cap B_{\bar{\varepsilon}}(\bar{z}) \quad \text{e} \quad c^T \hat{z} > c^T \bar{z}. \quad (3.37)$$

Pelo lema 3.3.1, existe $\hat{s} \in S_v(\hat{z})$. E, por (3.36)-(3.37), $\hat{s} \in S_v(\bar{z})$. Defina-se $z(\lambda) = \lambda \hat{z} + (1 - \lambda)\bar{z}$. Como \hat{z} e \bar{z} pertencem ao poliedro $Z(\hat{s})$, então $z(\lambda) \in Z(\hat{s})$ para todo $\lambda \in [0, 1]$. Logo, $(z(\lambda), \hat{s})$ é viável para (P) para todo $\lambda \in [0, 1]$. Mais ainda, por (3.37), obtém-se que

$$c^T z(\lambda) = \lambda c^T \hat{z} + (1 - \lambda)c^T \bar{z} = c^T \bar{z} + \lambda(c^T \hat{z} - c^T \bar{z}) > c^T \bar{z} \quad (3.38)$$

para todo $\lambda \in (0, 1]$. Como $z(\lambda) \rightarrow \bar{z}$ à medida que $\lambda \rightarrow 0$, tem-se que, para cada $\varepsilon > 0$, existe $\lambda \in (0, 1]$ tal que $(z(\lambda), \hat{s}) \in B_\varepsilon(\bar{z}, \hat{s})$ e é um ponto viável para (P) satisfazendo (3.38). Portanto, existe $\hat{s} \in S_v(\bar{z})$ tal que (\bar{z}, \hat{s}) não é solução local de (P) , o que contradiz a hipótese. Logo, \bar{z} deve ser uma solução local de PLDNP. ■

Esclarecemos que um ponto \bar{z} pode não ser uma solução local do problema PLDNP se (\bar{z}, \bar{s}) é uma solução local de (P) apenas para algum $\bar{s} \in S_v(\bar{z})$. Esta situação é ilustrada pelo exemplo (E1), apresentado na figura 3.1. O conjunto viável de PLDNP é mostrado em negrito. Então, está claro que o ponto A é a solução global

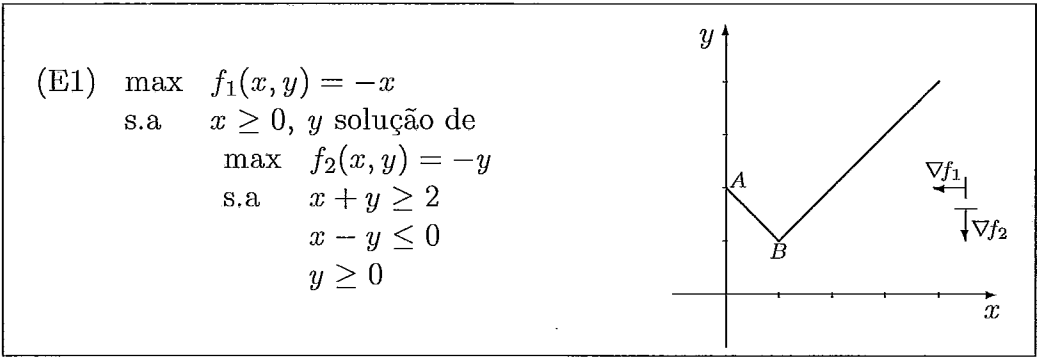


Figura 3.1:

e o ponto B , correspondendo a $\bar{z} = (\bar{x}, \bar{y}, \bar{w}_1, \bar{w}_2)^T = (1, 1, 0, 0)^T \in Z$, não é solução local de PLDNP. Entretanto, para $\bar{s} = (0, \bar{v}, \bar{u}_1, \bar{u}_2)^T = (0, 0, 0, 1)^T \in S_v(\bar{z})$, o ponto (\bar{z}, \bar{s}) é uma solução local de (P) . De fato, pelo corolário 3.3.1, qualquer ponto viável (z, s) em uma certa vizinhança de (\bar{z}, \bar{s}) deve satisfazer $\bar{s}^T z = 0$. Então, considerando-se $z = (x, y, w_1, w_2)^T$, deve-se ter $w_2 = 0$, ou seja, $x = y$. E como $x + y \geq 2$, segue-se que $x \geq 1$. Assim, $F(z, s) = f_1(x, y) = -x \leq -1 = F(\bar{z}, \bar{s})$. Portanto, (\bar{z}, \bar{s}) é uma solução local de (P) . Por outro lado, existe outro ponto $\hat{s} = (0, 0, 1, 0)^T \in S_v(\bar{z})$ tal que (\bar{z}, \hat{s}) não é solução local de (P) .

O teorema 3.3.1 sugere a busca de soluções locais do problema PLDNP entre soluções locais de (P) . Na verdade, estes pontos podem ser caracterizados através do problema penalizado, o que é computacionalmente atrativo.

De modo a estabelecer relações entre ótimos locais dos problemas (P) e $P(M)$, introduzimos o seguintes conceito.

Definição 3.3.1 Um ponto (\bar{z}, \bar{s}) é um **ponto de equilíbrio** do problema penalizado $P(M)$ se existe $\bar{M} \geq 0$ tal que, para cada $M \geq \bar{M}$, verifica-se

$$\max_{s \in S} F_M(\bar{z}, s) = F_M(\bar{z}, \bar{s}) = \max_{z \in Z} F_M(z, \bar{s}), \quad (3.39)$$

onde $F_M(z, s) = c^T z - M s^T z$.

A noção de equilíbrio requerida pela definição acima estende aquela usada pelo algoritmo *mountain climbing* (Konno [75]) para um problema bilinear. Realmente, em nosso caso, a igualdade (3.39) deve ser satisfeita para todo $M \geq \bar{M}$, significando que o equilíbrio tem de ser verificado por uma família de problemas bilineares para-

métricos $P(M)$. Esta característica gera a seguinte propriedade relacionada ao termo de complementaridade da função de penalidade.

Lema 3.3.3 *Se (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $P(M)$, então*

$$\min_{s \in S} \bar{z}^T s = \bar{s}^T \bar{z} = \min_{z \in Z} \bar{s}^T z. \quad (3.40)$$

Prova: Seja (\bar{z}, \bar{s}) um ponto de equilíbrio do problema penalizado. Por (3.39), pode-se tomar $M > \bar{M} \geq 0$ para o qual

$$c^T \bar{z} - M \bar{z}^T \bar{s} = \max\{c^T \bar{z} - M \bar{z}^T s : s \in S\}.$$

Então, deve-se ter

$$\bar{z}^T \bar{s} = \min\{\bar{z}^T s : s \in S\},$$

o que prova a igualdade da esquerda. Novamente por (3.39), tem-se que

$$c^T \bar{z} - M \bar{s}^T \bar{z} = \max\{c^T z - M \bar{s}^T z : z \in Z\} \quad \forall M \geq \bar{M}. \quad (3.41)$$

Suponha-se, por contradição, que $\bar{s}^T \bar{z} > \bar{s}^T \hat{z} = \min\{\bar{s}^T z : z \in Z\}$. Então, $\bar{s}^T(\bar{z} - \hat{z}) > 0$. Tome-se $M > \max\{\bar{M}, c^T(\bar{z} - \hat{z})/\bar{s}^T(\bar{z} - \hat{z})\}$. Assim,

$$c^T \hat{z} - M \bar{s}^T \hat{z} > c^T \bar{z} - M \bar{s}^T \bar{z},$$

contradizendo (3.41), pois $M > \bar{M}$. Portanto,

$$\bar{s}^T \bar{z} = \min\{\bar{s}^T z : z \in Z\},$$

e a demonstração está completa. ■

Note-se que a recíproca do lema acima não é verdadeira. De fato, podem existir $\bar{z}, \hat{z} \in Z$ e $\bar{s} \in S$ com $\bar{s}^T \bar{z} = \bar{s}^T \hat{z} = 0$ e $c^T \hat{z} > c^T \bar{z}$. Neste caso, (\bar{z}, \bar{s}) verifica a condição (3.40), mas não é ponto de equilíbrio.

Agora mostramos que um ponto de equilíbrio satisfaz a condição de complementaridade. Esta propriedade é essencial para caracterizar pontos de equilíbrio como soluções locais do problema (P) .

Proposição 3.3.1 *Se (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $P(M)$, então $\bar{s}^T \bar{z} = 0$.*

Prova: Seja (\bar{z}, \bar{s}) um ponto de equilíbrio do problema penalizado. Pelo lema 3.3.3, tem-se que

$$\min\{\bar{z}^T s : s \in S\} = \bar{s}^T \bar{z} = \min\{\bar{s}^T z : z \in Z\}. \quad (3.42)$$

Posto que $S \neq \emptyset$ e $\bar{z} \in Z$, pelo lema 3.1.1 existe $(\hat{z}, \hat{s}) \in Z \times S_v$ com

$$\bar{z}^T \hat{s} = \min\{\bar{z}^T s : s \in S\} \quad \text{e} \quad \hat{z}^T = \bar{z}^T - \bar{z}_E^T E^{-1} D, \quad (3.43)$$

onde E é um base relativa a $\hat{s} = ((E^{-1}d)^T, 0)^T$. Pelas duas primeiras equações em (3.42) e (3.43) e a expressão de \hat{s} segue-se que

$$\bar{z}^T \bar{s} = \bar{z}^T \hat{s} = \bar{z}_E^T E^{-1} d. \quad (3.44)$$

Então, usando-se o fato de que $\hat{z} \in Z$ na igualdade da direita em (3.42) e considerando-se a expressão de \hat{z} , (3.44) e $D\bar{s} = d$, obtém-se que

$$0 \leq \bar{z}^T \bar{s} \leq \hat{z}^T \bar{s} = \bar{z}^T \bar{s} - \bar{z}_E^T E^{-1} D \bar{s} = \bar{z}^T \bar{s} - \bar{z}_E^T E^{-1} d = 0.$$

Portanto, $\bar{s}^T \bar{z} = 0$. ■

A identificação de soluções locais do problema (P) pode ser também realizada através da noção de solução local estável do problema penalizado, apresentada por Önal [94] (ver definição 2.5.1). A seguir introduzimos um conceito similar.

Definição 3.3.2 *Um ponto (\bar{z}, \bar{s}) é uma solução local parcialmente estável do problema penalizado $P(M)$ se existe $\bar{M} \geq 0$ tal que (\bar{z}, \bar{s}) é uma solução local de $P(M)$ para todo $M \geq \bar{M}$.*

Em sua definição, Önal exige uma estabilidade dupla: tanto o ponto solução quanto o valor da função objetivo não devem se alterar a partir de um certo valor \bar{M} . Na verdade, mostra-se que soluções locais estáveis e parcialmente estáveis são equivalentes, se definidas especificamente para $P(M)$. A diferença entre os dois conceitos torna-se aparente quando eles são estendidos para o problema penalizado relativo ao problema em dois níveis com restrições no líder, que consideraremos na seção 3.4.

A equivalência entre as duas definições para $P(M)$ é consequência direta da seguinte caracterização de soluções locais do problema (P) .

Teorema 3.3.2 *Dado o problema (P), as seguintes afirmações são equivalentes:*

- (1) *o ponto (\bar{z}, \bar{s}) é uma solução local de (P);*
- (2) *o ponto (\bar{z}, \bar{s}) é uma solução local parcialmente estável do problema penalizado $P(M)$;*
- (3) *o ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $P(M)$.*

Prova:

(1) \Rightarrow (2): Seja $(\bar{z}, \bar{s}) \in Z \times S$ uma solução local de (P) e $B_\varepsilon(\bar{z}, \bar{s})$ uma ε -vizinhança onde (\bar{z}, \bar{s}) é globalmente ótima. Considerem-se os problemas (P) e $\mathcal{P}(M)$ dados na seção 3.2 com $\mathcal{Z} = Z \cap B_\varepsilon(\bar{z})$, $\mathcal{S} = S \cap B_\varepsilon(\bar{s})$ e $\mathcal{F}(z, s) = c^T z$. Como \mathcal{Z} e \mathcal{S} são conjuntos compactos não vazios, $\mathcal{P}(M)$ tem solução para todo $M \in \Re$. Note-se também que (\bar{z}, \bar{s}) é viável para (P) e que $V[\mathcal{P}] = c^T \bar{z}$. Então, pelo lema 3.2.1, existe $\bar{M} \geq 0$ tal que

$$V[\mathcal{P}(M)] = V[\mathcal{P}] = c^T \bar{z} \quad \forall M \geq \bar{M}.$$

Portanto, (\bar{z}, \bar{s}) é uma solução local parcialmente estável do problema penalizado.

(2) \Rightarrow (3): Seja $(\bar{z}, \bar{s}) \in Z \times S$ uma solução local parcialmente estável do problema penalizado. Então, a derivada direcional de F_M em (\bar{z}, \bar{s}) em qualquer direção viável é não positiva para todo $M \geq \bar{M}$, para algum $\bar{M} \geq 0$. Como $Z \times S$ é convexo, isto significa que deve-se ter

$$\nabla F_M(\bar{z}, \bar{s})^T \begin{pmatrix} z - \bar{z} \\ s - \bar{s} \end{pmatrix} = \begin{pmatrix} c - M\bar{s} \\ -M\bar{z} \end{pmatrix}^T \begin{pmatrix} z - \bar{z} \\ s - \bar{s} \end{pmatrix} \leq 0 \quad \forall (z, s) \in Z \times S, \forall M \geq \bar{M}. \quad (3.45)$$

A expressão acima com $z = \bar{z}$ gera

$$-M\bar{z}^T s \leq -M\bar{z}^T \bar{s} \quad \forall s \in S, \forall M \geq \bar{M}. \quad (3.46)$$

Portanto,

$$F_M(\bar{z}, \bar{s}) = \max\{F_M(\bar{z}, s) : s \in S\} \quad \forall M \geq \bar{M}. \quad (3.47)$$

Por outro lado, fazendo-se $s = \bar{s}$ em (3.45), obtém-se

$$c^T z - M\bar{s}^T z \leq c^T \bar{z} - M\bar{s}^T \bar{z} \quad \forall z \in Z, \forall M \geq \bar{M}. \quad (3.48)$$

Logo,

$$F_M(\bar{z}, \bar{s}) = \max\{F_M(z, \bar{s}) : z \in Z\} \quad \forall M \geq \bar{M}. \quad (3.49)$$

De (3.47) e (3.49), segue-se que (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado.

(3) \Rightarrow (1): Seja $(\bar{z}, \bar{s}) \in Z \times S$ um ponto de equilíbrio do problema penalizado. Então, existe $\bar{M} \geq 0$ tal que

$$c^T z - \bar{M} \bar{s}^T z \leq c^T \bar{z} - \bar{M} \bar{s}^T \bar{z} \quad \forall z \in Z. \quad (3.50)$$

Pela proposição 3.3.1, (\bar{z}, \bar{s}) é viável para (P) . Além disso, pelo corolário 3.3.1, existe $\varepsilon > 0$ tal que

$$\bar{s}^T z = \bar{s}^T \bar{z} = 0 \quad \forall (z, s) \in B_\varepsilon(\bar{z}, \bar{s}), \text{ viável para } (P). \quad (3.51)$$

Usando-se (3.50)-(3.51), conclui-se que $c^T z \leq c^T \bar{z}$ para todo $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$ viável para (P) . Portanto, (\bar{z}, \bar{s}) é uma solução local de (P) . ■

Os teoremas 3.3.1 e 3.3.2 nos permitem caracterizar as soluções locais de PLDNP como a seguir.

Teorema 3.3.3 *Um ponto \bar{z} é uma solução local do problema PLDNP se, e somente se, \bar{z} é viável para PLDNP e (\bar{z}, s) é um ponto de equilíbrio (ou solução local parcialmente estável) do problema penalizado $P(M)$ para todo vértice $s \in S_v(\bar{z})$.*

De acordo com esta caracterização, um ponto \bar{z} é claramente uma solução local do problema PLDNP quando (\bar{z}, \bar{s}) é um ponto de equilíbrio e \bar{s} é o único vértice em S complementar com \bar{z} . Esta situação pode ser identificada pesquisando-se os vértices adjacentes a \bar{s} , como mostra a seguinte resultado.

Corolário 3.3.2 *Seja (\bar{z}, \bar{s}) um ponto de equilíbrio do problema penalizado $P(M)$. Se $\bar{s} \in S_v$ e $\bar{z}^T s > 0$ para todo $s \in S_v$ adjacente a \bar{s} , então \bar{z} é uma solução local do problema PLDNP.*

Prova: Seja (\bar{z}, \bar{s}) um ponto de equilíbrio. Pela proposição 3.3.1, $\bar{s} \in S_v(\bar{z})$ e, por conseguinte, \bar{z} é viável para PLDNP. Então, o lema 3.3.1 garante que $S(\bar{z})$ é uma face de S . Por hipótese, $s \notin S(\bar{z})$ para todo vértice $s \in S_v$, adjacente a \bar{s} . Logo, $S_v(\bar{z}) = \{\bar{s}\}$. Assim, pelo teorema 3.3.3, segue-se que (\bar{z}, \bar{s}) é uma solução local de PLDNP. ■

Outra condição suficiente para otimalidade local em PLDNP é dada abaixo. Ela é mais fraca, porém mais difícil de se verificar computacionalmente.

Teorema 3.3.4 *Seja (\bar{z}, \bar{s}) um ponto de equilíbrio do problema penalizado $P(M)$. Se existe $\varepsilon > 0$ tal que*

$$\bar{s} \in \bigcap_z \{S(z) : z \in Z \cap B_\varepsilon(\bar{z}), S(z) \neq \emptyset\}, \quad (3.52)$$

então \bar{z} é uma solução local do problema PLDNP. Em particular, se $\bar{s} \in S_v$, pode-se substituir $S(z)$ por $S_v(z)$ em (3.52).

Prova: Seja (\bar{z}, \bar{s}) um ponto de equilíbrio. Pela proposição 3.3.1, tem-se que $\bar{s} \in S(\bar{z})$. Então, pela definição 3.3.1, segue-se que

$$\max\{c^T z - \bar{M} \bar{s}^T z : z \in Z\} = c^T \bar{z}$$

para algum $\bar{M} \geq 0$. Recorde-se que o conjunto viável de (P) pode ser definido como $\{z \in Z : S(z) \neq \emptyset\}$. Agora assumamos que existe $\varepsilon > 0$ satisfazendo (3.52), isto é, $\bar{s}^T z = 0$ para todo $z \in Z \cap B_\varepsilon(\bar{z})$ com $S(z) \neq \emptyset$. Então,

$$\begin{aligned} & \max\{c^T z : z \in Z \cap B_\varepsilon(\bar{z}), S(z) \neq \emptyset\} \\ &= \max\{c^T z - \bar{M} \bar{s}^T z : z \in Z \cap B_\varepsilon(\bar{z}), S(z) \neq \emptyset\} \\ &\leq \max\{c^T z - \bar{M} \bar{s}^T z : z \in Z\} = c^T \bar{z}. \end{aligned} \quad (3.53)$$

Como $\bar{z} \in Z \cap B_\varepsilon(\bar{z})$ e $S(\bar{z}) \neq \emptyset$, a igualdade se verifica em (3.53). Logo, \bar{z} é uma solução local de PLDNP. Além disso, como $S(z) \neq \emptyset$ implica $S_v(z) \neq \emptyset$ (lema 3.3.1), pode-se substituir $S(z)$ por $S_v(z)$ em (3.52) quando $\bar{s} \in S_v$. ■

Observe-se que a condição (3.52) significa que todo ponto viável do problema PLDNP, dentro de alguma vizinhança de \bar{z} , seja complementar a \bar{s} . Tal condição é realmente mais fraca que as suposições do corolário 3.3.2. De fato, estas hipóteses implicam que $S_v(\bar{z}) = \{\bar{s}\}$. Assim, (3.52) se verifica, pois, pelo lema 3.3.2, $S(z) \subseteq S(\bar{z})$ para todo $z \in Z \cap B_\varepsilon(\bar{z})$, para algum $\varepsilon > 0$. Entretanto, a condição (3.52) não é suficiente para $S_v(\bar{z}) = \{\bar{s}\}$, como mostra o exemplo (E2), figura 3.2.

Considere-se o ponto A , relativo a $\bar{z} = (\bar{x}, \bar{y}, \bar{w}_1, \bar{w}_2, \bar{w}_3)^T = (0, 2, 0, 2, 0)^T \in Z$. Para $\bar{s} = (0, \bar{v}, \bar{u}_1, \bar{u}_2, \bar{u}_3)^T = (0, 0, 1, 0, 0)^T \in S$, tem-se que (\bar{z}, \bar{s}) é um ponto de equilíbrio. Realmente, para todo $M \geq 0$, tem-se que $F_M(\bar{z}, \bar{s}) = 0$, $F_M(\bar{z}, s) \leq 0$ para todo $s \in S$ e $F_M(z, \bar{s}) \leq 0$ para todo $z \in Z$. Além disso, para $\varepsilon \in (0, 1)$, qualquer ponto $z = (x, y, w_1, w_2, w_3)^T \in B_\varepsilon(\bar{z})$, viável para (P) , tem $w_1 = 0$. Isto

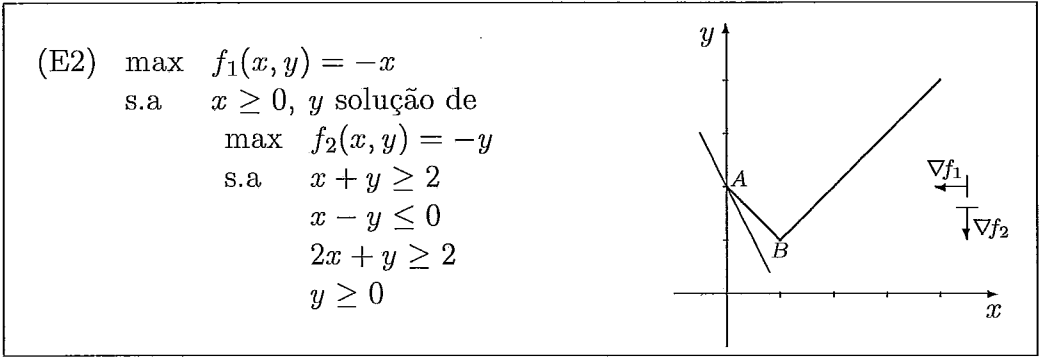


Figura 3.2:

implica que $\bar{s}^T z = 0$ e, portanto, (\bar{z}, \bar{s}) satisfaz (3.52). Todavia, as hipóteses do corolário 3.3.2 não são satisfeitas, pois $\hat{s}^T \bar{z} = 0$ para $\hat{s} = (0, 0, 0, 0, 1)^T \in S_v$, isto é, $\hat{s} \in S_v(\bar{z})$ com \hat{s} adjacente a \bar{s} .

Finalmente, vale a pena mencionar que, para um ponto de equilíbrio (\bar{z}, \bar{s}) , a expressão (3.52) fornece uma condição suficiente mas não necessária para \bar{z} ser uma solução local de PLDNP. De fato, no exemplo (E2) (ver figura 3.2), se a função objetivo do primeiro nível é substituída por $f_1(x, y) = -y$, a solução global será atingida no ponto B , que não satisfaz a condição (3.52).

3.3.2 Otimalidade local estrita

Nesta subseção caracterizamos soluções locais estritas do problema original e dos problemas auxiliares. A princípio, note-se que ótimos locais estritos de PLDNP e (P) podem ser relacionados exatamente como no teorema 3.3.1, apenas trocando-se o termo *local* por *local estrito*. A demonstração segue os mesmos argumentos. Entretanto, usando-se as seguintes definições, que são naturais neste contexto, pode-se apresentar uma caracterização mais simples.

Definição 3.3.3 *Um ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio estrito do problema penalizado $P(M)$ se é um ponto de equilíbrio e existe $M \geq \bar{M}$ tal que*

$$\{\bar{s}\} = \arg \max_{s \in S} F_M(\bar{z}, s) \quad e \quad \{\bar{z}\} = \arg \max_{z \in Z} F_M(z, \bar{s})$$

onde \bar{M} é dado pela definição 3.3.1.

Definição 3.3.4 Um ponto (\bar{z}, \bar{s}) é uma **solução local estrita parcialmente estável** do problema penalizado $P(M)$ se existe $\bar{M} \geq 0$ tal que (\bar{z}, \bar{s}) é uma solução local estrita de $P(M)$ para todo $M \geq \bar{M}$.

Demonstramos abaixo a equivalência entre solução local estrita de (P) , solução local estrita parcialmente estável e ponto de equilíbrio estrito do problema penalizado.

Teorema 3.3.5 Dado o problema (P) , as seguintes afirmações são equivalentes:

- (1) o ponto (\bar{z}, \bar{s}) é uma solução local estrita de (P) ;
- (2) o ponto (\bar{z}, \bar{s}) é uma solução local estrita parcialmente estável do problema penalizado $P(M)$;
- (3) o ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio estrito do problema penalizado $P(M)$.

Prova:

(1) \Rightarrow (2): Seja $(\bar{z}, \bar{s}) \in Z \times S$ uma solução local estrita de (P) . Definam-se os problemas (\mathcal{P}) e $\mathcal{P}(M)$ como na demonstração do teorema 3.3.2. Então, pelo lema 3.2.1, pode-se tomar $\bar{M} > M^* \geq 0$ tal que os problemas (\mathcal{P}) e $\mathcal{P}(M)$ têm o mesmo conjunto solução para todo $M \geq \bar{M}$. Assim, (\bar{z}, \bar{s}) é a única solução de (\mathcal{P}) e $\mathcal{P}(M)$ para todo $M \geq \bar{M}$. Logo, (\bar{z}, \bar{s}) é uma solução local estrita parcialmente estável do problema penalizado.

(2) \Rightarrow (3): Seja $(\bar{z}, \bar{s}) \in Z \times S$ uma solução local estrita parcialmente estável do problema penalizado. Então, existe $\bar{M} \geq 0$ tal que a inequação (3.45) verificasse estritamente para todo $M \geq \bar{M}$ e todo $(z, s) \in Z \times S$ com $(z, s) \neq (\bar{z}, \bar{s})$. Consequentemente, (3.46) e (3.48) são satisfeitas com desigualdades estritas para todo $M \geq \bar{M}$ e para todo $s \in S \setminus \{\bar{s}\}$ e todo $z \in Z \setminus \{\bar{z}\}$. Portanto, segue-se que

$$\{\bar{s}\} = \arg \max\{F_M(\bar{z}, s) : s \in S\} \quad \forall M \geq \bar{M}$$

e

$$\{\bar{z}\} = \arg \max\{F_M(z, \bar{s}) : z \in Z\} \quad \forall M \geq \bar{M}.$$

Logo, (\bar{z}, \bar{s}) é um ponto de equilíbrio estrito do problema penalizado.

(3) \Rightarrow (1): Seja $(\bar{z}, \bar{s}) \in Z \times S$ um ponto de equilíbrio estrito do problema penalizado. Então, existe $\bar{M} \geq 0$ tal que

$$c^T z - \bar{M} \bar{s}^T z < c^T \bar{z} - \bar{M} \bar{s}^T \bar{z} \quad \forall z \in Z \setminus \{\bar{z}\} \quad (3.54)$$

e

$$c^T \bar{z} - \bar{M} \bar{z}^T s < c^T \bar{z} - \bar{M} \bar{z}^T \bar{s} \quad \forall s \in S \setminus \{\bar{s}\}. \quad (3.55)$$

Além disso, pela proposição 3.3.1, tem-se que $\bar{z}^T \bar{s} = 0$. Assim, por (3.55), segue-se que $\bar{z}^T s > 0$ para todo $s \in S \setminus \{\bar{s}\}$, isto é, $S(\bar{z}) = \{\bar{s}\}$. Seja $\varepsilon > 0$ dado pelo corolário 3.3.1 e tome-se $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$, viável para (P) . Então,

$$\bar{z}^T s = \bar{s}^T z = \bar{s}^T \bar{z} = 0. \quad (3.56)$$

Suponha-se que $(z, s) \neq (\bar{z}, \bar{s})$. Como $s \in S(\bar{z}) = \{\bar{s}\}$, deve-se ter $s = \bar{s}$ e $z \neq \bar{z}$. Logo, por (3.54) e (3.56), segue-se que $c^T z < c^T \bar{z}$. Portanto, (\bar{z}, \bar{s}) é uma solução local estrita de (P) . ■

Finalmente, derivamos uma nova caracterização para as soluções locais estritas do problema PLDNP.

Teorema 3.3.6 *Um ponto \bar{z} é a solução local estrita do problema PLDNP se, e somente se, \bar{z} é viável para PLDNP e (\bar{z}, \bar{s}) é uma solução local estrita de (P) (ou ponto de equilíbrio estrito ou solução local estrita parcialmente estável do problema penalizado $P(M)$) para algum $\bar{s} \in S$.*

Prova: Por demonstração similar à do teorema 3.3.1, conclui-se que \bar{z} é uma solução local estrita de PLDNP se, e somente, se $S(\bar{z}) \neq \emptyset$ e (\bar{z}, s) é uma solução local estrita de (P) para todo $s \in S_v(\bar{z})$. Por outro lado, todo ponto de equilíbrio estrito (\bar{z}, \bar{s}) satisfaz

$$\emptyset \neq S_v(\bar{z}) \subseteq S(\bar{z}) = \{\bar{s}\},$$

ou seja, $S_v(\bar{z}) = \{\bar{s}\}$. Logo, a equivalência resulta do teorema 3.3.5. ■

3.3.3 Otimalidade do ponto de equilíbrio

Como visto nas subseções anteriores, as soluções locais de PLDNP estão associadas a pontos de equilíbrio do problema penalizado $P(M)$. Nesta seção, estudamos mais detalhadamente condições de otimalidade para pontos de equilíbrio. A intenção é obter propriedades que sejam computacionalmente mais simples que aquelas estabelecidas pelo corolário 3.3.2 e pelo teorema 3.3.4.

No decorrer desta seção, considera-se que (\bar{z}, \bar{s}) seja um ponto de equilíbrio de $P(M)$. Assim, pela definição 3.3.1, \bar{z} e \bar{s} são soluções dos seguintes problemas lineares paramétricos, para todo $M \geq \bar{M}$.

$$P(M; \bar{s}) \quad \max \quad F_M(z, \bar{s}) = c^T z - M \bar{s}^T z$$

$$\text{s.a} \quad z \in Z = \{z \geq 0 : Az = a\}$$

$$P(M; \bar{z}) \quad \max \quad F_M(\bar{z}, s) = c^T \bar{z} - M \bar{z}^T s$$

$$\text{s.a} \quad s \in S = \{s \geq 0 : Ds = d\}$$

Como soluções destes problemas ocorrem nos vértices, vai-se assumir também que $(\bar{z}, \bar{s}) \in Z_v \times S_v$.

Considerem-se, abaixo, os quadros simplex ótimos associados aos problemas $P(M; \bar{s})$ e $P(M; \bar{z})$, onde B e E são submatrizes básicas e N e R , não básicas.

	z_B^T	z_N^T			z_B^T	z_N^T		
	B	N	a	\longrightarrow	I	$\tilde{N} = B^{-1}N$	$\bar{z}_B = B^{-1}a$	
M	$-\bar{s}_B^T$	$-\bar{s}_N^T$	0		M	0	$-\tilde{s}_N^T = -\bar{s}_N^T + \bar{s}_B^T \tilde{N}$	$0 = \bar{s}_B^T \bar{z}_B$
	c_B^T	c_N^T	0			0	$\tilde{c}_N^T = c_N^T - c_B^T \tilde{N}$	$-c_B^T \bar{z}_B$

	s_E^T	s_R^T			s_E^T	s_R^T		
	E	R	d	\longrightarrow	I	$\tilde{R} = E^{-1}R$	$\bar{s}_E = E^{-1}d$	
M	$-\bar{z}_E^T$	$-\bar{z}_R^T$	0		M	0	$-\tilde{z}_R^T = -\bar{z}_R^T + \bar{z}_E^T \tilde{R}$	$0 = \bar{z}_E^T \bar{s}_E$

Nos quadros relativos a $P(M; \bar{s})$, a função objetivo $F_M(z, \bar{s}) = c^T z - M \bar{s}^T z$ está representada em duas linhas, sendo que a primeira corresponde ao termo de complementaridade $\bar{s}^T z$ e a segunda à parcela linear $c^T z$. Observe-se que é possível considerar M de forma implícita, otimizando prioritariamente a primeira linha da função objetivo, como no método simplex *big-M* (Bazaraa *et al.* [16]).

Já nos quadros referentes a $P(M; \bar{z})$, a função a maximizar é $F_M(\bar{z}, s) = c^T \bar{z} - M \bar{z}^T s$. Como o primeiro termo é constante, representa-se apenas o segundo. Neste caso, o parâmetro M não influencia o ponto solução.

Como (\bar{z}, \bar{s}) é ponto de equilíbrio, deduz-se do lema 3.3.3 que

$$\tilde{s}_N \geq 0 \quad \text{e} \quad \tilde{z}_R \geq 0.$$

Mais ainda, quando

$$\tilde{c}_N \leq 0, \quad (3.57)$$

segue-se que (\bar{z}, \bar{s}) é solução do problema relaxado e, por conseguinte, solução global de PLDNP. Por outro lado, se

$$\tilde{z}_R > 0, \quad (3.58)$$

então \bar{s} é o único ponto em S complementar a \bar{z} ($S(\bar{z}) = \{\bar{s}\}$) e, pelo teorema 3.3.3, conclui-se que \bar{z} é uma solução local de PLDNP.

Estudamos agora outras condições de otimalidade que podem ser inferidas quando estas situações particulares não ocorrem. A análise será desenvolvida em função das direções extremas de Z e S , a partir de \bar{z} e \bar{s} , que estão disponíveis nos quadros. Tais direções estão associadas às colunas das seguintes matrizes

$$G = \begin{bmatrix} -\tilde{N} \\ I_{n-m_2} \end{bmatrix} \in \mathfrak{R}^{n \times (n-m_2)} \quad \text{e} \quad H = \begin{bmatrix} -\tilde{R} \\ I_{n-n_2} \end{bmatrix} \in \mathfrak{R}^{n \times (n-n_2)},$$

formadas a partir das submatrizes não básicas pivoteadas, juntamente com matrizes identidade I_{n-m_2} e I_{n-n_2} de ordem $n - m_2$ e $n - n_2$ respectivamente. Note-se que

$$\tilde{c}_N^T = c^T G, \quad \tilde{s}_N^T = \bar{s}^T G \geq 0 \quad \text{e} \quad \tilde{z}_R^T = \bar{z}^T H \geq 0.$$

Para o desenvolvimento que se segue, é adequado considerar diretamente as colunas das matrizes G e H . Denote-se, então, por J_N e J_R os índices das variáveis não básicas em \bar{z} e \bar{s} e assumam-se, por simplicidade, que as matrizes G e H conservam a mesma indexação A e D . Assim, tem-se que

$$G_i = \begin{bmatrix} -\tilde{N}_i \\ e_i \end{bmatrix} \in \mathfrak{R}^n \quad \text{e} \quad H_j = \begin{bmatrix} -\tilde{R}_j \\ e_j \end{bmatrix} \in \mathfrak{R}^n,$$

são colunas de G e H para $i \in J_N$ e $j \in J_R$, onde $e_i \in \mathfrak{R}^{n-m_2}$ e $e_j \in \mathfrak{R}^{n-n_2}$ são colunas de I_{n-m_2} e I_{n-n_2} . Adicionalmente, escrevem-se os custos reduzidos como

$$\tilde{c}_i = c^T G_i, \quad \tilde{s}_i = \bar{s}^T G_i \geq 0 \quad \text{e} \quad \tilde{z}_j = \bar{z}^T H_j \geq 0, \quad (3.59)$$

para cada $i \in J_N$ e $j \in J_R$.

Defina-se, por último, os conjuntos

$$J_N^+ = \{i \in J_N : \tilde{c}_i > 0\} = \{i \in J_N : c^T G_i > 0\},$$

$$J_R^0 = \{j \in J_R : \tilde{z}_j = 0\} = \{j \in J_R : \bar{z}^T H_j = 0\}.$$

Recorde-se por (3.57)-(3.58) que \bar{z} é uma solução global de PLDNP, quando $J_N^+ = \emptyset$, ou local, se $J_R^0 = \emptyset$.

Mostramos a seguir que os pontos de Z podem ser expressos em função de \bar{z} e das direções G_i . O mesmo desenvolvimento é depois aplicado a S , \bar{s} e direções H_j .

Lema 3.3.4 *Se $z \in Z$, então $z = \bar{z} + \sum_{i \in J_N} z_i G_i$.*

Prova: Seja $z = (z_D^T, z_N^T)^T \in Z$. Então $Dz_D + Nz_N = a$, ou ainda, $z_D = \bar{z}_D - \tilde{N}z_N$. Como $\bar{z}_N = 0$, segue-se que

$$\begin{bmatrix} z_D \\ z_N \end{bmatrix} = \begin{bmatrix} \bar{z}_D \\ \bar{z}_N \end{bmatrix} + \begin{bmatrix} -\tilde{N} \\ I_{n-m_2} \end{bmatrix} z_N.$$

Por conseguinte, $z = \bar{z} + \sum_{i \in J_N} z_i G_i$. ■

Lema 3.3.5 *Se $s \in S$, então $s = \bar{s} + \sum_{j \in J_R} s_j H_j$. Em particular, se $s \in S(\bar{z})$, então*

$$s = \bar{s} + \sum_{j \in J_R^0: \bar{z}_j=0} s_j H_j.$$

Prova: Seguindo-se os mesmos passos da demonstração acima, conclui-se que $s = \bar{s} + \sum_{j \in J_R} s_j H_j$. Assuma-se, agora, que $s \in S(\bar{z})$. Defina $\bar{J} = \{j \in J_R : \bar{z}_j = 0\}$. Como $s \in S(\bar{z})$, se $j \in J_R \setminus \bar{J}$ então $s_j = 0$. Assim,

$$s = \bar{s} + \sum_{j \in \bar{J}} s_j H_j = \bar{s} + \sum_{j \in \bar{J} \cap J_R^0} s_j H_j + \sum_{j \in \bar{J} \setminus J_R^0} s_j H_j. \quad (3.60)$$

Mais ainda,

$$0 = \bar{z}^T s = \bar{z}^T \bar{s} + \sum_{j \in \bar{J} \cap J_R^0} s_j \bar{z}^T H_j + \sum_{j \in \bar{J} \setminus J_R^0} s_j \bar{z}^T H_j = \sum_{j \in \bar{J} \setminus J_R^0} s_j \bar{z}^T H_j.$$

Por (3.59) e pela definição de J_R^0 , tem-se que $\bar{z}^T H_j > 0$ para todo $j \in \bar{J} \setminus J_R^0 \subseteq J_R \setminus J_R^0$. Logo, $s_j = 0$ para todo $j \in \bar{J} \setminus J_R^0$. Segue-se, então, de (3.60) que

$$s = \bar{s} + \sum_{j \in \bar{J} \cap J_R^0} s_j H_j. \quad \blacksquare$$

Em todos os próximos resultados, vai-se considerar a seguinte hipótese:

[D] Se $i \in J_N^+$ e $\tilde{N}_{ki} > 0$, então $\bar{z}_k > 0$.

Esta suposição é trivialmente verificada quando o vértice \bar{z} é não degenerado. Na verdade, ela assegura que G_i ($i \in J_N^+$) é de fato uma direção viável de Z , como mostramos no lema abaixo.

Lema 3.3.6 *Se $i \in J_N^+$ então existe $L > 0$ tal que $z = \bar{z} + z_i G_i \in Z$ para todo $0 \leq z_i \leq L$.*

Prova: Seja $i \in J_N^+$. Tem-se $G_i = \begin{bmatrix} -\tilde{N}_i \\ e_i \end{bmatrix}$. Se $G_i \geq 0$, então $z = \bar{z} + z_i G_i \geq 0$ para todo $z_i \geq 0$. Neste caso, tome-se $L > 0$ arbitrário. Caso $G_i \not\geq 0$, existe $k \in J_B$ com $\tilde{N}_{ki} > 0$, onde J_B é o conjunto de índices das variáveis básicas em \bar{z} . Tome-se, então,

$$L = \min_{k \in J_B} \left\{ \frac{\bar{z}_k}{\tilde{N}_{ki}} : \tilde{N}_{ki} > 0 \right\}.$$

Sob a hipótese [D], tem-se que $L > 0$. Seja $z = \bar{z} + z_i G_i$, com $0 \leq z_i \leq L$. Particione-se $z = (z_B^T, z_N^T)^T$. Então,

$$z_B = \bar{z}_B - z_i \tilde{N}_i \quad \text{e} \quad z_N = z_i e_i.$$

Trivialmente, $z_N \geq 0$. Por outro lado, seja $k \in J_B$. Então $z_k = \bar{z}_k - z_i \tilde{N}_{ki}$. Se $\tilde{N}_{ki} \leq 0$, então $z_k \geq \bar{z}_k > 0$. Do contrário, tem-se também $z_k \geq 0$, pois

$$z_k = \bar{z}_k - z_i \tilde{N}_{ki} \geq \bar{z}_k - L \tilde{N}_{ki} \geq \bar{z}_k - \frac{\bar{z}_k}{\tilde{N}_{ki}} \tilde{N}_{ki} = 0.$$

Em qualquer caso, conclui-se que $z = \bar{z} + z_i G_i \geq 0$ para todo $0 \leq z_i \leq L$. Finalmente, verifica-se que

$$Az = A\bar{z} + z_i A G_i = a + z_i [B \quad N] \begin{bmatrix} -B^{-1} N_i \\ e_i \end{bmatrix} = a.$$

Logo, $z = \bar{z} + z_i G_i \in Z$ para todo $0 \leq z_i \leq L$. ■

No propósito de obter condições de otimalidade local para (\bar{z}, \bar{s}) , estabelecemos primeiramente o seguinte resultado auxiliar.

Lema 3.3.7 *Seja $i \in J_N^+$. Se $s \in S(\bar{z})$, então $s^T G_i \geq 0$. Em particular, $\bar{s}^T G_i > 0$.*

Prova: Sejam $i \in J_N^+$ e $s \in S(\bar{z})$. Pelo lema 3.3.6, existe $z = \bar{z} + z_i G_i \in Z$ com $z_i > 0$. Então,

$$0 \leq s^T z = s^T \bar{z} + z_i s^T G_i = z_i s^T G_i.$$

Como $z_i > 0$, deve-se ter $s^T G_i \geq 0$. Assuma-se agora que $s = \bar{s} \in S(\bar{z})$. Como (\bar{z}, \bar{s}) é ponto de equilíbrio, segue-se da definição 3.3.1 que $(c - M\bar{s})^T z \leq (c - M\bar{s})^T \bar{z}$ para algum $M \geq 0$. Assim,

$$0 \geq (c - M\bar{s})^T (z - \bar{z}) = z_i (c - M\bar{s})^T G_i.$$

Visto que $z_i > 0$ e $i \in J_N^+$, tem-se

$$M\bar{s}^T G_i \geq c^T G_i = \tilde{c}_i > 0$$

e, por conseguinte, $\bar{s}^T G_i > 0$. ■

Apresentamos agora uma condição necessária e suficiente para que \bar{z} seja um ótimo local de PLDNP, em função das direções extremas de Z a partir de \bar{z} .

Teorema 3.3.7 *O ponto \bar{z} é uma solução local de PLDNP se, e somente se, $J_N^+ = \emptyset$ ou $\min\{G_i^T s : s \in S(\bar{z})\} > 0$ para todo $i \in J_N^+$.*

Prova: Vai-se demonstrar, equivalentemente, que \bar{z} não é ótimo local de PLDNP se, e somente se, existe $i \in J_N^+$ tal que $\min\{G_i^T s : s \in S(\bar{z})\} \leq 0$.

Primeiro, assumamos que existe $i \in J_N^+$ tal que $\min\{G_i^T s : s \in S(\bar{z})\} \leq 0$. Então, $G_i^T s \leq 0$ para algum $s \in S(\bar{z})$. Pelo lema 3.3.7, $G_i^T s = 0$. Seja $\varepsilon > 0$ arbitrário. A partir do lema 3.3.6 conclui-se que existe $z = \bar{z} + z_i G_i \in Z \cap B_\varepsilon(\bar{z})$ com $z_i > 0$. Mais ainda, tem-se

$$z^T s = \bar{z}^T s + z_i G_i^T s = 0.$$

Assim, $z \in B_\varepsilon(\bar{z})$ é viável para PLDNP e $c^T z - c^T \bar{z} = z_i c^T G_i > 0$ ($i \in J_N^+$). Logo, \bar{z} não é ótimo local de PLDNP.

Assumamos, agora, que \bar{z} não é ótimo local de PLDNP e seja ε dado pelo lema 3.3.2. Então, existe $z \in B_\varepsilon(\bar{z})$, com $\emptyset \neq S(z) \subseteq S(\bar{z})$ e $c^T z > c^T \bar{z}$. Pelo lema 3.3.4,

$$z = \bar{z} + \sum_{i \in J_N} z_i G_i.$$

Assim,

$$0 < c^T z - c^T \bar{z} = \sum_{i \in J_N} z_i c^T G_i.$$

Logo, existe $k \in J_N$ com $z_k > 0$ e $c^T G_k > 0$. Portanto, $k \in J_N^+$. Deseja-se mostrar que $\min\{G_k^T s : s \in S(\bar{z})\} \leq 0$. De fato, seja $\hat{s} \in S(z) \subseteq S(\bar{z})$. Então, z e \bar{z} estão na face de Z definida por $\arg \min\{\hat{s}^T z : z \in Z\}$. Dado que $z_k > 0$ e considerando-se o lema 3.3.6, deduz-se que $\hat{z} = \bar{z} + \hat{z}_k G_k$ pertence a esta face para algum $0 < \hat{z}_k \leq z_k$. Por conseguinte,

$$0 = \hat{s}^T \hat{z} = \hat{s}^T \bar{z} + \hat{z}_k \hat{s}^T G_k,$$

implicando que $\hat{s}^T G_k = 0$. Portanto,

$$\min\{G_k^T s : s \in S(\bar{z})\} \leq G_k^T \hat{s} = 0.$$

■

Pelo teorema 3.3.3, se uma solução viável \bar{z} de PLDNP não é ótimo local, então (\bar{z}, s) não é ponto de equilíbrio para algum $s \in S(\bar{z})$. Relacionamos, a seguir, este resultado com o teorema 3.3.7 acima.

Corolário 3.3.3 *Se existem $i \in J_N^+$ e $s \in S(\bar{z})$ tais que $G_i^T s = 0$, então \bar{z} não é solução local e (\bar{z}, s) não é ponto de equilíbrio. Neste caso, existe $z \in Z(s)$, $z = \bar{z} + z_i G_i$, com $z_i > 0$ e $c^T z > c^T \bar{z}$.*

Prova: Sejam $i \in J_N^+$ e $s \in S(\bar{z})$ tais que $G_i^T s = 0$. A primeira implicação é decorrência direta do teorema 3.3.7. Adicionalmente, pelo lema 3.3.6, existe $z = \bar{z} + z_i G_i \in Z$, com $z_i > 0$. Assim,

$$c^T z - M s^T z = c^T \bar{z} - M s^T \bar{z} + \underbrace{z_i c^T G_i}_{>0} - M z_i \underbrace{s^T G_i}_{=0} > c^T \bar{z} - M s^T \bar{z} \quad (3.61)$$

para todo M . Logo (\bar{z}, s) não é ponto de equilíbrio. Mais ainda, como $s^T \bar{z} = 0$ e $s^T G_i = 0$, tem-se

$$s^T z = s^T \bar{z} + z_i s^T G_i = 0.$$

Logo, por (3.61), segue-se que $c^T z > c^T \bar{z}$. ■

Observe-se que verificar a condição do teorema 3.3.7 corresponde a resolver problemas lineares. Em alguns casos, pode-se reduzir este esforço computacional com mostramos abaixo.

Teorema 3.3.8 *Se o conjunto $\{(i, j) \in J_N^+ \times J_R^0 : \bar{z}_j = 0, G_i^T H_j < 0\}$ é vazio, então \bar{z} é um ótimo local de PLDNP.*

Prova: Suponha-se, por contraposição, que \bar{z} não é ótimo local de PLDNP. Deseja-se mostrar que existe $(i, j) \in J_N^+ \times J_R^0$ com $\bar{z}_j = 0$ e $G_i^T H_j < 0$. Pelos teorema 3.3.7 e lema 3.3.7, existem $i \in J_N^+$ e $s \in S(\bar{z})$ tais que $G_i^T s = 0$. O corolário 3.3.3 assegura que existe $z = \bar{z} + z_i G_i \in Z(s)$ com $c^T z > c^T \bar{z}$ e $z_i > 0$. Por outro lado, pelo lema 3.3.5,

$$s = \bar{s} + \sum_{j \in J_R^0: \bar{z}_j=0} s_j H_j.$$

Assim,

$$\begin{aligned} 0 = z^T s &= \bar{z}^T \bar{s} + \sum_{j \in J_R^0: \bar{z}_j=0} s_j \bar{z}^T H_j + z_i G_i^T \bar{s} + \sum_{j \in J_R^0: \bar{z}_j=0} z_i s_j G_i^T H_j \\ &= z_i G_i^T \bar{s} + \sum_{j \in J_R^0: \bar{z}_j=0} z_i s_j G_i^T H_j \end{aligned}$$

Como $z_i > 0$ e $G_i^T \bar{s} > 0$ pelo lema 3.3.7, conclui-se que $G_i^T H_j < 0$ para algum $j \in J_R^0$ com $\bar{z}_j = 0$. E como $i \in J_N^+$, o resultado segue-se. \blacksquare

Quando \bar{z} e \bar{s} são ambos não degenerados, uma condição ainda mais simples do ponto de vista computacional é apresentada pelo próximo corolário.

Corolário 3.3.4 *Assuma-se que \bar{z} e \bar{s} são vértices não degenerados e seja J_E o conjunto de índices das variáveis básicas em \bar{s} . Se o conjunto $\{(i, j) \in J_N^+ \times J_R^0 : i \in J_E, \bar{z}_j = 0, \tilde{R}_{ij} > 0\}$ é vazio, então \bar{z} é um ótimo local de PLDNP.*

Prova: Suponha-se, por contraposição, que \bar{z} não é ótimo local de PLDNP. Deseja-se mostrar que existe $(i, j) \in J_N^+ \times J_R^0$ com $i \in J_E, \bar{z}_j = 0$ e $\tilde{R}_{ij} > 0$. Pelo teorema 3.3.8, existe $(i, j) \in J_N^+ \times J_R^0$ com $\bar{z}_j = 0$ e $G_i^T H_j < 0$. Então vai-se mostrar que, no caso não degenerado, $i \in J_E$ e $G_i^T H_j = -\tilde{R}_{ij}$. De fato, seja J_B o conjunto de índices das variáveis básicas em \bar{z} e considerem-se as expressões de G_i e H_j .

$$G_{ki} = \begin{cases} -\tilde{N}_{ki}, & \text{se } k \in J_B \\ 0, & \text{se } k \in J_N \setminus \{i\} \\ 1, & \text{se } k = i \end{cases} \quad H_{kj} = \begin{cases} -\tilde{R}_{kj}, & \text{se } k \in J_E \\ 0, & \text{se } k \in J_R \setminus \{j\} \\ 1, & \text{se } k = j \end{cases}$$

Como $\bar{z}^T \bar{s} = 0$, segue-se que $J_B \subseteq J_R$ no caso não degenerado. Além disso, $J_B \subseteq J_R \setminus \{j\}$, pois $\bar{z}_j = 0$. Logo, se $k \in J_B$, então $H_{kj} = 0$. Assim, tem-se

$$0 > G_i^T H_j = \sum_{k \in J_B} G_{ki} \underbrace{H_{kj}}_{=0} + \sum_{k \in J_N \setminus \{i\}} \underbrace{G_{ki}}_{=0} H_{kj} + \underbrace{G_{ii}}_{=1} H_{ij} = H_{ij}.$$

Então, $H_{ij} < 0$. Finalmente, pela expressão de H_j , conclui-se que $i \in J_E$ e, por conseguinte,

$$0 > H_{ij} = -\tilde{R}_{ij}.$$

Logo, o conjunto $\{(i, j) \in J_N^+ \times J_R^0 : i \in J_E, \bar{z}_j = 0, \tilde{R}_{ij} > 0\}$ é não vazio. ■

3.4 Extensões

Nas duas seções anteriores derivamos resultados para PLDNP. Agora, identificamos as propriedades que podem ou não ser estendidas para PLDN, ou seja, o problema obtido de PLDNP com a adição de restrições no primeiro nível.

Os problemas auxiliares associados a PLDN são definidos similarmente como:

$$\begin{aligned} (\tilde{P}) \quad \max \quad & F(z, s) = c^T z \\ \text{s.a.} \quad & z \in \tilde{Z}, s \in S \\ & s^T z = 0 \end{aligned} \qquad \begin{aligned} \tilde{P}(M) \quad \max \quad & F_M(z, s) = c^T z - M s^T z \\ \text{s.a.} \quad & z \in \tilde{Z}, s \in S \end{aligned}$$

onde

$$\begin{aligned} \tilde{Z} &= \{z \in \mathbb{R}^n : B_1 x + B_2 y \leq b, Az = a, z^T = (x^T, y^T, w^T) \geq 0\} \subseteq Z \\ S &= \{s \in \mathbb{R}^n : Ds = d, s^T = (0, v^T, u^T) \geq 0\}. \end{aligned}$$

Como o conjunto S está associado ao problema dual do seguidor, sua definição não se altera em relação a PLDNP. Entretanto o conjunto \tilde{Z} é menor que Z , pois inclui também as restrições do líder.

De uma forma geral, todos os resultados obtidos para PLDNP, (P) e $P(M)$ podem ser estendidos para PLDN, (\tilde{P}) e $\tilde{P}(M)$ quando $B_2 = 0$, ou seja, quando as restrições do primeiro nível envolvem apenas a variável x . Neste caso, conforme corolário 1.2.4, tais restrições podem ser transferidas para o segundo nível, obtendo-se assim a situação anterior.

Analisamos agora o que acontece quando $B_2 \neq 0$, isto é, quando as restrições do primeiro nível dependem da variável y . Primeiramente, consideramos a análise global desenvolvida na seção 3.2.

Observe-se que a proposição 3.2.1 pode ser igualmente expressa em termos de (\tilde{P}) e $\tilde{P}(M)$. De fato, ela decorre diretamente do lema 3.2.1, que abrange também estes novos problemas. Isto garante um resultado similar à segunda parte do teorema 3.2.2, que enunciamos a seguir.

Teorema 3.4.1 *Se o problema PLDN (ou (\tilde{P})) é viável e o problema $\tilde{P}(M_0)$ tem solução para algum $M_0 \geq 0$, então PLDN tem solução. Neste caso, z^* é uma solução global de PLDN se, e somente se, existe s^* tal que (z^*, s^*) é solução global de $\tilde{P}(M)$ para todo $M > M^*$, para algum $M^* \geq M_0$.*

Infelizmente, nas situações não compreendidas pelas hipóteses do teorema acima, a equivalência entre PLDN e o problema penalizado correspondente não se verifica, pois as proposições 3.2.2 e 3.2.3 não podem ser estendidas para (\tilde{P}) e $\tilde{P}(M)$. Isto deve-se essencialmente ao fato de que o lema 3.1.1 não é válido quando se substitui Z por \tilde{Z} .

Na verdade, embora os três casos enumerados no teorema 3.2.1 também aconteçam com (\tilde{P}) e $\tilde{P}(M)$, existem ainda as seguintes possibilidades:

- (i) (\tilde{P}) é inviável e $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$;
- (ii) (\tilde{P}) é inviável e $\tilde{P}(M)$ tem solução para algum $M \geq 0$;
- (iii) (\tilde{P}) tem solução e $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$.

Note-se que estes casos podem ser escritos em função de PLDN em vez de (\tilde{P}) .

Estas novas situações são ilustradas através de exemplos. Para o caso (i) apresentamos o problema (E3), figura 3.3, onde se tem

$$\tilde{Z} = \{(x, y, w_1, w_2)^T \geq 0 : -x + y \geq 3, x + y - w_1 = 2, x - y + w_2 = 0\}$$

e

$$S = \{(0, v, u_1, u_2)^T \geq 0 : u_1 + u_2 + v = 1\}.$$

Então, $\bar{s} = (0, 0, 0, 1)^T \in S$ e $z(\alpha) = (\alpha, 3 + \alpha, 1 + 2\alpha, 3)^T \in \tilde{Z}$ para todo $\alpha \geq 0$. Como, para cada $M \in \Re$, $F_M(z(\alpha), \bar{s}) = \alpha - 3M \rightarrow +\infty$ quando $\alpha \rightarrow +\infty$, conclui-se que $\tilde{P}(M)$ é ilimitado. Por outro lado, note-se que qualquer ponto (x, y) , tal que y é solução do problema do seguidor para x fixo, não satisfaz a restrição do primeiro nível. Assim, o problema PLDN e, conseqüentemente, o problema (\tilde{P}) são inviáveis.

Para ilustrar o caso (ii), considere-se o exemplo (E4), obtido de (E3) pela substituição da função objetivo do primeiro nível por $f_1(x, y) = -y$. Assim, os conjuntos \tilde{Z} e S são os mesmos do exemplo anterior. Logo, o problema (\tilde{P}) permanece inviável. Por outro lado, como esta nova função é limitada superiormente em \tilde{Z} , $\tilde{P}(M)$ passa

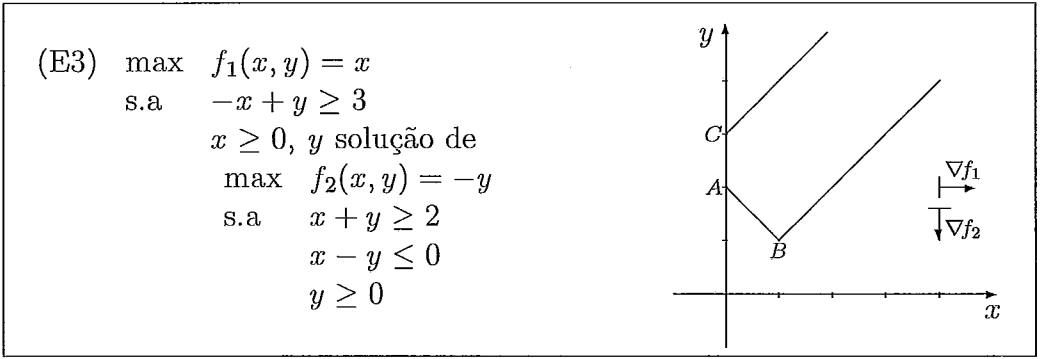


Figura 3.3:

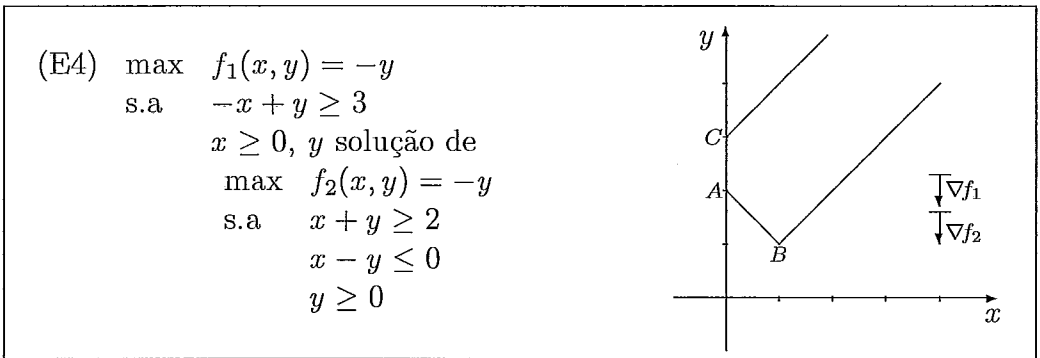


Figura 3.4:

a ter solução para todo $M \in \Re$. Realmente, para todo $M \geq 0$, a solução de $\tilde{P}(M)$ é $\bar{z} = (0, 3, 1, 3)^T, \bar{s} = (0, 0, 1, 0)$, com $F_M(\bar{z}, \bar{s}) = -3 - M$. Note-se que \bar{z} corresponde ao ponto C na figura 3.4.

Finalmente, para exemplificar o caso (iii), seja substituída em (E3) a restrição do primeiro nível por $-x + y \geq 1$. Com isto gera-se o exemplo (E5), figura 3.5, onde o conjunto S é mesmo que o de (E3), mas o conjunto \tilde{Z} é agora dado por

$$\tilde{Z} = \{(x, y, w_1, w_2)^T \geq 0 : -x + y \geq 1, x + y - w_1 = 2, x - y + w_2 = 0\}.$$

Note-se, assim, este novo conjunto \tilde{Z} contém o anterior. Logo, o conjunto viável de $\tilde{P}(M)$ neste exemplo é maior que o do exemplo (E3), implicando que $\tilde{P}(M)$ é novamente ilimitado para todo $M \in \Re$. Por outro lado, note-se que o problema (\tilde{P}) agora é viável. Na figura 3.5, o conjunto viável de (E3) corresponde ao segmento AC . Mais ainda, (E3) tem solução, que é atingida no ponto $C = (1/2, 3/2)$.

Examinamos agora os resultados da análise local realizada na seção 3.3 e as possíveis extensões para PLDN, (\tilde{P}) e $\tilde{P}(M)$. Como já mencionado, a generalização

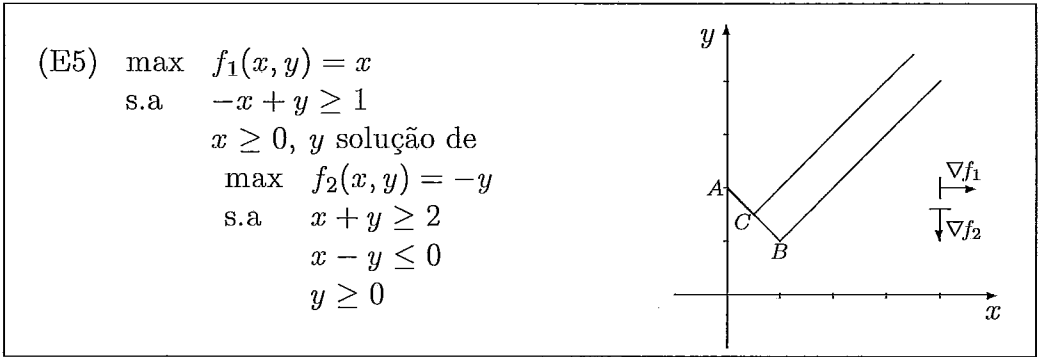


Figura 3.5:

é direta quando $B_2 = 0$. Passamos então ao outro caso.

Começamos pela caracterização de uma vizinhança viável para (\tilde{P}) em torno de um ponto $(\bar{z}, \bar{s}) \in \tilde{Z} \times S$. Uma tal caracterização pode ser dada como no corolário 3.3.1. De fato, resultado similar ao lema 3.3.2 é válido para as funções ponto-conjunto

$$S(z) = \{s \in S : z^T s = 0\} \quad \text{e} \quad \tilde{Z}(s) = \{z \in \tilde{Z} : s^T z = 0\} \subseteq Z(s).$$

Assim, derivamos a propriedade abaixo:

Teorema 3.4.2 *Seja $(\bar{z}, \bar{s}) \in \tilde{Z} \times S$. Existe $\varepsilon > 0$ tal que todo ponto $(z, s) \in B_\varepsilon(\bar{z}, \bar{s})$, viável para o problema (\tilde{P}) , deve satisfazer $\bar{s}^T z = \bar{z}^T s = 0$, isto é, (z, s) e (\bar{z}, \bar{s}) também são viáveis para (\tilde{P}) .*

A relação entre soluções locais de PLDNP e soluções locais de (P) , expressa no teorema 3.3.1, pode ser também generalizada para PLDN e (\tilde{P}) . Seguindo-se a demonstração deste teorema, com \tilde{Z} em lugar de Z , e levando-se em conta que o conjunto viável de PLDN é dado por $\{z \in \tilde{Z} : S(z) \neq \emptyset\}$, chega-se diretamente a tal resultado. Além disso, como visto no início da subseção 3.3.2, a mesma argumentação aplica-se a soluções locais estritas. Desta forma, tem-se a seguinte caracterização.

Teorema 3.4.3 *Um ponto \bar{z} é uma solução local (estrita) do problema PLDN se, e somente se, \bar{z} é viável para PLDN e (\bar{z}, s) é uma solução local (estrita) do problema (\tilde{P}) para todo vértice $s \in S_v(\bar{z})$.*

Podemos estender os conceitos de ponto de equilíbrio para o problema penalizado $\tilde{P}(M)$ como a seguir.

Definição 3.4.1 *Um ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $\tilde{P}(M)$ se existe $\bar{M} \geq 0$ tal que, para cada $M \geq \bar{M}$, verifica-se*

$$\max_{s \in S} F_M(\bar{z}, s) = F_M(\bar{z}, \bar{s}) = \max_{z \in \tilde{Z}} F_M(z, \bar{s}), \quad (3.62)$$

onde $F_M(z, s) = c^T - M s^T z$.

De forma análoga à demonstração do lema 3.3.3, obtemos a propriedade abaixo com relação ao termo de complementaridade.

Lema 3.4.1 *Se (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $\tilde{P}(M)$, então*

$$\min_{s \in S} \bar{z}^T s = \bar{s}^T \bar{z} = \min_{z \in \tilde{Z}} \bar{s}^T z.$$

Entretanto um ponto de equilíbrio de $\tilde{P}(M)$ pode não anular as folgas complementares e, por conseguinte, não ser viável para (\tilde{P}) . Este fato, ilustrado pelo exemplo (E6), figura 3.6, deve-se também à não validade do lema 3.1.1 quando se substitui Z por \tilde{Z} .

No exemplo (E6), tem-se

$$\tilde{Z} = \{(x, y, w_1, w_2)^T \geq 0 : x + 2y \geq 6, x + y - w_1 = 2, x - y + w_2 = 0\}$$

e

$$S = \{(0, v, u_1, u_2)^T \geq 0 : u_1 + u_2 + v = 1\}.$$

Seja, então, $\bar{z} = (0, 3, 1, 3)^T \in \tilde{Z}$, associado ao ponto D , e $\bar{s} = (0, 0, 1, 0)^T \in S$. Assim, $F_M(\bar{z}, \bar{s}) = -M$ para cada $M \in \Re$. Além disso, dado $(z, s) \in \tilde{Z} \times S$, obtém-se

$$F_M(\bar{z}, s) = -M(3v + u_1 + 3u_2) = -M - 2M(v + u_2) \leq -M$$

e

$$F_M(z, \bar{s}) = -x - M w_1 \leq -M.$$

Portanto, (\bar{z}, \bar{s}) é um ponto de equilíbrio de $\tilde{P}(M)$, porém não é viável para (\tilde{P}) , pois $\bar{z}^T \bar{s} = 1$.

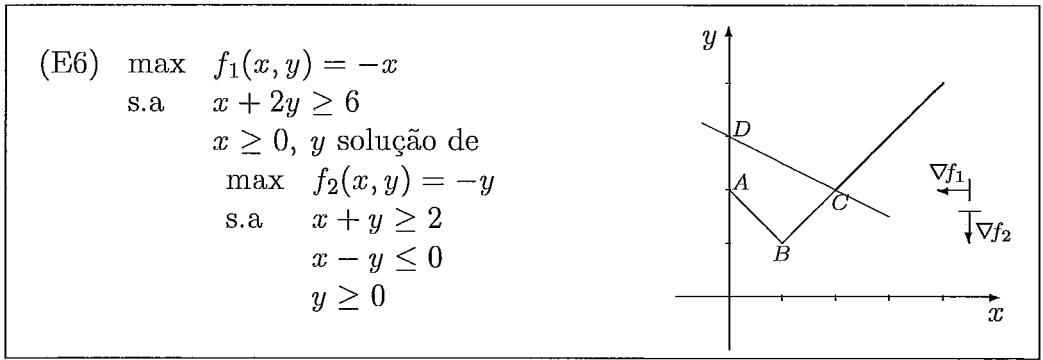


Figura 3.6:

Por outro lado, a caracterização de soluções locais de (\tilde{P}) como pontos de equilíbrio é ainda possível quando se exige explicitamente que a complementaridade seja satisfeita. Esta condição é requerida pela definição 2.5.1 de solução local estável de $P(M)$, introduzida por Önal [94], que estendemos agora para $\tilde{P}(M)$.

Definição 3.4.2 Um ponto (\bar{z}, \bar{s}) é uma solução local estável do problema penalizado $\tilde{P}(M)$ se $\bar{z}^T \bar{s} = 0$ e se existe $\bar{M} \geq 0$ tal que (\bar{z}, \bar{s}) é uma solução local de $\tilde{P}(M)$ para todo $M \geq \bar{M}$.

Obtemos então as seguintes equivalências, similares ao teorema 3.3.2.

Teorema 3.4.4 Dado o problema (\tilde{P}) , as seguintes afirmações são equivalentes:

- (1) o ponto (\bar{z}, \bar{s}) é uma solução local de (\tilde{P}) ;
- (2) o ponto (\bar{z}, \bar{s}) é uma solução local estável do problema penalizado $\tilde{P}(M)$;
- (3) o ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $\tilde{P}(M)$ e $\bar{z}^T \bar{s} = 0$.

Prova: Segue a demonstração do teorema 3.3.2, substituindo-se Z por \tilde{Z} e, consequentemente, (P) por (\tilde{P}) e $P(M)$ por $\tilde{P}(M)$. Além disso, na implicação (3) \Rightarrow (1), a viabilidade de (\bar{z}, \bar{s}) para (\tilde{P}) é garantida por hipótese e não através da proposição 3.3.1. ■

Vale a pena mencionar que continua válida para $\tilde{P}(M)$, como demonstrado para $P(M)$, a equivalência entre ponto de equilíbrio e solução local parcialmente estável, sendo este último conceito dado pela extensão direta da definição 3.3.2. Isto mostra a diferença entre solução local estável e parcialmente estável do problema penalizado $\tilde{P}(M)$.

Capítulo 4

Algoritmos propostos

A partir dos resultados teóricos derivados no capítulo anterior, propomos neste capítulo dois algoritmos para PLDNP: um local (seção 4.2) e outro global (seção 4.3). Ambos os algoritmos são baseados no conceito de pontos de equilíbrio, que são encontrados através de procedimentos propostos na seção 4.1. O segundo algoritmo também utiliza uma modificação de um método de otimização global chamado *outer approximation*.

O desempenho dos algoritmos local e global é avaliado na seção 4.4, para uma série de problemas-teste gerados aleatoriamente. A subseção 4.4.1 descreve os problemas utilizados e a subseção 4.4.2 apresenta os resultados obtidos.

4.1 Algoritmos para pontos de equilíbrio

Conforme estabelecem os teoremas 3.3.3 e 3.3.2, os ótimos locais de PLDNP encontram-se entre os pontos de equilíbrio do problema penalizado $P(M)$, que são, na verdade, soluções locais de (P) . Nesta seção propomos algoritmos para determinar um ponto de equilíbrio.

Recorde-se, da definição 3.3.1, que um ponto de equilíbrio (\bar{z}, \bar{s}) de $P(M)$ deve verificar a condição

$$\max_{s \in S} F_M(\bar{z}, s) = F_M(\bar{z}, \bar{s}) = \max_{z \in Z} F_M(z, \bar{s}),$$

a partir de um certo valor \bar{M} de M . Assim, \bar{z} e \bar{s} são soluções dos seguintes problemas lineares paramétricos

$$\begin{aligned} P(M; \bar{s}) \quad & \max \quad F_M(z, \bar{s}) = c^T z - M \bar{s}^T z \\ \text{s.a} \quad & z \in Z = \{z \geq 0 : Az = a\} \end{aligned}$$

$$P(M; \bar{z}) \quad \max \quad F_M(\bar{z}, s) = c^T \bar{z} - M \bar{z}^T s \\ \text{s.a.} \quad s \in S = \{s \geq 0 : Ds = d\}$$

Observe-se que o ponto solução de $P(M; \bar{z})$ independe do parâmetro M , pois o termo $c^T \bar{z}$ é uma constante. Logo, pode-se resolver este problema, por exemplo, pelo método simplex clássico. Ressalte-se ainda que $P(M; \bar{z})$ não pode ser ilimitado, uma vez que $\bar{z}^T s \geq 0$ para todo $s \in S$.

Quanto ao problema $P(M; \bar{s})$, embora sua solução dependa de M , este parâmetro pode ser considerado implicitamente um valor dominante como, por exemplo, no método simplex *big-M* (ver Bazaraa *et al.* [16, seção 4.3]). Tal consideração leva em conta que o equilíbrio deve manter-se para todo $M \geq \bar{M}$. Desta forma, quando $P(M; \bar{s})$ é ilimitado, para um valor dominante M , tem-se que $P(M)$ é ilimitado para todo $M \geq 0$. Neste caso, o teorema 3.2.1 garante que (P) é também ilimitado.

Na verdade, um ponto de equilíbrio pode ser caracterizado sem a utilização do parâmetro M , como demonstramos no teorema 4.1.1 abaixo, a partir do seguinte resultado auxiliar.

Lema 4.1.1 *Se $\bar{s} \in S$ e $\bar{z} \in \arg \max\{c^T z : z \in Z, \bar{s}^T z = 0\}$, então existe $M_0 \geq 0$ tal que*

$$\sup\{c^T z - M \bar{s}^T z : z \in Z\} < +\infty \quad \forall M \geq M_0.$$

Prova: Seja $\bar{s} \in S$ e $\bar{z} \in \arg \max\{c^T z : z \in Z, \bar{s}^T z = 0\}$. Suponha-se, por absurdo, que, para todo $M_0 \geq 0$, existe $M \geq M_0$ tal que

$$\sup\{c^T z - M \bar{s}^T z : z \in Z\} = +\infty.$$

Assim, a equação acima verifica-se para todo $M \geq 0$. Então, aplicando-se raciocínio semelhante à primeira parte da demonstração do lema 3.2.3 com os problemas (\mathcal{P}) e $\mathcal{P}(M)$ dados por $Z = Z_r$, $S = \{\bar{s}\}$ e $\mathcal{F}(z, s) = c^T z$ no lema 3.2.1, deduz-se que existe uma direção de recessão $h \in Z_r$ tal que

$$\bar{s}^T h = 0 \quad \text{e} \quad c^T h > 0.$$

Logo, $\sup\{c^T z : z \in Z, \bar{s}^T z = 0\} = +\infty$, contradizendo a hipótese. ■

Teorema 4.1.1 *O ponto (\bar{z}, \bar{s}) é um ponto de equilíbrio do problema penalizado $P(M)$ se, e somente se,*

$$\bar{s} \in S \quad \text{e} \quad \bar{z} \in \arg \max\{c^T z : z \in Z, \bar{s}^T z = 0\}.$$

Prova: Seja (\bar{z}, \bar{s}) um ponto de equilíbrio de $P(M)$. Pela definição 3.3.1, $(\bar{z}, \bar{s}) \in Z \times S$ e

$$c^T \bar{z} = \max\{c^T z - \bar{M} \bar{s}^T z : z \in Z\}$$

para algum $\bar{M} \geq 0$. Mais ainda, pela proposição 3.3.1, $\bar{s}^T \bar{z} = 0$. Logo,

$$c^T \bar{z} \leq \max\{c^T z : z \in Z, \bar{s}^T z = 0\} \leq \max\{c^T z - \bar{M} \bar{s}^T z : z \in Z\} = c^T \bar{z},$$

mostrando que $\bar{z} \in \arg \max\{c^T z : z \in Z, \bar{s}^T z = 0\}$.

Reciprocamente, assumamos que

$$\bar{z} \in \arg \max\{c^T z : z \in Z, \bar{s}^T z = 0\}. \quad (4.1)$$

Então, $\bar{z} \in Z$ e $\bar{s}^T \bar{z} = 0$. Como $\bar{z}^T s \geq 0 = \bar{z}^T \bar{s}$ para todo $s \in S$, conclui-se que

$$\max\{c^T \bar{z} - M \bar{z}^T s : s \in S\} = c^T \bar{z} - M \bar{z}^T \bar{s} \quad \forall M \geq 0.$$

Assim, pela definição 3.3.1 e levando-se em conta que $\bar{s}^T \bar{z} = 0$, resta mostrar que existe $\bar{M} \geq 0$ tal que

$$\max\{c^T z - M \bar{s}^T z : z \in Z\} = c^T \bar{z} \quad \forall M \geq \bar{M}. \quad (4.2)$$

De fato, o lema 4.1.1 garante a existência do máximo em (4.2) a partir de certo $M_0 \geq 0$. Assim,

$$\max\{c^T z - M \bar{s}^T z : z \in Z\} = \max\{c^T z - M \bar{s}^T z : z \in Z_v\} \quad (4.3)$$

para cada $M \geq M_0$. Considere-se, então, $Z_v(\bar{s}) = Z_v \cap Z(\bar{s})$ o conjunto de vértices de Z complementares a \bar{s} . Se $Z_v \setminus Z_v(\bar{s}) \neq \emptyset$, tome-se $\bar{M} \geq M_0$ com

$$\bar{M} > \max\{(c^T z - c^T \bar{z}) / \bar{s}^T z : z \in Z_v \setminus Z_v(\bar{s})\}.$$

Assim, para todo $M \geq \bar{M}$, segue-se que

$$c^T z - M \bar{s}^T z < c^T \bar{z} \quad \forall z \in Z_v \setminus Z_v(\bar{s}).$$

Logo, para todo $M \geq \bar{M}$, tem-se

$$\begin{aligned} \max\{c^T z - M \bar{s}^T z : z \in Z_v\} &= \max\{c^T z - M \bar{s}^T z : z \in Z_v(\bar{s})\} \\ &= \max\{c^T z : z \in Z, \bar{s}^T z = 0\}. \end{aligned}$$

Caso $Z_v \setminus Z_v(\bar{s}) = \emptyset$, obtêm-se trivialmente as igualdades acima para todo $M \geq M_0$. Assim, a expressão desejada (4.2) segue-se destas igualdades, de (4.1) e (4.3). ■

A seguir propomos dois algoritmos para obter um ponto de equilíbrio, partindo de um ponto inicial $z^0 \in Z$ ou $s^0 \in S$. Na verdade, embora a caracterização do teorema 4.1.1 não seja usada na descrição do procedimento, ela será útil na demonstração de convergência. A descrição é feita em função dos problemas paramétricos, para se adequar à formulação do algoritmo local apresentado na seção seguinte.

Algoritmo 1 (Ponto de equilíbrio)

Passo 0 Sejam $Z \times S \neq \emptyset$ e $z^0 \in Z$.

Passo 1 Resolva $P(M; z^0)$ pelo método simplex, obtendo uma solução \bar{s} .

Passo 2 Resolva $P(M; \bar{s})$, pelo método simplex *big-M*. Obtenha uma solução \bar{z} ou verifique que o problema é ilimitado. No primeiro caso, (\bar{z}, \bar{s}) é um ponto de equilíbrio; no segundo, conclua que $P(M)$ é ilimitado para todo $M \geq 0$.

Mencione-se que este algoritmo fará parte do algoritmo local (ver próxima seção), que fornecerá as condições requeridas pelo passo 0, ou seja, verificará se Z ou S são vazios ou obterá um ponto inicial z^0 . Note-se ainda que o problema $P(M; z^0)$ do passo 1 não pode ser ilimitado, pois $(z^0)^T s \geq 0$ para todo $s \in S$.

A demonstração de otimalidade do algoritmo 1 baseia-se no seguinte resultado.

Lema 4.1.2 *Sejam $\bar{s} \in S$ e \bar{z} solução do problema paramétrico $P(M; \bar{s})$. Se $Z(\bar{s}) \neq \emptyset$, então*

(i) $\bar{s}^T \bar{z} = 0$ e

(ii) (\bar{z}, \bar{s}) é um ponto de equilíbrio.

Prova: Sejam \bar{z} e \bar{s} satisfazendo as hipóteses e $\hat{z} \in Z(\bar{s})$.

(i) Suponha-se, por absurdo, que $\bar{s}^T \bar{z} > 0$. Tome-se, então, $\hat{M} \geq 0$ tal que

$$\hat{M} > (c^T \bar{z} - c^T \hat{z}) / \bar{s}^T \bar{z}.$$

Logo, para todo $M \geq \hat{M}$, tem-se

$$c^T \bar{z} - M \bar{s}^T \bar{z} < c^T \hat{z} = c^T \hat{z} - M \bar{s}^T \hat{z},$$

o que contradiz o fato de \bar{z} ser solução de $P(M; \bar{s})$.

(ii) Como \bar{z} é solução de $P(M; \bar{s})$ e $\bar{s}^T \bar{z} = 0$, tem-se que

$$c^T \bar{z} = \max\{c^T z - M \bar{s}^T z : z \in Z\}, \quad (4.4)$$

para M suficientemente grande. Por outro lado,

$$\max\{c^T z - M \bar{s}^T z : z \in Z\} \geq \max\{c^T z : z \in Z, \bar{s}^T z = 0\} \geq c^T \bar{z}.$$

para todo $M \geq 0$. Assim, por (4.4), as igualdades se verificam na expressão acima.

E pelo teorema 4.1.1, (\bar{z}, \bar{s}) é um ponto de equilíbrio. ■

Proposição 4.1.1 *O algoritmo 1 termina com uma das seguintes possibilidades:*

- (i) *encontra um ponto de equilíbrio do problema penalizado $P(M)$ e, equivalentemente, uma solução local de (P) , ou*
- (ii) *verifica que $P(M)$ é ilimitado para todo $M \geq 0$ e, equivalentemente, que (P) é ilimitado.*

Prova: Claramente o algoritmo pára com um ponto (\bar{z}, \bar{s}) ou verifica que $P(M; \bar{s})$ é ilimitado para um valor dominante de M . Neste último caso, segue-se que $P(M)$ é ilimitado para todo $M \geq 0$. Considere-se, agora, o primeiro caso. Como \bar{s} é solução de $P(M; z^0)$, tem-se que $\bar{s} \in \arg \min\{(z^0)^T s : s \in S\}$. Então, pelo corolário 3.1.1, deduz-se que $Z(\bar{s}) \neq \emptyset$. Assim, visto que \bar{z} é solução de $P(M; \bar{s})$, o lema 4.1.2 garante que (\bar{z}, \bar{s}) é um ponto de equilíbrio. As equivalências com (P) são dadas pelos teoremas 3.2.1 e 3.3.2. ■

Analogamente ao algoritmo 1, pode-se definir outro procedimento para determinar um ponto de equilíbrio, a partir de um ponto inicial em $s^0 \in S$.

Algoritmo 2 (Ponto de equilíbrio)

Passo 0 Sejam $Z \times S \neq \emptyset$ e $s^0 \in S$.

Passo 1 Resolva $P(M; s^0)$, pelo método simplex *big-M*. Obtenha uma solução \hat{z} ou verifique que o problema é ilimitado. No segundo caso, conclua que $P(M)$ é ilimitado para todo $M \geq 0$.

Passo 2 Resolva $P(M; \hat{z})$ pelo método simplex, obtendo uma solução \bar{s} .

Passo 3 Resolva $P(M; \bar{s})$, pelo método simplex *big-M*. Obtenha uma solução \bar{z} ou verifique que o problema é ilimitado. No primeiro caso, (\bar{z}, \bar{s}) é um ponto de equilíbrio; no segundo, conclua que $P(M)$ é ilimitado para todo $M \geq 0$.

Na verdade, considerando-se o lema 4.1.2, pode-se concluir que (\hat{z}, s^0) é um ponto de equilíbrio, já no passo 1, se $\hat{z}^T s^0 = 0$. De qualquer modo, esta situação será verificada posteriormente no passo 3. Preferiu-se não introduzir tal teste no passo 1 porque na descrição do algoritmo local, que usa o algoritmo 2, vai-se considerar os quadros simplex relativos a ambos os problemas $P(M; \bar{s})$ e $P(M; \bar{z})$.

Por raciocínio similar à demonstração da proposição 4.1.1, deduz-se que:

Proposição 4.1.2 *O algoritmo 2 termina com uma das seguintes possibilidades:*

- (i) *encontra-se um ponto de equilíbrio do problema penalizado $P(M)$ e, equivalentemente, uma solução local de (P) , ou*
- (ii) *verifica-se que $P(M)$ é ilimitado para todo $M \geq 0$ e, equivalentemente, que (P) é ilimitado.*

Note-se que os algoritmos 1 e 2 determinam um ponto de equilíbrio para o problema penalizado $P(M)$. Por outro lado, o conceito de ponto de equilíbrio foi estendido, na seção 3.4, para o problema penalizado $\tilde{P}(M)$, que está associado ao problema com restrições no primeiro nível (PLDN).

A seguir apresentamos um algoritmo para encontrar um ponto de equilíbrio de $\tilde{P}(M)$, a partir de um ponto $s^0 \in S$. Procedimento similar, pode ser definido partindo-se de um ponto $z^0 \in Z$. Na descrição abaixo, $\tilde{P}(M; \bar{s})$ e $\tilde{P}(M; \bar{z})$ denotam o problema $\tilde{P}(M)$ com $s = \bar{s}$ e $z = \bar{z}$ respectivamente.

Algoritmo 3 (Ponto de equilíbrio)

Passo 0 Sejam $Z \times S \neq \emptyset$ e $s^0 \in S$. Faça $i = 0$.

Passo 1 Resolva $\tilde{P}(M; s^i)$ pelo método simplex *big-M*. Obtenha uma solução z^i ou verifique que o problema é ilimitado. No segundo caso, conclua que $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$.

Passo 2 Resolva $\tilde{P}(M; z^i)$, obtendo uma solução s^{i+1} . Se $F_M(z^i, s^i) = F_M(z^i, s^{i+1})$, pare: (z^i, s^i) é um ponto de equilíbrio.

Passo 3 Resolva $\tilde{P}(M; s^{i+1})$ pelo método simplex *big-M*. Obtenha uma solução z^{i+1} ou verifique que o problema é ilimitado. No segundo caso, conclua que $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$. No primeiro, se $F_M(z^i, s^{i+1}) = F_M(z^{i+1}, s^{i+1})$, pare: (z^i, s^{i+1}) é um ponto de equilíbrio; senão, faça $i = i + 1$ e volte ao passo 2.

Este procedimento generaliza o algoritmo *mountain climbing* proposto por Konno [75] para encontrar pontos satisfazendo as condições de Karush-Kuhn-Tucker para um problema bilinear com conjunto viável compacto. A generalização se dá no sentido de que não se pede compacidade e de que um “equilíbrio paramétrico” é considerado.

A demonstração de convergência do algoritmo 3 usa o seguinte resultado.

Lema 4.1.3 *Sejam $\bar{s} \in S$ e \bar{z} solução do problema paramétrico $\tilde{P}(M; \bar{s})$. Então, $\bar{s}^T z \geq \bar{s}^T \bar{z}$ para todo $z \in \tilde{Z}$.*

Prova: Sejam \bar{z} e \bar{s} satisfazendo as hipóteses. Suponha-se, por absurdo, que existe $\hat{z} \in \tilde{Z}$ tal que $\bar{s}^T \hat{z} < \bar{s}^T \bar{z}$. Tome-se, então, $\hat{M} \geq 0$ tal que

$$\hat{M} > (c^T \bar{z} - c^T \hat{z}) / (\bar{s}^T \bar{z} - \bar{s}^T \hat{z}).$$

Logo, para todo $M \geq \hat{M}$, tem-se

$$c^T \bar{z} - M \bar{s}^T \bar{z} < c^T \hat{z} - M \bar{s}^T \hat{z},$$

o que contradiz o fato de \bar{z} ser solução de $\tilde{P}(M; \bar{s})$. ■

Proposição 4.1.3 *O algoritmo 3 pára em um número finito de iterações.*

Prova: Suponha-se, por absurdo, que o algoritmo 3 gera uma sequência infinita de pontos $\{(z^i, s^i)\}$. Posto que o algoritmo enumera vértices de $\tilde{Z} \times S$ e que o número de vértices é finito, existem pontos repetidos na sequência. Sejam, então, dois pontos (z^i, s^i) e (z^{i+1}, s^{i+1}) gerados em iterações consecutivas. Então, pelo passo 2, conclui-se que $F_M(z^i, s^i) \neq F_M(z^i, s^{i+1})$. E como s^{i+1} é solução de $\tilde{P}(M; z^i)$, tem-se que $F_M(z^i, s^i) < F_M(z^i, s^{i+1})$, ou seja,

$$F_M(z^i, s^i) = c^T z^i - M (z^i)^T s^i < c^T z^i - M (z^i)^T s^{i+1} = F_M(z^i, s^{i+1}).$$

Considerando-se que $M \geq 0$, segue-se que

$$(z^i)^T s^i > (z^i)^T s^{i+1}. \quad (4.5)$$

Mais ainda, como z^{i+1} é solução de $\tilde{P}(M; s^{i+1})$ (passo 3), deduz-se, do lema 4.1.3, que

$$(z^i)^T s^{i+1} \geq (z^{i+1})^T s^{i+1}. \quad (4.6)$$

Por (4.5)-(4.6), conclui-se que os pontos da sequência não se repetem: uma contradição. ■

Proposição 4.1.4 *O algoritmo 3 encontra um ponto de equilíbrio do problema penalizado $\tilde{P}(M)$ ou verifica que $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$.*

Prova: Pela proposição 4.1.3, o algoritmo pára eventualmente. Suponha-se, a princípio, que a parada ocorre no passo 1 ou 3, porque se detecta que $\tilde{P}(M; s)$ é ilimitado para um valor dominante de M . Neste caso, $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$. Outra possibilidade do algoritmo terminar acontece, no passo 2, com a verificação de que $F_M(z^i, s^i) = F_M(z^i, s^{i+1})$. Note-se que z^i é solução $\tilde{P}(M; s^i)$ e s^{i+1} é solução de $\tilde{P}(M; z^i)$. Então, tem-se

$$F_M(z^i, s^i) \geq F_M(z, s^i) \quad \forall z \in \tilde{Z}$$

e

$$F_M(z^i, s^i) = F_M(z^i, s^{i+1}) \geq F_M(z^i, s) \quad \forall s \in S.$$

Logo, (z^i, s^i) é um ponto de equilíbrio. Finalmente, o último critério de parada encontra-se no passo 3, quando $F_M(z^i, s^{i+1}) = F_M(z^{i+1}, s^{i+1})$. Por desenvolvimento similar ao do caso anterior, conclui-se que (z^i, s^{i+1}) é um ponto de equilíbrio. ■

Recorde-se que um ponto de equilíbrio do problema $\tilde{P}(M)$ pode não satisfazer as condições de complementaridade (ver figura 3.6). Assim sendo, o algoritmo 3 não encontra ótimos locais de (\tilde{P}) .

4.2 Algoritmo local

A partir de um ponto de equilíbrio (\bar{z}, \bar{s}) do problema penalizado $P(M)$, podem-se avaliar as condições de otimalidade derivadas na subseção 3.3.3, que permitem

descobrir se \bar{z} é uma solução local de PLDNP ou fornecem uma direção para sair do ponto de equilíbrio, melhorando a função objetivo. Estas condições são lembradas a seguir, juntamente com a notação.

Considere-se que $(\bar{z}, \bar{s}) \in Z_v \times S_v$ seja um ponto de equilíbrio encontrado pelo algoritmo 1 ou 2, como solução dos problemas lineares paramétricos $P(M; \bar{s})$ e $P(M; \bar{z})$. Seja B uma base associada a \bar{z} e particione-se $A = [B \ N]$. Similarmente, considere-se $D = [E \ R]$, onde R é uma matriz básica relativa a \bar{s} . Defina-se ainda J_B, J_N e J_E, J_R os índices das variáveis em z e s correspondentes às respectivas submatrizes.

Os quadros simplex associados às soluções ótimas dos problemas $P(M; \bar{s})$ e $P(M; \bar{z})$ são apresentados abaixo, conforme descrição discutida na subseção 3.3.3.

	z_B^T	z_N^T		s_E^T	s_R^T	
	I	$B^{-1}N$	$\bar{z}_B = B^{-1}a$	I	$E^{-1}R$	$\bar{s}_E = E^{-1}d$
M	0	$-\bar{s}_N^T + \bar{s}_B^T B^{-1}N$	0	0	$-\bar{z}_R^T + \bar{z}_E^T E^{-1}R$	0
	0	$c_N^T - c_B^T B^{-1}N$	$-c_B^T B^{-1}a$			

O subconjunto de variáveis não básicas J_N^+ que podem incrementar a função objetivo do líder ao entrar na base é determinado a partir dos custos reduzidos da última linha do quadro simplex associado a $P(M; \bar{s})$. Efetivamente,

$$J_N^+ = \{i \in J_N : c_i - c_B^T B^{-1}N_i > 0\}, \quad (4.7)$$

onde N_i é a i -ésima coluna da matriz N .

Já as variáveis não básicas que podem gerar soluções alternativas complementares a \bar{z} tem custos reduzidos nulo no quadro simplex relativo a $P(M; \bar{z})$, definindo assim o subconjunto

$$J_R^0 = \{j \in J_R : \bar{z}_j - \bar{z}_E^T E^{-1}R_j = 0\}, \quad (4.8)$$

onde R_j a j -ésima coluna de R .

Associada às colunas não básicas de $A \in \mathfrak{R}^{m_2 \times n}$, define-se a matriz

$$G = \begin{bmatrix} -B^{-1}N \\ I \end{bmatrix} \in \mathfrak{R}^{n \times (n-m_2)}. \quad (4.9)$$

As direções extremas de $Z = \{z \geq 0 : Az = a\}$ a partir de \bar{z} são combinações lineares não negativas das colunas G_i ($i = 1, 2, \dots, n - m_2$) de G . Em particular, se \bar{z} é não degenerado ($\bar{z}_B > 0$), tais direções são exatamente as colunas de G .

No que se segue, considere-se a hipótese de não degeneração abaixo:

[D] Se $i \in J_N^+$ e $G_{ki} < 0$, então $\bar{z}_k > 0$.

Sob esta suposição, as seguintes condições de otimalidade foram demonstradas na subseção 2.5.2:

(1) O ponto \bar{z} é uma solução local de PLDNP se, e somente se, $J_N^+ = \emptyset$ ou

$$\delta_i = \min\{G_i^T s : s \in S(\bar{z}), s^T = (s_B^T, s_N^T)\} > 0 \quad (4.10)$$

para todo $i \in J_N^+$ (note-se que $\delta_i \geq 0$ sempre).

(2) Se $J_N^+ = \emptyset$, então \bar{z} é uma solução global de PLDNP (\bar{z} é solução do problema relaxado);

(3) Se $J_R^0 = \emptyset$, então \bar{z} é uma solução local de PLDN (\bar{s} é o único ponto de S complementar a \bar{z});

(4) Se existem $i \in J_N^+$ e $s \in S(\bar{z})$ tais que $G_i^T s = 0$, então \bar{z} não é solução local e (\bar{z}, s) não é ponto de equilíbrio. Neste caso, existe $z \in Z(s)$, $z = \bar{z} + z_i G_i$, com $z_i > 0$ e $c^T z > c^T \bar{z}$ (a variável z_i pode entrar na base para gerar um ponto viável de PLDNP melhor que \bar{z}).

Observe-se que as propriedades (2)-(4) são decorrentes da caracterização (1). Entretanto, elas estão evidenciadas por suas implicações algorítmicas:

- (i) a verificação de que J_N^+ ou J_R^0 são vazios é simples e evita a resolução dos problema lineares definidos em (4.10) e
- (ii) a última condição mostra que qualquer $i \in J_N^+$ com $\delta_i = 0$ pode ser usado para iniciar a geração de um novo ponto de equilíbrio.

Para se calcular cada valor δ_i , pode-se considerar o quadro simplex associado a $P(M; \bar{z})$ e, similarmente ao que se faz no quadro simplex relativo a $P(M; \bar{s})$, inclui-se outra linha de função objetivo referente a $G_i^T s$. Então, permite-se que entre na

base apenas as variáveis indexadas por J_R^0 . Isto garante a obediência à restrição $s \in S(\bar{z})$.

Feitas estas considerações, descrevemos a seguir o algoritmo local proposto.

Algoritmo (Algoritmo local)

Passo 1 Se $Z \times S = \emptyset$, pare: PLDNP é inviável. Senão encontre $\hat{s} \in S$.

Passo 2 Aplique o algoritmo 2 de equilíbrio a partir de \hat{s} . Se $P(M)$ é ilimitado para todo $M \geq 0$, pare: PLDNP é ilimitado. Do contrário, seja (\bar{z}, \bar{s}) o ponto de equilíbrio encontrado.

Passo 3 Considere os quadros simplex relativos aos problemas $P(M; \bar{s})$ e $P(M; \bar{z})$. Sejam J_N^+ e J_R^0 como definidos em (4.7) e (4.8). Se $J_N^+ = \emptyset$ ou $J_R^0 = \emptyset$, pare: \bar{z} é ótimo global ou local de PLDNP.

Passo 4 Calcule δ_i como definido em (4.10). Se existem $i \in J_N^+$ e $\hat{s} \in S(\bar{z})$ tais que $\delta_i = G_i^T \hat{s} = 0$, volte ao passo 2. Do contrário, pare: \bar{z} é um ótimo local de PLDNP.

No passo 1, a verificação de que Z ou S são ou não vazios pode ser realizada através da primeira fase do método simplex. Desta forma também se encontra o ponto inicial $\hat{s} \in S$. Na verdade, pode-se alternativamente começar o algoritmo local a partir de um ponto $\hat{z} \in Z$ e aplicar o algoritmo 1 na primeira execução do passo 2. Em particular, \hat{z} pode ser a solução do problema relaxado, caso exista.

Proposição 4.2.1 *O algoritmo local pára em um número finito de iterações.*

Prova: Suponha-se, por absurdo, que o algoritmo local não satisfaz os critérios de parada. Observe-se que este fato não se deve à avaliação dos conjuntos J_N^+ e J_R^0 no passo 3 ou ao cálculo dos δ_i no passo 4. Mais ainda, o algoritmo de equilíbrio é finito e enumera vértices de $Z \times S$. Assim, conclui-se que há pontos repetidos na sequência de pontos de equilíbrio gerada. Por outro lado, encontrado um ponto de equilíbrio (\bar{z}, \bar{s}) , uma nova aplicação do algoritmo de equilíbrio ocorre, no passo 4, quando existem $i \in J_N^+$ e $\hat{s} \in S(\bar{z})$ tais que $G_i^T \hat{s} = 0$. Assim, pelo corolário 3.3.3, existe $\hat{z} \in Z$ tal que

$$\hat{s}^T \hat{z} = 0 \quad \text{e} \quad c^T \hat{z} > c^T \bar{z}. \quad (4.11)$$

Considere-se, então, a nova aplicação do algoritmo 2 a partir de \hat{s} . Como $Z(\hat{s}) \neq \emptyset$, deduz-se da descrição do algoritmo 2, do lema 4.1.2 e do teorema 4.1.1 que o novo ponto de equilíbrio (\bar{z}', \bar{s}') satisfaz

$$\bar{s}' = \hat{s} \quad \text{e} \quad \bar{z}' \in \arg \max\{c^T z : z \in Z, \hat{s}^T z = 0\}.$$

Logo, segue-se de (4.11) que

$$c^T \bar{z}' \geq c^T \hat{z} > c^T \bar{z}.$$

Portanto, os pontos de equilíbrio nunca se repetem, o que é uma contradição. ■

Proposição 4.2.2 *O algoritmo local encontra uma solução local de PLDNP ou verifica que o problema é ilimitado ou inviável.*

Prova: Pela proposição 4.2.1, o algoritmo local pára eventualmente. Em particular, os critérios de parada que detectam inviabilidade (passo 1) ou ilimitação (passo 2) de PLDNP são corretos pela equivalência global deste problema com (P) e pelo item (ii) da proposição 4.1.2. Por outro lado, a corretude dos teste de otimalidade local dos passos 3 e 4 é assegurada com o teorema 3.3.7 e o lema 3.3.7. ■

Mencione-se que o algoritmo local não pode ser estendido para o problema PLDN, usando o algoritmo 3 em lugar do algoritmo 2. De fato, como observamos na subseção anterior, um ponto de equilíbrio obtido pelo algoritmo 3 pode não corresponder a uma solução viável de PLDN. Tal situação pode acontecer quando existem restrições do líder que dependem da variável y (ver figura 3.6).

4.3 Algoritmo global

Neste seção propomos um algoritmo iterativo para encontrar ε -soluções globais de PLDNP, que segue a mesma estratégia do algoritmo local apresentado anteriormente, ou seja, determina pontos de equilíbrio do problema penalizado $P(M)$. Além disso, o algoritmo proposto incorpora um procedimento de otimização global chamado *outer approximation*, adaptado a sua necessidade.

A iteração inicial ($k = 0$) encontra um ponto de equilíbrio (z^0, s^0) de $P(M)$, que fornece um limite inferior $c^T z^0$ para PLDNP. Cada iteração k seguinte ($k \geq 1$),

passa a considerar o problema $\tilde{P}(M)$, obtido de $P(M)$ com o acréscimo da restrição linear

$$c^T z \geq F^* + \varepsilon, \quad (4.12)$$

onde $\varepsilon > 0$ e F^* é o melhor limite inferior conhecido até a iteração $k - 1$. Este tipo de corte é usado por Júdice e Faustino [73], Önal [94], Tuy *et al.* [111] e Amouzegar e Moshirvaziri [2].

O problema $\tilde{P}(M)$ é gerado, assim, pela penalização das condições de complementaridade do problema (\tilde{P}) , definido por (P) restrito a (4.12). Em outros termos, têm-se os problemas

$$\begin{array}{ll} (\tilde{P}) & \max F(z, s) = c^T z \\ & \text{s.a. } z \in \tilde{Z}, s \in S \\ & s^T z = 0 \end{array} \quad \tilde{P}(M) \quad \max F_M(z, s) = c^T z - M s^T z \\ & \text{s.a. } z \in \tilde{Z}, s \in S$$

onde

$$\begin{aligned} \tilde{Z} &= Z \cap \{z \in \mathfrak{R}^n : c^T z \geq F^* + \varepsilon\} \\ &= \{z \in \mathfrak{R}^n : c^T z \geq F^* + \varepsilon, Az = a, z \geq 0\} \\ S &= \{s \in \mathfrak{R}^n : Ds = d, s \geq 0\}. \end{aligned}$$

O problema (\tilde{P}) pode ser visto como sendo a reformulação de um problema em dois níveis com a restrição (4.12) no primeiro nível, ou seja, com $B_1 = c_1^T$, $B_2 = c_2^T$ e $b = F^* + \varepsilon$. Assim, os problemas (\tilde{P}) e $\tilde{P}(M)$ acima estão definidos de acordo com a seção 3.4.

Como visto nessa referida seção, um ponto de equilíbrio de $\tilde{P}(M)$ pode não satisfazer a complementaridade de folgas, não sendo viável para (\tilde{P}) . Neste caso, uma forma de se encontrar uma solução viável para (\tilde{P}) é resolver o seguinte problema bilinear

$$\begin{array}{ll} (PC) & \min s^T z \\ & \text{s.a. } (z, s) \in \tilde{Z} \times S \end{array}$$

Observe-se que (PC) não pode ser ilimitado, pois $z^T s \geq 0$ para todo $(z, s) \in \tilde{Z} \times S$. Se (PC) é inviável ou seu valor ótimo é não nulo, então (\tilde{P}) é inviável. Do contrário, qualquer solução ótima (\hat{z}, \hat{s}) é viável para (\tilde{P}) .

Na verdade, como o conjunto viável de PLDNP é conexo, se ainda existe solução viável em \tilde{Z} , pelo menos uma delas está sobre o corte, ou seja, satisfaz a igualdade

$c^T z = F^* + \varepsilon$. Assim, o problema (PC) é viável se, e somente se, o problema (PC') , formulado a seguir, é viável.

$$(PC') \quad \begin{aligned} \min \quad & s^T z \\ \text{s.a} \quad & z \in Z' = \{z \geq 0 : A'z = a'\} \\ & s \in S \end{aligned}$$

onde $A' = \begin{bmatrix} A \\ c^T \end{bmatrix}$ e $a' = \begin{bmatrix} a \\ F^* + \varepsilon \end{bmatrix}$.

Duas são as razões para se considerar (PC') em lugar de (PC) . Primeiro, o espaço de soluções de (PC') é menor. Além disso, Z' é compacto se as curvas de nível da função do líder $c^T z$ são limitadas sobre Z . Estas duas propriedades são interessantes para o método de solução a ser empregado. Destaque-se ainda que a descrição do conjunto viável $Z' \times S$ através de restrições de igualdade facilita o desenvolvimento que se segue.

Por ser um problema bilinear, uma solução global de (PC') , se existir, ocorre em um vértice de $Z' \times S$ (Horst e Tuy [69, proposição IX.1]). De fato, (PC') pode ser visto como a minimização da função côncava

$$g(z) = \inf\{z^T s : s \in S\}$$

sobre o poliedro Z' , ou seja, (PC') é equivalente a

$$(PC') \quad \min_{z \in Z'} g(z)$$

Note-se que a função g é côncava, tendo em vista que é definida pelo ínfimo de uma família de funções lineares (Rockafellar [97, teorema 5.5]). Mais ainda, se (PC') é viável, então g assume valores reais em Z' , pois $z^T s \geq 0$ para todo $(z, s) \in Z' \times S$.

Vários métodos de otimização global para resolver problemas bilineares são descritos por Horst e Tuy [69, capítulo IX]. Entre os métodos propostos, escolhemos a *outer approximation*, que é capaz de lidar com poliedros não compactos. Na verdade, vamos utilizar uma modificação deste método conveniente ao nosso contexto. Mencione-se que algoritmos tipo *outer approximation*, aplicados diretamente ao problema de programação linear em dois níveis, são desenvolvidos por Tuy *et al.* [111, 112], conforme descrito na subseção 2.6.2.

A idéia básica da *outer approximation* é minimizar a função côncava sobre uma série de aproximações poliedrais externas do conjunto viável, cada vez mais estreitas.

O poliedro de uma nova iteração é obtido do anterior pelo acréscimo de uma restrição. Para cada poliedro, determinam-se seus vértices e calcula-se o valor da função sobre eles, de modo a se obter o mínimo. Ressalte-se que os vértices do novo poliedro são obtidos através do conjunto de vértices, já conhecido, do poliedro anterior.

Uma explanação detalhada do funcionamento do método pode ser encontrada em Horst e Tuy [69, seção IX.1.5]. A seguir, apresentamos uma descrição específica no contexto do problema (PC')

Assuma-se que tenha sido encontrado um ponto de equilíbrio $(\bar{z}, \bar{s}) \in Z'_v \times S_v$, com $\bar{z}^T \bar{s} > 0$ e, por isso, deseja-se resolver (PC') . Seja B uma base associada a \bar{z} e particione-se $A' = [B \ N]$. Expressando-se as variáveis básicas z_B em função das não básicas $z \equiv z_N$, o problema (PC') é reescrito como:

$$\begin{aligned} (PC') \quad \min \quad & \bar{s}^T \bar{z} - s_B^T Q z + s_N^T z \\ \text{s.a} \quad & Q z \leq q \\ & s = (s_B^T, s_N^T)^T \in S \end{aligned}$$

onde $Q = B^{-1}N$ e $q = \bar{z}_B = B^{-1}a' \geq 0$.

Alternativamente, (PC') é dado pela formulação

$$\begin{aligned} (PC') \quad \min \quad & \tilde{g}(z) \\ \text{s.a} \quad & z \in Z' = \{z \geq 0 : Q_i^T z \leq q_i \ i = 1, 2, \dots, m_2 + 1\} \end{aligned}$$

onde

$$\tilde{g}(z) = \min\{\bar{s}^T \bar{z} + z^T U^T s : s \in S\},$$

Q_i^T é a i -ésima linha da matriz Q e $U^T = [-Q^T \ I]$. Ressalte-se que \tilde{g} está bem definida com o mínimo, pois (PC') é viável ($(0, \bar{s}) \in Z' \times S$). Note-se ainda que

$$Z' \subseteq \mathfrak{R}^{n'}, \quad n' = n - m_2 - 1,$$

e que a origem de $\mathfrak{R}^{n'}$ é um vértice de Z' .

O método *outer approximation* para resolver (PC') , a partir de (\bar{z}, \bar{s}) , é descrito abaixo.

Algoritmo (*Outer Approximation*)

Passo 0 Construa um politopo (limitado) \mathcal{W}^1 contendo todos os vértices de Z' .

Calcule o conjunto de vértices \mathcal{W}_v^1 . Seja $J_1 = \{1, 2, \dots, m_2 + 1\}$ o conjunto de índices das restrições que definem Z' . Faça $l = 1$.

Passo 1 Escolha $\tilde{z} \in \arg \max\{\tilde{g}(z) : z \in \mathcal{W}_v^l\}$.

Passo 2 Se $Q_i^T \tilde{z} \leq q_i$ para todo $i \in J_l$, pare: (\tilde{z}, \tilde{s}) é solução de (PC') , onde \tilde{s} é a solução que fornece $\tilde{g}(\tilde{z})$, ou seja,

$$\tilde{s} \in \arg \min\{\tilde{s}^T \tilde{z} + \tilde{z}^T U^T s : s \in S\}.$$

(No espaço original, (\tilde{z}, \tilde{s}) é solução de (PC') , onde $\tilde{z}_N = \tilde{z}$ e $\tilde{z}_B = \bar{z}_B - Q\tilde{z}_N$).

Passo 3 Escolha $j \in \arg \max\{Q_j^T \tilde{z} - q_j : j \in J_l\}$. Faça $\mathcal{W}^{l+1} = \mathcal{W}^l \cap \{z : Q_j^T z \leq q_j\}$. Determine o conjunto de vértices \mathcal{W}_v^{l+1} a partir de \mathcal{W}_v^l . Faça $J_{l+1} = J_l \setminus \{j\}$, $l = l + 1$ e volte ao passo 1.

Note-se que este algoritmo termina em $m_2 + 1$ iterações no máximo, onde $m_2 + 1$ é o número de restrições que definem Z' (Horst e Tuy [69, observação IX.1]).

Claramente, é desejável que o politopo inicial \mathcal{W}^1 seja uma boa aproximação de Z' e que seja fácil calcular seu conjunto de vértices \mathcal{W}_v^1 . Em particular, quando Z' é limitado, esta tarefa é simples. De fato, neste caso, Z' está contido no simplex

$$\mathcal{W}^1 = \left\{ z \geq 0 : \sum_{i=1}^{n'} z_i \leq L \right\},$$

onde $L = \max\{\sum_{i=1}^{n'} z_i : z \in Z'\}$. Mais ainda, $\mathcal{W}_v^1 = \{0, Le_i (i = 1, 2, \dots, n')\}$, sendo e_i o i -ésimo vetor unitário em $\mathfrak{R}^{n'}$.

Em geral, se Z' é ilimitado e não é simples definir um politopo abrangendo Z'_v , pode-se tomar alternativamente \mathcal{W}^1 como sendo um poliedro (não limitado) contendo Z' . Neste caso, além dos vértices de \mathcal{W}^1 , deve-se também determinar suas direções de recessão (ver Horst e Tuy [69, algoritmo IX.4]). Em última instância, pode-se começar com \mathcal{W}^1 igual ao primeiro ortante de $\mathfrak{R}^{n'}$.

O novo conjunto \mathcal{W}_v^{l+1} , atualizado no passo 3, é formado pelos vértices de \mathcal{W}^l que satisfazem a restrição acrescentada $h(z) = Q_j^T z - q_j \leq 0$, juntamente com os vértices gerados pelo hiperplano $h(z) = 0$. Considerem-se, então, as seguintes definições:

$$\begin{aligned} \mathcal{W}_v^+ &= \{z \in \mathcal{W}_v^l : h(z) > 0\} \\ \mathcal{W}_v^- &= \{z \in \mathcal{W}_v^l : h(z) < 0\} \\ \bar{\mathcal{W}} &= \mathcal{W}^l \cap \{z \in \mathfrak{R}^{n'} : h(z) = 0\} \end{aligned}$$

Assim,

$$\mathcal{W}_v^{l+1} = \mathcal{W}_v^- \cup \bar{\mathcal{W}}_v.$$

O conjunto de vértices $\bar{\mathcal{W}}_v$ sobre o hiperplano $h(z) = 0$ pode ser determinado através de vários procedimentos (ver por exemplo Matheiss e Rubin [85], Horst *et al.* [70], Chen *et al.* [45], Horst *et al.* [68, seção 3.3.4]). Todos os procedimentos propostos para o caso onde \mathcal{W}^l é limitado se baseiam na seguinte caracterização (Horst *et al.* [68, teorema 3.1]):

Proposição 4.3.1 *Um ponto $z \in \bar{\mathcal{W}}_v$ se, e somente se, z é um vértice de \mathcal{W}^l em $\bar{\mathcal{W}}$ ou z é interseção de $\bar{\mathcal{W}}$ com uma aresta $[z^-, z^+]$ de \mathcal{W}^l , onde $z^- \in \mathcal{W}_v^-$ e $z^+ \in \mathcal{W}_v^+$.*

Uma versão desta propriedade englobando o caso em que \mathcal{W}^l é ilimitado, quando também devem ser consideradas as direções de recessão, é apresentada por Horst e Tuy [69, lema II.1].

O algoritmo proposto por Chen *et al.* [45] para encontrar $\bar{\mathcal{W}}_v$ mostra-se o mais efetivo dentre aqueles citados, conforme experimentos computacionais realizados pelos autores.

O princípio básico do algoritmo é usar listas de adjacências entre os vértices de \mathcal{W}^l para encontrar as arestas que interceptam $\bar{\mathcal{W}}$ e calcular os pontos de interseção. Para tal, mantém-se uma lista $\text{adj}(z)$ dos vértices adjacentes a $z \in \mathcal{W}^l$ e registra-se o conjunto $J(z)$ dos índices das restrições de \mathcal{W}^l ativas em z . A descrição abaixo assume não degeneração de \mathcal{W}^l .

Algoritmo (Chen *et al.* [45])

Passo 1 Determine \mathcal{W}_v^- e \mathcal{W}_v^+ . Faça $\bar{\mathcal{W}}_v = \emptyset$.

Passo 2 Para cada $z^- \in \mathcal{W}_v^-$ e $z^+ \in \mathcal{W}_v^+ \cap \text{adj}(z^-)$, determine $\alpha = h(z^+) / (h(z^+) - h(z^-))$ e o ponto interseção $z = \alpha z^- + (1 - \alpha)z^+$. Faça $\bar{\mathcal{W}}_v = \bar{\mathcal{W}}_v \cup \{z\}$, $\text{adj}(z^-) = \text{adj}(z^-) \setminus \{z^+\} \cup \{z\}$, $\text{adj}(z) = \{z^-\}$ e $J(z) = (J(z^-) \cap J(z^+)) \cup \{j\}$, onde $h(z) = Q_j^T z - q_j$.

Passo 3 Para cada $z' \in \bar{\mathcal{W}}_v$ e $z'' \in \bar{\mathcal{W}}_v$, com $z' \neq z''$ e $|J(z') \cap J(z'')| = n' - 1$, faça $\text{adj}(z') = \text{adj}(z') \cup \{z''\}$ e $\text{adj}(z'') = \text{adj}(z'') \cup \{z'\}$.

Observe-se que no passo 2 as adjacências são pesquisadas a partir dos vértices em \mathcal{W}_v^- . Esta estratégia deve ser preferida quando $|\mathcal{W}_v^-| \leq |\mathcal{W}_v^+|$. Do contrário, a pesquisa deve ser realizada a partir de \mathcal{W}_v^+ e os papéis de \mathcal{W}_v^- e \mathcal{W}_v^+ são trocados no algoritmo.

A construção das listas de adjacências dos novos vértices é necessária para a aplicação seguinte do procedimento. Lembre-se que ele está inserido no algoritmo *outer approximation*. Note-se ainda que as adjacências no politopo inicial \mathcal{W}^1 devem ser determinadas, o que é trivial no caso de um simplex.

O passo de maior complexidade do algoritmo é o terceiro. Para uma implementação eficiente, Chen *et al.* [45, pág. 405] sugerem o uso de códigos hash (Langsam *et al.* [77, seção 7.4]). Por outro lado, no caso degenerado, o teste $|J(z') \cap J(z'')| = n' - 1$ não é apropriado para detectar adjacência entre z' e z'' . Neste caso, os autores indicam duas alternativas: uma pequena perturbação no hiperplano $h(z) = 0$ ou a verificação de que existem $n' - 1$ restrições linearmente independentes em $J(z') \cap J(z'')$.

Como se pode perceber, a *outer approximation* é um procedimento computacionalmente caro, característica esperada de um método de otimização global. Em nosso contexto, porém, sua aplicação ao problema (PC') não pretende necessariamente determinar uma solução ótima. Na verdade, é suficiente encontrar uma solução $(\hat{z}, \hat{s}) \in \tilde{Z} \times S$ tal que $\hat{s}^T \hat{z} < \bar{s}^T \bar{z}$, de modo a poder reiniciar o algoritmo de equilíbrio.

Por este motivo, pode-se parar tão logo esta situação seja atingida, o que é possível de ser verificado com a introdução das seguintes modificações no algoritmo *outer approximation*.

Caso o teste de parada não tenha sido satisfeito no passo 2, vai-se calcular a interseção do segmento $[0, \tilde{z}]$ com a fronteira de Z' e verificar a complementaridade relativa ao ponto viável obtido. Com efeito, determina-se o ponto $\hat{z} = \alpha \tilde{z}$, onde α é o máximo passo que se pode dar nesta direção dentro do conjunto Z' . Dado que $\tilde{z} \geq 0$, α é o maior valor tal que $\alpha Q \tilde{z} \leq q$.

Como $q \geq 0$, se $Q_i^T \tilde{z} \leq 0$, então a restrição i não oferece limite para α . Esta condição se verifica, em particular, para todo $i \in J_1 \setminus J_l$. Caso $Q_i^T \tilde{z} > 0$, deve-se ter $\alpha \leq q_i / Q_i^T \tilde{z}$. Note-se que esta situação acontece sempre que o algoritmo não pára

no passo 2, pois $Q_i^T \tilde{z} > q_i \geq 0$ para algum $i \in J_l$. Assim, segue-se que

$$\alpha = \min_{i \in J_l} \left\{ \frac{q_i}{Q_i^T \tilde{z}} : Q_i^T \tilde{z} > 0 \right\}. \quad (4.13)$$

Adicionalmente, note-se que não é preciso calcular exatamente o argumento do máximo no passo 1. Isto implicaria a resolução de um problema linear para cada vértice em \mathcal{W}_v^l . Na verdade, quando se encontra um vértice \tilde{z} com $\tilde{g}(\tilde{z}) < \tilde{g}(0)$, pode existir um ponto viável $\hat{z} \in [0, \tilde{z}]$ que verifique a condição desejada.

Incluindo-se estas duas alterações no algoritmo *outer approximation* obtém-se o seguinte procedimento.

Algoritmo (*Outer Approximation Adaptado*)

Passo 0 Construa um polítopo (limitado) \mathcal{W}^1 contendo todos os vértices de Z' .

Calcule o conjunto de vértices \mathcal{W}_v^1 . Seja $J_1 = \{1, 2, \dots, m_2 + 1\}$ o conjunto de índices das restrições que definem Z' . Faça $l = 1$.

Passo 1 Encontre $\tilde{z} \in \mathcal{W}_v^l$ tal que $\tilde{g}(\tilde{z}) < \tilde{g}(0)$. Se não existe tal vértice, pare: $(0, \bar{s}) \in Z' \times S$ é solução de (PC') . Retorne $(\bar{z}, \bar{s}) \in Z' \times S$.

Passo 2 Se $Q_i^T \tilde{z} \leq q_i$ para todo $i \in J_l$, pare: $\tilde{z}^T \bar{s} < \bar{z}^T \bar{s}$, onde $\tilde{z}_N = \tilde{z}$, $\tilde{z}_B = \bar{z}_B - Q \tilde{z}_N$ e $\bar{s} \in \arg \min \{ \bar{s}^T \bar{z} + \tilde{z}^T U^T s : s \in S \}$. Retorne (\tilde{z}, \bar{s}) .

Passo 3 Calcule α conforme (4.13) e faça $\hat{z} = \alpha \tilde{z}$. Determine $\tilde{g}(\hat{z})$, obtendo $\hat{s} \in S$. Seja $\hat{z}_N = \hat{z}$ e $\hat{z}_B = \bar{z}_B - Q \hat{z}_N$ (note-se que $\tilde{g}(\hat{z}) = \hat{z}^T \hat{s}$). Se $\hat{z}^T \hat{s} < \bar{z}^T \bar{s}$, pare. Retorne (\hat{z}, \hat{s}) .

Passo 4 Escolha $j \in \arg \max \{ Q_i^T \tilde{z} - q_i : i \in J_l \}$. Faça $\mathcal{W}^{l+1} = \mathcal{W}^l \cap \{ z : Q_j^T z \leq q_j \}$. Determine o conjunto de vértices \mathcal{W}_v^{l+1} a partir de \mathcal{W}_v^l . Faça $J_{l+1} = J_l \setminus \{j\}$, $l = l + 1$ e volte ao passo 1.

Note-se que este procedimento adaptado também termina em um número finito de iterações. Realmente, as modificações introduzidas apenas podem antecipar o final do algoritmo.

O algoritmo completo para encontrar uma ε -solução global de PLDNP é descrito abaixo.

Algoritmo (Algoritmo Global)

Passo 1 Se $S = \emptyset$, vá para o passo 6. Senão encontre $\hat{s} \in S$ e considere $\tilde{Z} = Z$.
Faça $k = 0$.

Passo 2 Se $\tilde{Z} = \emptyset$, vá para o passo 6.

Passo 3 Aplique o algoritmo 3 de equilíbrio a partir de \hat{s} . Se $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$, pare: PLDNP é ilimitado. Do contrário, seja (\bar{z}, \bar{s}) o ponto de equilíbrio encontrado.

Passo 4 Se $\bar{s}^T \bar{z} = 0$, então (\bar{z}, \bar{s}) é solução viável de PLDNP; faça $(z^*, s^*) = (\bar{z}, \bar{s})$ e $F^* = F(\bar{z}, \bar{s})$; aplique um corte linear, definindo $\tilde{Z} = Z \cap \{z \in \mathfrak{R}^n : f_1(x, y) \geq F^* + \varepsilon\}$, com $\varepsilon > 0$; faça $\hat{s} = \bar{s}$ e $k = k + 1$; volte ao passo 2.

Passo 5 Se $\bar{s}^T \bar{z} > 0$, aplique o algoritmo *outer approximation* adaptado, obtendo (\hat{z}, \hat{s}) . Se $\hat{s}^T \hat{z} \geq \bar{s}^T \bar{z}$, vá para o passo 6. Senão, volte ao passo 3.

Passo 6 Pare. Se $k = 0$, PLDN é inviável. Senão, (x^*, y^*) é uma ε -solução global de PLDNP.

A constatação de que \tilde{Z} ou S são ou não vazios, bem como a obtenção do ponto inicial $\hat{s} \in S$, podem ser realizadas, como no algoritmo local, através da primeira fase do método simplex. Mais ainda, a partir da iteração $k = 1$, a verificação de que $\tilde{Z} \neq \emptyset$ é melhor efetuada com o método dual-simplex.

No passo 3, em vez do algoritmo 3, poderia ser usado um algoritmo similar, que começasse de um ponto $\hat{z} \in \tilde{Z}$. Neste caso, no passo 2, deveria ser determinado este ponto inicial. Tal estratégia é particularmente interessante após a aplicação do corte. Note-se, ainda, que a reaplicação do passo 3 a partir do passo 5 também é possível a partir de $\hat{z} \in \tilde{Z}$.

Observe-se que na iteração inicial, antes de se passar o primeiro corte, qualquer um dos algoritmos 1, 2 ou 3 poderiam ser aplicados. Em qualquer caso, o ponto de equilíbrio obtido (o primeiro) satisfaz a restrição de complementaridade.

Em toda iteração onde se encontra um ponto de equilíbrio (\bar{z}, \bar{s}) com $\bar{s}^T \bar{z} = 0$, as condições de otimalidade local, usadas no algoritmo local, poderiam ser verificadas e o corte só seria aplicado caso \bar{z} fosse efetivamente um ótimo local de PLDNP. Embora esta estratégia possa retardar a definição do corte, ela acarreta custos de

computação adicionais.

Como último comentário a respeito do algoritmo global, mencione-se que o valor usado para $\varepsilon > 0$ tanto pode ser constante quanto variar no decorrer do algoritmo. Uma alternativa interessante, adotada por Júdice e Faustino [73], é tomar $\varepsilon = \gamma F^*$, onde $\gamma \in (0, 1)$ representa o desvio percentual máximo aceito entre a solução encontrada pelo algoritmo e a solução real do problema.

Demonstramos agora a convergência e otimalidade do algoritmo global.

Proposição 4.3.2 *O algoritmo global pára em um número finito de iterações.*

Prova: Note-se que um corte é aplicado no passo 4, eliminando um ponto \bar{z} do conjunto viável Z , sempre que $\bar{z}^T \bar{s} = 0$. Neste caso, $\bar{z} \in \tilde{Z}$ é um vértice de Z . Então, posto que o número de vértices de Z é finito, o algoritmo pára se o número de iterações entre um corte e outro é finito. Suponha-se que tal situação não acontece. Logo, a partir de uma certa iteração k , o passo 4 não é mais executado. Assim, o único laço possível ocorre entre os passos 3 e 5. Suponha-se, por absurdo, que este laço ocorre e é infinito. Então é gerada uma sequência infinita de pontos de equilíbrio no passo 3. Como estes pontos são vértices de $\tilde{Z} \times S$, conclui-se que existem pontos repetidos na sequência. Considerem-se, então, (\bar{z}, \bar{s}) e (\hat{z}, \hat{s}) em $\tilde{Z} \times S$ o ponto de equilíbrio e ponto resultante da *outer approximation* obtidos em uma certa execução deste laço. Por conseguinte,

$$\hat{s}^T \hat{z} < \bar{s}^T \bar{z}$$

e o algoritmo 3 é reaplicado a partir de \hat{s} . Pelo lema 4.1.3, o ponto z^1 , obtido pelo passo 1 do algoritmo 3, satisfaz

$$\hat{s}^T z^1 \leq \hat{s}^T \hat{z}.$$

Mais ainda, por (4.5)-(4.6), conclui-se que, se outros pontos são gerados pelo algoritmo 3, a complementaridade não cresce. Assim, o novo ponto de equilíbrio (\bar{z}', \bar{s}') verifica

$$(\bar{s}')^T \bar{z}' \leq \hat{s}^T z^1.$$

Das três desigualdades acima, segue-se que os pontos de equilíbrio da sequência têm complementaridade estritamente decrescente. Portanto, não se repetem, o que é uma contradição. ■

Proposição 4.3.3 *O algoritmo global encontra uma ε -solução global de PLDNP ou verifica que o problema é ilimitado ou inviável.*

Prova: Pela proposição 4.3.2, o algoritmo global pára eventualmente. Em particular, o teste de parada que detecta inviabilidade (passo 6, com $k = 0$) é correto pela equivalência entre PLDNP e (P) no caso inviável. Já o teste que verifica ilimitação (passo 3) também é válido. De fato, neste caso, a proposição 4.1.4 garante o reconhecimento de que $\tilde{P}(M)$ é ilimitado para todo $M \geq 0$. E como $\tilde{P}(M)$ é mais restrito que $P(M)$, conclui-se que $P(M)$ é também ilimitado para todo $M \geq 0$. Então, pelo teorema 3.2.1, tem-se que (P) é ilimitado e, por conseguinte, PLDNP é ilimitado. Finalmente, se estes testes não se verificam, o algoritmo pára porque não existe mais solução viável para (P) em

$$\tilde{Z} = Z \cap \{z \in \mathbb{R}^n : f_1(x, y) \geq F^* + \varepsilon\},$$

onde $F^* = f_1(x^*, y^*)$. Neste caso, deduz-se que (x^*, y^*) é uma ε -solução global de PLDNP. ■

4.4 Resultados computacionais

4.4.1 O gerador de problemas

Infelizmente não existe disponível uma biblioteca de problemas-teste em dois níveis que possam ser usados para avaliar a eficiência de novos algoritmos propostos e compará-la a de procedimentos já existentes na literatura. O que se observa até o momento é que cada pesquisador constrói seus próprios problemas e não os mantém disponíveis.

Por outro lado, geradores de problemas aleatórios foram desenvolvidos por Calamai e Vicente [34] e Moshirvaziri *et al.* [89]. Em nosso caso, optamos por utilizar o gerador proposto por Calamai e Vicente, que está disponível na internet. Este programa foi usado em um recente trabalho pelos próprios autores do segundo gerador (ver Amouzegar e Moshirvaziri [2]).

O método proposto por Calamai e Vicente [34] constitui-se de duas etapas: a primeira constrói um PLDN separável e a segunda elimina a separabilidade através de uma mudança de variáveis. Na verdade, gera-se um PLDNP, pois as restrições do

primeiro nível, que só dependem de x , são transferidas para o segundo. O método permite controlar a esparsidade e o condicionamento da matriz de restrições e a quantidade de ótimos locais e globais do problema gerado, como descrevemos brevemente a seguir.

Definido o número de variáveis do líder (n_1) e do seguidor (n_2), a primeira etapa constrói $p = \min\{n_1, n_2\}$ subproblemas em dois níveis, cada um com duas variáveis (uma no líder e outra no seguidor) e quatro restrições. Os subproblemas podem ser de cinco tipos distintos: nos três primeiros há somente ótimos globais (um, no primeiro e terceiro tipos, e dois, no segundo); nos dois últimos tipos, tem-se um ótimo global e um local.

Em seguida, os p subproblemas são agrupados, formando um PLDNP separável, que também inclui no seu segundo nível uma restrição limitante para cada uma das outras $p' = \max\{n_1, n_2\} - p$ variáveis. Assim, gera-se, ao final da primeira etapa, um PLDNP com $n_{12} = n_1 + n_2$ variáveis, $m = 4p + p'$ restrições, 2^{p_2} ótimos globais e $2^{p_2+p_4+p_5} - 2^{p_2}$ ótimos locais não globais, onde p_i é o número de subproblemas do tipo i (Calamai e Vicente [34, propriedades 1 e 2]). Mencione-se ainda que o problema relaxado correspondente é ilimitado.

A segunda etapa elimina a separabilidade do problema com a seguinte mudança de variáveis

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = HD^{-1}H \begin{bmatrix} x \\ y \end{bmatrix},$$

onde

$$D = I_{xy}d_{xy} \quad \text{e} \quad H = \begin{bmatrix} I_x - 2v_x v_x^T & 0 \\ 0 & I_y - 2v_y v_y^T \end{bmatrix},$$

sendo I_x , I_y , I_{xy} matrizes identidade de ordens n_1 , n_2 , n_{12} respectivamente, e $v_x \in \mathfrak{R}^{n_1}$, $v_y \in \mathfrak{R}^{n_2}$, $d_{xy} \in \mathfrak{R}^{n_{12}}$ vetores gerados aleatoriamente com $v_x^T v_x = v_y^T v_y = 1$ e $d_{xy} > 0$. Note-se que H é uma matriz bloco-diagonal formada por submatrizes de Householder.

A geração dos vetores aleatórios é realizada de modo que o número de elementos não nulos em $v_{xy}^T = (v_x^T, v_y^T)$ seja igual a τ e que a matriz D tenha número de condicionamento na norma euclideana igual a 10^δ , onde τ e δ são parâmetros fornecidos.

Com esta mudança de variáveis, a matriz de restrições do problema fica multiplicada à direita por $(HD^{-1}H)^{-1} = HDH$. Esta transformação linear estabelece

uma correspondência biunívoca entre os ótimos locais e globais do problema original e do problema transformado (Calamai e Vicente [34, proposição 4.1]).

Outra transformação que usaremos para remover a separabilidade do PLDNP original substitui a matriz H por

$$\bar{H} = I_{xy} - 2v_{xy}v_{xy}^T, \quad \text{com } v_{xy}^T v_{xy} = 1.$$

Ela foi sugerida numa versão não publicada do artigo de Calamai e Vicente [34]. Ressalte-se que esta segunda transformação não mantém a correspondência entre os ótimos dos problemas.

Duas características do gerador, com respeito às restrições do PLDNP obtido, são indesejáveis em nosso caso. Primeiro, o número de restrições é relativamente grande se comparado ao número de variáveis. Por exemplo, quando $n_1 = n_2 = 50$, tem-se $m = 200$. Segundo, o problema não inclui restrições de não negatividade, usualmente consideradas em métodos para resolver problemas lineares. Na verdade, o problema original conserva os mesmos ótimos locais e globais se as variáveis são restritas ao ortante não negativo. Todavia, a mesma propriedade não se pode assegurar após a transformação.

Calamai e Vicente [34] apontam que algumas restrições podem ser excluídas sem alterar as soluções do problema. Seguindo sugestões dos autores, introduzimos modificações no gerador para obter problemas menos restritos. Efetivamente, nos subproblemas do primeiro tipo, deixa-se uma única restrição; no terceiro tipo, duas; e nos demais, três. Assim, gera-se um problema com n_{12} variáveis e $m = p_1 + 2p_3 + 3(p_2 + p_4 + p_5) + 1$ restrições. Ressalte-se que se as restrições de não negatividade são consideradas, o problema gerado conserva os mesmos ótimos locais e globais antes da transformação.

Vale a pena mencionar que detectamos algumas incorreções no código disponível na internet. Informados aos autores, os problemas foram confirmados e o código alterado apropriadamente.

4.4.2 Testes com os algoritmos propostos

Os algoritmo local e global propostos nas seções 4.2 e 4.3 foram implementados em C e executados em um microcomputador AMD K6-2/400MHz. Os problemas-teste

Algoritmo	n_{12}	$(n_2/n_{12}) * 100$		$(m/n_{12}) * 100$		densidades (%)
	min-max	min-max	med	min-max	med	
Bard e Moore [13]	40-100	30-60	43	40-40	40	40
Júdice e Faustino [73]	80-170	23-40	32	29-40	30	17;40
Hansen <i>et al.</i> [60]	70-130	40-50	45	40-40	40	8;17;40

Tabela 4.1: Características de problemas-teste usados em outros trabalhos

foram construídos usando-se o gerador modificado, com dimensões compatíveis com as de outras instâncias usadas na literatura.

Os trabalhos tomados como base são aqueles que apresentam os melhores resultados computacionais, quais sejam: Bard e Moore [13], Júdice e Faustino [73] e Hansen *et al.* [60].

A tabela 4.1 mostra as características dos problemas resolvidos nos trabalhos citados. A segunda coluna descreve o número total mínimo e máximo de variáveis consideradas. A coluna seguinte apresenta a proporção de variáveis atribuídas ao seguidor. Além dos valores mínimo e máximo, é também avaliado o valor médio. O mesmo acontece na quarta coluna com respeito à relação entre o número de restrições e o número total de variáveis. Mencione-se que Júdice e Faustino [73] consideram ainda alguns problemas esparsos com 400 variáveis ($n_1 = 250$, $n_2 = 150$) e 150 restrições, mas que não conseguem garantir a obtenção de um ótimo.

A última coluna da tabela 4.1 retrata as diversas densidades impostas para a matriz de restrições. Os valores apresentados correspondem ao percentual de elementos não nulos na matriz. Júdice e Faustino [73] utilizam outra medida de densidade para alguns problemas: a relação entre o número de elementos não nulos e n_{12} . Em tais problemas, este número varia entre 3,5 e 7,5.

Na tabela 4.2, apresentamos os resultados obtidos pelo algoritmo local. A coluna T^0 expressa o tempo (em segundos) consumido pelo problema não transformado. A coluna \bar{T} representa o tempo médio (em segundos) relativo a 8 problemas transformados, que foram obtidos pelas diferentes combinações dos parâmetros $\tau \in \{0, 8n_{12}; 0, 4n_{12}\}$, $\delta \in \{1; 3\}$ e matrizes H e \bar{H} . Assim, um total de 126 problemas foram testados.

Observe-se que o número de restrições é igual a 40% do número de variáveis ($m = 0, 4n_{12}$). Esta é a mesma proporção usada por Bard e Moore [13] e Hansen

n_{12}	n_2	m	T^0	\bar{T}
60	20	24	< 0,01	< 0,01
60	40	24	< 0,01	< 0,01
90	30	36	< 0,01	0,01
90	60	36	< 0,01	0,03
120	40	48	0,06	0,04
120	80	48	0,11	0,07
150	50	60	0,11	0,08
150	100	60	0,16	0,14
200	65	80	0,27	0,19
200	135	80	0,44	0,36
300	100	120	0,94	0,78
300	200	120	1,48	1,31
500	165	200	4,67	3,81
500	335	200	7,58	6,63

Tabela 4.2: Resultados computacionais obtidos pelo algoritmo local

et al. [60]. Note-se ainda que para um mesmo número n_{12} de variáveis, atribuiu-se ora 1/3 ora 2/3 delas ao seguidor. Nos dois trabalhos referidos acima, os autores apontam que o tempo consumido por seus algoritmos aumenta à medida que a relação n_2/n_{12} cresce. Vale a pena destacar que estamos considerando um proporção maior de variáveis no seguidor que nos outros três trabalhos.

Os tempos descritos na tabela 4.2 mostram que o algoritmo local é bastante efetivo. Embora cresça com número de variáveis, o que é esperado, o tempo gasto ainda é bem pequeno para problemas com $n_{12} = 500$. Constata-se também que, para problemas com a mesma relação m/n_{12} , o esforço computacional é maior quando mais variáveis são atribuídas ao líder. Isto reforça as observações de Bard e Moore [13] e Hansen *et al.* [60].

Comparações deste algoritmo com outros algoritmos locais propostos na literatura são praticamente impossíveis, tendo em vista que não são relatados resultados computacionais para eles.

Passamos agora a avaliação de desempenho do algoritmo global. Mencione-se que foi usado $\varepsilon = 0.013$ em todas as execuções. Destaque-se ainda que na *outer approximation* o procedimento para encontrar vértices foi implementado usando-se os códigos hash sugeridos por Chen *et al.* [45, pág. 405].

Num primeiro experimento, aplicou-se o algoritmo aos mesmos problemas da

tabela 4.2. Entretanto, só foi possível resolver as instâncias com até 90 variáveis. Para $n_{12} = 60$, conseguiu-se solucionar 11 dos 18 problemas, obtendo-se um tempo médio de 0,99s antes da transformação e 38,38s depois. Já para $n_{12} = 90$, foram resolvidos 7 dos 18 problemas testados, com média de 41,80s e 76,123s antes e depois da mudança de variáveis.

A justificativa para o mau desempenho do algoritmo global nestes problemas está no fato de que a *outer approximation* depende bastante do número de restrições. De fato, quanto mais recortado o poliedro, mais lentamente se consegue uma boa aproximação sua. Constata-se, assim, que o bom funcionamento do algoritmo global é limitado pela quantidade de restrições do problema.

Uma segunda experiência foi realizada com instâncias onde a relação m/n_{12} é menor. Avaliou-se ainda, neste caso, a influência do número de ótimo locais no desempenho do algoritmo global.

Lembre-se que o número de restrições de um problema obtido com o gerador de Calamai e Vicente [34] é basicamente função de $p = \min\{n_1, n_2\}$. Assim, para se ter menos restrições é necessário que $0 < n_2/n_{12} \ll 1/2$ ou $1/2 \ll n_2/n_{12} < 1$. Neste sentido, foram gerados 60 problemas, onde esta relação é aproximadamente $1/4$ ou $3/4$.

Os primeiros 30 problemas, apresentados na tabela 4.3, foram obtidos pelo gerador modificado sem a mudança de variáveis. Neste caso, tem-se o controle da quantidade de ótimos globais e locais (não globais), conforme especificado na tabela. Os três números da coluna m correspondem respectivamente à quantidade de restrições dos problemas associados às colunas $1 + 1$, $1 + 3$ e $1 + 7$.

Observe-se na tabela 4.3 que os tempos consumidos foram ligeiramente maiores quando mais variáveis são atribuídas ao seguidor. Embora esta diferença tenha sido imperceptível no algoritmo local para problemas dessas dimensões, verificou-se tal tendência também naquele caso para dimensões maiores.

Ainda pelos resultados da tabela 4.3, verifica-se que o tempo gasto aproximadamente quadruplica com a duplicação do número de ótimos (locais e globais). Esta análise não inclui o problema que demandou 24,99s, pois destoa dos demais.

Aos problemas da tabela 4.3 foi, em seguida, aplicada a transformação relativa à matriz H , com parâmetros $\tau = 0,8n_{12}$ e $\delta = 1$. Os resultados para estas 30

Problemas			global + locais		
n_{12}	n_2	m	1+1	1+3	1+7
50	12	15,17,19	0,33	0,17	0,66
50	38	15,17,19	0,05	0,17	0,72
60	15	18,20,22	0,11	0,28	1,21
60	45	18,20,22	0,11	0,33	1,26
70	17	20,22,24	0,11	0,55	2,03
70	53	20,22,24	0,16	0,55	2,20
80	20	23,25,27	0,22	0,83	3,29
80	60	23,25,27	0,27	0,88	24,99
90	22	25,27,29	0,33	1,20	5,16
90	68	25,27,29	0,38	1,32	5,27

Tabela 4.3: Resultados computacionais obtidos pelo algoritmo global para problemas não transformados

Problemas			global + locais		
n_{12}	n_2	m	1+1	1+3	1+7
50	12	15,17,19	< 0,01	0,22	0,11
50	38	15,17,19	5,43	5,00	8,24
60	15	18,20,22	< 0,01	< 0,01	< 0,01
60	45	18,20,22	0,05	44,71	70,03
70	17	20,22,24	< 0,01	< 0,01	0,06
70	53	20,22,24	0,82	68,93	–
80	20	23,25,27	< 0,01	< 0,01	0,82
80	60	23,25,27	0,44	96,82	44,38
90	22	25,27,29	< 0,01	< 0,01	0,99
90	68	25,27,29	53,01	–	–

Tabela 4.4: Resultados computacionais obtidos pelo algoritmo global para problemas transformados

iter.	vértices	adjacências
	$ \bar{\mathcal{W}}_v $	$ \bar{\mathcal{W}}_v * \frac{(n_{12}-2)^\dagger}{2}$
1	264	8976
2	573	19482
3	333	11322
4	1937	65858
5	5081	172754

(†) $Z' \subseteq \mathbb{R}^{n'} \equiv \mathbb{R}^{n_{12}-1}$

Tabela 4.5: Iterações do algoritmo *outer approximation* adaptado

novas instâncias são apresentados na tabela 4.4. Mencione-se que, neste caso, não há mais controle do número de ótimos locais e globais. A denominação das colunas na tabela 4.4 apenas mantém a correspondência com os problemas originais.

Note-se na tabela 4.4 que os problemas com menos variáveis atribuídas ao seguidor ($n_2/n_{12} \approx 1/4$) foram facilmente resolvidos. Quando se usa a proporção complementar, a dificuldade de resolução dos problemas cresce consideravelmente. Este resultado confirma a tendência verificada no algoritmo local.

Observe-se também que não foi possível resolver 3 das 30 instâncias consideradas, em um tempo limite de 900s. Esta situação deve-se ao fato de que a *outer approximation* não foi bem sucedida nestes problemas.

Para se entender a razão desta dificuldade, considere-se, por exemplo, o problema não resolvido onde $n_{12} = 70$. A tabela 4.5 mostra o número de vértices gerados nas primeiras cinco iterações do procedimento *outer approximation* adaptado e o consequente número de adjacências que devem ser determinadas pelo passo 3 do algoritmo de Chen *et al.* [45]. É fácil perceber que, neste caso, o tempo de computação é excessivo.

É importante ressaltar que os problemas da tabela 4.4 são densos ($\tau = 0.8n_{12}$), o que dificulta sua resolução, conforme experiências computacionais de Hansen *et al.* [60]. De fato, note-se que as instâncias da tabela 4.3, mais esparsas, foram resolvidos eficientemente. Na verdade, conseguiu-se resolver problemas não transformados com até 125 restrições e 500 variáveis ($n_2/n_{12} \in \{1/4, 3/4\}$).

Comparações com outros algoritmos globais são difíceis, tendo em vista que não se pôde testá-los com o mesmo conjunto de problemas. Entretanto, para se ter minimamente uma idéia da qualidade dos nossos resultados, reproduzimos na

Problemas			densidade da matriz		
n_{12}	n_2	m	40%	17%	8%
50	30	20	92,4	13,3	
70	20	28	13,0	12,2	
70	28	28	947,5	7,9	
70	35	28	339,7	104,5	6,4
90	30	36	136,2	12,4	9,0
90	45	36	1292,3	58,2	58,6
100	30	40	54,2	25,1	10,8
100	40	40	160,8	35,5	18,0
100	50	40			56,8
120	50	48			112,3
120	60	48			126,0
130	60	52			154,8

Tabela 4.6: Resultados computacionais obtidos por Hansen *et al.*

tabela 4.6 os resultados computacionais obtidos pelo algoritmo BR5 de Hansen *et al.* [60, tabelas 7a e 7b].

Este algoritmo foi escolhido como parâmetro de comparação porque, nos experimentos realizados por Hansen *et al.*, ele mostra-se superior ao algoritmo de Bard e Moore [13] e competitivo com o algoritmo de Júdice e Faustino [73]. Além disso, é o algoritmo para o qual se dispõe de mais resultados computacionais.

A denominação de cada uma das três últimas colunas da tabela 4.6 representa o percentual de elementos não nulos na matriz de restrições dos problemas. Cada valor nestas colunas expressa o tempo médio (em segundos) gasto por 10 problemas aleatórios. Os tempos foram obtidos em uma SUN 3/50.

Confrontando-se as tabelas 4.4 e 4.6 e considerando-se problemas com o mesmo número de variáveis, verifica-se que os tempos do algoritmo global são em geral menores que os de BR5. Mais ainda, note-se que as instâncias da tabela 4.4 são bem mais densas e que uma proporção maior de variáveis é atribuída ao seguidor.

Não obstante os problemas considerados possuam características diversas e os tempos tenham sido obtidos em máquinas diferentes, as constatações acima sugerem que o algoritmo global seja competitivo.

Capítulo 5

Erros na literatura: contra-exemplos e modificações

Tem sido bastante frequente na literatura o aparecimento de notas apontando erros em artigos sobre programação linear em dois níveis. Muitas vezes estes erros se propagam com a utilização de resultados incorretos em publicações subsequentes.

Neste capítulo identificamos incorreções em dois artigos publicados em 1993. Tais trabalhos pretendem resolver PLDNP globalmente, através da penalidade estudada no capítulo 3. Na seção 5.1 indicamos problemas na análise teórica apresentada por Önal [94] e suas implicações no algoritmo proposto. Na seção 5.2 apontamos a invalidade de uma das hipóteses consideradas por White e Anandalingam [127] e o comprometimento dos resultados teóricos obtidos. Estes resultados são reestabelecidos sob uma hipótese mais fraca. Adicionalmente, imprecisões na definição de alguns passos do algoritmo são contornadas. Em ambas as seções, exemplos ilustram as observações feitas.

É importante destacar que a investigação destes erros e o estudo de alternativas para contorná-los foram o ponto de partida para o desenvolvimento teórico apresentado no capítulo 3.

5.1 Artigo de Önal

Önal [94] sugere uma abordagem para determinar ε -soluções globais de PLDNP, usando o problema penalizado

$$P(M) \quad \max \quad F_M(x, y, w, u, v) = c_1^T x + c_2^T y - M(v^T y + u^T w) \quad (5.1)$$

$$\text{s.a.} \quad A_1 x + A_2 y + w = a \quad (5.2)$$

$$x \geq 0, y \geq 0, w \geq 0 \quad (5.3)$$

$$A_2^T u - v = d \quad (5.4)$$

$$u \geq 0, v \geq 0 \quad (5.5)$$

Os sistema (5.2)-(5.3) e (5.4)-(5.5) serão referenciados respectivamente como primal (variáveis de PLDNP) e dual (variáveis duais do seguidor).

A seguir identificamos algumas imprecisões nessa abordagem, tanto no aspecto teórico quanto algorítmico. Na subseção 5.1.1, apontamos os pontos falhos na teoria apresentada e indicamos sob qual hipótese a análise desenvolvida seria válida. Mesmo com a correção teórica, mostramos que o algoritmo pode não convergir para um ótimo global como afirma o autor. Na subseção 5.1.2, introduzimos alguns exemplos para ilustrar todas as observações realizadas.

5.1.1 Problemas observados

Como visto na subseção 2.5.1, o procedimento proposto por Önal [94] compõe-se de duas fases. A primeira aplica o algoritmo de Beale [18] para encontrar uma solução local inicial do problema penalizado $P(M)$, para um valor dominante do parâmetro M , que é considerado implicitamente. Esta solução $(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$ é estável e constitui um ótimo local do problema (P) , conforme o teorema 3.3.2. Neste caso, soluções locais estáveis e parcialmente estáveis são equivalentes.

Observe-se todavia que, de acordo com a seção 3.3, (\bar{x}, \bar{y}) não é necessariamente um ótimo local de PLDNP, a menos que (\bar{v}, \bar{u}) seja o único ponto dual complementar a (\bar{y}, \bar{w}) .

A partir de uma solução local $(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$ de (P) , a segunda fase considera o problema $P'(M)$, obtido de $P(M)$ com o acréscimo do corte linear

$$c_1^T x_1 + c_2^T x_2 \geq f_1(\bar{x}, \bar{y}) + \varepsilon. \quad (5.6)$$

Esta restrição exclui o ótimo local $(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$, mas mantém viáveis todas as possíveis melhores soluções, dentro de uma tolerância $\varepsilon > 0$. Desta forma, a segunda fase pretende encontrar uma melhor solução local de (P) , ou concluir que (\bar{x}, \bar{y}) é uma ε -solução de PLDNP.

Segundo Önal [94], uma das seguintes situações ocorre ao novo problema penalizado:

- (i) $P'(M)$ tem solução local estável,
- (ii) $P'(M)$ é inviável, ou
- (iii) $P'(M)$ é instável, quer dizer, o valor da função (5.1) depende de M .

No caso (i), a segunda fase é repetida com o corte atualizado. No caso (ii), tem-se que (\bar{x}, \bar{y}) é uma ε -solução de PLDNP. Esta mesma conclusão é atribuída por Önal à situação (iii), pois, segundo ele, tal ocorrência significa que as restrições (5.2)-(5.6) e a complementaridade de folgas não podem ser satisfeitas ao mesmo tempo, implicando que não existem soluções viáveis de PLDNP satisfazendo o corte.

Algumas observações devem ser feitas sobre a análise desenvolvida por Önal. O caso (i) significa que *pelo menos uma* das soluções locais de $P'(M)$ é estável. Então, existem duas possibilidades que complementam (i) e (ii):

- (iii.1) $P'(M)$ tem solução local, mas *qualquer* delas é instável, ou
- (iii.2) $P'(M)$ é viável, mas não tem soluções locais.

A situação (iii.1) sugere o sentido de (iii) e, aparentemente, a possibilidade (iii.2) não é considerada por Önal. Note-se, por outro lado, que o autor não assume limitação do problema.

Consideremos primeiramente a situação em que PLDNP é ilimitado. É fácil ver que o caso (ii) nunca ocorre. Todavia, os casos (iii.1) e (iii.2) são ambos possíveis (ver subseção seguinte, figura 5.1). Assim sendo, as conclusões de Önal não são válidas em geral. Na verdade, de acordo com o teorema 3.3.3, estes dois subcasos garantem que (\bar{x}, \bar{y}) é uma ε -solução entre os ótimos locais de PLDNP.

Assumamos agora que PLDNP não é ilimitado. Nesta situação, o caso (iii.2) não acontece, pois $P(M)$ e, conseqüentemente, $P'(M)$ não podem ser ilimitados para todo $M \geq 0$ (ver teorema 3.2.2). Desta forma, os casos (ii) e (iii.1) asseguram realmente a ε -otimalidade de (\bar{x}, \bar{y}) , pois toda solução global de PLDNP está associada a uma solução local estável, conforme o teorema 3.3.3.

Pelo exposto, as conclusões de Önal só são corretas quando PLDNP não é ilimitado. Além disso, o significado do caso (iii) corresponde à situação descrita por (iii.1). Essa análise do problema $P'(M)$ é usada na definição do algoritmo proposto.

Entretanto, esse procedimento, descrito a seguir, não reconhece apropriadamente o caso (iii).

Algoritmo (Önal [94])

Passo 0 Inicialize $k = 0$ e $F_0^* = -\infty$ (F_k^* é o valor da solução local encontrada na iteração k).

Passo 1 Se (5.2)-(5.3) ou (5.4)-(5.5) são inviáveis, vá para o passo 5. Senão, encontre soluções básicas viáveis $(\bar{x}, \bar{y}, \bar{w})$ e (\bar{u}, \bar{v}) .

Passo 2 Substitua os valores das variáveis básicas na função (5.1), expressando-a em termos do valor corrente $F^0 = F_M(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v})$ e das variáveis não básicas (x_N, y_N, w_N) e (u_N, v_N) . Calcule as derivadas parciais da função em relação às variáveis não básicas e depois iguale estas variáveis a zero, obtendo

$$\left(\frac{\partial F_M}{\partial x_N}, \frac{\partial F_M}{\partial y_N}, \frac{\partial F_M}{\partial w_N} \right) \Big|_0 = (P_x, P_y, P_w) \quad \text{e} \quad \left(\frac{\partial F_M}{\partial u_N}, \frac{\partial F_M}{\partial v_N} \right) \Big|_0 = (-M P_u, -M P_v).$$

Se $(P_x, P_y, P_w) \leq 0$ e $(P_u, P_v) \geq 0$, vá para o passo 4. Senão, vá para o passo 3.

Passo 3 Escolha para entrar na base a variável (primal ou dual) com o maior coeficiente positivo em $(P_x, P_y, P_w, -M P_u, -M P_v)$. Determine a variável a sair pelo critério usual do método simplex. Pivoteie e volte ao passo 1.

Passo 4 A solução corrente é localmente ótima. Se F^0 envolve M (solução instável), vá para o passo 5. Senão, atualize $k = k + 1$ e $F_k^* = F^0$. Escolha $\epsilon > 0$ e forme o corte (5.6) com $f_1(\bar{x}, \bar{y}) = F^0$. Defina uma nova variável de folga e acrescente a nova equação ao sistema primal. Volte ao passo 1.

Passo 5 Pare. Se $k = 0$, o problema é inviável. Se $k \geq 1$, a solução obtida na iteração k é ótima.

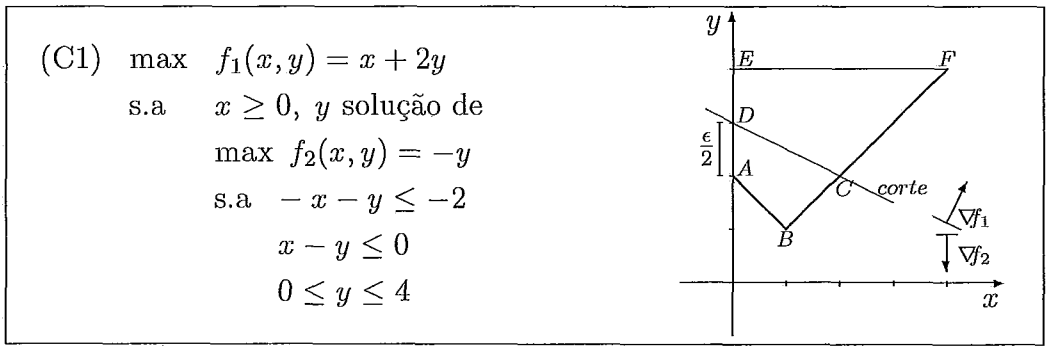
Os problemas com o algoritmo aparecem nos passos 4-5. A obtenção de uma solução local instável no passo 4 leva a uma conclusão equivocada no passo 5. Visto que esta ocorrência pode não corresponder ao caso (iii.1), pois $P'(M)$ ainda pode ter uma solução local estável, o algoritmo retorna neste caso uma ϵ -solução global errada. Esta situação é ilustrada na próxima subseção com o exemplo 5.1.1.

Embora Önal [94] não assuma explicitamente a hipótese de não ilimitação de PLDNP, as conclusões do passo 5 indicam esta suposição. Também na subseção seguinte, mostramos que o algoritmo não é capaz de reconhecer o caso ilimitado.

5.1.2 Contra-exemplos

O exemplo a seguir, apresentado por Campêlo e Scheimberg [37], mostra que o algoritmo de Önal [94] pode não encontrar uma ε -solução global. Simples modificações neste exemplo também ilustram a ocorrência dos casos (iii.1)-(iii.2) e como o algoritmo se comporta nestas situações.

Exemplo 5.1.1



O gráfico acima será usado para acompanhar o caminho percorrido pelo algoritmo no espaço primal. Note-se que a região viável de (C1) está representada em negrito.

Neste PLDNP tem-se $c_1 = 1, c_2 = 2, d = -1, a = (-2, 0, 4)^T, A_1 = (-1, 1, 0)^T$ e $A_2 = (-1, -1, 1)^T$. O problema penalizado é definido como

$$\begin{aligned} \max \quad & F_M(x, y, w, u, v) = x + 2y - M(vy + u_1w_1 + u_2w_2 + u_3w_3) \\ \text{s.a} \quad & x + y - w_1 = 2 \\ & x - y + w_2 = 0 \\ & y + w_3 = 4 \\ & u_1 + u_2 - u_3 + v = 1 \\ & x, y, w_1, w_2, w_3, u_1, u_2, u_3, v \geq 0 \end{aligned}$$

Na iteração $k = 0$ do algoritmo, sejam as soluções viáveis iniciais de (5.2)-(5.3) e (5.4)-(5.5) dadas por $(\bar{x}, \bar{y}, \bar{w}_1, \bar{w}_2, \bar{w}_3) = (0, 2, 0, 2, 2)$ e $(\bar{u}_1, \bar{u}_2, \bar{u}_3, \bar{v}) = (1, 0, 0, 0)$. Estas soluções, requeridas no passo 1, podem ser encontradas aplicando-se a primeira fase do método simplex. Observe-se que a solução primal está associada ao ponto A no gráfico.

Expressando-se as variáveis básicas (y, w_2, w_3) e u_1 em função das não básicas (x, w_1) e (u_2, u_3, v) , como solicitado no passo 2, a função de penalidade torna-se

$$F_M(x, y, w, u, v) = 4 - x + 2w_1 - M [v(2 - x + w_1) + w_1(1 - u_2 + u_3 - v) + u_2(2 - 2x + w_1) + u_3(2 + x - w_1)]$$

com $F^0 = F_M(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v}) = 4$.

As derivadas parciais de F_M em relação às variáveis não básicas são:

$$\begin{aligned} \frac{\partial F_M}{\partial x} &= -1 - M(-v - 2u_2 + u_3), & \frac{\partial F_M}{\partial w_1} &= 2 - M \\ \frac{\partial F_M}{\partial u_2} &= -2M(1 - x), & \frac{\partial F_M}{\partial u_3} &= -M(2 + x), & \frac{\partial F_M}{\partial v} &= -M(2 - x). \end{aligned}$$

Anulando-se as variáveis não básicas e considerando-se M um valor dominante, obtém-se que

$$(P_x, P_{w_1}) = (-1, 2 - M) \leq 0 \quad \text{e} \quad (P_{u_2}, P_{u_3}, P_v) = (2, 2, 2) \geq 0.$$

Assim, segue-se para o passo 4, onde se conclui que a solução corrente é localmente ótima para $P(M)$ e estável. Então, para se eliminar este primeiro ótimo local, atingido no ponto A , define-se o corte

$$x + 2y - w_4 = 4 + \varepsilon, \tag{5.7}$$

onde $w_4 \geq 0$ é a nova variável de folga e $\varepsilon > 0$ (ver gráfico). Inicia-se, pois, a iteração $k = 1$.

Acrescentada a nova restrição ao quadro, uma iteração do método dual-simplex pode ser aplicada para se obter a seguinte solução primal viável

$$(\bar{x}, \bar{y}, \bar{w}_1, \bar{w}_2, \bar{w}_3, \bar{w}_4) = \frac{1}{2}(0, 4 + \varepsilon, \varepsilon, 4 + \varepsilon, 4 - \varepsilon, 0),$$

que satisfaz (5.7) para todo $0 \leq \varepsilon \leq 4$. Este ponto encontra-se sobre o segmento $(A, E]$ no gráfico. Quanto à solução dual, pode-se manter a anterior, que continua viável.

A partir dessa solução primal-dual no passo 2, obtém-se a seguinte expressão de F_M em função das variáveis não básicas (x, w_4) e (u_2, u_3, v) :

$$\begin{aligned} F_M(x, y, w, u, v) &= 4 + \varepsilon + w_4 - M/2 [v(4 + \varepsilon - x + w_4) \\ &\quad + (\varepsilon + x + w_4)(1 - u_2 + u_3 - v) \\ &\quad + u_2(4 + \varepsilon - 3x + w_4) + u_3(4 - \varepsilon + x - w_4)], \end{aligned}$$

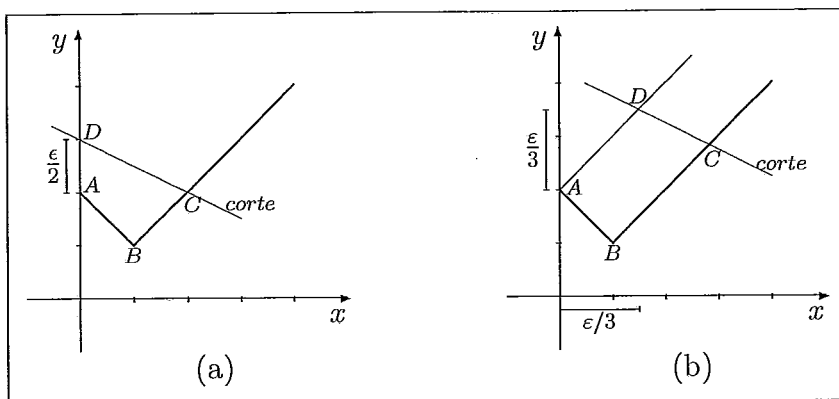


Figura 5.1: Ilustração dos casos (iii.1) e (iii.2)

sendo $F^0 = F_M(\bar{x}, \bar{y}, \bar{w}, \bar{u}, \bar{v}) = 4 + \varepsilon - M\varepsilon/2$.

As derivadas parciais em relação às variáveis não básicas duais (u_2, u_3, v) continuam as mesmas. Com respeito às variáveis primais, tem-se agora:

$$\frac{\partial F_M}{\partial x} = -M(-v - 2u_2 + u_3 + 1/2) \quad \text{e} \quad \frac{\partial F_M}{\partial w_4} = 1 - M/2.$$

Anulando-se as variáveis não básicas e considerando-se M um valor dominante chega-se a

$$(P_x, P_{w_1}) = (-M/2, 1 - M/2) \leq 0 \quad \text{e} \quad (P_{u_2}, P_{u_3}, P_v) = (2, 2, 2) \geq 0.$$

Então, vai-se para o passo 4 e de lá para o passo 5, pois F^0 depende de M . O algoritmo pára e conclui que o ponto da iteração anterior $(\bar{x}, \bar{y}) = (0, 2)$ é uma ε -solução global de (C1). Entretanto, esta resposta é incorreta, pois o ótimo global é $(x^*, y^*) = (4, 4)$ e

$$f_1(4, 4) - f_1(0, 2) = 12 - 4 > \varepsilon.$$

O erro deve-se ao fato do algoritmo considerar que ocorreu o caso (iii.1). No gráfico, observe-se que, depois da adição do corte, com $0 < \varepsilon \leq 4$, o novo ótimo local D de $P'(M)$ estará no intervalo de pontos instáveis $(A, E]$. Por isso, o algoritmo termina. Todavia, ainda existe solução local estável no politopo $CDEF$, a saber, a solução global F .

Introduzimos agora modificações no exemplo (C1) para ilustrar a ocorrência dos casos (iii.1) e (iii.2) quando PLDNP é ilimitado.

Descartando-se a restrição $y \leq 4$ no exemplo (C1), obtém-se a situação descrita na figura 5.1a. A partir dos mesmos pontos iniciais, o algoritmo encontra o primeiro

ótimo local de $P(M)$ novamente no ponto A . Então, adicionado o corte, com $\varepsilon > 0$, a única solução local remanescente de $P'(M)$ está associada a $D = (0, 2 + \varepsilon/2)$. Visto que este ponto é instável, o algoritmo retornará A como ε -solução global, embora o problema não tenha solução. Este exemplo mostra que mesmo a ocorrência do caso (iii.1) não é suficiente para assegurar a ε -otimalidade da solução anterior.

Considere-se, finalmente, que a restrição $y \leq 4$ do exemplo (C1) foi substituída por $-x + y \leq 2$, como mostra a figura 5.1b. Após atingida a primeira solução local em A e acrescentado o corte, com $\varepsilon > 3$, o algoritmo irá atingir o ponto $D = (\varepsilon/3, 2 + \varepsilon/3)$, que não é ótimo local do problema penalizado. Na verdade, não existe mais nenhum ótimo local, o que caracteriza o caso (iii.2). Novamente a solução anterior A não é ε -ótima.

5.2 Artigo de White e Anandalingam

Para manter uma correspondência com a notação utilizada por White e Anandalingam [127], nesta seção vai-se considerar o problema penalizado $P(M)$ sem as variáveis de folga das restrições (5.2) e (5.4). Sendo assim, este problema é reescrito como

$$P(M) \quad \max_{(x,y,u)} \quad \hat{F}(x, y, u, M) = c_1^T x + c_2^T y - M\pi(x, y, u) \quad (5.8)$$

$$\text{s.a.} \quad (x, y) \in W, \quad u \in U \quad (5.9)$$

onde

$$W = \{(x, y) \geq 0 : A_1 x + A_2 y \leq a\}, \quad U = \{u \geq 0 : A_2^T u \geq d\}$$

e

$$\pi(x, y, u) = (A_2^T u - d)^T y + u^T (a - A_1 x - A_2 y) = u^T (a - A_1 x) - d^T y$$

é o gap de dualidade do problema do seguidor.

White and Anandalingam [127] propõem um algoritmo para resolver o problema PLDNP globalmente, através da resolução do problema $P(M)$ para uma sequência finita de valores de M . Os autores trabalham sob as seguintes hipóteses:

[A1] Se x^* é uma solução ótima do líder, então o conjunto solução do seguidor $\Psi(x^*)$ é um único ponto.

[A2] Os poliedros W e U são não vazios e compactos.

Identificamos alguns problemas nesta abordagem. Primeiro, a hipótese [A2] não é válida, o que compromete os resultados teóricos e numéricos. Além disso, o algoritmo não está bem definido com relação aos cortes usados para descartar soluções locais e ao teste para identificar possíveis soluções melhores.

Na subseção 5.2.1 substituímos as hipóteses [A1] e [A2] pela suposição de que o problema PLDNP tem solução. Sob esta nova condição, reestabelecemos as propriedades do problema penalizado. Também nesta subseção, discutimos os problemas observados no algoritmo e introduzimos modificações para contorná-los. Entretanto, observamos que a completa boa definição do algoritmo necessita de uma hipótese de compacidade sobre W ou U . A subseção 5.2.2 apresenta exemplos ilustrando todas estas observações.

5.2.1 Problemas observados

Em vez de abordar $P(M)$ diretamente através da formulação (5.8)-(5.9), White e Anandalingam [127] consideram o problema equivalente

$$\max_{u \in U} \Theta(u, M), \quad (5.10)$$

onde

$$\Theta(u, M) = \max_{x,y} \{ \hat{F}(x, y, u, M) : (x, y) \in W \}. \quad (5.11)$$

A fundamentação teórica do algoritmo de White e Anandalingam [127] está baseada nos seguintes resultados.

Teorema 5.2.1 *Dados $u \in U \subseteq \mathbb{R}^{m_2}$ e $M \geq 0$, a função $\Theta(u, M)$ está bem definida e $\Theta(\cdot, M)$ é uma função convexa em \mathbb{R}^{m_2} . Além disso, uma solução do problema (5.10) é atingida em algum vértice $u^* \in U$ ($u^* \in U_v$).*

Teorema 5.2.2 *Para $M \geq 0$ fixo, uma solução ótima de $P(M)$ é atingida em $W_v \times U_v$ e $W_v \times U_v = (W \times U)_v$.*

Teorema 5.2.3 *Existe um valor finito $M^* \geq 0$ de M para o qual uma solução ótima do problema $P(M)$ gera uma solução ótima do problema PLDNP para todo $M \geq M^*$.*

Teorema 5.2.4 *Se $(x(M), y(M), u(M))$ resolve $P(M)$ como função de M , então a função objetivo do líder $f_1(x(M), y(M))$ e o gap de dualidade $\pi(x(M), y(M), u(M))$ do problema do seguidor são monotonicamente não crescentes com respeito ao parâmetro M .*

As demonstrações dos teoremas acima podem ser encontradas em Anandalingam e White [3]. Elas são efetuadas sob a hipótese [A2], sendo que a prova do teorema 5.2.3 também utiliza a condição [A1] (ver White e Anandalingam [127, pág. 403]). Entretanto, a suposição [A2] nunca é satisfeita, como mostra o resultado abaixo, apresentado por Charnes *et al.* [44].

Teorema 5.2.5 *Se $Y = \{y \geq 0 : Ay \leq q\}$ é não vazio e compacto, então $U = \{u \geq 0 : A^T u \geq d\}$ é não vazio e ilimitado; também, se U é não vazio e compacto, então Y é não vazio e ilimitado.*

Embora a condição [A2] não seja válida, os teoremas 5.2.1-5.2.4 podem ser reestabelecidos sob uma hipótese mais fraca e natural neste caso, qual seja:

[A] O problema PLDNP tem solução.

Destaque-se, ainda, que [A2] será descartada.

Como a suposição [A] permite que o problema relaxado seja ilimitado, o problema penalizado $P(M)$ pode não ter solução para certos valores de M (em particular para $M = 0$). Entretanto, o teorema 3.2.1 garante que existe $M_0 \geq 0$ tal que $P(M)$ tem solução para todo $M \geq M_0$.

Considerando-se esse valor M_0 do parâmetro de penalidade, os teoremas 5.2.1 e 5.2.2 podem ser reescritos para $M \geq M_0$. De fato, como o problema $P(M)$ é dado igualmente por (5.10), segue-se que a função Θ está bem definida para $M \geq M_0$. Já a convexidade de $\Theta(\cdot, M)$ decorre do fato de que esta função é dada pelo máximo de uma família de funções lineares (Rockafellar [97, teorema 5.5]). Finalmente, a segunda parte do teorema 5.2.1 está incluída no teorema 5.2.2, que é um resultado conhecido em programação bilinear (Horst e Tuy [69, proposição IX.1]).

Quanto aos teoremas 5.2.3 e 5.2.4, seus enunciados não precisam ser modificados. O primeiro é uma consequência direta do teorema 3.2.1, e o segundo resulta do lema 9.2.1 de Bazaraa *et al.* [17].

Mesmo sendo reestabelecidos os resultados teóricos, a boa definição do algoritmo de White e Anandalingam [127] ainda depende de alguns ajustes. Antes, reescrevemos o algoritmo original, seguido de uma explanação, em linhas gerais, do seu funcionamento.

Seja $(x(u, M), y(u, M))$ uma solução de (5.11) e defina-se

$$\begin{aligned}\Phi(u', u, M) &= (u' - u)^T(a - A_1x(u, M)) \\ &= \pi(x(u, M), y(u, M), u') - \pi(x(u, M), y(u, M), u)\end{aligned}\quad (5.12)$$

para cada $M \geq 0$ e $u, u' \in U$ (no novo contexto deve-se tomar $M \geq M_0$). O algoritmo proposto por White e Anandalingam [127] é descrito abaixo.

Algoritmo (White e Anandalingam [127])

Passo 0 Escolha M (grande) e $u^1 \in U_v$. Inicialize $\Theta^1 = -\infty$ e $\bar{u}^1 = u^1$.

Passo 1 Determine $\Theta(u^1, M)$, obtendo $(x(u^1, M), y(u^1, M))$.

$$\text{Atualize } \bar{u}^1 = \begin{cases} u^1 & \text{se } \Theta(u^1, M) > \Theta^1 \\ \bar{u}^1 & \text{se } \Theta(u^1, M) \leq \Theta^1 \end{cases}$$

e $\Theta^1 = \max\{\Theta^1, \Theta(u^1, M)\}$.

Passo 2 Seja $\{u^{1s} : 1 \leq s \leq N(u^1)\}$ o conjunto dos vértices adjacentes a u^1 .

Se $\Theta(u^{1s}, M) > \Theta^1$ para algum s , então

atualize $u^1 = u^{1s}$, $\bar{u}^1 = u^{1s}$, $\Theta^1 = \Theta(u^{1s}, M)$ e repita o passo 2.

Passo 3 Se $\Theta(u^{1s}, M) \leq \Theta^1$, para todo s ,

determine $\Gamma(u^1, M) = \min\{\Phi(u, u^1, M) : u \in U\}$, obtendo $u^*(u^1, M)$.

Passo 4 Se $\Gamma(u^1, M) < 0$ então faça $u^1 = u^*(u^1, M)$ e

volte ao passo 1.

Passo 5 Se $\Gamma(u^1, M) \geq 0$,

seja $\{t^{1s} : 1 \leq s \leq N(u^1)\}$ o conjunto dos raios extremos unitários a

partir de u^1 e determine $\alpha_s = \max\{\alpha \geq 0 : \Theta(u^1 + \alpha t^{1s}, M) \leq \Theta^1\}$,

$1 \leq s \leq N(u^1)$.

Passo 6 Sejam $\nu^{1s} = u^1 + \alpha_s t^{1s}$, $1 \leq s \leq N(u^1)$,

$$\Lambda(u^1) = \left\{ \lambda = (\mu, \sigma) \in \mathbb{R}^m \times \mathbb{R} : \sigma \in \{1, -1\}, \mu^T u^1 - \sigma \leq 0, \right. \\ \left. \mu^T \nu^{1s} - \sigma \geq 0, 1 \leq s \leq N(u^1) \right\}$$

e, para cada $u \in U$:

$$G(u, u^1) = \min\{\mu^T u - \sigma : \lambda \in \Lambda(u^1)\}.$$

Passo 7 Tome $u^{1*} \in \arg \max\{G(u, u^1) : u \in U\}$.

Passo 8 Se $G(u^{1*}, u^1) \leq 0$ então

$\bar{u}^1 \in \arg \max\{\Theta(u, M) : u \in U\}$, e o valor ótimo de $\Theta(\cdot, M)$ foi atingido para este M particular, com solução $(x(\bar{u}^1, M), y(\bar{u}^1, M))$.

Então, vá para o passo 10.

Passo 9 Se $G(u^{1*}, u^1) > 0$, faça $u^1 = u^{1*}$ e volte ao passo 1.

Passo 10 Se $\pi(x(\bar{u}^1), y(\bar{u}^1), \bar{u}^1) > 0$, faça $M = M + \Delta$ e volte ao passo 1.

Senão, $\pi(x(\bar{u}^1), y(\bar{u}^1), \bar{u}^1) = 0$ e $(x(\bar{u}^1), y(\bar{u}^1))$ resolve PLDNP.

Os passos 1-4 determinam um ótimo local de $\Theta(\cdot, M)$, como descrito por White e Anandalingam [127, pág. 404]. Em particular, este ótimo local $u^1 \in U_v$ é tal que os vértices adjacentes estão sob a curva de nível $\Theta(u, M) = \Theta^1$ que passa por u^1 (ver figura 5.2a).

Os passos 5-9 constituem modificações de um algoritmo proposto por Tuy [108]. O passo 5 estende os raios que saem de u^1 até atingir a curva de nível $\Theta(u, M) = \Theta^1$, determinando os pontos ν^{1s} no passo 6 (ver figura 5.2b), que são chamados Θ^1 -extensões dos vértices u^{1s} com respeito a u^1 (Horst e Tuy [69, definição V.1]). Neste momento ocorre o primeiro problema com o algoritmo, pois alguns dos raios podem nunca atingir a curva de nível (ver figura 5.2e). Quando tal situação acontece, o conjunto $\Lambda(u^1)$ precisa ser redefinido.

O conjunto $\Lambda(u^1)$ é formado por hiperplanos que separam u^1 dos pontos ν^{1s} , estabelecendo cortes para o ótimo local u^1 . A “distância” de um ponto $u \in U$ a um corte $(\mu, \sigma) \in \Lambda(u^1)$ é dado por $\mu^T u - \sigma$. Na verdade, $\mu^T u - \sigma \geq 0$, se u satisfaz o corte, e $\mu^T u - \sigma < 0$, caso contrário. Assim, o passo 7 procura determinar um vértice $u^{1*} \in U_v$ que verifique todos os cortes.

Quando U é ilimitado, o passo 7 pode falhar (ver exemplo 5.2.2, próxima subseção). Caso contrário, existem três possibilidades para o vértice u^{1*} determinado. Se $G(u^{1*}, u^1) > 0$, tem-se que u^{1*} verifica todos os cortes, e o processo é repetido a partir deste vértice. Quando $G(u^{1*}, u^1) < 0$, todo ponto de U não satisfaz pelo menos um corte e, por conseguinte, o máximo de $\Theta(u, M)$ foi atingido para o valor corrente de M . Finalmente, no caso $G(u^{1*}, u^1) = 0$, o procedimento também deveria

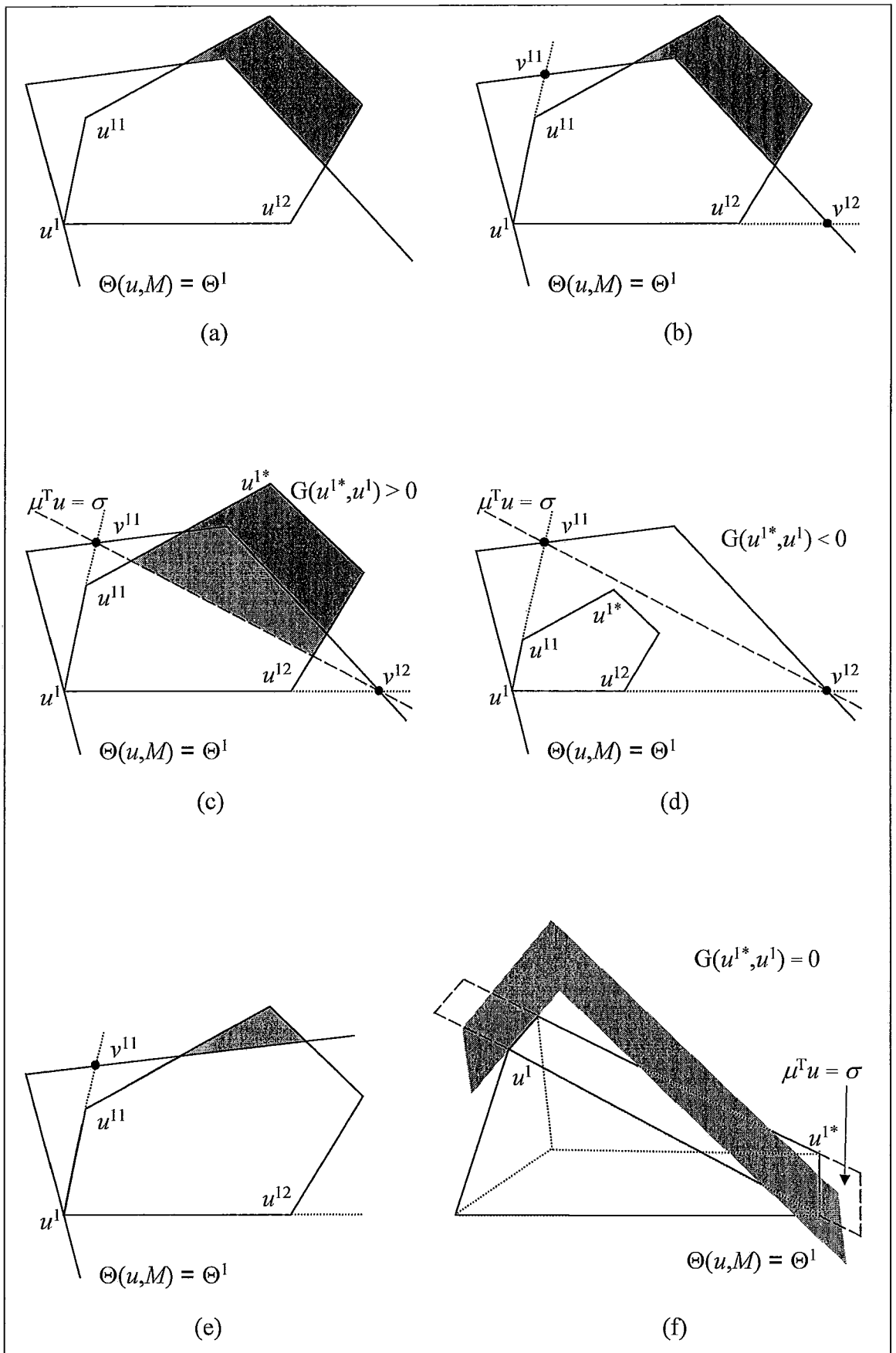


Figura 5.2:

ser repetido a partir de u^{1*} , ao contrário do que estabelece o passo 8.

As três situações são ilustradas nos desenhos (c), (d) e (f) da figura 5.2. Por simplicidade, está representado um único hiperplano $(\mu, \sigma) \in \Lambda(u^1)$ em cada caso. Em particular, na figura 5.2f, note-se que u^1 não fornece o máximo de $\Theta(u, M)$ para M fixo e que u^{1*} é atingido no único vértice acima da curva de nível $\Theta(u, M) = \Theta^1$. Entretanto, $G(u^{1*}, u^1) = 0$, pois

$$0 \leq G(u^{1*}, u^1) \leq \mu^T u^{1*} - \sigma = 0,$$

onde (μ, σ) representa o hiperplano indicado.

Pelo visto, o único caso que garante ser $(x(u^1, M), y(u^1, M), u^1)$ uma solução ótima de $P(M)$ é $G(u^{1*}, u^1) < 0$. Além disso, se o termo de penalidade é nulo, isto é, $\pi(x(u^1, M), y(u^1, M), u) = 0$, segue-se que $(x(u^1, M), y(u^1, M))$ é solução de PLDNP. Do contrário, o valor do parâmetro ainda não é suficientemente grande, sendo então incrementado e o algoritmo reiniciado.

Procedemos agora aos ajustes necessários e demonstramos a corretude das alterações introduzidas. Ressalte-se que tais modificações não visam a melhorar o desempenho computacional do algoritmo, mas apenas a torná-lo bem definido. Neste sentido, procura-se conservar as idéias originais o máximo possível.

No passo inicial, pode-se supor que a escolha de M garante a avaliação da função $\Theta(\cdot, M)$ durante o algoritmo. Se em algum momento descobre-se que este não é o caso, incrementa-se o valor de M e volta-se ao passo 1. Este processo pode se repetir no máximo até que se tenha $M \geq M_0$.

Os passos 1-4 continuam bem definidos quando se assume $[A]$ em vez de $[A2]$. O único ponto a se considerar é o cálculo do mínimo no passo 3. Usando-se a definição (5.12) e levando-se em conta que $\pi(x, y, u) \geq 0$ para todo $(x, y, u) \in W \times U$, segue-se que $\Gamma(\cdot, M)$ está bem definido. De fato,

$$\Gamma(u^1, M) = \min\{\Phi(u, u^1, M) : u \in U\} \leq \Phi(u^1, u^1, M) = 0$$

e

$$\begin{aligned} \Gamma(u^1, M) &= \min\{\pi(x(u^1, M), y(u^1, M), u) : u \in U\} - \pi(x(u^1, M), y(u^1, M), u^1) \\ &\geq -\pi(x(u^1, M), y(u^1, M), u^1). \end{aligned}$$

Logo, o mínimo é sempre atingido. Em particular, $\Gamma(u^1, M) = 0$ com $u^*(u^1, M) = u^1$ quando $\pi(x(u^1, M), y(u^1, M), u^1) = 0$.

Na verdade, a função Φ também depende de (x, y) , uma vez que o conjunto de soluções de (5.11) pode não ser unitário. Para se considerar todos os vértices deste conjunto, poder-se-ia redefinir

$$\Gamma(u^1, M) = \min\{\pi(x, y, u) - \pi(x, y, u^1) : u \in U, (x, y) \in W_v(u^1, M)\},$$

onde $W_v(u^1, M) = \arg \max\{\hat{F}(x, y, u^1, M) : (x, y) \in W_v\}$. Esta alteração criaria mais chances do algoritmo sair do ótimo local u^1 sem a necessidade de um corte.

No passo 5, calculam-se os passos α_s ao longo dos raios extremos t^{1s} ($1 \leq s \leq N(u^1)$) que partem de u^1 . Segundo Horst e Tuy [69, proposição IX.3], α_s é igual ao valor ótimo do seguinte problema de programação linear:

$$\alpha_s = \min_{(r, w, v)} (\Theta^1 + M a^T u^1) r - (c_1 + M A_1^T u^1)^T w - (c_2 + M d)^T v \quad (5.13)$$

$$\text{s.a. } M(a^T t^{1s}) r - M(A_1^T t^{1s}) w = -1 \quad (5.14)$$

$$-ar + A_1 w + A_2 v \leq 0 \quad (5.15)$$

$$r \geq 0, w \geq 0, v \geq 0 \quad (5.16)$$

onde $w \in \mathfrak{R}^{n_1}$, $v \in \mathfrak{R}^{n_2}$, $r \in \mathfrak{R}$.

Para se ter o ponto ν^{1s} bem definido no passo 6, é preciso que $0 < \alpha_s < +\infty$. Mostramos a seguir que cada passo α_s é estritamente positivo.

Proposição 5.2.1 *Se $M \geq M_0$, então $\alpha_s > 0$ para todo $s \in [1, N(u^1)]$.*

Prova: Seja $s \in [1, N(u^1)]$. Se t^{1s} não é uma direção de recessão de U , tem-se

$$u^{1s} = u^1 + \hat{\alpha} t^{1s}, \quad \text{com } \hat{\alpha} > 0.$$

E como $\Theta(u^{1s}, M) \leq \Theta^1$, segue-se que $\alpha_s \geq \hat{\alpha} > 0$. Agora, assumamos que t^{1s} seja uma direção de recessão e defina-se

$$\Theta_s(\alpha, M) = \Theta(u^1 + \alpha t^{1s}, M).$$

Suponha-se, por contradição, que $\alpha_s = 0$. Então, $\Theta_s(\alpha, M) > \Theta_s(0, M)$ para todo $\alpha > 0$. Como $\Theta_s(\cdot, M)$ é convexa em \mathfrak{R} , a proposição 1.1.4 em Hiriart-Urruty e Lemarechal [66] assegura que

$$\Theta_s(\alpha, M) \geq \frac{\Theta_s(\bar{\alpha}, M) - \Theta_s(0, M)}{\bar{\alpha}} \alpha + \Theta_s(0, M), \quad \forall \alpha \geq \bar{\alpha} > 0.$$

Assim,

$$\begin{aligned} \lim_{\alpha \rightarrow +\infty} \Theta(u^1 + \alpha t^{1s}, M) &= \lim_{\alpha \rightarrow +\infty} \Theta_s(\alpha, M) \\ &\geq \lim_{\alpha \rightarrow +\infty} \left(\frac{\Theta_s(\bar{\alpha}, M) - \Theta_s(0, M)}{\bar{\alpha}} \alpha + \Theta_s(0, M) \right) = +\infty, \end{aligned}$$

contradizendo o fato de que $\Theta(u, M)$ é limitada superiormente em U para todo $M \geq M_0$. ■

Por outro lado, na próxima subseção, mostramos que se pode ter $\alpha_s = +\infty$ para algum $s \in [1, N(u^1)]$, seja ou não t^{1s} uma direção de recessão do conjunto dual U . Na verdade, apresentamos exemplos onde o problema (5.13)-(5.16) é inviável. Neste caso, não é possível definir algum ponto ν^{1s} .

Para contornar tal dificuldade, faz-se necessário a redefinição dos passos 5-6. Seguindo Horst e Tuy [69, capítulo III], redefinimos estes passos como abaixo:

Passo 5' Se $\Gamma(u^1, M) \geq 0$,

seja $\{t^{1s} : 1 \leq s \leq N(u^1)\}$ o conjunto dos raios extremos unitários a partir de u^1 e determine $\alpha_s = \sup\{\alpha \geq 0 : \Theta(u^1 + \alpha t^{1s}, M) \leq \Theta^1\}$, $1 \leq s \leq N(u^1)$.

Passo 6' Sejam $I = \{s : 1 \leq s \leq N(u^1), \alpha_s < +\infty\}$,

$$\nu^{1s} = u^1 + \alpha_s t^{1s}, \quad \forall s \in I,$$

$$\Lambda(u^1) = \left\{ \lambda = (\mu, \sigma) \in \mathbb{R}^m \times \mathbb{R} : \sigma \in \{1, -1\}, \mu^T u^1 - \sigma \leq 0, \right. \\ \left. \mu^T \nu^{1s} - \sigma \geq 0, s \in I, \mu^T t^{1s} \geq 0, s \notin I \right\}$$

e, para cada $u \in U$:

$$G(u, u^1) = \min\{\mu^T u - \sigma : \lambda \in \Lambda(u^1)\}.$$

O novo conjunto $\Lambda(u^1)$ está bem definido, como demonstramos a seguir.

Proposição 5.2.2 *O conjunto $\Lambda(u^1)$ é não vazio. Em particular, existe $(\mu, \sigma) \in \Lambda(u^1)$ com $\mu^T u^1 - \sigma < 0$.*

Prova: Seja $\mathcal{U} = \{u = \sum_{s \in I} \rho_s \alpha_s t^{1s} + \sum_{s \notin I} \rho_s t^{1s} : \sum_s \rho_s \geq 1, \rho_s \geq 0 \forall s\}$. Tem-se que $0 \notin \mathcal{U}$, pois $\alpha_s > 0$ para todo $s \in I$, $\rho_s > 0$ para algum s e o cone convexo gerado por $\{t^{1s}\}$ não contém retas. Seja $\mathcal{V} = u^1 + \mathcal{U}$. Então, \mathcal{V} é um conjunto não vazio, convexo e fechado, e $u^1 \notin \mathcal{V}$. Logo, existe (μ, σ) , com $\mu \neq 0$, tal que $\mu^T u^1 < \sigma$ e $\mu^T u \geq \sigma$ para todo $u \in \mathcal{V}$ (Bazaraa *et al.* [17, teorema 2.4.4]). Como,

para cada $s \in I$, $\nu^{1s} = u^1 + \alpha_s t^{1s} \in \mathcal{V}$, segue-se que $\mu^T \nu^{1s} \geq \sigma$. Adicionalmente, dado $s \notin I$, tem-se que $u^1 + t^{1s} \in \mathcal{V}$ e, por conseguinte, $\mu^T(u^1 + t^{1s}) \geq \sigma$. Então $\mu^T t^{1s} \geq \sigma - \mu^T u^1 > 0$. Portanto, $(\mu, \sigma) \in \Lambda(u^1)$. ■

No próximo resultado mostramos que o conjunto $\Lambda(u^1)$ estabelece cortes que justificam a definição de $G(u, u^1)$ no passo 6.

Proposição 5.2.3 *Sejam $u \in U$ e $(\mu, \sigma) \in \Lambda(u^1)$. Se $\mu^T u - \sigma < 0$, então $\Theta(u, M) \leq \Theta^1$.*

Prova: Seja $u \in U$. Então existem escalares $\rho_s \geq 0$ tais que

$$u = u^1 + \sum_s \rho_s t^{1s} = \left(1 - \sum_{s \in I} \frac{\rho_s}{\alpha_s}\right) u^1 + \sum_{s \in I} \frac{\rho_s}{\alpha_s} \nu^{1s} + \sum_{s \notin I} \rho_s t^{1s}.$$

E a condição requerida para $(\mu, \sigma) \in \Lambda(u^1)$ implica

$$\begin{aligned} 0 > \mu^T u - \sigma &= \left(1 - \sum_{s \in I} \frac{\rho_s}{\alpha_s}\right) (\mu^T u^1 - \sigma) + \sum_{s \in I} \frac{\rho_s}{\alpha_s} (\mu^T \nu^{1s} - \sigma) + \sum_{s \notin I} \rho_s \mu^T t^{1s} \\ &\geq \left(1 - \sum_{s \in I} \frac{\rho_s}{\alpha_s}\right) (\mu^T u^1 - \sigma). \end{aligned}$$

Portanto, $0 \leq \sum_{s \in I} (\rho_s / \alpha_s) < 1$. Seja o conjunto convexo $C = \{u \in U : \Theta(u, M) \leq \Theta^1\}$. Como $u^1 \in C$, $\nu^{1s} \in C$ para todo $s \in I$ e t^{1s} é direção de recessão de C para todo $s \notin I$, segue-se que $u \in C$. ■

O problema de maximização do passo 7 pode ser formulado em termos do conjunto de vértices $\{(\mu^j, \sigma^j) : 1 \leq j \leq T(u^1)\}$ de $\Lambda(u^1)$. Sejam M a matriz cujas colunas são $\{\mu^j\}$ e q o vetor com componentes $\{\sigma^j\}$. Então, de acordo com White e Anandalingam [127, pág 408], o problema mencionado é equivalente a:

$$\tau = \min_{(\beta, \gamma)} -q^T \beta - d^T \gamma \quad (5.17)$$

$$\text{s.a. } e^T \beta = 1 \quad (5.18)$$

$$M\beta + A_2 \gamma \leq 0 \quad (5.19)$$

$$\beta \geq 0, \gamma \geq 0 \quad (5.20)$$

onde $\beta \in \Re^{T(u^1)}$, $\gamma \in \Re^{n_2}$ e $e^T = (1, 1, \dots, 1) \in \Re^{T(u^1)}$. Esta formulação pode ser resolvida usando-se o método de geração de colunas.

Se o conjunto U é limitado, o problema acima sempre tem solução $\tau = G(u^{1*}, u^1)$ para algum $u^{1*} \in U$. Se $\tau < 0$, a solução de $P(M)$ já foi encontrada para o

M corrente, de acordo com a proposição 5.2.3. Todavia, se $\tau = 0$, pode existir ainda uma solução melhor (ver exemplo 5.2.1). Neste caso, o passo 8 irá retornar uma solução incorreta. Por isso, o teste $G(u^{1*}, u^1) \leq 0$ deve ser substituído pela desigualdade estrita.

Quando u^{1*} não satisfaz este novo teste de parada, ou seja, $G(u^{1*}, u^1) \geq 0$, tem-se sempre $u^{1*} \neq u^1$. De fato, $G(u^1, u^1) < 0$ pela proposição 5.2.2. Portanto, o algoritmo pode continuar a partir de u^{1*} .

Ressalvamos, por outro lado, que o problema (5.17)-(5.20) pode ser inviável quando U é ilimitado (ver exemplo 5.2.2). Neste caso, $\tau = +\infty$ e o passo 7 falha por não conseguir determinar um novo vértice u^{1*} para dar prosseguimento ao algoritmo. Assim, de modo a se ter o algoritmo bem definido, vai-se acrescentar a seguinte hipótese:

[B] U é compacto.

É importante ressaltar que as condições [A] e [B] podem ser verificadas ao mesmo tempo. Mais ainda, elas podem ser simplesmente substituídas pela hipótese de que W é não vazio e compacto. De fato, em lugar da função Θ , o algoritmo poderia usar a função $\phi(x, y, M) = \max\{\hat{F}(x, y, w, M) : u \in U\}$, definida para todo $(x, y) \in W$ e $M \geq 0$. Com isto, os mesmos resultados teóricos são obtidos e o algoritmo estará bem definido.

5.2.2 Contra-exemplos

Esta subseção apresenta dois exemplos introduzidos por Campêlo *et al.* [35], que ilustram os problemas com o algoritmo de White e Anandalingam [127]. O primeiro descreve um caso onde o passo 6 original não está bem definido, pois se obtém $\alpha_s = +\infty$ para algum s , e o passo 8 irá retornar uma solução incorreta para $P(M)$, devido a ocorrência de $\tau = 0$. Neste exemplo, o conjunto dual U é limitado. No segundo exemplo, tem-se U ilimitado e, novamente, ocorre que $\alpha_s = +\infty$ para algum s . Além disso, o passo 7 irá falhar, pois $\tau = +\infty$.

Exemplo 5.2.1

$$\begin{aligned}
 (C2) \quad & \max_{(x,y)} -0.4x - 6y_1 - 5y_2 \\
 \text{s.a} \quad & x \geq 0, \quad y = (y_1, y_2, y_3, y_4) \text{ solução de} \\
 & \max_y 0.5y_2 - y_3 - 2y_4 \\
 \text{s.a} \quad & -0, 1x - y_1 - y_2 \leq -1 \\
 & 0.2x + 1.25y_2 - y_4 \leq -1 \\
 & -x + 6y_1 + y_2 - 2y_3 \leq 1 \\
 & y_1, y_2, y_3, y_4 \geq 0
 \end{aligned}$$

Neste exemplo, tem-se $c_1 = -0.4$, $c_2 = (-6, -5, 0, 0)^T$, $d = (0, 0.5, -1, -2)^T$, $a = (-1, -1, 1)^T$, $A_1 = (-0.1, 0.2, -1)^T$ e $A_2 = \begin{pmatrix} -1 & -1 & 0 & 0 \\ 0 & 1.25 & 0 & -1 \\ 6 & 1 & -2 & 0 \end{pmatrix}$.

O conjunto dual viável do seguidor e seus vértices são:

$$\begin{aligned}
 U = \{u = (u_1, u_2, u_3) \geq 0 : & -u_1 + 6u_3 \geq 0, \\
 & -u_1 + 1.25u_2 + u_3 \geq 0.5, -2u_3 \geq -1, -u_2 \geq -2\},
 \end{aligned}$$

$$U_v = \{(2.5, 2, 0.5), (2.4, 2, 0.4), (0, 2, 0.5), (0, 0, 0.5), (0, 2, 0), (0, 0.4, 0)\}.$$

A função de penalidade torna-se

$$\begin{aligned}
 \hat{F}(x, y, u, M) = & (-0.4 - 0.1Mu_1 + 0.2Mu_2 - Mu_3)x \\
 & -6y_1 + (-5 + 0.5M)y_2 - My_3 - 2My_4 + M(u_1 + u_2 - u_3).
 \end{aligned}$$

Assuma-se que o algoritmo começa com $u^1 = (2.5, 2, 0.5)$ e $M = 10$. No passo 1, determina-se

$$\Theta(u^1, M) = -5,$$

valor atingido unicamente no vértice $(x(u^1, M), y(u^1, M)) = (0, 0, 1, 0, 2.25)$.

Os vértices adjacentes a u^1 são:

$$u^{11} = (2.4, 2, 0.4), \quad u^{12} = (0, 2, 0.5), \quad u^{13} = (0, 0, 0.5),$$

que produzem, no passo 2,

$$\Theta(u^{11}, M) = -5, \quad \Theta(u^{12}, M) = -26, \quad \Theta(u^{13}, M) = -50.$$

Como $\Theta(u^{1s}, M) \leq -5$, $1 \leq s \leq 3$, segue-se para o passo 3. Obtém-se $\Gamma(u^1, M) = 0$, pois $\pi(x(u^1), y(u^1), u^1) = 0$.

No passo 5 calculam-se os raios

$$t^{11} = (-0.1, 0, -0.1), \quad t^{12} = (-2.5, 0, 0), \quad t^{13} = (-2.5, -2, 0),$$

em direção aos vértices u^{11} , u^{12} e u^{13} respectivamente.

Cada passo α_s ao longo do raio t^{1s} , $1 \leq s \leq 3$, é calculado por (5.13)-(5.16).

Assim,

$$\alpha_s = \inf\{-45r + 3.9w + 6v_1 + 10v_3 + 20v_4 : (r, w) \in C_s, (r, w, v_1, v_2, v_3, v_4) \in D\},$$

onde

$$\begin{aligned} C_1 &= \{(r, w) \geq 0 : -1.1w = -1\}, & C_2 &= \{(r, w) \geq 0 : 25r - 2.5w = -1\}, \\ C_3 &= \{(r, w) \geq 0 : 45r + 1.5w = -1\} \end{aligned}$$

e

$$\begin{aligned} D &= \{(r, w, v_1, v_2, v_3, v_4) \geq 0 : -0.1w - v_1 - v_2 \leq -r, \\ &\quad 0.2w + 1.25v_2 - v_4 \leq -r, -w + 6v_1 + v_2 - 2v_3 \leq r\}. \end{aligned}$$

Calculando-se os ínfimos, obtêm-se $\alpha_1 = 61/55$, $\alpha_2 = 3.16$, e $\alpha_3 = +\infty$, pois $C_3 = \emptyset$. Perceba-se que um dos passos é infinito, embora U seja compacto. A partir dos passos calculados, o conjunto de cortes é definido, de acordo com o passo 6', como

$$\begin{aligned} \Lambda(u^1) &= \{(\mu_1, \mu_2, \mu_3, \sigma) : 2.5\mu_1 + 2\mu_2 + 0.5\mu_3 \leq \sigma, -5.4\mu_1 + 2\mu_2 + 0.5\mu_3 \geq \sigma, \\ &\quad 13.14\mu_1 + 11\mu_2 + 2.14\mu_3 \geq 5.5\sigma, -2.5\mu_1 - 2\mu_2 \geq 0, \sigma \in \{1, -1\}\}. \end{aligned}$$

Tomem-se $(\mu^1, \sigma^1) = (0, -0.5, 0, -1) \in \Lambda(u^1)$ e $(\mu^2, \sigma^2) = (-2, 2.5, 2, 1) \in \Lambda(u^1)$. Sejam $u^2 = (0, 2, 0)$ e $u^3 = (0, 0.4, 0)$. Então, para cada vértice de U , obtêm-se:

$$(\mu^1)^T u^1 = (\mu^1)^T u^{11} = (\mu^1)^T u^{12} = (\mu^1)^T u^2 = \sigma^1$$

$$(\mu^2)^T u^{13} = (\mu^2)^T u^3 = \sigma^2$$

Portanto, $G(u, u^1) = \min\{\mu^T u - \sigma : \lambda \in \Lambda(u^1)\} \leq 0$, para todo $u \in U$. Assim, $\tau \leq 0$ e o algoritmo concluirá, no passo 8, que u^1 é solução de (5.10). Entretanto, esta não é a resposta correta, uma vez que $\Theta(u^2, M) = -4$.

Exemplo 5.2.2

$$\begin{aligned}
 (C3) \quad & \max_{(x,y)} x + y_1 - 4y_2 \\
 \text{s.a} \quad & x \geq 0, \quad y = (y_1, y_2) \text{ solução de} \\
 & \max_y y_2 \\
 \text{s.a} \quad & x + y_1 + y_2 \leq 3 \\
 & -x - y_1 + y_2 \leq -1 \\
 & -x + y_1 + y_2 \leq 1 \\
 & x - y_1 + y_2 \leq 1 \\
 & y_2 \leq 1/2 \\
 & y_1 \geq 0, \quad y_2 \geq 0
 \end{aligned}$$

Este problema é uma modificação do exemplo 5.2.2, onde os coeficientes são $c_1 = 1$, $c_2 = (1, -4)^T$, $d = (0, 1)^T$, $a = (3, -1, 1, 1, 0.5)^T$, $A_1 = (1, -1, -1, 1, 0)^T$ e $A_2 = \begin{pmatrix} 1 & -1 & 1 & -1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}^T$.

O conjunto dual viável do seguidor é dado por

$$U = \{u = (u_1, u_2, u_3, u_4, u_5) \geq 0 : u_1 - u_2 + u_3 - u_4 \geq 0, u_1 + u_2 + u_3 + u_4 + u_5 \geq 1\}.$$

Os conjuntos de vértices dos poliedros primal e dual são:

$$Z_v = \{(0, 1, 0), (0.5, 1, 0.5), (1, 0.5, 0.5), (1, 1.5, 0.5), (1.5, 1, 0.5), (2, 1, 0), (1, 2, 0), (1, 0, 0)\}$$

$$W_v = \{(0.5, 0.5, 0, 0, 0), (0.5, 0, 0, 0.5, 0), (0, 0.5, 0.5, 0, 0), (0, 0, 0.5, 0.5, 0), (0, 0, 1, 0, 0), (0, 0, 0, 0, 1), (1, 0, 0, 0, 0)\}$$

e a função de penalidade escreve-se como

$$\begin{aligned}
 \hat{F}(x, y, u, M) = & (1 + Mu_1 - Mu_2 - Mu_3 + Mu_4)x + y_1 + (-4 + M)y_2 \\
 & -M(3u_1 - u_2 + u_3 + u_4 + 0.5u_5).
 \end{aligned}$$

Assuma-se que o algoritmo comece com $u^1 = (0, 0.5, 0.5, 0, 0)$ e M suficientemente grande. No passo 1, encontra-se uma única solução $(x(u^1, M), y(u^1, M)) = (0, 1, 0)$, que produz

$$\Theta(u^1, M) = 1.$$

Os vértices adjacentes a u^1 são:

$$\begin{aligned}
 u^{11} &= (0.5, 0.5, 0, 0, 0), & u^{12} &= (0, 0, 0.5, 0.5, 0), \\
 u^{13} &= (0, 0, 0, 0, 1), & u^{14} &= (0, 0, 1, 0, 0)
 \end{aligned}$$

com

$$\begin{aligned}\Theta(u^{11}, M) &= \Theta(u^{12}, M) = (1 - M)/2, \\ \Theta(u^{13}, M) &= 1/2, \quad \Theta(u^{14}, M) = 1 - M.\end{aligned}$$

Como $\Theta(u^{1s}, M) < 1$, $1 \leq s \leq 4$ para M suficientemente grande, segue-se para o passo 3, onde se obtém $\Gamma(u^1, M) = 0$, já que $\pi(x(u^1), y(u^1), u^1) = 0$.

No passo 5, calculam-se os raios

$$\begin{aligned}t^{11} &= (1, 0, -1, 0, 0), & t^{12} &= (0, -1, 0, 1, 0), \\ t^{13} &= (0, -1, -1, 0, 2), & t^{14} &= (0, -1, 1, 0, 0),\end{aligned}$$

em direção aos vértices u^{11} , u^{12} , u^{13} e u^{14} dados acima.

O quinto raio extremo partindo de u^1 é $t^{15} = (0, 1, 1, 0, 0)$, que é uma direção de recessão de U . De fato, considerem-se as expressões das variáveis básicas em u^1 em função das não básicas, ou seja:

$$\begin{aligned}u_2 &= 0.5 - u_4 - 0.5u_5 - 0.5f_1 + 0.5f_2, \\ u_3 &= 0.5 - u_1 - 0.5u_5 + 0.5f_1 + 0.5f_2,\end{aligned}$$

onde f_1 e f_2 são as variáveis de folga das inequações que definem U . Então, para todo $(f_1, f_2) = (0, 2\alpha) \geq 0$, tem-se

$$(0, 0.5 + \alpha, 0.5 + \alpha, 0, 0) = u^1 + \alpha t^{15} \in U.$$

Cada passo α_s é determinado por

$$\alpha_s = \inf\{r - (1 - M)w - v_1 - (-4 + M)v_2 : (r, w) \in C_s, (r, w, v_1, v_2) \in D\},$$

onde

$$\begin{aligned}C_1 = C_2 &= \{(r, w) \geq 0 : 2Mr - 2Mw = -1\}, \\ C_3 &= \{(r, w) \geq 0 : Mr - 2Mw = -1\}, \\ C_4 &= \{(r, w) \geq 0 : 2Mr = -1\}, \\ C_5 &= \{(r, w) \geq 0 : 2Mw = -1\}\end{aligned}$$

e

$$\begin{aligned}D &= \{(r, w, v_1, v_2) \geq 0 : w + v_1 + v_2 \leq 3r, -w - v_1 + v_2 \leq -r, \\ &\quad -w + v_1 + v_2 \leq r, w - v_1 + v_2 \leq r, v_2 \leq 0.5r\}.\end{aligned}$$

Como $C_4 = C_5 = \emptyset$ para todo $M \geq 0$, segue-se que $\alpha_4 = \alpha_5 = +\infty$. E, com $M = 10$, obtém-se $\alpha_1 = \alpha_2 = 0.9$ e $\alpha_3 = 0.525$. Novamente, tem-se um passo infinito. Então o passo 6' é usado para definir

$$\begin{aligned} \Lambda(u^1) = \{ & (\mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \sigma) : \mu_2 + \mu_3 \leq 2\sigma, 9\mu_1 + 5\mu_2 - 4\mu_3 \geq 10\sigma, \\ & -4\mu_2 + 5\mu_3 + 9\mu_4 \geq 10\sigma, -\mu_2 - \mu_3 + 42\mu_5 \geq 40\sigma, \\ & -\mu_2 + \mu_3 \geq 0, \mu_2 + \mu_3 \geq 0, \sigma \in \{1, -1\}\}. \end{aligned}$$

Da primeira e última inequações, percebe-se que é impossível se ter $\sigma = -1$. Sendo assim, pode-se fixar $\sigma = 1$.

O problema de maximização do passo 7 é resolvido através do modelo (5.17)-(5.20). Neste exemplo, tal modelo é inviável, ou seja, $\tau = +\infty$. Esta verificação pode ser feita usando-se o seguinte problema, associado à primeira fase do método simplex:

$$\begin{aligned} \min \quad & \beta_0 \\ \text{s.a} \quad & e^T \beta + \beta_0 = 1 \\ & M\beta + A_2\gamma \leq 0 \\ & \beta \geq 0, \gamma \geq 0, \beta_0 \geq 0 \end{aligned}$$

onde β_0 é uma variável artificial introduzida na restrição (5.18).

Substituindo-se o valor de A_2 e tomando-se a coluna de M definida pelo vértice $(\mu^1, 1) = (1, 1, 1, 1, 1, 1)$ de $\Lambda(u^1)$, obtém-se o seguinte problema auxiliar:

$$\begin{aligned} \min \quad & \beta_0 \\ \text{s.a} \quad & \beta_1 + \beta_0 = 1 \\ & \beta_1 + \gamma_1 + \gamma_2 \leq 0 \\ & \beta_1 - \gamma_1 + \gamma_2 \leq 0 \\ & \beta_1 + \gamma_1 + \gamma_2 \leq 0 \\ & \beta_1 - \gamma_1 + \gamma_2 \leq 0 \\ & \beta_1 + \gamma_2 \leq 0 \\ & \beta_0, \beta_1, \gamma_1, \gamma_2 \geq 0 \end{aligned}$$

cuja solução é $\beta_0^* = 1$ com multiplicadores simplex $\delta = 1$, associado à primeira restrição, e $\epsilon = (-1, 0, 0, 0, 0)^T$, relativo às cinco desigualdades seguintes.

Pode-se mostrar que esta é também a solução do problema (5.2.2), que considera toda a matriz M . De fato, cada coluna da matriz de restrições deste problema, associada a uma variável β_j , $j > 1$, é dada por

$$\bar{\mu}^j = \begin{pmatrix} 1 \\ \mu^j \end{pmatrix}$$

onde $(\mu^j, 1)$ é um vértice de $\Lambda(u^1)$. Sendo assim, o custo reduzido de β_j com relação à solução básica obtida vale

$$0 - \begin{pmatrix} \delta & \epsilon^T \end{pmatrix} \bar{\mu}^j = -\delta - \epsilon^T \mu^j.$$

Logo, o menor dos custos reduzidos, considerando-se implicitamente toda a matriz M , é igual a

$$\min\{-\epsilon^T \mu - \delta : (\mu, 1) \in \Lambda(u^1)\} = 0,$$

implicando que a solução ótima de (5.2.2) é atingida realmente em $\beta_0^* = 1$. Conclui-se assim que o sistema (5.18)-(5.20) é inconsistente. Por conseguinte, $\tau = +\infty$, e o algoritmo não pode prosseguir, pois não dispõe de um novo vértice u^{1*} .

Vale a pena mencionar que o pacote LINDO foi usado para resolver os dois problemas lineares acima, para calcular os passos em ambos os exemplos e para determinar $\Theta(\cdot, 10)$ em relação aos vértices do primeiro exemplo.

Conclusões e perspectivas

Neste trabalho estudamos o problema de programação linear em dois níveis (PLDN). Realizamos, primeiramente, uma revisão bibliográfica e, em seguida, desenvolvemos uma análise teórica do problema e propomos novos algoritmos.

Na parte introdutória, ressaltamos as características de PLDN que tornam difícil sua resolução, enumeramos formulações alternativas e apresentamos relações com outros problemas. Além disso, resumimos a teoria já desenvolvida e descrevemos os principais algoritmos existentes.

Essa parte inicial traz as primeiras contribuições do trabalho, na medida em que apresentamos novas demonstrações para resultados já conhecidos e estendemos para o problema com restrições no primeiro nível a propriedade de que a região viável de PLDN é formada por faces do conjunto viável relaxado.

A revisão bibliográfica revelou três aspectos deficitários na literatura: uma certa falta de formalismo matemático, o uso de hipóteses restritivas e a escassez de análises locais do problema.

A pouca rigorosidade encontrada em boa parte das publicações compromete a clareza do que é exposto e suscita dúvidas quanto à validade de alguns resultados. Na verdade, notas apontando erros têm aparecido com frequência. Em particular, neste trabalho identificamos incorreções em dois artigos e sugerimos formas de contorná-las, alterando minimamente o desenvolvimento original.

Quanto às hipóteses consideradas, constatamos que a quase totalidade dos trabalhos assume, implícita ou explicitamente, alguma condição de compacidade ou de existência de solução. Tais condições são também usadas nos algoritmos, que são incapazes de reconhecer quando o problema é ilimitado. Em especial, a maioria deles supõe que o problema relaxado tem solução.

O terceiro ponto levantado tem uma importância prática, tendo em vista a difi-

culdade de se encontrar soluções globais para o problema.

Este trabalho apresentou contribuições nos vários aspectos citados acima. Especificamente, procuramos prezar pela rigor em todas as demonstrações, estudamos as situações de ilimitação e obtivemos condições de otimalidade global e local. Mais ainda, ressaltamos que toda a teoria desenvolvida está unificada por uma metodologia única, que também propicia um enfoque algorítmico direto.

O método de penalidade que norteia nosso desenvolvimento tem sido usado em outros trabalhos como forma de encontrar soluções globais para o problema sem restrições no primeiro nível (PLDNP).

Aqui, empregamos o problema penalizado em um contexto mais amplo. Ele serviu de instrumento não só para a caracterização de soluções globais de PLDNP, quanto para a identificação dos casos de inviabilidade e ilimitação e ainda para fornecer condições de otimalidade local e local estrita. Adicionalmente, as propriedades obtidas são generalizadas, sempre que possível, para o problema com restrições no líder.

Particularmente, a análise local é bastante detalhada e as condições de otimalidade derivadas favorecem a abordagem computacional. Esta característica deve-se à noção de ponto de equilíbrio do problema penalizado, introduzida neste trabalho.

A partir dos resultados teóricos, propusemos dois algoritmos para PLDNP que se estruturam sobre o conceito de ponto de equilíbrio. O primeiro encontra ótimos locais e o segundo, ϵ -soluções globais. Na verdade, o segundo algoritmo também usa um procedimento de otimização global chamado *outer approximation*, adaptado a sua necessidade.

Os resultados computacionais obtidos pelo algoritmo local mostram a sua eficiência para os 126 problemas testados. Menos de oito segundos foram gastos para encontrar um ótimo local do problema com maiores dimensões, ou seja, 200 restrições e 500 variáveis, 335 das quais atribuídas ao líder. Isto indica que o algoritmo é capaz de tratar problemas bem maiores em tempo satisfatório.

Os experimentos com o algoritmo global não são conclusivos. Para um grupo de 30 problemas com até 90 variáveis e 25 restrições, o desempenho obtido foi bastante animador. Todas as instâncias, à exceção de uma, consumiram menos de seis segundos. Já para um outro grupo de 30 problemas, com as mesmas dimensões, os

resultados foram bem diferentes. Metade deles foi resolvida quase instantaneamente, outros demandaram mais de um minuto e alguns não puderam ser resolvidos em um tempo limite de quinze minutos.

Além do tamanho do problema, verificamos, a princípio, que os seguintes fatores influenciaram o comportamento do algoritmo: a proporção de variáveis que são atribuídas ao seguidor, a densidade da matriz de restrições e o número de ótimos locais. O quanto cada um destes aspectos contribui para a dificuldade de resolução do problema é um campo para maiores pesquisas.

O fraco desempenho do algoritmo global em algumas das instâncias testadas deveu-se essencialmente à complexidade do método *outer approximation*, usado para encontrar soluções viáveis após a introdução do corte. Entretanto, qualquer outro procedimento que desempenhe este papel pode ser perfeitamente usado pelo algoritmo. O estudo de alternativas à *outer approximation* constitui também uma possibilidade de trabalhos futuros.

Finalmente, apontamos a dificuldade em se realizar comparações com outros algoritmos conhecidos. A impossibilidade de acesso aos códigos e a inexistência de uma biblioteca de problemas-teste usados por todos são fatores que dificultam a realização de comparações mais verdadeiras. A construção de um gerador de problemas com características variadas e/ou a manutenção de uma biblioteca de problemas-teste, a exemplo do que existe em outras áreas da programação matemática, são necessidades que se apresentam no momento.

Referências Bibliográficas

- [1] F. Al-Khayyal. An implicit enumeration procedure for the general linear complementarity problem. *Mathematical Programming Studies*, 31:1–20, 1987.
- [2] M. Amouzegar e K. Moshirvazari. A penalty method for linear bilevel programming problems. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 251–271, Kluwer Academic Publishers, 1998.
- [3] G. Anandalingam e D. White. A solution method for the linear static Stackelberg problem using penalty functions. *IEEE Transactions on Automatic Control*, 35:1170–1173, 1990.
- [4] C. Audet, P. Hansen, B. Jaumard e G. Savard. Links between linear bilevel and mixed 0–1 programming problems. *Journal of Optimization: Theory and Applications*, 93(2):273–300, 1997.
- [5] C. Audet, P. Hansen, B. Jaumard e G. Savard. On the linear maxmin and related programming problems. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 181–208, Kluwer Academic Publishers, 1998.
- [6] J. Bard. An efficient point algorithm for a linear two-stage optimization problem. *Operations Research*, 31:670–684, 1983.
- [7] J. Bard. An algorithm for solving the general bilevel programming problem. *Mathematics of Operations Research*, 8:260–272, 1983.
- [8] J. Bard. Coordination of a multidivisional organization through two levels of management. *OMEGA*, 11:457–468, 1983.
- [9] J. Bard. An investigation of the linear three level programming problem. *IEEE Transactions on Systems, Man, and Cybernetics*, 14:711–717, 1984.
- [10] J. Bard. Optimality conditions for the bilevel programming problem. *Naval Research Logistics Quarterly*, 31:13–26, 1984.
- [11] J. Bard. Some properties of the bilevel programming problem. *Journal of Optimization Theory and Applications*, 68:371–378, 1991. Technical Note.
- [12] J. Bard e J. Falk. An explicit solution to the multi-level programming problem. *Computers and Operations Research*, 9:77–100, 1982.

- [13] J. Bard e J. Moore. A branch and bound algorithm for the bilevel programming problem. *SIAM Journal on Scientific and Statistical Computing*, 11:281–292, 1990.
- [14] J. Bard e J. Moore. An algorithm for the discrete bilevel programming problem. *Naval Research Logistics*, 39:419–435, 1992.
- [15] J. Bard, J. Plummer e J. Sourie. Determining tax credits for converting non-food crops to biofuels: an application of bilevel programming. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 23–50, Kluwer Academic Publishers, 1998.
- [16] M. Bazaraa, J. Jarvis e H. Sherali. *Linear programming and network flows*. John Wiley & Sons, Inc., Singapura, 2a. edição, 1990.
- [17] M. Bazaraa, H. Sherali e C. Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, Inc., New York, 2a. edição, 1993.
- [18] E. Beale. On quadratic programming. *Naval Research Logistics Quarterly*, 6:227–243, 1959.
- [19] O. Ben-Ayed. Bilevel linear programming. *Computers and Operations Research*, 20:485–501, 1993.
- [20] O. Ben-Ayed e C. Blair. Computational difficulties of bilevel linear programming. *Operations Research*, 38:556–560, 1990.
- [21] O. Ben-Ayed, C. Blair, D. Boyce e L. LeBlanc. Construction of a real-world bilevel linear programming model of the highway design problem. *Annals of Operations Research*, 34:219–254, 1992.
- [22] O. Ben-Ayed, D. Boyce e C. Blair. A general bilevel linear programming formulation of the network design problem. *Transportation Research*, 22 B:311–318, 1988.
- [23] H. Benson. On the structure and properties of a linear multilevel programming problem. *Journal of Optimization Theory and Applications*, 60:353–373, 1989.
- [24] W. Bialas e M. Karwan. *Multilevel linear programming*. Technical Report 78-1, State University of New York at Buffalo, Operations Research Program, 1978.
- [25] W. Bialas e M. Karwan. On two-level optimization. *IEEE Transactions on Automatic Control*, 27:211–214, 1982.
- [26] W. Bialas e M. Karwan. Two-level linear programming. *Management Science*, 30:1004–1020, 1984.
- [27] W. Bialas, M. Karwan e J. Shaw. *A parametric complementary pivot approach for two-level linear programming*. Technical Report 80-2, State University of New York at Buffalo, Operations Research Program, 1980.

- [28] J. Bisschop, W. Candler, J. Duloy e G. O'Mara. The indus basin model: a special application of two-level linear programming. *Mathematical Programming Study*, 20:30–38, 1982.
- [29] J. Bracken, J. Falk e F. Miercort. A strategic weapons exchange allocation model. *Operations Research*, 25:968–976, 1977.
- [30] J. Bracken e J. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21:37–44, 1973.
- [31] J. Bracken e J. McGill. Defense applications of mathematical programs with optimization problems in the constraints. *Operations Research*, 22:1086–1096, 1974.
- [32] J. Bracken e J. McGill. A method for solving mathematical programs with nonlinear programs in the constraints. *Operations Research*, 22:1097–1101, 1974.
- [33] R. Burton e B. Obel. The multilevel approach to organizational issues of the firm: a critical review. *Omega*, 5(4):395–444, 1977.
- [34] P. Calamai e L. Vicente. Generating linear and linear-quadratic bilevel programming problems. *SIAM Journal on Scientific and Statistical Computing*, 14:770–782, 1993.
- [35] M. Campêlo, S. Dantas e S. Scheimberg. *A note on a penalty function approach for solving bilevel linear programs*. Technical Report ES-467/98, Universidade Federal do Rio de Janeiro, COPPE/Programa de Engenharia de Sistemas e Computação, 1998.
- [36] M. Campêlo e S. Scheimberg. *An analysis of the bilevel linear problem by a penalty approach*. Technical Report ES-479/98, Universidade Federal do Rio de Janeiro, COPPE/Programa de Engenharia de Sistemas e Computação, 1998.
- [37] M. Campêlo e S. Scheimberg. A note on a modified simplex approach for solving bilevel linear programming problems. *European Journal of Operational Research Society*, 1999. a aparecer.
- [38] W. Candler. A linear bilevel programming algorithm: a comment. *Computers and Operations Research*, 15:297–298, 1988.
- [39] W. Candler, J. Fortuny-Amat e B. McCarl. The potential role of multilevel programming in agricultural economics. *American Journal of Agricultural Economics*, 63:521–531, 1981.
- [40] W. Candler e R. Norton. *Multilevel programming*. Technical Report 20, World Bank Development Research Center, Washington D.C., 1977.
- [41] W. Candler e R. Norton. *Multilevel programming and development policy*. Technical Report 258, World Bank Development Research Center, Washington D.C., 1977.

- [42] W. Candler e R. Townsley. A linear two-level programming problem. *Computers and Operations Research*, 9:59–76, 1982.
- [43] R. Cassidy, M. Kirby e W. Raike. Efficient distribution of resources through three levels of government. *Management Science*, 17:462–473, 1971.
- [44] A. Charnes, W. Cooper e G. Thompson. Some properties of redundant constraints and extraneous variables in direct and dual linear programming problems. *Operations Research*, 10(5):711–723, 1960.
- [45] P. Chen, P. Hansen e B. Jaumard. On-line and off-line vertex enumeration by adjacency lists. *Operations Research Letters*, 10(7):403–409, 1991.
- [46] Y. Chen e M. Florian. Congested O-D trip demand adjustment problem: bilevel programming formulation and optimality conditions. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 1–22, Kluwer Academic Publishers, 1998.
- [47] P. Clarke e A. Westerberg. A note on the optimality conditions for the bilevel programming problem. *Naval Research Logistics*, 35:413–418, 1988.
- [48] S. Dempe. A simple algorithm for the linear bilevel programming problem. *Optimization*, 18:373–385, 1987.
- [49] S. Dempe. Computing optimal incentives via bilevel programming. *Optimization*, 33:29–42, 1995.
- [50] A. deSilva. *Sensitivity formulas for nonlinear factorable programming and their application to the solution of an implicitly defined optimization model of US crude oil production*. PhD thesis, George Washington University, 1978.
- [51] J. Falk. A linear max-min problem. *Mathematical Programming*, 5:169–188, 1973.
- [52] M. Florian e Y. Chen. *A bilevel programming approach to estimating O-D matrix by traffic counts*. Technical Report CRT-750, Centre de Recherche sur les Transports, 1991.
- [53] M. Florian e Y. Chen. *A coordinate descent method for bilevel O-D matrix estimation problems*. Technical Report CRT-807, Centre de Recherche sur les Transports, 1993.
- [54] J. Fortuny-Amat e B. McCarl. A representation and economic interpretation of a two-level programming problem. *Journal of the Operational Research Society*, 32:783–792, 1981.
- [55] J. Friedman. *Oligopoly Theory*. Cambridge University Press, Cambridge, 1983.
- [56] T. Friesz, T. Miller e R. Tobin. Algorithms for spatially competitive network-facility location. *Environment Planning*, 15B:, 1988.

- [57] T. Friesz, C. Suwansirikul e R. Tobin. Equilibrium decomposition optimization: a heuristic for the continuous equilibrium network design problem. *Transportation Science*, 21:254–263, 1987.
- [58] T. Friesz, R. Tobin e T. Miller. Theory and algorithms for equilibrium network facility location. *Environment Planning*, 15B:191–203, 1988.
- [59] G. Gallo e A. Ülkcü. Bilinear programming: an exact algorithm. *Mathematical Programming*, 12:173–194, 1977.
- [60] P. Hansen, B. Jaumard e G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on Scientific and Statistical Computing*, 13:1194–1217, 1992.
- [61] A. Haurie, R. Loulou e G. Savard. A two-level systems analysis model of power cogeneration under asymmetric pricing. In *Proceedings of IEEE Automatic Control Conference*, San Diego, May 1990.
- [62] A. Haurie, G. Savard e D. White. A note on: an efficient point algorithm for a linear two-stage optimization problem. *Operations Research*, 38:553–555, 1990.
- [63] J. Herskovits. Feasible direction interior-point technique for nonlinear optimization. *Journal of Optimization: Theory and Applications*, 99(1):121–146, 1998.
- [64] J. Herskovits e A. Leontiev. An interior point algorithm for convex bilevel programming problems. In *Annales 1er Encuentro Latino Iberoamericano de Optimización*, pages 389–390, Concepción, Chile, 1997.
- [65] J. Herskovits, A. Leontiev e G. Santos. A mathematical programming algorithm for optimal design of elastic solids in contact. In *Analys 2nd World Congress of Structural and Multidisciplinary optimization*, Zakopane, Polônia, 1997.
- [66] J. Hiriart-Urruty e C. Lemarechal. *Convex analysis and minimization algorithms I*. Springer-Verlag, Berlin, 1993.
- [67] B. Hobbs e S. Nelson. A nonlinear bilevel model for analysis of electric utility demand-side planning issues. *Annals of Operations Research*, 34:255–274, 1992.
- [68] R. Horst, P. Pardalos e N. Thoai. *Introduction to global optimization*. John Wiley & Sons, Inc., Holanda, 1995.
- [69] R. Horst e H. Tuy. *Global optimization: deterministic approaches*. Springer-Verlag, Berlin, 1990.
- [70] R. Horst, J. Vries e N. Thoai. On finding new vertices and redundant constraints in cutting plane algorithms for global optimization. *Operations Research Letters*, 7(2):85–90, 1988.

- [71] R. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical Programming*, 32:146–164, 1985.
- [72] J. Júdice e A. Faustino. A sequential LCP algorithm for bilinear and concave quadratic programming. *Investigação Operacional*, 8(2):67–87, 1988.
- [73] J. Júdice e A. Faustino. A sequential LCP method for bilevel linear programming. *Annals of Operations Research*, 34:89–106, 1992.
- [74] J. Júdice e G. Mitra. Reformulation of mathematical programming problems as linear complementarity problems and investigation of their solution methods. *Journal of Optimization: Theory and Applications*, 57(1):123–149, 1988.
- [75] H. Konno. A cutting plane algorithm for solving bilinear programs. *Mathematical Programming*, 11:14–27, 1976.
- [76] F. Kyland. Hierarchical decomposition in linear economic model. *Management Science*, 21:1029–1039, 1975.
- [77] Y. Langsam, M. Augenstein e A. Tenenbaum. *Data structures using C and C++*. Prentice-Hall, Inc., New Jersey, 2a. edição, 1996.
- [78] L. Leblanc e D. Boyce. A bilevel programming algorithm for exact solution of the network design problem with user-optimal flows. *Transportation Research*, 20 B:259–265, 1986.
- [79] Y. Liu e S. Hart. Characterizing an optimal solution to the linear bilevel programming problem. *European Journal of Operational Research*, 73:164–166, 1994.
- [80] P. Loridan e J. Morgan. Weak via strong Stackelberg problems. *Journal of Global Optimization*, 8:263–287, 1996.
- [81] P. Marcotte. Network design problem with congestion effects: a case of bilevel programming. *Mathematical Programming*, 34:142–162, 1986.
- [82] P. Marcotte. A note on bilevel programming algorithm by LeBlanc and Boyce. *Transportation Research*, 22 B:233–237, 1988.
- [83] P. Marcotte e G. Marquis. Efficient implementation of heuristics for the continuous network design problem. *Annals of Operations Research*, 34:163–176, 1992.
- [84] P. Marcotte e G. Savard. A note on the pareto optimality of solutions to the linear bilevel programming problem. *Computers and Operations Research*, 18:355–359, 1991.
- [85] T. Matheiss e D. Rubin. A survey and comparison of methods for finding all vertices of convex polyhedral sets. *Mathematics of Operations Research*, 5(2):167–185, 1980.
- [86] A. Migdalas. Bilevel programming in traffic planning: models, methods and challenge. *Journal of Global Optimization*, 7:381–405, 1995.

- [87] T. Miller, T. Friesz e R. Tobin. Heuristic algorithms for delivered price spatially competitive network facility location problems. *Annals of Operations Research*, 34:177–202, 1992.
- [88] J. Moore e J. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38:911–921, 1990.
- [89] K. Moshirvaziri, M. Amouzegar e S. Jacobsen. Test problem construction for linear bilevel programming problems. *Journal of Global Optimization*, 8:235–243, 1996.
- [90] K. Murty. *Linear complementarity, linear and nonlinear programming*. Heldermann Verlag, Berlin, 1988.
- [91] M. Nicholls. The application of non-linear bi-level programming to the aluminium industry. *Journal of Global Optimization*, 8:245–261, 1996.
- [92] M. Nicholls. Optimizing the operations of an aluminium smelter using non-linear bi-level programming. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 117–148, Kluwer Academic Publishers, 1998.
- [93] H. Önal. Computational experience with a mixed solution method for bilevel linear/quadratic programs. 1992. (Preprint from the University of Illinois at Urbana-Champaign).
- [94] H. Önal. A modified simplex approach for solving bilevel linear programming problems. *European Journal of Operational Research*, 67:126–135, 1993.
- [95] H. Önal, D. Darmawan e S. Johnson. A multilevel analysis of agricultural credit distribution in East Java, Indonesia. *Computers and Operations Research*, 22:227–236, 1995.
- [96] F. Parraga. *Hierarchical programming and applications to economic policy*. PhD thesis, University of Arizona, Systems and Industrial Engineering Department, 1981.
- [97] R. Rockafellar. *Convex Analysis*. Princeton University Press, Nova Jersey, 2a. edição, 1972.
- [98] G. Savard. *Contributions à la programmation mathématique à deux niveaux*. PhD thesis, Université de Montréal, École Polytechnique, 1989.
- [99] H. Serali e C. Shetty. A finitely convergent algorithm for bilinear programming problems using polar cuts and disjunctive face cuts. *Mathematical Programming*, 19:14–31, 1980.
- [100] K. Shimizu, Y. Ishizuka e J. Bard. *Nondifferentiable and two-level mathematical programming*. Kluwer Academic Publishers, 1997.
- [101] M. Simaan e J. Cruz. On the Stackelberg strategy in nonzero-sum games. *Journal of Optimization: Theory and Applications*, 11(5):533–555, 1973.

- [102] M. Simaan e J. Cruz. Additional aspects of the Stackelberg strategy in nonzero-sum games. *Journal of Optimization: Theory and Applications*, 11(6):613–626, 1973.
- [103] H. Stackelberg. *The theory of the market economy*. Oxford University Press, New York, Oxford, 1952.
- [104] G. Stavroulakis e H. Günzel. Optimal structure design in nonsmooth mechanics. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 91–115, Kluwer Academic Publishers, 1998.
- [105] S. Suh e T. Kim. Solving nonlinear bilevel programming models of the equilibrium network design problem: a comparative review. *Annals of Operations Research*, 34:203–218, 1992.
- [106] T. Thieu. A note on the solution of bilinear programming problems by reduction to concave minimization. *Mathematical Programming*, 41:249–260, 1988.
- [107] R. Tobin e T. Friesz. Spatial competition facility location models: definition, formulation and solution approach. *Annals of Operations Research*, 6:49–74, 1986.
- [108] H. Tuy. Concave programming under linear constraints. *Soviet Mathematics*, 5:1437–1440, 1964.
- [109] H. Tuy. Bilevel linear programming, multiobjective programming and monotonic reverse convex programming. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 295–314, Kluwer Academic Publishers, 1998.
- [110] H. Tuy e S. Ghannadan. A new branch-and-bound method for bilevel linear programs. In A. Migdalas, P. Pardalos e P. Värbrand, editores, *Multilevel optimization: algorithms and applications*, Nonconvex Optimization and its Applications 20: 231–249, Kluwer Academic Publishers, 1998.
- [111] H. Tuy, A. Migdalas e P. Värbrand. A global optimization approach for the linear two-level program. *Journal of Global Optimization*, 3:1–23, 1993.
- [112] H. Tuy, A. Migdalas e P. Värbrand. A quasiconcave minimization method for solving linear two-level programs. *Journal of Global Optimization*, 4:243–263, 1994.
- [113] G. Ünlü. A linear bilevel programming algorithm based on bicriteria programming. *Computers and Operations Research*, 14:173–179, 1987.
- [114] H. Vaish e C. Shetty. The bilinear programming problem. *Naval Research Logistics Quarterly*, 23:303–309, 1976.

- [115] L. Vicente. *Bilevel programming*. Master's thesis, University of Coimbra, Department of Mathematics, 1992. Written in Portuguese.
- [116] L. Vicente e P. Calamai. Bilevel and multilevel programming: a bibliography review. *Journal of Global Optimization*, 5:291–306, 1994.
- [117] L. Vicente, G. Savard e J. Júdice. Descent approaches for quadratic bilevel programming. *Journal of Optimization Theory and Applications*, 81:379–399, 1994.
- [118] L. Vicente, G. Savard e J. Júdice. Discrete linear bilevel programming problem. *Journal of Optimization: Theory and Applications*, 89(3):597–614, 1996.
- [119] U. Wen. *Mathematical methods for multilevel linear programming*. PhD thesis, State University of New York at Buffalo, Department of Industrial Engineering, 1981.
- [120] U. Wen e S. Hsu. Linear bi-level programming problems - a review. *Journal of the Operational Research Society*, 42:125–133, 1991.
- [121] U. Wen e S. Hsu. Efficient solutions for the linear bilevel programming problem. *European Journal of Operational Research*, 62:354–362, 1992.
- [122] U. Wen e S. Hsu. A note on a linear bilevel programming algorithm based on bicriteria programming. *Computers and Operations Research*, 16:79–83, 1989.
- [123] U. Wen e C. Jiang. A multilevel programming approach in comission rate setting problem. *J. Chin. Inst. Ind. Engrs*, 5:43–49, 1988.
- [124] U. Wen e S. Lin. Finding an efficient solution to linear bilevel programming problem: An Effective approach. *Journal of Global Optimization*, 8:295–306, 1996.
- [125] U. Wen e Y. Yang. Algorithms for solving the mixed integer two-level linear programming problem. *Computers and Operations Research*, 17:133–142, 1990.
- [126] D. White. A linear programming approach to solving bilinear programmes. *Mathematical Programming*, 56:45–50, 1992.
- [127] D. White e G. Anandalingam. A penalty function approach for solving bi-level linear programs. *Journal of Global Optimization*, 3:397–419, 1993.
- [128] D. Wolf e Y. Smeers. Optimal dimensioning of pipe networks with application to gas transmission networks. *Operations Research*, 44(4):596–608, 1996.
- [129] M. Zeleny. *Multiple Criteria Decision Making*. McGraw-Hill, New York, 1982.
- [130] J. Zhang e D. Zhu. A bilevel programming method for pipe network optimization. *SIAM Journal on Optimization*, 6(3):838–857, 1996.
- [131] R. Zhang. Problems of hierarchical optimization in finite dimensions. *SIAM Journal on Optimization*, 4(3):521–536, 1994.