



ALGORITMOS DE DIREÇÕES VIÁVEIS PARA OTIMIZAÇÃO NÃO DIFERENCIÁVEL

Mario Tanaka Filho

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Mecânica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Mecânica.

Orientadores: José Herskovits Norman
Anatoli Leontiev

Rio de Janeiro
Março de 2011

ALGORITMOS DE DIREÇÕES VIÁVEIS PARA OTIMIZAÇÃO NÃO
DIFERENCIÁVEL

Mario Tanaka Filho

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE)
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM CIÊNCIAS EM ENGENHARIA MECÂNICA.

Examinada por:

Prof. José Herskovits Norman, D.Ing.

Prof. Anatoli Leontiev, Ph.D.

Prof. Nestor Alberto Zouain Pereira, D.Sc.

Prof. Jean Rodolphe Roche , Ph.D.

Prof. José Mario Martínez, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2011

Tanaka Filho, Mario

Algoritmos de Direções Viáveis para Otimização Não Diferenciável/Mario Tanaka Filho. – Rio de Janeiro: UFRJ/COPPE, 2011.

XII, 134 p.: il.; 29,7cm.

Orientadores: José Herskovits Norman

Anatoli Leontiev

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Mecânica, 2011.

Referências Bibliográficas: p. 110 – 116.

1. Otimização não diferenciável. 2. Algoritmo de direções viáveis e ponto interior. 3. Métodos de Plano de corte. 4. Métodos de Feixe. I. Norman, José Herskovits *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Mecânica. III. Título.

A meu filho recém-nascido

LUCAS.

*A meus pais, irmãos
e a minha esposa Solange.*

Agradecimentos

A Deus, por permitir este momento tão singular em minha vida.

Ao professor José Herskovits Norman pela orientação, amizade e conhecimentos transmitidos ao longo da realização deste trabalho, bem como, por seu exemplo profissional e humano.

Ao Prof. Anatoli Leontiev pela orientação, por sua atenção ao me receber quando ainda não tinha certeza de onde fazer o doutoramento e por me indicar ao Laboratório.

Aos Profs. Jean R. Roche (Nancy Université, França) e Napsu Karmita (University of Turku, Finlândia) pelo suporte, amizade e simpatia.

Aos professores, o pessoal do administrativo e amigos do Programa de Engenharia Mecânica PEM-COOPE/UFRJ, sempre dispostos a ajudar e por proporcionar um excelente ambiente de convívio.

Agradeço a meus pais: Mário Tanaka e Djelma Neves Tanaka, por todo esforço e dedicação em manter as condições necessárias para meus estudos. E portanto, hoje sem dúvida nenhuma, considero que este momento também é deles.

A minha esposa, Solange Tanaka, por seu apoio, pela preocupação de sempre tentar proporcionar um ambiente favorável a meus estudos e por estar ao meu lado me dando força.

A família Penha: Sogra, sogra e cunhadas pelo constante incentivo.

A todos os colegas e amigos do Laboratório OptimizE: Arminda, Alfredo Canelas, Henry Cortés, Miguel Aroztegui, Jorge Zerpa, Elmer, Pavel e Helena, Sandro, Evandro e aos outros que passaram pelo Laboratório, pelo agradável e sempre descontraído ambiente de estudo proporcionado.

Agradeço o auxílio financeiro concedido pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), através das bolsas de doutorado e Iniciação Científica; A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes), pela bolsa de Mestrado, fundamentais para o suporte de meus estudos nestes longos anos de formação.

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

ALGORITMOS DE DIREÇÕES VIÁVEIS PARA OTIMIZAÇÃO NÃO DIFERENCIÁVEL

Mario Tanaka Filho

Março/2011

Orientadores: José Herskovits Norman
Anatoli Leontiev

Programa: Engenharia Mecânica

Problemas de otimização não diferenciável aparecem em muitas aplicações práticas, como por exemplo, em Mecânica, Economia e Controle Ótimo. Contudo, a aplicação direta de métodos diferenciáveis, ou seja, métodos baseados em informações do gradiente, não é recomendada devido a natureza não diferenciável dos problemas. Com isso em vista, neste trabalho são apresentadas técnicas para a resolução de três problemas diferentes que envolvem funções não necessariamente diferenciáveis. Inicialmente, considera-se o caso convexo sem restrições. Depois, o método é estendido para abranger o caso não convexo, onde são consideradas funções localmente Lipschitz contínuas. Em seguida, introduz-se o problema convexo com restrições convexas não diferenciáveis. São também apresentados os resultados numéricos da experiência computacional preliminar e uma aplicação em Otimização Topológica de estruturas reticuladas robustas.

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

FEASIBLE DIRECTIONS ALGORITHMS FOR NONSMOOTH OPTIMIZATION

Mario Tanaka Filho

March/2011

Advisors: José Herskovits Norman
Anatoli Leontiev

Department: Mechanical Engineering

Nonsmooth optimization problems appear in many practical applications, such as in Mechanics, Economics, Optimal Control and Engineering. However, the direct application of methods differentiable, i.e., methods based on gradient information, is not recommended due to the nonsmooth nature of the problems. With this in mind, this study presents techniques for solving three different problems involving functions not necessarily differentiable. Initially, we consider the convex case without restrictions. Then the method is extended to cover the case non-convex, where they are considered locally Lipschitz continuous functions. Then introduces the problem convex with convex constraints not differentiable. We also present numerical results of preliminary computational experience and an application on Robust Truss Topology Design.

Sumário

Lista de Figuras	x
Lista de Tabelas	xi
Lista de Símbolos	xii
Introdução	1
1 Preliminares	4
1.1 Notações e Definições Básicas	4
1.2 Análise Não Diferenciável	8
1.3 Condições de Otimalidade	11
1.4 Otimização Convexa Não Diferenciável	13
1.5 Otimização Não Convexa Não Diferenciável	25
1.6 Otimização Convexa Não Diferenciável com Restrições	27
1.7 Algoritmo de Pontos Interiores e Direções Viáveis	29
2 Algoritmo para Otimização Convexa Não Diferenciável	35
2.1 Método	35
2.2 Análise da Convergência	41
2.3 Resultados Numéricos	48
3 Algoritmo para Otimização Não Convexa e Não Diferenciável	54
3.1 Método	55
3.2 Análise da convergência	61
3.3 Resultados Numéricos	70
4 Uma Técnica para Otimização Restrita Não Diferenciável	74
4.1 Método	75
4.2 Análise da Convergência	82
4.3 Resultados Numéricos	89

5	Otimização Topológica Robusta de Estruturas Reticuladas	93
5.1	Introdução	93
5.2	Modelos em Otimização Topológica	94
5.3	Modelo de Otimização Topológica Robusta	97
6	Conclusões	106
6.1	Contribuições deste trabalho	106
6.2	Trabalhos Futuros	108
	Referências Bibliográficas	110
A	Problemas Irrestritos	117
A.1	Problemas Teste Convexos	118
A.2	Problemas Teste Não Convexos	122
B	Problemas Com Restrição	126
B.1	Problemas com restrições	127

Lista de Figuras

1.1	Semi-continuidade interior do ε -subdiferencial	16
1.2	Direções de descida: caso diferenciável e não diferenciável	17
1.3	Iterações do método de planos de corte	19
1.4	Introdução de uma função afim quase “horizontal”	20
1.5	Direção de busca do FDIPA	32
2.1	Performance Profiles: número de iterações	53
2.2	Performance Profiles: número de avaliações da função	53
3.1	Encontrando o próximo iterado do Algoritmo.	57
3.2	Determinação de um plano viável.	58
5.1	Treliça do Exemplo 1.	100
5.2	Estrutura ótima obtida no Exemplo 1.	101
5.3	Exemplo 1 - Evolução dos quatro maiores auto-valores.	101
5.4	Treliça do Exemplo 2.	102
5.5	Estrutura ótima obtida no Exemplo 2.	102
5.6	Evolução dos quatro maiores auto-valores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$	102
5.7	Estrutura ótima obtida no Exemplo 3	103
5.8	Evolução dos seis maiores autovalores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$	103
5.9	Estrutura ótima obtida no Exemplo 4.	104
5.10	Evolução dos seis maiores auto-valores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$	104

Lista de Tabelas

2.1	Tabela de Problemas	49
2.2	Solvers utilizados na comparação	49
2.3	Resultados para o número de iterações	50
2.4	Resultados para o número de avaliações da função	50
2.5	Resultados para o valor da função objetivo	51
3.1	Problemas teste.	72
3.2	Solvers utilizados na comparação	72
3.3	Resultados da Experiência Numérica.	73
4.1	Problemas Teste	91
4.2	Resultados Preliminares do NFDCA	92
4.3	Valores do RELACC	92
5.1	Resultados dos exemplos de otimização.	105
5.2	Volumes das barras de estrutura ótima.	105
A.1	Tabela de Problemas	117
B.1	Problemas Teste	126
B.2	parâmetros	133

Lista de Símbolos

A, B, G	matrizes, p. 7
$B(\mathbf{x}, r)$	Bola aberta de centro \mathbf{x} e raio r , p. 4
$N_C(\bar{\mathbf{x}})$	cone normal, p. 5
$T_C(\bar{\mathbf{x}})$	cone tangente, p. 5
\mathbb{R}^n	Espaço Euclidiano n-dimensional, p. 4
$\text{conv } S$	fecho convexo de S , p. 5
$\nabla f(\mathbf{x})$	gradiente de f em \mathbf{x} , p. 6
$\nabla^2 f(\mathbf{x})$	matriz Hessiana, p. 7
$\partial f(\mathbf{x})$	subdiferencial de f em \mathbf{x} , p. 9
\mathbf{x}, \mathbf{y}	vetores (coluna), p. 4
\mathbf{x}^T	vetor transposto, p. 4
$\mathbf{x}^T \mathbf{y}$	produto interno, p. 4
$\{\mathbf{x}^k\}, (\mathbf{x}^k)$	sequencia, p. 6
$f'(\mathbf{x}; \mathbf{d})$	derivada direcional de $f : \mathbb{R}^n \rightarrow \mathbb{R}$ em \mathbf{x} na direção \mathbf{d} , p. 6
$f^\circ(\mathbf{x}; \mathbf{d})$	derivada direcional generalizada de $f : \mathbb{R}^n \rightarrow \mathbb{R}$ em \mathbf{x} na direção \mathbf{d} , p. 9
x_i	componente i do vetor \mathbf{x} , p. 4

Introdução

A teoria clássica de otimização presume certa diferenciabilidade e fortes hipóteses de regularidade como pode ser visto em Fletcher [1]. Contudo, estas hipóteses estão longe de acontecer na prática, onde em muitos casos a própria física do problema impõe um modelo não diferenciável. Problemas de otimização não diferenciáveis aparecem em muitos campos de aplicação, como por exemplo, em Mecânica [2], Economia [3], Controle Ótimo [4]. A origem da não diferenciabilidade pode ser dividida em quatro classes, [5]: *inerente*, *tecnológica*, *metodológica* e *numérica*.

No caso da não diferenciabilidade *inerente*, considera-se que o fenômeno original contém nele mesmo várias descontinuidades e irregularidades. Um exemplo típico, são as mudanças de fase do material no processo de moldagem contínua do aço (veja [6]) e modelos lineares por parte em economia (veja [7]).

A não diferenciabilidade *tecnológica* num modelo é usualmente causada por algumas restrições tecnológicas extras. Estas restrições podem causar uma dependência não diferenciável entre as variáveis e as funções, ainda que, as funções originais sejam continuamente diferenciáveis. Estes tipos de exemplos são chamados de problemas de obstáculo em otimização de formas (veja [8]).

Exemplos de não diferenciabilidade *metodológica* aparecem no métodos de função penalidade exata, e métodos de decomposição de Lagrange.

Finalmente, existem problemas que podem ser analiticamente diferenciáveis mas *numericamente* não diferenciáveis. Estes problemas são, em geral, chamados de “*stiff problems*” os quais são numericamente instáveis e se comportam como problemas não diferenciáveis.

Para problemas onde não se tem a diferenciabilidade em todos os pontos do domínio da função, existe uma área da Programação Matemática denominada Otimização Não Diferenciável (*Nonsmooth Optimization*), que utiliza técnicas, que substituem o Cálculo Diferencial clássico, oriundas de uma área da Matemática chamada de Análise Convexa, [9].

Existem grandes dificuldades quando se lida com funções não diferenciáveis e, em muitos casos, essas funções tem mínimo onde o gradiente não está definido. Ratificando tal ideia, não é preciso ir muito longe para entender melhor as dificuldades causadas pela não diferenciabilidade. Para tanto, basta considerar a

função valor absoluto $f(\mathbf{x}) = |\mathbf{x}|$, com $\mathbf{x} \in \mathbb{R}$, nota-se que f não é diferenciável na origem, justamente o ponto onde ocorre o mínimo. Então, se para esta função, for aplicado qualquer método de otimização diferenciável, este nem sequer reconhecerá o ponto de mínimo $\bar{\mathbf{x}} = 0$, pois nestes métodos o mínimo precisa satisfazer o famoso resultado de Fermat

$$\nabla f(\bar{\mathbf{x}}) = 0,$$

que é, na verdade, uma condição de otimalidade necessária.

Diante do exposto, vem sendo desenvolvidos uma série de métodos para lidar com este tipo de problema como pode ser visto, por exemplo, nos trabalhos de Kelley [10], Kiwiel [11], Lemaréchal [12] e Mäkelä [13], entre outros.

Os métodos considerados mais eficientes e confiáveis quando a função objetivo é convexa, são os métodos de Feixes, (veja [11, 14] ou [15]), estes métodos são baseados na teoria de subdiferenciais desenvolvida por Rockafellar [9] e Clarke [16]. O que estes métodos tem em comum é que, em cada iteração, requerem a utilização de um único subgradiente além do valor da função, onde suas ideias básicas se concentram em aproximar o subdiferencial (que é o conjunto dos subgradientes) da função objetivo, usando informações dos subgradientes armazenados em iterações anteriores no chamado feixe. A história destes métodos começa com o método chamado ε -*steepest descent* apresentado por Lemaréchal em 1976, [17]. Para ter uma melhor compreensão da discussão do caso convexo, veja [11] ou [18].

Problemas envolvendo funções não convexas e não diferenciáveis são mais difíceis de lidar. Contudo, durante as últimas três décadas, consideráveis progressos tem sido realizados nessa área, Um passo crucial foi a tese de doutorado de Clarke em 1973. Corroborando com tal progresso sabe-se que um cálculo eficiente, bem como condições de otimalidade aplicáveis, em termos de construções convexas locais, pode ser desenvolvido para a classe de funções localmente Lipschitz contínuas, veja [16], [19] ou [20].

Problemas convexos com restrições não diferenciáveis são ainda mais complexos e poucos métodos são encontrados na literatura. Problemas convexos com restrições consideradas “*fáceis*”, tais como restrições lineares, podem ser resolvidos inserindo-se tais restrições diretamente em cada problema quadrático (veja [21, 22]). Para o problema convexo com restrições mais gerais, uma forma bem popular, para métodos de feixe, é encontrar um mínimo irrestrito para a chamada *improvement function* [23], esta abordagem foi utilizada em [24] e nos capítulos 5 e 6 de [11]. Outra estratégia é resolver um problema irrestrito equivalente com o uso de uma função objetivo de penalidade exata [25]. Em [26, 27] uma estratégia de filtros [28] é sugerida como alternativa ao uso da função de penalidade num método de feixes.

Neste trabalho são apresentadas três alternativas de algoritmos para Otimização Não Diferenciável: O primeiro algoritmo, considera o caso convexo sem restrições,

o qual foi inicialmente proposto por Freire [29] em sua tese de doutoramento em 2005. Em seguida, apresenta-se um algoritmo que lida com problemas não convexos sem restrições. O terceiro e último algoritmo considera problemas convexos com restrições de desigualdade convexas.

Os métodos são híbridos, neles são utilizadas algumas ideias do método clássico de planos de corte de Kelley [10], para realizar aproximações das funções não diferenciáveis. E em cada iteração, utiliza-se o método de pontos interiores e direções viáveis FDIPA, desenvolvido por Herskovits em [30], para gerar direções de descida viáveis.

Este trabalho está organizado da seguinte forma: No Capítulo 1, reservados as preliminares, são definidas a notação e alguns resultados básicos da Análise Não Diferenciável, generaliza-se as condições de otimalidade para o caso não diferenciável, em seguida, faz-se uma revisão bibliográfica para Otimização Não Diferenciável onde estão incluídas algumas técnicas mais recentes e por fim é apresentado o *Feasible Direction Interior Point Algorithm* (FDIPA).

No Capítulo 2 apresenta-se o algoritmo para problemas não diferenciáveis convexos sem restrições. No Capítulo 3 propõe-se um método para a resolução do problema

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \text{sujeito a } \mathbf{x} \in \mathbb{R}^n \end{cases}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função localmente Lipschitz contínua. Constam ainda, o estudo da convergência global e os resultados da experiência computacional. No Capítulo 4 propõe-se um método para a resolução do problema restrito

$$\begin{cases} \text{minimize}_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \\ \text{sujeito a } c(\mathbf{x}) \leq 0 \end{cases}$$

onde $f, c : \mathbb{R}^n \rightarrow \mathbb{R}$ são funções convexas em geral não diferenciáveis. Este método é uma extensão direta do método desenvolvido por Freire em [29], pois são acrescentadas restrições de desigualdade como acima e são resolvidas uma sequência de problemas auxiliares que vão sendo definidos a medida em que as restrições funcionais vão sendo aproximadas por planos de corte.

No Capítulo 5 é feita uma aplicação do método apresentado no Capítulo 2 para Otimização Topológica de estruturas reticuladas robustas.

Finalmente, são apresentadas as conclusões do trabalho obtidas até o presente momento e os possíveis trabalhos futuros.

Capítulo 1

Preliminares

Neste Capítulo, são introduzidas algumas notações, conceitos e resultados básicos necessários ao estudo da análise não diferenciável, cujas ideias estão baseadas principalmente nos trabalhos de Clarke [16], Makela [13] e Lemarechal [31]. Então generaliza-se os conceitos de diferencial para funções convexas e localmente Lipschitz contínuas, respectivamente. Com isso, são generalizadas as condições clássicas de otimalidade para o caso não diferenciável. Em seguida, faz-se uma pequena descrição de alguns métodos clássicos bem como alguns métodos mais recentes, para otimização não diferenciável. E finalmente, apresenta-se o *Feasible Direction Interior Point Algorithm* (FDIPA), um método para otimização não linear desenvolvido por Herskovits em [30]. As demonstrações dos resultados apresentados neste capítulo podem ser encontradas nos trabalhos citados acima.

1.1 Notações e Definições Básicas

Todos os vetores são considerados como vetores coluna. Denota-se o **produto interno** usual por $\mathbf{x}^T \mathbf{y}$ e por $\|\mathbf{x}\|$ a **norma** no espaço Euclidiano n -dimensional, i.e.,

$$\|\mathbf{x}\| = (\mathbf{x}^T \mathbf{x})^{\frac{1}{2}} = \left(\sum_{i=1}^n \mathbf{x}_i^2 \right)^{\frac{1}{2}},$$

onde $\mathbf{x} \in \mathbb{R}^n$ e $\mathbf{x}_i \in \mathbb{R}$ é a i -ésima componente do vetor \mathbf{x} .

A bola unitária com centro em $\mathbf{x} \in \mathbb{R}^n$ e raio $r > 0$ é denotada por

$$B(\mathbf{x}; r) = \{\mathbf{y} \in \mathbb{R}^n \mid \|\mathbf{y} - \mathbf{x}\| < r\}.$$

Um conjunto $S \subset \mathbb{R}^n$ é dito **convexo** se $\lambda \mathbf{x} + (1 - \lambda) \mathbf{y} \in S$, onde \mathbf{x} e $\mathbf{y} \in S$ e $\lambda \in [0, 1]$. Geometricamente, se esta dizendo que todo segmento de reta que une os pontos \mathbf{x} e \mathbf{y} está inteiramente contido em S sempre que $\mathbf{x}, \mathbf{y} \in S$. Se S_1 e S_2 são conjuntos convexos em \mathbb{R}^n e $\lambda_1, \lambda_2 \in \mathbb{R}$, então $\lambda_1 S_1 + \lambda_2 S_2$ também é convexo. Se

$S_i \subset \mathbb{R}^n$ são conjuntos convexos para $i = 1, 2, \dots, m$, então sua interseção $\bigcap_{i=1}^m S_i$ é também um conjunto convexo.

Denota-se por $\text{conv}(S)$ o **fecho convexo** de $S \subset \mathbb{R}^n$, i.e., a interseção de todos os conjuntos convexos que contém S , ou ainda,

$$\text{conv}(S) = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{x} = \sum_{i=1}^k \lambda_i \mathbf{x}_i, \sum_{i=1}^k \lambda_i = 1, \mathbf{x}_i \in \mathbf{R}^n, \lambda_i \geq 0 \right\}.$$

O fecho convexo de um conjunto S é o menor conjunto convexo contendo S e S é convexo se e somente se $S = \text{conv}(S)$. Além disso, o fecho convexo de um conjunto compacto é compacto.

Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita **convexa** se

$$f(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y}), \quad (1.1)$$

onde \mathbf{x} e \mathbf{y} estão em \mathbb{R}^n e $\lambda \in [0, 1]$. Se a desigualdade for estrita em (1.1) para todo $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ tal que $\mathbf{x} \neq \mathbf{y}$ e $\lambda \in (0, 1)$, a função f é dita **estritamente convexa**.

Um conjunto C é denominado **cone** se contém todos os múltiplos positivos de seus elementos, i.e., se $\mathbf{x} \in C$ e $\lambda > 0$, então $\lambda \mathbf{x} \in C$. Pela definição, se C é um cone não vazio, necessariamente $0 \in C$.

O **cone tangente** (de direções tangentes) de um conjunto convexo C pode ser definido como

$$T_C(\bar{\mathbf{x}}) := \left\{ \mathbf{d} \in \mathbb{R}^n \mid \begin{array}{l} \forall \{t_k\} \subset \mathbb{R}_+, \{t_k\} \rightarrow 0_+, \\ \exists \{\mathbf{d}^k\} \subset \mathbb{R}^n, \{\mathbf{d}^k\} \rightarrow \mathbf{d}, \text{ tal que} \\ \bar{\mathbf{x}} + t_k \mathbf{d}^k \in C \text{ para todo } k \in \mathbb{N} \end{array} \right\}.$$

O **cone normal** (cone de direções normais) de um conjunto convexo C em $\bar{\mathbf{x}} \in C$ é o conjunto

$$N_C(\bar{\mathbf{x}}) := \left\{ \mathbf{d} \in \mathbb{R}^n \mid \mathbf{d}^T (\mathbf{x} - \bar{\mathbf{x}}) \leq 0 \quad \forall \mathbf{x} \in C \right\}.$$

Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita **localmente Lipschitz continua** com constante $L > 0$ em $\mathbf{x} \in \mathbb{R}^n$ se existe um número positivo ε tal que

$$|f(\mathbf{y}) - f(\mathbf{z})| \leq L \|\mathbf{y} - \mathbf{z}\|,$$

para todo $\mathbf{y}, \mathbf{z} \in B(\mathbf{x}, \varepsilon)$. No que segue utiliza-se a denominação curta localmente Lipschitz.

Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa em $\mathbf{x} \in \mathbb{R}^n$, então f é localmente Lipschitz em \mathbf{x} .

Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita **inferiormente semi-continua** (resp. superior) num ponto $\mathbf{x} \in \mathbb{R}^n$ se, para cada sequência $(\mathbf{x}_k) \subset \mathbb{R}^n$ com $(\mathbf{x}_k) \rightarrow \mathbf{x}$, tem-se

$$f(\mathbf{x}) \leq \liminf_{k \rightarrow \infty} f(\mathbf{x}_k) \left(\text{resp. } f(\mathbf{x}) \geq \liminf_{k \rightarrow \infty} f(\mathbf{x}_k) \right).$$

Uma função que é semi-continua superior e inferior é uma função contínua.

Agora serão revistos alguns conceitos básicos da teoria das funções diferenciáveis: Uma função é dita **diferenciável** em $\mathbf{x} \in \mathbb{R}^n$ se existe um vetor $\nabla f(\mathbf{x}) \in \mathbb{R}^n$ e uma função $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}$ tal que para todo $\mathbf{d} \in \mathbb{R}^n$,

$$f(\mathbf{x} + \mathbf{d}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{d} + \|\mathbf{d}\| \varepsilon(\mathbf{d}),$$

onde o vetor $\nabla f(\mathbf{x})$ é o **vetor gradiente** de f em \mathbf{x} e $\varepsilon(\mathbf{d}) \rightarrow 0$ sempre que $\|\mathbf{d}\| \rightarrow 0$. O vetor gradiente tem a seguinte fórmula

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_n} \right)^T,$$

onde as componentes $\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_i}$, $i = 1, 2, \dots, n$ são as chamadas **derivadas parciais** da função f . Se uma função é diferenciável e suas derivadas parciais são todas contínuas, então a função é dita **continuamente diferenciável**.

O limite

$$f'(\mathbf{x}; \mathbf{d}) = \lim_{t \rightarrow 0} \frac{f(\mathbf{x} + t\mathbf{d}) - f(\mathbf{x})}{t}$$

(se existe) é chamado de **derivada direcional** de f em $\mathbf{x} \in \mathbb{R}^n$ na direção $\mathbf{d} \in \mathbb{R}^n$. Se uma função f é diferenciável em \mathbf{x} , então a derivada direcional existe em cada direção $\mathbf{d} \in \mathbb{R}^n$ e

$$f'(\mathbf{x}; \mathbf{d}) = \nabla f(\mathbf{x})^T \mathbf{d}.$$

Se, ainda, f for convexa, então para todo $\mathbf{y} \in \mathbb{R}^n$

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}).$$

Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita **duas vezes diferenciável** em $\mathbf{x} \in \mathbb{R}^n$ se existe um vetor $\nabla f(\mathbf{x}) \in \mathbb{R}^n$, uma matriz simétrica $\nabla^2 f(\mathbf{x}) \in \mathbb{R}^{n \times n}$ e uma função $\varepsilon : \mathbb{R}^n \rightarrow \mathbb{R}$ tal que para todo $\mathbf{d} \in \mathbb{R}^n$,

$$f(\mathbf{x} + \mathbf{d}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T \nabla^2 f(\mathbf{x}) \mathbf{d} + \|\mathbf{d}\| \varepsilon(\mathbf{d}),$$

onde $\varepsilon(\mathbf{d}) \rightarrow 0$ sempre que $\|\mathbf{d}\| \rightarrow 0$. A matriz $\nabla^2 f(\mathbf{x})$ é chamada de **Hessiana** de f em \mathbf{x} e é definida por

$$\nabla^2 f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{bmatrix}.$$

onde as componentes $\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}$, $i, j = 1, 2, \dots, n$ são as chamadas **derivadas parciais de segunda ordem** da função f . Se uma função é duas vezes diferenciável e suas derivadas parciais de segunda ordem são todas contínuas, então a função é dita **duas vezes continuamente diferenciável**.

Uma matriz $A \in \mathbb{R}^{n \times n}$ é chamada **definida positiva** se $A = A^T$, i.e., A é **simétrica** e

$$\mathbf{x}^T A \mathbf{x} > 0$$

para todo vetor $\mathbf{x} \in \mathbb{R}^n$.

Considere um mapeamento F que associa cada $\mathbf{x} \in X \subset \mathbb{R}^n$ a um conjunto do \mathbb{R}^n , ou seja, $\mathbf{x} \mapsto F(\mathbf{x}) \subset \mathbb{R}^n$. Denomina-se tal mapeamento de **multi-função**.

O **domínio de F** é definido por $\text{dom } F := \{\mathbf{x} \in X : F(\mathbf{x}) \neq \emptyset\}$.

Diz-se que F é **fechada** se seu gráfico (i.e., a união de $\{\mathbf{x}\} \times F(\mathbf{x}) \subset X \times \mathbb{R}^n$) é um conjunto fechado. Dize-se ainda que ela é **localmente limitada** perto de \mathbf{x}^* se para alguma vizinhança V de \mathbf{x}^* e algum conjunto limitado $B \subset \mathbb{R}^n$, tem-se $V \subset \text{dom } F$ e $F(V) \subset B$.

Seja F uma multi-função fechada e localmente limitada em \mathbf{x}^* . Então, F é **semi-contínua exterior** (resp. semi-contínua interior) se, para todo $\varepsilon > 0$, existe uma vizinhança $V_{\mathbf{x}^*}$ de \mathbf{x}^* tal que $\mathbf{x} \in V_{\mathbf{x}^*}$ implica

$$F(\mathbf{x}) \subset F(\mathbf{x}^*) + B(0; \varepsilon) \left(F(\mathbf{x}^*) \subset F(\mathbf{x}) + B(0; \varepsilon) \right).$$

Além disso, se F é semi-contínua exterior e interior, então ela é contínua.

1.2 Análise Não Diferenciável

A teoria da Análise não diferenciável para funções convexas está baseada na chamada Análise Convexa, e por esta razão faz-se uma breve introdução baseada nos conceitos de convexidade (veja Rockafellar [9]). Primeiramente define-se o conceito de subdiferencial de uma função convexa e depois estende-se os resultados para funções localmente Lipschitz não convexas.

Definição 1.1. *O **subdiferencial** de uma função convexa $f : \mathbb{R}^n \rightarrow \mathbb{R}$ num ponto $\mathbf{x} \in \mathbb{R}^n$ é o conjunto dos vetores $\mathbf{s} \in \mathbb{R}^n$ tais que*

$$\partial_c f(\mathbf{x}) = \left\{ \mathbf{s} \in \mathbb{R}^n \mid f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{s}^T(\mathbf{y} - \mathbf{x}) \text{ para todo } \mathbf{y} \in \mathbb{R}^n \right\}.$$

Teorema 1.2. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa. Então a derivada direcional existe em qualquer direção $\mathbf{d} \in \mathbb{R}^n$ e satisfaz*

$$f'(\mathbf{x}; \mathbf{d}) = \inf_{t>0} \frac{f(\mathbf{x} + t\mathbf{d}) - f(\mathbf{x})}{t}.$$

Apresenta-se algumas relações existentes entre o subdiferencial e a derivada direcional.

Teorema 1.3. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa. Então para todo $\mathbf{x} \in \mathbb{R}^n$*

$$(i) \quad f'(\mathbf{x}; \mathbf{d}) = \max \left\{ \mathbf{s}^T \mathbf{d} \mid \mathbf{s} \in \partial_c f(\mathbf{x}) \right\} \text{ para todo } \mathbf{d} \in \mathbb{R}^n,$$

$$(ii) \quad \partial_c f(\mathbf{x}) = \left\{ \mathbf{s} \in \mathbb{R}^n \mid f'(\mathbf{x}; \mathbf{d}) \geq \mathbf{s}^T \mathbf{d} \text{ para todo } \mathbf{d} \in \mathbb{R}^n \right\},$$

(iii) $\partial_c f(\mathbf{x})$ é um conjunto não vazio, convexo e compacto tal que $\partial_c f(\mathbf{x}) \subset B(0; L)$, onde $L > 0$ é a constante de Lipschitz de f em \mathbf{x} .

O próximo teorema mostra que realmente o subdiferencial é uma generalização da derivada clássica.

Teorema 1.4. *Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa e diferenciável em \mathbf{x} , então*

$$\partial_c f(\mathbf{x}) = \{ \nabla f(\mathbf{x}) \}.$$

Teorema 1.5. *Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa então para todo $\mathbf{y} \in \mathbb{R}^n$*

$$f(\mathbf{y}) = \max \left\{ f(\mathbf{x}) + \mathbf{s}^T(\mathbf{y} - \mathbf{x}) \mid \mathbf{x} \in \mathbb{R}^n, \mathbf{s} \in \partial_c f(\mathbf{x}) \right\}. \quad (1.2)$$

Em Otimização não diferenciável, os chamados métodos de feixe são baseados na teoria do ε -subdiferencial, cuja definição é uma adaptação do conceito de subdiferencial.

Definição 1.6. Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa. A ε -**derivada direcional** de f em $\mathbf{x} \in \mathbb{R}^n$ na direção de $\mathbf{d} \in \mathbb{R}^n$ é definida por

$$f'_\varepsilon(\mathbf{x}; \mathbf{d}) = \inf_{t>0} \frac{f(\mathbf{x} + t\mathbf{d}) - f(\mathbf{x}) + \varepsilon}{t}.$$

Definição 1.7. Seja $\varepsilon > 0$, o ε -**subdiferencial** de uma função convexa $f : \mathbb{R}^n \rightarrow \mathbb{R}$ em $\mathbf{x} \in \mathbb{R}^n$ é o conjunto

$$\partial_\varepsilon f(\mathbf{x}) = \left\{ \mathbf{s} \in \mathbb{R}^n \mid f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{s}^T(\mathbf{y} - \mathbf{x}) - \varepsilon \text{ para todo } \mathbf{y} \in \mathbb{R}^n \right\}.$$

Cada elemento $\mathbf{s} \in \partial_\varepsilon f(\mathbf{x})$ é chamado ε -**subgradiente** de f em \mathbf{x} .

Para funções localmente Lipschitz não necessariamente existe a noção de derivada direcional clássica, por isso, primeiramente define-se uma derivada direcional generalizada (veja Clarke[16]). Dando sequência estende-se o conceito de subdiferencial para funções localmente Lipschitz.

Definição 1.8 (Clarke). Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz em $\mathbf{x} \in \mathbb{R}^n$. A **derivada direcional generalizada** de f em \mathbf{x} na direção $\mathbf{d} \in \mathbb{R}^n$ é definida por

$$f^\circ(\mathbf{x}; \mathbf{d}) = \limsup_{\substack{\mathbf{y} \rightarrow \mathbf{x} \\ t \rightarrow 0}} \frac{f(\mathbf{y} + t\mathbf{d}) - f(\mathbf{y})}{t}.$$

Definição 1.9 (Clarke). Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz em $\mathbf{x} \in \mathbb{R}^n$. O **subdiferencial** de f em \mathbf{x} é o conjunto de vetores $\mathbf{s} \in \mathbb{R}^n$ tal que

$$\partial f(\mathbf{x}) = \left\{ \mathbf{s} \in \mathbb{R}^n \mid f^\circ(\mathbf{x}; \mathbf{d}) \geq \mathbf{s}^T \mathbf{d} \text{ para todo } \mathbf{d} \in \mathbb{R}^n \right\}.$$

Cada vetor $\mathbf{s} \in \partial f(\mathbf{x})$ é chamado de **subgradiente** de f em \mathbf{x} .

Teorema 1.10. Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz em $\mathbf{x} \in \mathbb{R}^n$ com constante L . Então

- (i) $f^\circ(\mathbf{x}; \mathbf{d}) = \max \left\{ \mathbf{s}^T \mathbf{d} \mid \mathbf{s} \in \partial f(\mathbf{x}) \right\}$ para todo $\mathbf{d} \in \mathbb{R}^n$,
- (ii) $\partial f(\mathbf{x})$ é um conjunto não vazio, convexo e compacto tal que $\partial f(\mathbf{x}) \subset B(0; L)$.

O próximo teorema mostra que o subdiferencial para funções localmente Lipschitz é uma generalização do subdiferencial de uma função convexa.

Teorema 1.11. Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa. Então

- (i) $f'(\mathbf{x}; \mathbf{d}) = f^\circ(\mathbf{x}; \mathbf{d})$ para todo $\mathbf{d} \in \mathbb{R}^n$ e
- (ii) $\partial_c f(\mathbf{x}) = \partial f(\mathbf{x})$.

Os dois seguintes teoremas mostram que o subdiferencial realmente é uma generalização da derivada clássica.

Teorema 1.12. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz e diferenciável em $\mathbf{x} \in \mathbb{R}^n$. Então*

$$\nabla f(\mathbf{x}) \in \partial f(\mathbf{x}).$$

Teorema 1.13. *Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é continuamente diferenciável em $\mathbf{x} \in \mathbb{R}^n$. Então*

$$\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}.$$

Teorema 1.14 (Rademacher). *Seja $U \subset \mathbb{R}^n$ um conjunto aberto. Uma função $f : U \rightarrow \mathbb{R}$ que é localmente Lipschitz em U é diferenciável em quase todos os pontos de U (i.e., é diferenciável a menos de um conjunto de medida nula).*

Devido ao Teorema de Rademacher sabe-se que para funções localmente Lipschitz o gradiente existe em quase todos os pontos do domínio da função. De posse desta informação, pode-se reconstruir o subdiferencial como sendo o fecho convexo todos os possíveis limites de gradientes nos pontos $\{\mathbf{x}^k\}$ que convergem a \mathbf{x} .

E usa-se a notação Ω_f para representar o conjunto dos pontos onde f não é diferenciável.

Teorema 1.15. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz em $\mathbf{x} \in \mathbb{R}^n$. Então*

$$\partial f(\mathbf{x}) = \text{conv} \left\{ \mathbf{s} \in \mathbb{R}^n \mid \text{existe } \{\mathbf{x}^k\} \subset \mathbb{R}^n \setminus \Omega_f \text{ tal que } \mathbf{x}^k \rightarrow \mathbf{x} \text{ e } \nabla f(\mathbf{x}^k) \rightarrow \mathbf{s} \right\}. \quad (1.3)$$

Agora define-se o **Goldstein ε -subdiferencial** de uma função localmente Lipschitz, de modo análogo ao que fizemos para funções convexas.

Definição 1.16. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz em $\mathbf{x} \in \mathbb{R}^n$ e seja $\varepsilon > 0$. Então o **Goldstein ε -subdiferencial** de f é o conjunto*

$$\partial_\varepsilon^G f(\mathbf{x}) = \text{conv} \{ \partial f(\mathbf{y}) \mid \mathbf{y} \in B(\mathbf{x}; \varepsilon) \}.$$

Cada elemento $\mathbf{s} \in \partial_\varepsilon^G f(\mathbf{x})$ é chamado **ε -subgradiente** de f em \mathbf{x} .

1.3 Condições de Otimalidade

Nesta seção generaliza-se as condições de otimalidade de primeira ordem clássicas para os casos de otimização restrita e irrestrita. E mostra-se as condições necessárias para uma função localmente Lipschitz atingir um mínimo local.

Definição 1.17. Um ponto $\mathbf{x} \in \mathbb{R}^n$ é um **mínimo local** de f , se existir $\varepsilon > 0$ tal que $f(\mathbf{x}) < f(\mathbf{y})$ para todo $\mathbf{y} \in B(\mathbf{x}, \varepsilon)$.

Definição 1.18. Um ponto $\mathbf{x} \in \mathbb{R}^n$ é um **mínimo global** de f , se satisfaz $f(\mathbf{x}) < f(\mathbf{y})$ para todo $\mathbf{y} \in \mathbb{R}^n$.

Inicia-se com as condições necessárias básicas para o problema de otimização irrestrito. Nota-se que para funções convexas as condições também são suficientes.

Teorema 1.19. Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é Localmente Lipschitz em \mathbf{x} e atinge um mínimo local em \mathbf{x} , então

- (i) $0 \in \partial f(\mathbf{x})$,
- (ii) $f^\circ(\mathbf{x}; \mathbf{d}) \geq 0$ para todo $\mathbf{d} \in \mathbb{R}^n$.

Um ponto $\mathbf{x} \in \mathbb{R}^n$ satisfazendo $0 \in \partial f(\mathbf{x})$ é chamado um **ponto crítico** ou **estacionário** de f .

Teorema 1.20. Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa, então as seguintes condições são equivalentes:

- (i) f atinge um mínimo global em \mathbf{x} ,
- (ii) $0 \in \partial_c f(\mathbf{x})$,
- (iii) $f'(\mathbf{x}; \mathbf{d}) \geq 0$ para todo $\mathbf{d} \in \mathbb{R}^n$.

Teorema 1.21. Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é Localmente Lipschitz em \mathbf{x} e atinge um mínimo local em \mathbf{x} , então

$$0 \in \partial_\varepsilon^G f(\mathbf{x}).$$

Teorema 1.22. Se $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa, então as seguintes condições são equivalentes:

- (i) $0 \in \partial_\varepsilon^G f(\mathbf{x})$,
- (ii) \mathbf{x} minimiza f com $\varepsilon > 0$, i.e., $f(\mathbf{x}) \leq f(\mathbf{y}) + \varepsilon$ para todo $\mathbf{y} \in \mathbb{R}^n$.

Agora, mostra-se as condições necessárias correspondentes para o caso de problemas com restrições. Uma dificuldade existente em métodos de otimização iterativos é a de encontrar uma direção tal que os valores da função objetivo vão sempre decrescendo quando nos movimentamos naquela direção.

Definição 1.23. A direção $\mathbf{d} \in \mathbb{R}^n$ é uma **direção de descida** para uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ num ponto $\mathbf{x} \in \mathbb{R}^n$, se existe $t_\varepsilon > 0$ tal que para todo $t \in (0, t_\varepsilon]$,

$$f(\mathbf{x} + t\mathbf{d}) < f(\mathbf{x}).$$

Lema 1.24. Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função localmente Lipschitz contínua em $\mathbf{x} \in \mathbb{R}^n$. A direção $\mathbf{d} \in \mathbb{R}^n$ é uma direção de descida para f em \mathbf{x} se $\mathbf{s}^T \mathbf{d} < 0$ para todo $\mathbf{s} \in \partial f(\mathbf{x})$.

Contudo, em otimização com restrições não é suficiente encontrar qualquer direção de descida, pois não se pode violar as restrições. Logo, precisa-se definir a noção de direção viável. Considere o seguinte problema com restrições de desigualdade

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \mathbf{x} \in \mathbb{R}^n \\ \text{sujeito a } c(\mathbf{x}) \leq 0 \end{cases} \quad (1.4)$$

onde $f, c : \mathbb{R}^n \rightarrow \mathbb{R}$.

Definição 1.25. A direção $\mathbf{d} \in \mathbb{R}^n$ é uma **direção viável** para um problema do tipo (1.4) se existe $t_\varepsilon > 0$ tal que para todo $t \in (0, t_\varepsilon]$

$$\mathbf{x} + t\mathbf{d} \in \Omega$$

onde $\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid c(\mathbf{x}) \leq 0\}$ é a região viável.

Lema 1.26. Seja $c : \mathbb{R}^n \rightarrow \mathbb{R}$, uma função localmente Lipschitz contínua em $\mathbf{x} \in \Omega$. A direção $\mathbf{d} \in \mathbb{R}^n$ é uma direção viável em \mathbf{x} para o problema (1.4) se $\mathbf{r}^T \mathbf{d} < 0$ para todo $\mathbf{r} \in \partial c(\mathbf{x})$ tal que $c(\mathbf{x}) = 0$.

Teorema 1.27 (Condição de otimalidade para minimizar uma função convexa num conjunto convexo). Sejam $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa e $C \subset \mathbb{R}^n$ um conjunto convexo. Então $\bar{\mathbf{x}} \in \mathbb{R}^n$ é um minimizador de f em C se, e somente se,

$$\exists \mathbf{s} \in \partial f(\bar{\mathbf{x}}) \text{ tal que } \mathbf{s}^T (\mathbf{x} - \bar{\mathbf{x}}) \geq 0 \quad \forall \mathbf{x} \in C \quad (1.5)$$

ou equivalentemente

$$0 \in \partial f(\bar{\mathbf{x}}) + N_C(\bar{\mathbf{x}}), \quad (1.6)$$

onde $N_C(\bar{\mathbf{x}})$ é o cone de direções normais.

Demonstração. Veja em Solodov [32], página 168. □

De fato, na prática, o conjunto C definido pela restrição tem uma forma especial. O Corolário generaliza as condições de otimalidade necessárias Karush-Kuhn-Tucker (KKT)

Corolário 1.28. *Seja $c : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função tal que $c(\mathbf{x}) = \max\{c_i(\mathbf{x}) \mid i = 1, \dots, m\}$ onde cada $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$ é convexa e suponha que $c(\mathbf{z}) < 0$ para algum $\mathbf{z} \in \mathbb{R}^n$. Seja*

$$C := \{\mathbf{x} \in \mathbb{R}^n \mid c(\mathbf{x}) \leq 0\}.$$

Suponha que $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é convexa. Então as seguintes condições são equivalentes:

- (i) f atinge seu mínimo global sobre C em $\bar{\mathbf{x}}$,*
- (ii) Existe $\mu_i \geq 0$ para $i = 1, \dots, m$ tal que $\mu_i c_i(\bar{\mathbf{x}}) = 0$ e*

$$0 \in \partial f(\bar{\mathbf{x}}) + \sum_{i=1}^m \mu_i \partial c_i(\bar{\mathbf{x}}). \quad (1.7)$$

Demonstração. Veja [13], página 74. □

Um ponto \mathbf{x} é chamado um **ponto KKT** associado ao problema (1.4) se é viável e satisfaz a condição de otimalidade KKT do Teorema 1.28.

1.4 Otimização Convexa Não Diferenciável

Inicia-se com uma pequena introdução aos métodos clássicos para otimização convexa não diferenciável. A não diferenciabilidade produz uma série de dificuldades adicionais: A primeira aparece na determinação da direção de busca, pois, nem sempre, a direção obtida é de descida e conseqüentemente, a busca linear não faz sentido. Outra, é a dificuldade de estabelecer critérios de parada implementáveis.

Os métodos abordados nesta seção se concentram na resolução do seguinte problema irrestrito

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \text{sujeito a } \mathbf{x} \in \mathbb{R}^n \end{cases} \quad (1.8)$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa não necessariamente diferenciável.

De modo geral, os algoritmos que serão vistos estão baseados na geração de iterados \mathbf{x}^k através da busca de possíveis direções de descida \mathbf{d}^k , tamanhos de passo t^k e conseqüente atualização $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k \mathbf{d}^k$. Nota-se ainda que estes métodos diferem principalmente nas estratégias que conduzem a determinação das direções de descida.

1.4.1 Métodos de Descida

Os métodos de descida se baseiam na geração de uma sequência $\{\mathbf{x}^k\}$ com a garantia de decréscimo da função objetivo f em cada iteração. As direções a serem

tomadas são, portanto, as de descida, caracterizadas na Secção 1.3. O Algoritmo é descrito a seguir:

Algoritmo 1.1. *Método de Descida*

1. Tome $\mathbf{x}^1 \in \mathbb{R}^n$ e seja $k = 1$.
2. (critério de parada formal) Se $0 \in \partial f(\mathbf{x}^k)$, pare.
3. (Descida) Encontre uma direção de descida \mathbf{d}^k de f em \mathbf{x}^k .
4. (Busca linear) Encontre um tamanho de passo $t^k > 0$ tal que

$$f(\mathbf{x}^k + t^k \mathbf{d}^k) < f(\mathbf{x}^k).$$

5. (Próximo iterado) Defina $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k \mathbf{d}^k$.
6. (Loop) Tome $k = k + 1$ e vá para o passo 2.

Observa-se que o critério de parada oferecido pelo passo 2 é puramente formal, pois a obtenção de todo o subdiferencial é algo excessivo. Critérios de parada implementáveis serão vistos posteriormente com detalhes nos algoritmos de planos de corte e métodos de feixe. Por ora, nota-se apenas que num ponto \mathbf{x} dado, a melhor descida \mathbf{d}^k possível (i.e., à direção de máxima descida) é a solução do problema

$$\min_{\|\mathbf{d}\|=1} f'(\mathbf{x}^k; \mathbf{d}) \quad \text{ou} \quad \min_{\|\mathbf{d}\|=1} \max_{\mathbf{s} \in \partial f(\mathbf{x})} \langle \mathbf{s}; \mathbf{d} \rangle.$$

Geometricamente, isso significa que a direção de máxima descida é justamente a que está relacionada com ao hiperplano H ortogonal a projeção de $\{0\}$ em $\partial f(\mathbf{x}^k)$. Mais precisamente, $\mathbf{d}^k = -\gamma^k / \|\gamma^k\|$, onde $\gamma^k = P_{\partial f(\mathbf{x}^k)}(0)$.

Contudo há um problema que norteia o método de máxima descida, é o fato da sequência de iterados $\{\mathbf{x}^k\}$ poder oscilar e convergir para um ponto não ótimo. Referindo-se a [31](seção VII.2.2) para verificar, através de um exemplo numérico, que o método pode, de fato não convergir. Para uma melhor compreensão, lembra-se que o algoritmo de descida converge se $(f(\mathbf{x}^k))$ é decrescente e se $\{\mathbf{x}^k\}$ possui um ponto de acumulação \mathbf{x}^* , que é minimizador de f . Considera-se, então, a seguinte sequência:

$$\delta_k := \left\{ \text{dist}(0; \partial f(\mathbf{x}^k)) \right\}.$$

O subdiferencial $\partial f(\mathbf{x})$ visto como uma multi-função, é fechado, i.e., possui o gráfico fechado:

$$\left\{ \begin{array}{l} \mathbf{x}^k \rightarrow \mathbf{x}^* \\ \mathbf{s}^k \in \partial f(\mathbf{x}^k) \rightarrow \mathbf{s}^* \end{array} \right\} \implies \mathbf{s}^* \in \partial f(\mathbf{x}^*). \quad (1.9)$$

Dessa forma, se $\delta_k \rightarrow 0$, então $0 \in \partial f(\mathbf{x}^*)$ e \mathbf{x}^* é ponto de mínimo. Para assegurar que $\delta_k \rightarrow 0$, a multi-função $\mathbf{x} \mapsto \partial f(\mathbf{x})$ deveria ser contínua (i.e., semi-continua interior e exterior).

A semi-continuidade exterior está garantida pela propriedade (1.9). Contudo, o subdiferencial não possui a propriedade da semi-continuidade interior, ou seja, não é verdade que

$$\begin{cases} \mathbf{x}^k \rightarrow \mathbf{x}^* \\ \mathbf{s}^* \in \partial f(\mathbf{x}^*) \end{cases} \implies \text{existe } \mathbf{s}^k \rightarrow \mathbf{s}^* \text{ tal que } \mathbf{s}^* \in \partial f(\mathbf{x}^k). \quad (1.10)$$

para mais detalhes veja [15] seção 8.2.2.

Daí a importância do ε -subdiferencial, pois, além de aproximar o subdiferencial $\partial f(\mathbf{x})$, já que $\partial f(\mathbf{x}) \subset \partial_\varepsilon f(\mathbf{x})$, visto como uma multi-função, é semi-contínuo interior e exterior: A semi-continuidade exterior está garantida, pois seu gráfico é fechado:

$$\begin{cases} \mathbf{x}^k \rightarrow \mathbf{x}^* \\ \varepsilon_k \rightarrow \varepsilon_* \\ \mathbf{s}^k \in \partial f(\mathbf{x}^k) \rightarrow \mathbf{s}^* \end{cases} \implies \mathbf{s}^* \in \partial_{\varepsilon_*} f(\mathbf{x}^*). \quad (1.11)$$

Como f é localmente Lipschitz contínua, tem-se para $\varepsilon > 0$ fixado que

$$\forall r > 0, \exists \delta > 0 : \|\mathbf{x}^k - \mathbf{x}^*\| \leq \delta \implies \partial_\varepsilon f(\mathbf{x}^*) \subset \partial_\varepsilon f(\mathbf{x}^k) + B(0, r),$$

o que garante que $(\varepsilon, \mathbf{x}) \mapsto \partial_\varepsilon f(\mathbf{x})$ é semi-contínua interior em \mathbf{x}^* .

Portanto, uma possível maneira de contornar a não convergência do método de descida é utilizar $\partial_\varepsilon f(\cdot)$, com $\varepsilon > 0$, no lugar de $\partial f(\mathbf{x})$. Tais algoritmos são denominados de ε -descida [18]. Desta forma, estes algoritmos geram sequências (\mathbf{x}^k) tais que $(f(\mathbf{x}^k))$ é decrescente e $\{\text{dist}(0; \partial_{\varepsilon_k} f(\mathbf{x}^k))\} \rightarrow 0$ com $\varepsilon_k \rightarrow 0$.

Veja o algoritmo:

Algoritmo 1.2. *Método de ε -Descida*

1. Tome $\mathbf{x}^1 \in \mathbb{R}^n$, $\varepsilon > 0$ e seja $k = 1$.
2. (critério de parada formal) Se $0 \in \partial_\varepsilon f(\mathbf{x}^k)$, pare.
3. (Descida) Encontre uma direção de ε -descida \mathbf{d}^k de f em \mathbf{x}^k .
4. (Busca linear) Encontre um tamanho de passo $t^k > 0$ tal que

$$f(\mathbf{x}^k + t^k \mathbf{d}^k) < f(\mathbf{x}^k) - \varepsilon.$$

5. (Próximo iterado) Defina $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k \mathbf{d}^k$.
6. (Loop) Tome $k = k + 1$ e vá para o passo 2.

Neste algoritmo, $f(\mathbf{x}^k) \rightarrow -\infty$ ou o algoritmo termina numa iteração k_* tal que \mathbf{x}^{k_*} é ε -ótimo. Esse é o algoritmo de ε -descida mais simples, existem variantes do método que permitem escolhas de $\varepsilon = \varepsilon_k$ a cada iteração.

Nota-se que o ε -subdiferencial também não é usualmente conhecido por inteiro e portanto o algoritmo ainda não é implementável. Nas próximas seções, serão vistos outros algoritmos que tentam contornar essa questão.

Concluí-se esta seção mostrando, através de um exemplo, que a propriedade de semi-continuidade interior de $\partial_\varepsilon f(\mathbf{x})$ é válida apenas para $\varepsilon > 0$. Considera-se a função modulo dada por $f(\mathbf{x}) = |\mathbf{x}|$.

Na figura abaixo o gráfico da esquerda ilustra a multi-função $\partial f(\mathbf{x})$, enquanto que a da direita representa $\partial_\varepsilon f(\mathbf{x})$ para $\varepsilon > 0$ fixo. Nota-se por exemplo, que o conjunto $\partial f(0) = [-1, 1]$ é muito "maior" do que $\partial f(\mathbf{x}) = \{1\}$, quando $\mathbf{x} > 0$. Por outro lado, $\partial_\varepsilon f(\mathbf{x})$ não explode quando \mathbf{x} se aproxima do ponto 0.

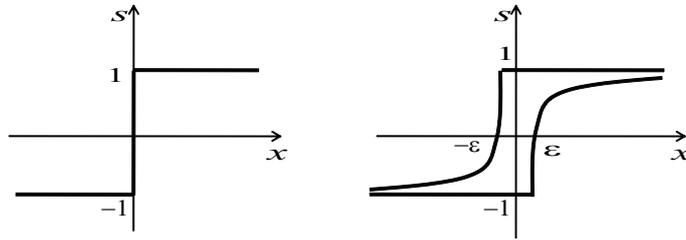


Figura 1.1: Semi-continuidade interior do ε -subdiferencial

1.4.2 Método de Subgradientes

Como observado na seção anterior, a determinação de todo o subgradiente é algo excessivo ou impossível computacionalmente falando, e uma forma de contornar esse problema é exigir menos, ou seja, pedir o cálculo de apenas um único subgradiente. Tal exigência está relacionada ao que chama-se de *caixa preta*, que é utilizado como base em diversos algoritmos para problemas não diferenciáveis, onde dado $\mathbf{x}^k \in \mathbb{R}^n$, a caixa preta é responsável por gerar $f(\mathbf{x}^k)$ e um subgradiente $\mathbf{s}^k \in \partial f(\mathbf{x}^k)$.

A ideia do método de subgradientes provém do método de Cauchy, lembre-se portanto deste método diferenciável: De posse das devidas hipóteses, o método de Cauchy encontra uma direção \mathbf{d}^k tal que $f(\mathbf{x}^k + \mathbf{d}^k) < f(\mathbf{x}^k)$. A intenção é escrever o problema (1.8) na forma

$$\begin{cases} \text{minimize} & f(\mathbf{x}^k + \mathbf{d}) - f(\mathbf{x}^k) \\ \text{sujeito a} & \mathbf{d} \in \mathbb{R}^n \end{cases}$$

que tem uma solução $\mathbf{d}^k \neq 0$. Devido a expansão de primeira ordem de Taylor

$$f(\mathbf{x}^k + \mathbf{d}) - f(\mathbf{x}^k) = f'(\mathbf{x}^k; \mathbf{d}) + \|\mathbf{d}\| \varepsilon(\mathbf{d}),$$

onde $\varepsilon(\mathbf{d}) \rightarrow 0$ quando $\|\mathbf{d}\| \rightarrow 0$ e a identidade $f'(\mathbf{x}^k; \mathbf{d}) = \nabla f(\mathbf{x}^k)^T \mathbf{d}$, obtém-se as seguintes versões equivalentes

$$\begin{cases} \text{minimize } f'(\mathbf{x}^k; \mathbf{d}) \\ \text{sujeito a } \|\mathbf{d}\| \leq 1 \end{cases} \Leftrightarrow \begin{cases} \text{minimize } \nabla f(\mathbf{x}^k)^T \mathbf{d} \\ \text{sujeito a } \|\mathbf{d}\| \leq 1. \end{cases}$$

onde a restrição adicional $\|\mathbf{d}\| \leq 1$ é necessária, pois a função $f'(\mathbf{x}^k; \cdot)$ é positivamente homogênea. Daí considera-se a direção $\mathbf{d}^k = -\nabla f(\mathbf{x}^k) / \|\nabla f(\mathbf{x}^k)\|$.

Com isso, no caso não diferenciável, a ideia do método de subgradientes é considerar o vetor oposto ao subgradiente fornecido pela caixa preta, i.e.,

$$\mathbf{d}^k = -\mathbf{s}^k / \|\mathbf{s}^k\|.$$

Porém, tal direção não é necessariamente de descida, conforme pode ser visto na Figura 1.2 abaixo, que mostra curvas de nível para funções minimizadas em zero, direções de descida devem fazer produto escalar negativo com o subdiferencial $\partial f(\mathbf{x})$ inteiro e não apenas com um único subgradiente \mathbf{s} .

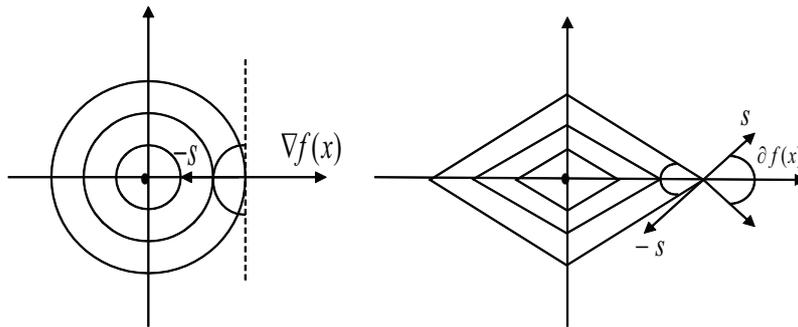


Figura 1.2: Direções de descida: caso diferenciável e não diferenciável

A Figura da direita (caso não diferenciável), mostra um exemplo em que a direção \mathbf{s} , fornecida pela caixa preta, é um vetor extremo do cone associado a $\partial f(\mathbf{x})$ e em tal exemplo, a direção oposta $-\mathbf{s}$ claramente não é descida.

Apesar deste algoritmo não assegurar necessariamente o decréscimo da função objetivo a cada iteração, escolhas adequadas dos tamanhos de passo podem garantir a convergência ao método. Contudo, a fixação do comprimento de passo para uma iteração k antes que o ponto \mathbf{x}^k seja calculado, dificilmente pode dar uma boa escolha. Diante o exposto, esta maneira de remover a dificuldade associada ao decréscimo da função objetivo é puramente formal visto que se trata de uma medida paliativa, apenas para resolver a análise de convergência teórica.

Veja o algoritmo:

Algoritmo 1.3. *Método de Subgradientes*

1. Tome $\mathbf{x}^1 \in \mathbb{R}^n$ e seja $k = 1$.
2. (caixa preta) Calcule \mathbf{s}^k e $f(\mathbf{x}^k)$.
3. (critério de parada) Se $0 \in \partial f(\mathbf{x}^k)$, pare.
4. (Busca linear) Encontre um tamanho de passo $t^k > 0$ adequado.
5. (Próximo iterado) Defina $\mathbf{x}^{k+1} = \mathbf{x}^k - t^k \frac{\mathbf{s}^k}{\|\mathbf{s}^k\|}$.
6. (Loop) Tome $k = k + 1$ e vá para o passo 2.

Para compreender o Passo 4 do Algoritmo 1.3, define-se primeiro \mathbf{x}^* como o conjunto de soluções ótimas. Da definição de subdiferencial, se $\mathbf{s}^k \in \partial f(\mathbf{x}^k)$ então

$$f(\mathbf{x}) - f(\mathbf{x}^k) \geq (\mathbf{s}^k)^T (\mathbf{x} - \mathbf{x}^k)$$

o que implica em $(-\mathbf{s}^k)^T (\mathbf{x}^* - \mathbf{x}^k) \geq f(\mathbf{x}^k) - f(\mathbf{x}^*) \geq 0 \quad \forall \mathbf{x} \in \mathbb{R}^n$.

Desta observação, conclui-se que o ângulo entre $-\mathbf{s}^k$ e $\mathbf{x}^* - \mathbf{x}^k$ é agudo e, portanto, para $t^k > 0$ suficiente pequeno, tem-se \mathbf{x}^{k+1} mais próximo de \mathbf{x}^* do que \mathbf{x}^k . E isso motiva a escolha de uma sequência $\{t^k\}$ satisfazendo $\lim_{k \rightarrow \infty} t^k = 0$.

Mais precisamente vale o seguinte resultado:

Lema 1.29. *Seja \mathbf{x}^* uma solução de $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$. Então $\|\mathbf{x}^{k+1} - \mathbf{x}^*\| < \|\mathbf{x}^k - \mathbf{x}^*\|$ sempre que $0 < t^k < 2 \frac{[f(\mathbf{x}^k) - f(\mathbf{x}^*)]}{\|\mathbf{s}^k\|}$.*

No entanto ressaltasse que essa estratégia não garante a convergência do método. Outra observação deve ser feita na escolha de t^k : Como $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| = t^k$ segue que se \mathbf{x}^1 é o ponto inicial, então $\|\mathbf{x}^{k+1} - \mathbf{x}^1\| \leq \sum_{j=1}^k t_j$.

Logo, para todo, $k \in \mathbb{N}$, $\mathbf{x}^k \in B(\mathbf{x}^1, r)$ onde $r = \sum_{k=1}^{\infty} t_k$.

Assim, o método encontrará a solução \mathbf{x}^* somente se $\mathbf{x}^k \in B(\mathbf{x}^1, r)$. Para contornar este problema, impõe-se $\sum_{k=1}^{\infty} t_k = +\infty$.

Teorema 1.30. *Suponha que o conjunto \mathbf{x}^* das soluções do problema $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$, seja não vazio e limitado. Então para qualquer ponto inicial \mathbf{x}^1 , a sequência (\mathbf{x}^k) definida por $\mathbf{x}^k = \mathbf{x}^k + t^k \frac{\mathbf{s}^k}{\|\mathbf{s}^k\|}$ com t^k satisfazendo $\lim_{k \rightarrow \infty} t^k = 0$ e $\sum_{k=1}^{\infty} t_k = +\infty$ é limitada e todos os seus pontos de acumulação pertencem a \mathbf{x}^* .*

Outra deficiência consiste no fato dos métodos de subgradiente não possuírem nenhuma regra de parada confiável. Por exemplo, um critério de parada que utiliza $\|\nabla f(\mathbf{x}^k)\| < \varepsilon$ para algum $\varepsilon > 0$ fixado, não podem ser utilizados em métodos não diferenciáveis pois esta situação pode nunca acontecer. Para ver isto basta considerar $f : \mathbb{R} \rightarrow \mathbb{R}$ definida por $f(\mathbf{x}) = |\mathbf{x}|$. Tem-se $\|\nabla f(\mathbf{x}^k)\| = 1, \quad \forall \mathbf{x}^k \neq 0$, não importando o quanto \mathbf{x}^k esteja próximo da solução $\mathbf{x}^* = 0$.

A única atração de métodos de subgradiente é sua simplicidade (quando o cálculo de subgradientes é fácil). Nem sequer requerem uma busca linear. Um estudo profundo sobre os métodos de subgradiente pode ser encontrado em [15], [14] ou [18].

1.4.3 Método de Planos de Corte

Ao contrário do método de subgradientes, a ideia do método de planos de corte é aproveitar as informações obtidas nas iterações anteriores para definir um modelo da função objetivo. Esse modelo será útil para obter candidatos a direções de descida, logo, a cada iteração k , tem-se um modelo linear por partes \check{f}_k de f construído da seguinte forma:

$$\check{f}_k(\mathbf{x}) = \max_{i=1,\dots,k} \{f(\mathbf{x}^i) + (\mathbf{s}^i)^T(\mathbf{x} - \mathbf{x}^i)\}. \quad (1.12)$$

Nota-se que a igualdade acima se deve ao teorema 1.5 e que a cada iteração k , adiciona-se ao modelo uma função afim $f(\mathbf{x}^k) + (\mathbf{s}^k)^T(\mathbf{x} - \mathbf{x}^k)$.

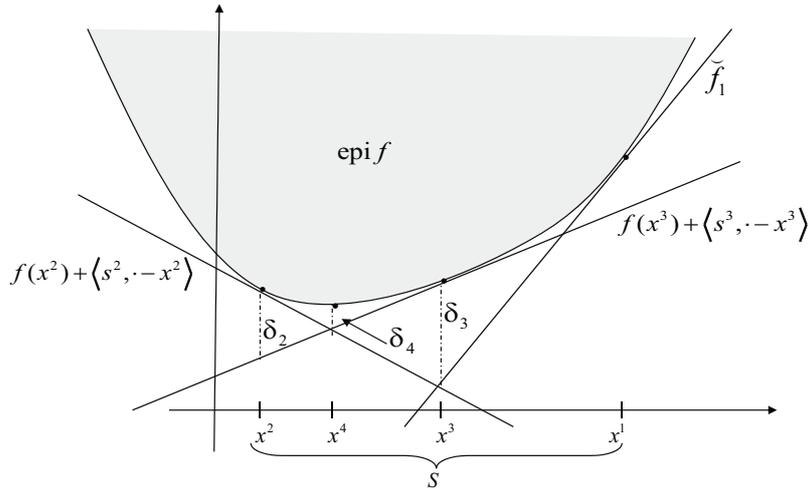


Figura 1.3: Iterações do método de planos de corte

O máximo de todas as funções que definem \check{f}_k é claramente uma função convexa e linear por partes. Além disso,

$$\check{f}_k \leq \check{f}_{k+1} \quad \text{e} \quad \check{f}_k \leq f$$

para todo k , ou seja, \check{f}_k se aproxima de f por baixo a cada iteração.

Feito isso, pode-se utilizar o modelo \check{f}_k para encontrar o próximo iterado \mathbf{x}^{k+1} , como solução do seguinte problema

$$\mathbf{x}^{k+1} \in \arg \min_{\mathbf{x} \in S} \check{f}_k(\mathbf{x}), \quad (1.13)$$

onde S é um conjunto compacto, convexo que contém um ponto mínimo de f . Ademais, o modelo ainda nos permite ter um critério de parada implementável através do cálculo do *decréscimo nominal*, definido por

$$\delta_k := f(\mathbf{x}^k) - \check{f}_{k-1}(\mathbf{x}^k). \quad (1.14)$$

Observa-se que o algoritmo termina quando δ_k é pequeno.

Vejam agora como fica o algoritmo.

Algoritmo 1.4. *Método de Planos de Corte*

1. *Sejam $tol \geq 0$ uma tolerância dada e $S \subset \mathbb{R}^n$.*
2. *Tome $\mathbf{x}^1 \in S$ e seja $k = 1$. Defina $\check{f}_0 \equiv -\infty$.*
3. *(caixa preta) Calcule \mathbf{s}^k e $f(\mathbf{x}^k)$.*
4. *(Decréscimo nominal) Calcule $\delta_k := f(\mathbf{x}^k) - \check{f}_{k-1}(\mathbf{x}^k)$.*
5. *(Critério de parada) Se $\delta_k \leq tol$, pare.*
6. *(Próximo iterado) Defina $\mathbf{x}^{k+1} \in \arg \min_{\mathbf{x} \in S} \check{f}_k(\mathbf{x})$.*
7. *(Loop) Tome $k = k + 1$ e vá para o passo 3.*

Assim como no método de subgradientes, o método de planos de corte não garante o decréscimo da função objetivo a cada iteração. Tal fato pode ser observado quando introduz-se uma função afim “quase horizontal” ao modelo \check{f}_k , por esse motivo diz-se que o algoritmo não está livre de instabilidades. Veja a Figura 1.4 abaixo.

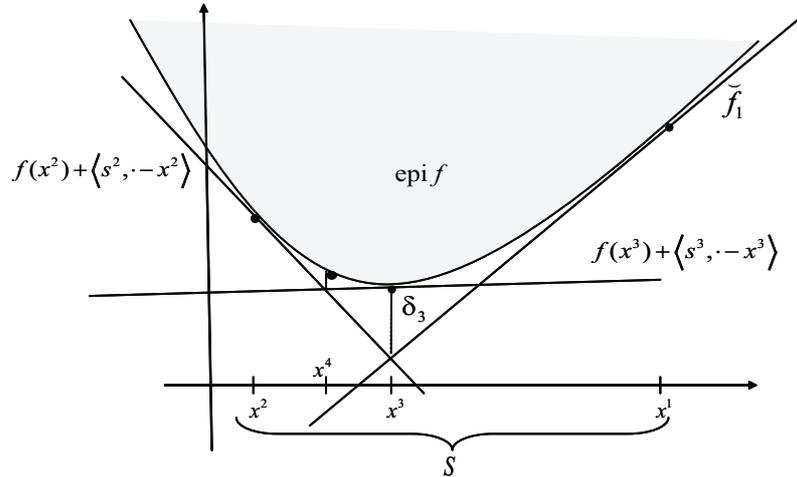


Figura 1.4: Introdução de uma função afim quase “horizontal”

Observe que a introdução da função afim $f(\mathbf{x}^3) + (\mathbf{s}^3)^T(\cdot - \mathbf{x}^3)$ gera um ponto \mathbf{x}^4 tal que $f(\mathbf{x}^4) > f(\mathbf{x}^3)$.

Verifica-se ainda que o modelo acumula um número crescente de funções afins que definem o modelo, o que dificulta a resolução dos problemas do passo 6, mesmo

se essas forem lineares. Além disso, em alguns momentos várias dessas restrições ficam quase idênticas umas as outras.

1.4.4 Método de Feixe

O método de Feixe reconcilia as características dos métodos de descida e planos de corte, garantindo ao mesmo tempo o decréscimo da função objetivo e a estabilização. O modelo é construído de modo análogo ao utilizado no método de planos de corte. Porém, tenta-se evitar o acúmulo grande de funções afins que causam mau condicionamento. E isso é feito adotando-se duas sequências de pontos distintas:

Uma sequência formada pelos chamados *centros de estabilização* $\{\mathbf{x}^k\} \subset \mathbb{R}^n$ que decrescem de fato a função objetivo. E supõe-se que, em adição para um ponto de iteração corrente \mathbf{x}^k , tem-se alguns *pontos testes* $\mathbf{y}_j \in \mathbb{R}^n$ (das últimas iterações) e subgradientes $\mathbf{s}_j \in \partial f(\mathbf{y}_j)$ para $j \in J_k$, onde o conjunto de índices J_k é um subconjunto não-vazio de $\{1, \dots, k\}$. Assim como nos planos de corte, a função objetivo é aproximada por um modelo de planos de corte

$$\check{f}_k(\mathbf{x}) = \max_{j \in J_k} \left\{ f(\mathbf{y}_j) + \mathbf{s}_j^T (\mathbf{x} - \mathbf{y}_j) \right\}. \quad (1.15)$$

O próximo iterado candidato é então definido por

$$\mathbf{y}_{k+1} := \mathbf{x}^k + \mathbf{d}^k,$$

onde a direção de busca \mathbf{d}^k é calculada por

$$\mathbf{d}^k := \arg \min_{\mathbf{d} \in \mathbb{R}^n} \left\{ \check{f}_k(\mathbf{x}^k + \mathbf{d}) + \frac{1}{2} \mathbf{d}^T M_k \mathbf{d} \right\}. \quad (1.16)$$

O papel do termo de estabilização $\frac{1}{2} \mathbf{d}^T M_k \mathbf{d}$ é para garantir a existência da solução \mathbf{d}_k e manter a aproximação local suficiente. Como instabilidades ocorrem quando o movimento a partir de \mathbf{x}^k é muito grande, a matriz M_k simétrica regular $n \times n$ destina-se a acumular informação a respeito da curvatura de f numa bola ao redor de \mathbf{x}^k .

Definição 1.31. *Um iterado (ponto candidato) \mathbf{y}_{k+1} torna-se um centro de estabilização (i.e., $\mathbf{x}^{k+1} := \mathbf{y}_{k+1}$) somente se uma condição do tipo Armijo é satisfeita, ou seja,*

$$f(\mathbf{y}_{k+1}) \leq f(\mathbf{x}^k) - m\delta_k$$

onde δ_k é o decréscimo nominal calculado na iteração k e $m \in (0, 1)$. Nesse caso, o passo realizado é chamado **passo sério**. Caso contrário, é tomado $\mathbf{x}^{k+1} := \mathbf{x}^k$ e o passo é denominado **passo nulo**.

O respectivo decréscimo nominal será $\delta_k = \check{f}_k(\mathbf{y}_{k+1}) - f(\mathbf{x}^k)$. Em ambos os casos, na Definição 1.31, um plano de corte é acrescentado ao modelo \check{f}_{k+1} , pois define-se o conjunto $J_{k+1} := J_k \cup \{k+1\}$.

Com as definições acima é possível descrever o método.

Algoritmo 1.5. Método de Feixe

1. Sejam $tol \geq 0$ uma tolerância dada e $m \in (0, 1)$.
2. Tome \mathbf{x}^1 e sejam $k = 1$ e $\delta_k := \infty$.
3. (caixa preta) Calcule \mathbf{s}^1 e $f(\mathbf{x}^1)$.
4. (Construção do modelo) Construa o modelo \check{f}_1 .
5. (Critério de parada) Se $\delta_k \leq tol$, pare.
6. (Ponto candidato) Calcule $\mathbf{y}_{k+1} := \mathbf{x}^k + \mathbf{d}$.
7. (Decréscimo nominal) Defina $\delta_k := \check{f}_k(\mathbf{y}_{k+1}) - f(\mathbf{x}^k)$.
8. (Teste de Descida) Se $f(\mathbf{y}_{k+1}) \leq f(\mathbf{x}^k) - m\delta_k$:
 Faça $\mathbf{x}^{k+1} := \mathbf{y}_{k+1}$, (passo sério).
 Caso contrário faça $\mathbf{x}^{k+1} := \mathbf{x}^k$ (passo nulo).
9. (Atualização do modelo) Construa \check{f}_{k+1} adicionando \mathbf{s}_{k+1} ao modelo.
10. (Loop) Tome $k = k + 1$ e vá para o passo 5.

Nota-se que o problema estabilizado (1.16) é de programação quadrática convexa, usualmente resolvido através do seu dual, que possui um conjunto viável com uma estrutura mais simples. Para mostrar esse problema dual, convém antes considerar o modelo \check{f}_k referindo-o ao centro de estabilização. Para isso, considere as seguintes definições:

Definição 1.32. Dada uma iteração k do método, o **feixe** é o conjunto das informações obtidas, i.e.,

$$\{(\mathbf{y}_j, f(\mathbf{y}_j), \mathbf{s}_j) : \mathbf{s}_j \in \partial f(\mathbf{y}_j), j \in J_k\}.$$

Definição 1.33. Considere uma iteração k do método. Para cada índice j do feixe, o **erro de linearização** é dado por $e_j^k := f(\mathbf{x}^k) - f(\mathbf{y}_j) - \mathbf{s}_j^T(\mathbf{x}^k - \mathbf{y}_j)$.

Feito isso, pode-se substituir a caracterização do modelo \check{f}_k dado em (1.15) pelo seguinte modelo:

$$\check{f}_k(\mathbf{x}) = f(\mathbf{x}^k) + \max_{i \in J_k} \{-e_i^k + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^k)\}. \quad (1.17)$$

Apesar do problema (1.16) ser um problema de otimização não diferenciável, devido a sua natureza linear por partes, é possível reescreve-lo como um

(diferenciável) subproblema de programação quadrática

$$\begin{cases} \text{minimize } \nu + \frac{1}{2} \mathbf{d}^T M_k \mathbf{d} \\ \text{sujeito a } e_j^k + \mathbf{s}_j^T \mathbf{d} \leq \nu \quad \text{para todo } j \in J_k \end{cases} \quad (\text{PQ})$$

Dualizando este problema chega-se a um problema equivalente quando se determina múltiplos λ_j^k para $j \in J_k$ resolvendo o problema

$$\begin{cases} \text{minimize } \frac{1}{2} \left[\sum_{j \in J_k} \lambda_j^k \mathbf{s}_j \right]^T M_k^{-1} \left[\sum_{j \in J_k} \lambda_j^k \mathbf{s}_j \right] + \sum_{j \in J_k} \lambda_j e_j^k \\ \text{sujeito a } \sum_{j \in J_k} \lambda_j = 1, \quad \lambda_j \geq 0 \end{cases} \quad (\text{PD})$$

Os problemas (PQ) e (PD) são equivalentes e tem soluções únicas, para mais detalhes veja [13]. Do ponto de vista computacional, a escolha do conjunto de índices é um ponto crucial, pois se os métodos armazenam todos os subgradientes anteriores, ou seja, se $J_k = \{1, \dots, k\}$, isso poderia causar grandes dificuldades como por exemplo um problema (PQ) muito grande.

No que segue, apresenta-se uma pequena descrição de alguns métodos de feixe onde é possível ver suas modificações. Para evitar detalhes técnicos, procura-se dar foco as principais diferenças na escolha da aproximação por plano de cortes \check{f}_k , o erro de linearização e_j^k ou a matriz de estabilização M_k .

1.4.5 Métodos de Feixe com Métrica Variável Diagonal

A ideia dos métodos com métrica variável diagonal está baseada na introdução de um parâmetro de ponderação no termo quadrático da função objetivo do problema quadrático, para acumular alguma informação de segunda ordem sobre a curvatura de f ao redor de x_k . Portanto a matriz de métrica variável M_k é considerada na forma diagonal

$$M_k = u_k I,$$

com o parâmetro de ponderação $u_k > 0$.

Baseado no algoritmo de ponto proximal de [33] e no trabalho de [23], o *método de feixe proximal* foi apresentado em [34], onde também uma técnica de interpolação quadrática para atualizar u_k foi introduzida.

Um resultado similar foi concluído em [19], onde o método de *feixe com regiões de confiança* foi desenvolvido combinando a ideia do feixe com o clássico método de regiões de confiança de [35, 36].

Além disso, o método *quase-Newton diagonal “poor man’s”* de [37] e o *método de feixe proximal* de [38] baseados na Regularização de Moreau-Yosida estão nesta classe de métodos de feixe.

Estes métodos diferenciam-se basicamente nas estratégias para atualizar o parâmetro u_k .

1.4.6 Métodos de Feixe com Métrica Variável

O desenvolvimento de métodos de segunda ordem para otimização não diferenciável tem chamado a atenção de alguns pesquisadores durante toda a sua história. Várias tentativas de empregar

$$M_k \quad \text{como um matriz "cheia"}$$

com algum tipo de atualização tem sido propostos por alguns autores. Já em seu trabalho pioneiro [12], Lemaréchal apresenta uma versão do método de Feixe com métrica variável, utilizando a formulação clássica do BFGS com atualização secante para otimização diferenciável (veja [39], p.135). Devido ao desapontamento com os resultados numéricos obtidos em [40], esta ideia foi abandonada por aproximadamente duas décadas. Depois, baseado na regularização de Moreau-Yosida, atualização BFGS e na técnica de busca em curva, um método quase-Newton foi proposto em [37]. De acordo com as experiências numéricas em [41], vê-se que os métodos de Feixe com métrica variável trabalham razoavelmente bem. Um trabalho mais recente baseado no método de feixe com métricas variáveis usando atualização BFGS, foi proposto em [42], a ideia do método é usar somente três subgradientes (dois calculados em \mathbf{x}^k e \mathbf{y}_{k+1} , e um agregado, contendo informações das últimas iterações). O propósito disso, segue do fato que a dimensão do problema quadrático é três e, com isso, o problema pode então ser resolvido com cálculos simples. Os testes numéricos em [42] mostram que a técnica é comparável com os métodos de feixe com métrica variável diagonal em número de avaliações da função, contudo o tempo computacional pode ser significativamente menor.

1.4.7 Métodos de Feixe - Newton

Um avanço na direção de métodos de feixe com informações de segunda ordem foi dado em [43], onde ao invés de um modelo de planos de corte linear por partes (1.15), os autores introduzem um modelo quadrático da forma

$$\tilde{f}_k(\mathbf{x}) := \max_{j \in J_k} \left\{ f(\mathbf{y}_j) + \mathbf{s}_j^T (\mathbf{x} - \mathbf{y}_j) + \frac{1}{2} \rho_j (\mathbf{x} - \mathbf{y}_j)^T M_j (\mathbf{x} - \mathbf{y}_j) \right\}. \quad (1.20)$$

onde ρ_j é um parâmetro de amortecimento. A direção de busca é encontrada no problema (1.16) foi então trocada pelo problema

$$\mathbf{d}^k := \arg \min_{\mathbf{d} \in \mathbb{R}^n} \left\{ \tilde{f}_k(\mathbf{x}^k + \mathbf{d}) \right\}. \quad (1.21)$$

1.4.8 Métodos de Feixe com Dados inexatos

Existe também variantes do método de feixes para casos onde a função objetivo e o subgradiente possuem dados inexatos. Supõe-se que para cada ponto $\mathbf{x} \in \mathbb{R}^n$ e $\varepsilon > 0$ é possível calcular o valor aproximado da função $f_\varepsilon(\mathbf{x})$ satisfazendo

$$f(\mathbf{x}) - \varepsilon \leq f_\varepsilon(\mathbf{x}) \leq f(\mathbf{x})$$

e um ε -subgradiente \mathbf{s}_ε do ε -subdiferencial

$$\partial_\varepsilon f(\mathbf{x}) = \{ \mathbf{s} \in \mathbb{R}^n \mid f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{s}^T(\mathbf{y} - \mathbf{x}) - \varepsilon, \text{ para todo } \mathbf{y} \in \mathbb{R}^n \}.$$

Então o modelo de planos de corte em (1.15) pode ser substituído por um *modelo de planos de corte aproximado*

$$\bar{f}_k(\mathbf{x}) = \max_{j \in J_k} \left\{ f_{\varepsilon_j}(\mathbf{y}_j) + (\mathbf{s}_{\varepsilon_j})^T(\mathbf{x} - \mathbf{y}_j) \right\}.$$

onde $\mathbf{s}_{\varepsilon_j} \in \partial_{\varepsilon_j} f(\mathbf{y}_j)$ para todo $j \in J_k$ e $\varepsilon \rightarrow 0$.

Em [44] o modelo de plano de corte generalizado de [11] foi estendido para dados inexatos, em outras palavras

$$M_k = I.$$

Depois o método de feixe proximal de [34] com

$$M_k = u_k I$$

foi estendido em [45] onde também alguma experiência numérica foi reportada.

1.5 Otimização Não Convexa Não Diferenciável

Nesta seção considera-se o seguinte problema de otimização irrestrito

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \text{sujeito a } \mathbf{x} \in \mathbb{R}^n \end{cases} \quad (1.22)$$

onde a função objetivo $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é suposta localmente Lipschitz contínua. Note que f não precisa ser convexa. O subdiferencial de f agora é definido por

$$\partial f(\mathbf{x}) = \text{conv}\{\lim_{i \rightarrow \infty} \nabla f(\mathbf{x}^i) \mid \mathbf{x}^i \rightarrow \mathbf{x} \text{ and } \nabla f(\mathbf{x}^i) \text{ existe.}\} \quad (1.23)$$

Lembrando que para as funções localmente Lipschitz contínuas tem-se a seguinte condição necessária de otimalidade: Se f , localmente Lipschitz contínua, atinge um mínimo local em \mathbf{x}^* , então

$$\mathbf{0} \in \partial f(\mathbf{x}^*). \quad (1.24)$$

A não convexidade trás algumas dificuldades, como por exemplo: Os métodos não podem garantir a otimalidade local das soluções, pois somente alguns candidatos, chamados pontos estacionários, satisfazem a condição (1.24). Outro fato que deve ser levado em consideração, é que no caso convexo, o modelo de planos de corte era uma estimação da função objetivo e o erro de linearização não negativo media o quão boa era a aproximação do problema original. No caso não convexo, estas propriedades não são mais válidas: e_j^k pode ser muito pequeno ou negativo embora o ponto teste \mathbf{y}_j esteja longe do ponto de iteração corrente \mathbf{x}^k e assim o subgradiente correspondente \mathbf{s}_j seria inútil. Por estas razões, as principais modificações para os métodos não convexos se concentram no erro de linearização.

1.5.1 Regras de eliminação de subgradientes

Alguns métodos são propostos simplesmente ignorando o erro de linearização e_j^k , Definição (1.33). Este procedimento já era feito nos métodos de gradiente conjugado [46], onde por exemplo, o seguinte tipo de regra de eliminação é proposta

$$J_k := \{1 \leq j \leq k \mid \|\mathbf{x}^k - \mathbf{y}_j\| \leq \delta_k\}$$

onde δ_k tende para zero. Em [47] um método de feixe transladado, é proposto sem qualquer tipo de regra de eliminação, pois o erro de linearização é sempre não-negativo, ou seja,

$$\beta_j^k = f(\mathbf{y}_j) - f(\mathbf{x}^k) \geq 0.$$

Um regra de eliminação mais complicada é proposta em [11], Capítulo 4, para métodos de planos de corte generalizados. O erro de linearização é substituído por seu valor absoluto

$$\beta_j^k := |e_j^k| = |f(\mathbf{x}^k) - f(\mathbf{y}_j) - \mathbf{s}_j^T(\mathbf{x}^k - \mathbf{y}_j)| \quad \text{para todo } j \in J_k,$$

e o algoritmo elimina a informação dos subgradientes anteriores sempre que

$$\|\mathbf{d}_k\| \leq m_S \cdot \max_{j \in J_k} \{s_j^k\},$$

onde $m_S > 0$ é uma tolerância para resetar, definida pelo usuário e

$$s_j^k := \|\mathbf{x}^j - \mathbf{y}_j\| + \sum_{i=j}^{k-1} \|\mathbf{x}^{i+1} - \mathbf{x}^i\| \quad (1.25)$$

é a medida da distância estimada de $\|\mathbf{x}^k - \mathbf{y}_j\|$ sem a necessidade de armazenar os pontos testes \mathbf{y}_j .

1.5.2 Subgradiente com medida local

No que segue, introduz-se outra estratégia popular para evitar as dificuldades causadas pela não convexidade. Para adicionar alguma informação local ao modelo, o erro de linearização é substituído pelo *subgradiente com medida local*

$$\beta_j^k := \max \left\{ e_j^k, \gamma \|\mathbf{x}^k - \mathbf{y}_j\|^2 \right\} \quad (1.26)$$

em [48] para o método ε -*steepest descent*. O parâmetro de medida da distância $\gamma \geq 0$ pode ser definido como zero quando a função f é convexa. Os autores também propõe o uso de uma medida da distância (1.25) evitando o armazenamento dos pontos testes \mathbf{y}_j , ou seja, trocando (1.26) por

$$\beta_j^k := \max \left\{ e_j^k, \gamma (s_j^k)^2 \right\}.$$

1.6 Otimização Convexa Não Diferenciável com Restrições

Considera-se o seguinte problema de otimização

$$\begin{cases} \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} & f(\mathbf{x}) \\ \text{sujeito a} & c_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, m, \end{cases} \quad (1.27)$$

onde a função objetivo $f : \mathbb{R}^n \rightarrow \mathbb{R}$ e as restrições funcionais $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$ são supostas convexas. Para este problema, considera-se que ele satisfaz a *Condição de Qualificação de Slater*, i.e., se

$$c(\mathbf{y}) < 0$$

para algum $\mathbf{y} \in \mathbb{R}^n$, onde $c : \mathbb{R}^n \rightarrow \mathbb{R}$ é a *função restrição total*, definida por

$$c(\mathbf{x}) := \max_{i=1, \dots, m} c_i(\mathbf{x}). \quad (1.28)$$

Teorema 1.34. *Se o problema (1.27) é convexo e satisfaz a condição de qualificação de Slater, então (1.27) atinge seu mínimo em \mathbf{x}^* se e somente se existe o multiplicador de Lagrange $0 \leq \mu \in \mathbb{R}^m$ tal que $\mu_i c_i(\mathbf{x}^*) = 0$ para todo $i = 1, \dots, m$ e*

$$0 \in \partial f(\mathbf{x}^*) + \sum_{i=1}^m \mu_i \partial c_i(\mathbf{x}^*).$$

Demonstração. Veja [13]. □

1.6.1 Método de Linearização de Restrições

Nos métodos de feixe, a forma mais popular de se lidar com as restrições é encontrar um mínimo irrestrito da chamada *improvement function* [23],

$$H(\mathbf{x}; \mathbf{y}) = \max\{f(\mathbf{x}) - f(\mathbf{y}), c(\mathbf{x})\}.$$

O modelo, a partir de planos de corte, da função $H(\mathbf{x}; \mathbf{y})$ é determinado linearizando a função objetivo e a restrição, considerando

$$\check{H}_k(\mathbf{x}) = \max\{\check{f}(\mathbf{x}) - f(\mathbf{x}_k), \check{c}_k(\mathbf{x})\}, \quad (1.29)$$

onde

$$\check{c}_k(\mathbf{x}) = \max_{j \in J_k} \{c(\mathbf{y}_j) + \mathbf{r}_j^T (\mathbf{x} - \mathbf{y}_j)\}. \quad (1.30)$$

e $\mathbf{r}_j \in \partial c(\mathbf{y}_j)$. A direção de busca encontrada em (1.16) é substituída por

$$\mathbf{d}_k := \arg \min_{\mathbf{d} \in \mathbb{R}^n} \left\{ \check{H}_k(\mathbf{x}_k + \mathbf{d}) + \frac{1}{2} \mathbf{d}^T M_k \mathbf{d} \right\}. \quad (1.31)$$

Esta abordagem foi empregada no contexto do método de plano de corte generalizado, ou seja, para

$$M_k = I,$$

no trabalho de Mifflin [24]. Versões salvando e armazenando subgradientes com estratégias de agregação foram introduzidas em [11], Capítulos 5 e 6.

Em [13] a abordagem de linearização da restrição foi usada com o método de feixe proximal, ou seja, com

$$M_k = u_k I,$$

com $u_k > 0$.

1.6.2 Método de Função Penalidade Exata

Uma outra estratégia em otimização não diferenciável com restrições é utilizar a função de penalidade exata [49],

$$e(\mathbf{x}; \alpha) := f(\mathbf{x}) + \sum_{i=1}^m \alpha_i \cdot \max\{c_i(\mathbf{x}), 0\}.$$

O modelo de planos de corte da função de penalidade exata é então definido por

$$\check{e}_k(\mathbf{x}; \alpha_k) := \check{f}_k(\mathbf{x}) + \sum_{i=1}^m \alpha_i^k \cdot \max\{\check{c}_i^k(\mathbf{x}), 0\}.$$

onde

$$\check{c}_i^k(\mathbf{x}) = \max_{j \in J_k} \left\{ c_i(\mathbf{y}_j) + \mathbf{r}_{i,j}^T (\mathbf{x} - \mathbf{y}_j) \right\}$$

e $\mathbf{r}_{i,j} \in \partial c_i(\mathbf{y}_j)$. A direção de busca é então obtida por

$$\mathbf{d}_k := \arg \min_{\mathbf{d} \in \mathbb{R}^n} \left\{ \check{e}_k(\mathbf{x}_k + \mathbf{d}) + \frac{1}{2} \mathbf{d}^T M_k \mathbf{d} \right\}. \quad (1.32)$$

1.7 Algoritmo de Pontos Interiores e Direções Viáveis

O algoritmo que apresentado nesta seção foi proposto por José Herkovits em [50, 51], e denominado FDIPA (*Feasible Direction Interior Point Algorithm*). Proposto para lidar com problemas de otimização não linear diferenciável, o FDIPA converge globalmente para pontos Karush-Kuhn-Tucker. É um método diferenciado, pois não é necessária a solução de subproblemas quadráticos, e não se trata de um método de penalidades ou barreira, ou filtros.

Apesar de ser uma técnica para problemas diferenciáveis, neste trabalho, os sistemas internos do FDIPA são amplamente utilizados na determinação de direções viáveis de descida para os métodos que estão sendo apresentados nos capítulos seguintes.

Considera-se o seguinte problema de otimização não linear diferenciável com restrições de desigualdade:

$$\begin{cases} \text{minimize} & f(\mathbf{x}) \\ & \mathbf{x} \in \mathbb{R}^n \\ \text{sujeito a} & g(\mathbf{x}) \leq 0 \end{cases} \quad (1.33)$$

e caracteriza-se suas soluções, onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ e $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ são funções diferenciáveis. Denota-se por $\nabla g(\mathbf{x}) \in \mathbb{R}^{n \times m}$ a matriz das derivadas de g , $\Omega =$

$\{\mathbf{x} \in \mathbb{R}^n \mid g(\mathbf{x}) \leq 0\}$ o conjunto viável e introduzindo a variável auxiliar $\boldsymbol{\lambda} \in \mathbb{R}^m$, chamada variável dual ou multiplicador de Lagrange, defini-se a função Lagrangiana associada ao problema (1.33)

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T g(\mathbf{x}),$$

cuja matriz Hessiana é dada por

$$H(\mathbf{x}, \boldsymbol{\lambda}) = \nabla^2 f(\mathbf{x}) + \sum_{i=1}^m \lambda_i \nabla^2 g_i(\mathbf{x}).$$

Representa-se por $I(\mathbf{x}) = \{i \mid g_i(\mathbf{x}) = 0\}$ o conjunto de índices cujas restrições são ativas e se diz que \mathbf{x} é um *ponto regular* se os vetores $\nabla g_i(\mathbf{x})$ para $i \in I(\mathbf{x})$ forem linearmente independentes.

Dado um ponto interior inicial, o FDIPA gera uma sequência $\{\mathbf{x}^k\}$ de pontos interiores tais que

$$f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k) \text{ e } g_i(\mathbf{x}^k) < 0 \text{ para } i = 1, \dots, m$$

que converge para o conjunto de pontos KKT do problema. Em cada iteração uma direção é definida de modo que também seja de descida para a função objetivo, em seguida, o sistema é perturbado para introduzir uma deflexão na direção de descida no sentido do interior da região viável de modo a obter ainda uma direção de descida viável de Ω . Uma busca linear inexata nessa direção para a função objetivo determina o próximo ponto interior, garantindo a convergência global.

Para isso, são assumidas as seguintes hipóteses:

Hipótese 1.1. *Seja $\Omega \equiv \{\mathbf{x} \in \mathbb{R}^n \mid g(\mathbf{x}) \leq 0\}$ o conjunto viável. Então existe um número real a tal que o conjunto $\Omega_a \equiv \{\mathbf{x} \in \Omega \mid f(\mathbf{x}) \leq a\}$ é compacto e tem interior denotado por $\text{int}(\Omega_a)$.*

Hipótese 1.2. *Cada $\mathbf{x} \in \text{int}(\Omega_a)$ satisfaz $g(\mathbf{x}) < 0$.*

Hipótese 1.3. *As funções f e g são continuamente diferenciáveis em Ω_a e suas derivadas satisfazem uma condição de Lipschitz.*

Hipótese 1.4. *(Condição de Regularidade) Um ponto $\mathbf{x} \in \Omega_a$ é chamado regular se os gradientes $\nabla g_i(\mathbf{x})$, para i tal que $g_i(\mathbf{x}) = 0$, são linearmente independentes. O FDIPA requer a hipótese de regularidade numa solução local de (1.33).*

Definição 1.35. *Um campo de direções $\mathbf{d}(\mathbf{x})$ definido em Ω é chamado um campo uniforme de direções viáveis do problema (1.33), se existe um tamanho de passo $\tau > 0$ tal que $\mathbf{x} + t\mathbf{d}(\mathbf{x}) \in \Omega$ for all $t \in [0, \tau]$ e para todo $\mathbf{x} \in \Omega$.*

Herkovits mostrou em [51] que d é uma direção viável se $\mathbf{d}^T \nabla g_i(\mathbf{x}) < 0$ para qualquer i tal que $g_i(\mathbf{x}) = 0$. A Definição 1.35 introduz uma condição para que um campo de vetores $\mathbf{d}(\mathbf{x})$, que é mais forte do que simplesmente viabilidade de

qualquer elemento de $\mathbf{d}(\mathbf{x})$. Quando $d(x)$ constitui um campo uniforme de direções viáveis, ele suporta um segmento viável $[\mathbf{x}, \mathbf{x} + \theta(\mathbf{x})\mathbf{d}(\mathbf{x})]$, tal que $\theta(\mathbf{x})$ é limitado inferiormente em Ω por $\tau > 0$.

Se \mathbf{x}^* é um ponto regular e mínimo local para o problema (1.33), então existe $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ tal que

$$\nabla f(\mathbf{x}^*) + \nabla g(\mathbf{x}^*)\boldsymbol{\lambda}^* = 0 \quad (1.34)$$

$$G(\mathbf{x}^*)\boldsymbol{\lambda}^* = 0 \quad (1.35)$$

$$\boldsymbol{\lambda}^* \geq 0 \quad (1.36)$$

$$g(\mathbf{x}^*) \leq 0 \quad (1.37)$$

onde $G(\mathbf{x}) = \text{diag}[g_1(\mathbf{x}), \dots, g_m(\mathbf{x})]$ é uma matriz $m \times m$.

Considerando as equações (1.35) e (1.36), e fazendo

$$\mathbf{y} = (\mathbf{x}, \boldsymbol{\lambda}) \quad \text{e} \quad \Phi(\mathbf{y}) = (\nabla f(\mathbf{x}) + \nabla g(\mathbf{x})\boldsymbol{\lambda}, G(\mathbf{x})\boldsymbol{\lambda})$$

obtém-se

$$\nabla \Phi(\mathbf{y}) = \begin{bmatrix} H(\mathbf{x}, \boldsymbol{\lambda}) & \nabla g(\mathbf{x}) \\ \Lambda \nabla g(\mathbf{x})^T & G(\mathbf{x}) \end{bmatrix}$$

onde $\Lambda = \text{diag}[\lambda_1, \dots, \lambda_m]$ é uma matriz $m \times m$.

Uma iteração de Newton para resolver o sistema de equações lineares $\Phi(\mathbf{y}) = 0$, com o ponto $\mathbf{y}^k = (\mathbf{x}^k, \boldsymbol{\lambda}^k)$ na iteração k , define um novo ponto $\mathbf{y}^{k+1} = (\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1})$ solução do sistema linear

$$\nabla \Phi(\mathbf{y}^k)(\mathbf{y} - \mathbf{y}^k)^T = -\Phi(\mathbf{y}^k)^T$$

que pode ser reescrito como

$$\begin{bmatrix} B^k & \nabla g(\mathbf{x}^k) \\ \Lambda^k \nabla g(\mathbf{x}^k)^T & G(\mathbf{x}^k) \end{bmatrix} \begin{bmatrix} \mathbf{x} - \mathbf{x}^k \\ \boldsymbol{\lambda} - \boldsymbol{\lambda}^k \end{bmatrix} = - \begin{bmatrix} \nabla f(\mathbf{x}^k) + \nabla g(\mathbf{x}^k)\boldsymbol{\lambda}^k \\ G(\mathbf{x}^k)\boldsymbol{\lambda}^k \end{bmatrix}, \quad (1.38)$$

onde substitui-se $H^k(\mathbf{x}^k, \boldsymbol{\lambda}^k)$ por B^k .

De fato, a matriz simétrica $B^k \in \mathbb{R}^{n \times n}$ pode ser considerada como uma aproximação da matriz Hessiana obtida por alguma técnica quase-Newton ou pela própria matriz identidade. Contudo, como requerimento para a convergência global do presente algoritmo, B^k deve ser definida positiva.

Introduzindo algumas modificações na iteração (1.38) obtém-se, para um dado par $(\mathbf{x}^k, \boldsymbol{\lambda}^k)$, uma nova estimativa com valor da função objetivo menor. Com isto, defini-se uma direção \mathbf{d}_α no espaço primal por $\mathbf{d}_\alpha = \mathbf{x}^{k+1} - \mathbf{x}^k$.

Então obtém-se o sistema linear

$$B^k \mathbf{d}_\alpha + \nabla g(\mathbf{x}^k)\boldsymbol{\lambda}_\alpha = -\nabla f(\mathbf{x}^k) \quad (1.39)$$

$$\Lambda^k \nabla g(\mathbf{x}^k)^T \mathbf{d}_\alpha + G(\mathbf{x}^k)\boldsymbol{\lambda}_\alpha = 0 \quad (1.40)$$

A solução $(\mathbf{d}_\alpha, \boldsymbol{\lambda}_\alpha)$ deste sistema, fornece uma direção \mathbf{d}_α , e uma nova estimativa $\boldsymbol{\lambda}_\alpha$ para $\boldsymbol{\lambda}$. Em [51], Herskovits provou que \mathbf{d}_α é uma direção de descida para f , ou seja, que $\mathbf{d}_\alpha^T \nabla f(\mathbf{x}^k) < 0$.

Entretanto, \mathbf{d}_α pode não ser uma direção viável, pois quando alguma restrição se aproxima de zero, \mathbf{d}_α tende a uma direção tangente ao conjunto viável.

De fato, reescrevendo a equação (1.40) tem-se

$$\lambda_i^k \nabla g_i(\mathbf{x}^k)^T \mathbf{d}_\alpha + g_i(\mathbf{x}^k) \lambda_{\alpha i} = 0 \quad i = 1, \dots, m$$

o que implica em $\nabla g_i(\mathbf{x}^k)^T \mathbf{d}_\alpha = 0$ para todo i tal que $g_i(\mathbf{x}^k) = 0$.

Para evitar este problema, adiciona-se um vetor negativo no lado direito de (1.40) e define-se um novo sistema em $(\mathbf{d}, \bar{\boldsymbol{\lambda}})$

$$B^k \mathbf{d} + \nabla g(\mathbf{x}^k) \bar{\boldsymbol{\lambda}} = -\nabla f(\mathbf{x}^k) \quad (1.41)$$

$$\Lambda^k \nabla g(\mathbf{x}^k)^T \mathbf{d} + G(\mathbf{x}^k) \bar{\boldsymbol{\lambda}} = -\rho^k \Lambda^k \quad (1.42)$$

onde $\rho^k > 0$, \mathbf{d} é a nova direção e $\bar{\boldsymbol{\lambda}}$ é a nova estimativa de $\boldsymbol{\lambda}$.

Neste caso, (1.42) é equivalente a

$$\lambda_i^k \nabla g_i(\mathbf{x}^k)^T \mathbf{d} + g_i(\mathbf{x}^k) \bar{\lambda}_i = -\rho^k \lambda_i^k \quad \text{para } i = 1, \dots, m$$

e, conseqüentemente

$$\nabla g_i(\mathbf{x}^k)^T \mathbf{d} = -\rho^k < 0$$

para as restrições ativas tais que $\lambda_i^k \neq 0$. Logo \mathbf{d} é uma direção viável.

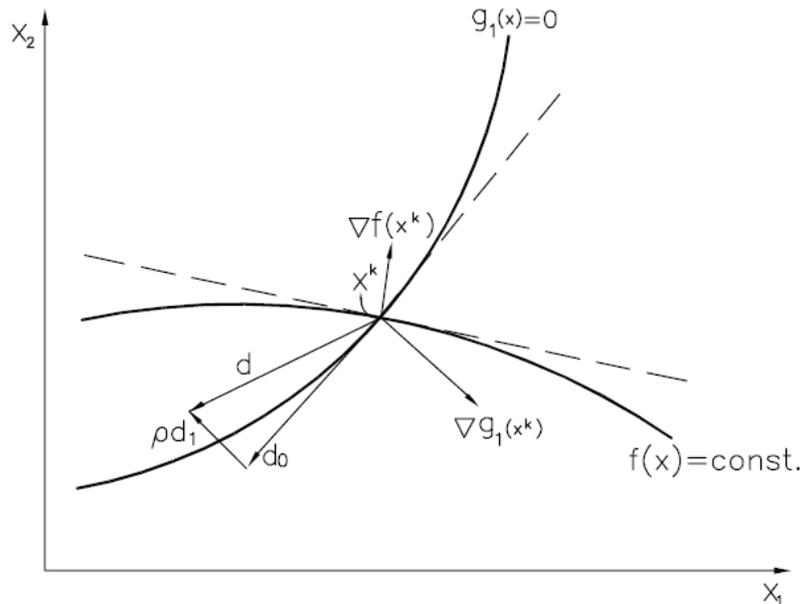


Figura 1.5: Direção de busca do FDIPA

A inclusão do termo negativo do lado direito da equação (1.40) produz uma deflexão em \mathbf{d} , proporcional à ρ , na direção do interior da região viável. Para garantir que \mathbf{d} seja uma direção de descida, ρ deve ser escolhido convenientemente, veja [51].

Observa-se que à direção \mathbf{d} e $\bar{\boldsymbol{\lambda}}$ podem ser obtidos resolvendo-se os sistemas de equação lineares (1.39)-(1.40) e (1.41)-(1.42) de modo que são definidos

$$\mathbf{d} = \mathbf{d}_\alpha + \rho \mathbf{d}_\beta \quad \text{e} \quad \bar{\boldsymbol{\lambda}} = \boldsymbol{\lambda}_\alpha + \rho \boldsymbol{\lambda}_\beta.$$

Agora se está em condições de apresentar o FDIPA.

1.7.1 Feasible Direction Interior Point Algorithm

ALGORITMO - FDIPA

Parâmetros: $\xi, \mu \in (0, 1)$, $\varphi > 0$, $\nu \in (0, 1)$.

Dados: (Inicialização)

$\mathbf{x}^0 \in \Omega^\circ$, $0 < \boldsymbol{\lambda}_0 < \mathbb{R}^n$, $B^0 \in \mathbb{R}^{n \times n}$ simétrica e definida positiva.

Passo 1: (Cálculo da direção de busca)

i) Calcule $(\mathbf{d}_\alpha, \boldsymbol{\lambda}_\alpha)$ resolvendo o sistema linear

$$\begin{cases} B\mathbf{d}_\alpha + \nabla g(\mathbf{x})\boldsymbol{\lambda}_\alpha = -\nabla f(\mathbf{x}) \\ \Lambda \nabla g(\mathbf{x}^k)^T \mathbf{d}_\alpha + G(\mathbf{x})\boldsymbol{\lambda}_\alpha = 0 \end{cases}$$

Se $\mathbf{d}_\alpha = 0$ pare.

ii) Calcule $(\mathbf{d}_\beta, \boldsymbol{\lambda}_\beta)$ resolvendo o sistema linear

$$\begin{cases} B\mathbf{d}_\beta + \nabla g(\mathbf{x})\boldsymbol{\lambda}_\beta = 0 \\ \Lambda \nabla g(\mathbf{x})^T \mathbf{d}_\beta + G(\mathbf{x})\boldsymbol{\lambda}_\beta = -\Lambda \end{cases}$$

iii) Se $\mathbf{d}_\beta^T \nabla f(\mathbf{x}) > 0$ então

$$\rho = \min \left\{ \varphi \|\mathbf{d}_\alpha\|_2^2; (\xi - 1) \frac{\mathbf{d}_\alpha^T \nabla f(\mathbf{x})}{\mathbf{d}_\beta^T \nabla f(\mathbf{x})} \right\}$$

Caso contrário,

$$\rho = \varphi \|\mathbf{d}_\alpha\|_2^2$$

iv) Calcule a direção de busca $\mathbf{d} = \mathbf{d}_\alpha + \rho \mathbf{d}_\beta$ e $\bar{\boldsymbol{\lambda}} = \boldsymbol{\lambda}_\alpha + \rho \boldsymbol{\lambda}_\beta$

Passo 2: (Busca Linear)

Calcule t , o primeiro elemento da seqüência $\{1, \nu, \nu^2, \nu^3, \dots\}$ que satisfaz

$$\begin{aligned}
& f(\mathbf{x} + t\mathbf{d}) \leq f(\mathbf{x}) + t\eta \nabla f(\mathbf{x})^T \mathbf{d}, \\
\text{e} \quad & g_i(\mathbf{x} + t\mathbf{d}) < 0 \quad \text{se } \bar{\lambda} \geq 0, \\
\text{ou} \quad & g_i(\mathbf{x} + t\mathbf{d}) \leq g_i(\mathbf{x}) \text{ caso contrário, onde } i = 1, \dots, m.
\end{aligned}$$

Passo 3: (Atualizações)

i) Faça $\mathbf{x} := \mathbf{x} + t\mathbf{d}$

e defina os novos valores de $\lambda > 0$ e B simétrica, definida positiva.

ii) Vá para o passo 1.

A seguinte atualização para λ foi considerada em [51],

$$\lambda_i := \max [\lambda_{\alpha i}; \epsilon \|\mathbf{d}_\alpha\|^2], \quad \epsilon > 0, \quad \text{para } i = 1, \dots, m.$$

Capítulo 2

Algoritmo para Otimização Convexa Não Diferenciável

Introduz-se o algoritmo para otimização convexa não diferenciável, que foi apresentado inicialmente em 2005, na tese de doutorado de Freire [29]. Contudo, neste trabalho, foram realizadas algumas modificações em relação ao trabalho de Freire, como por exemplo, a introdução do “passo sério de descida máxima”, hipóteses, e conseqüentemente partes da demonstração da convergência global do algoritmo. Por fim, foram feitos novos testes numéricos e uma análise comparativa do desempenho com outros algoritmos traçando seu performance profiles, usando como medida de performance o número de iterações e avaliações da função.

Ressalta-se que não serão citados os detalhes das modificações entre o algoritmo apresentado por Freire e o que por hora está sendo apresentado, uma vez que, o objetivo deste trabalho é dar continuidade na pesquisa e no aprimoramento das técnicas inicialmente apresentadas por Freire em sua tese de doutoramento.

2.1 Método

Neste estudo considera-se o seguinte problema de otimização:

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \text{sujeito a } \mathbf{x} \in \mathbb{R}^n, \end{cases} \quad (\text{P.1})$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa, não necessariamente diferenciável. Seja $\partial f(\mathbf{x})$ o subdiferencial [16] de f em \mathbf{x} . No que segue, assumi-se que um subgradiente arbitrário $\mathbf{s} \in \partial f(\mathbf{x})$ pode ser calculado em qualquer ponto $\mathbf{x} \in \mathbb{R}^n$.

Uma característica especial da otimização não diferenciável é o fato de que o gradiente $\nabla f(\mathbf{x})$ pode mudar descontinuamente, e ainda, não precisa ser necessariamente “pequeno” na vizinhança de um extremo local da função objetivo,

como o leitor pode ver em [15, 31]. Por esta razão, que os métodos de otimização clássicos, baseados no gradiente, não podem ser utilizados de forma direta. Tendo isso em vista, novos métodos tem sido propostos para resolver o problema (P.1), como pode ser visto no capítulo 2 ou, por exemplo, nos trabalhos de Kiwiel [11], Makëla [13].

Nesta nova proposta de algoritmo, são combinadas algumas ideias do clássico método de planos de corte de Kelley [10], os passos “sério” e “nulo” do método de feixe tradicional [11] com os sistemas internos do FDIPA - algoritmo de pontos interiores e direções viáveis desenvolvido por Hershkovits [30, 50], para criar um método adequado a resolução de problemas convexos não diferenciáveis. A escolha desta abordagem foi feita com base na não utilização de qualquer tipo de subproblema de programação quadrática, funções de penalidade, barreira, ou mesmo filtros.

A metodologia utilizada está baseada na substituição do problema (P.1) por um problema equivalente restrito (PE.1) com uma função objetivo linear e com restrições de desigualdade não diferenciáveis,

$$\begin{cases} \text{minimize} & F(\mathbf{x}, z) = z \\ & (\mathbf{x}, z) \in \mathbb{R}^{n+1} \\ \text{sujeito a} & f(\mathbf{x}) \leq z. \end{cases} \quad (\text{PE.1})$$

onde $z \in \mathbb{R}$ é uma variável auxiliar. Dessa forma pretende-se criar um algoritmo que gere uma sequência decrescente de pontos viáveis $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}} \subset \text{int}(\text{epi } f)$ que convirja para o mínimo de $f(\mathbf{x})$. E para isso, o algoritmo gera uma sequência que satisfaz

$$z^{k+1} < z^k, \quad \text{e} \quad f(\mathbf{x}^k) < z^k \quad \text{para todo } k.$$

Utilizando as ideias do método de planos de corte [10], o algoritmo gera uma aproximação linear por partes das restrições de (PE.1) da seguinte forma: Considere o plano de corte

$$g_i(\mathbf{x}, z) = f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) - z, \quad i = 0, 1, \dots, \ell$$

onde $\mathbf{y}_i \in \mathbb{R}^n$ são pontos auxiliares, $\mathbf{s}_i \in \partial f(\mathbf{y}_i)$ são subgradientes nestes pontos e ℓ representa o número de planos corrente. Agora defina,

$$\bar{\mathbf{g}}_\ell(\mathbf{x}, z) \equiv [g_0(\mathbf{x}, z), \dots, g_\ell(\mathbf{x}, z)]^T, \quad \bar{\mathbf{g}}_\ell : \mathbb{R}^n \times \mathbb{R} \longrightarrow \mathbb{R}^{\ell+1}$$

e o problema auxiliar corrente

$$\begin{cases} \text{minimize} & F(\mathbf{x}, z) = z \\ & (\mathbf{x}, z) \\ \text{sujeito a} & \bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq 0. \end{cases} \quad (\text{PA.1})$$

Ao invés de resolver este problema, o presente algoritmo utiliza os sistemas internos do FDIPA para determinar uma direção de busca \mathbf{d}_ℓ^k para (PA.1). Com o FDIPA a direção de descida \mathbf{d}_ℓ^k pode ser calculada ainda se o problema auxiliar (PA.1) não tiver um mínimo finito. E portanto, não é necessária a utilização de um termo quadrático estabilizante como é feito nos métodos de feixe, como pode ser visto em [15].

Quando usado para lidar com problemas de programação linear (como (PA.1)), o FDIPA se comporta de forma similar aos métodos de ponto interiores para programação linear [30], desta forma, esta é uma boa alternativa para resolver este tipo de problema.

Em cada iteração, um programa auxiliar (PA.1) linear é definido substituindo a restrição $f(\mathbf{x}) \leq z$ por planos de corte. Com isso, obtém-se uma direção de descida viável para (PA.1), e um tamanho de passo é calculado.

O maior passo viável é dado por $t \equiv \max\{t \mid \bar{\mathbf{g}}_\ell^k((\mathbf{x}^k, z^k) + t\mathbf{d}_\ell^k) \leq 0\}$. Como t não é sempre finito, considera-se

$$t_\ell^k := \min\{t_{max}, t\}.$$

Então, um novo iterado $(\mathbf{x}^{k+1}, z^{k+1})$ é definido de acordo com o seguinte procedimento: O algoritmo produz pontos auxiliares $(\mathbf{y}_\ell, w_\ell)$ e quando um ponto auxiliar está no interior do epígrafo de f , se diz que foi realizado um “passo sério” e este ponto será o novo iterado. Caso contrário, continua-se com o mesmo iterado e se diz que foi realizado um “passo nulo”. Em todos os casos, um novo plano de corte é adicionado e o procedimento se repete até que um passo sério seja obtido.

Com a direção de descida e um tamanho de passo calculados, um novo ponto auxiliar com respeito a (PA.1) é calculado. Para isso, considera-se os pontos da forma:

$$(\mathbf{x}_{\ell+1}^k, z_{\ell+1}^k) = (\mathbf{x}^k, z^k) + t_\ell^k \mathbf{d}_\ell^k \quad (2.1)$$

viáveis com respeito ao problema auxiliar (PA.1). Depois calcula-se o próximo ponto auxiliar fazendo,

$$(\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k) = (\mathbf{x}^k, z^k) + \mu t_\ell^k \mathbf{d}_\ell^k, \quad (2.2)$$

onde $\mu \in (0, 1)$. Se $(\mathbf{y}_{\ell+1}, w_{\ell+1})$ é viável com respeito a (PE.1) e descente para f , atualiza-se a solução (i.e. faz-se $(\mathbf{x}_{\ell+1}, z_{\ell+1}) = (\mathbf{y}_{\ell+1}, w_{\ell+1})$) e diz-se que o passo é um **passo sério de descida viável**.

Se o novo ponto auxiliar é viável com relação a (PE.1), mas não é de descida para f , considera-se que o ponto de iteração corrente (\mathbf{x}^k, z^k) está longe da fronteira do epígrafo de f . E neste caso, ao invés de usar a direção calculada pelo FDIPA,

utiliza-se a direção de descida “rápida” $-\mathbf{e}_z$ (onde $\mathbf{e}_z = [0, \dots, 0, 1]^T \in \mathbb{R}^{n+1}$) para obter um ponto ainda estritamente viável e suficientemente próximo da fronteira do epígrafo. Assim, para a próxima iteração faz-se $f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k)$ e pode-se provar que a nova direção de busca gerada pelo FDIPA é também de descida para f . Este passo é chamado de *passo sério de máxima descida*.

Se nenhum dos casos acima citados acontece, o algoritmo realiza um *passo nulo*. Neste caso, a solução não é atualizada, mas um novo plano de corte é calculado em $(\mathbf{y}_{\ell+1}, w_{\ell+1})$ e uma nova direção viável com respeito a (PA.1) é calculada usando o FDIPA. Então o procedimento se inicia novamente.

2.1.1 Nonsmooth Feasible Direction Algorithm

ALGORITMO - NFDA

Parâmetros: Escolha uma tolerância aproximada final $\varepsilon > 0$. Selecione os parâmetros de controle $\varphi > 0$ e $\xi \in (0, 1)$ para o limite da deflexão. Selecione os multiplicadores $\mu \in (1/2, 1)$ para o tamanho de passo e o tamanho máximo de passo $t_{\max} > 0$.

Dados: (Inicialização) Escolha um ponto estritamente viável $(\mathbf{x}^0, z^0) \in \text{int}(\text{epi } f)$, um vetor positivo inicial $\boldsymbol{\lambda}^0 \in \mathbb{R}^\ell$ e uma matriz simétrica definida positiva $B^0 \in \mathbb{R}^{(n+1) \times (n+1)}$. Seja $\mathbf{y}_0^0 = \mathbf{x}^0$, $k = 0$ e $\ell = 0$. Calcule $f(\mathbf{x}^0)$.

Passo 1: (Plano de Corte inicial) Calcule $\mathbf{s}_0^k \in \partial f(\mathbf{x}^k)$ e um novo plano de corte

$$g_0^k(\mathbf{x}^k, z^k) = f(\mathbf{x}^k) - z^k.$$

Considere

$$\nabla g_0^k(\mathbf{x}^k, z^k) = \begin{bmatrix} \mathbf{s}_0^k \\ -1 \end{bmatrix} \in \mathbb{R}^{n+1},$$

defina

$$\begin{aligned} \bar{g}_0^k(\mathbf{x}^k, z^k) &= [g_0^k(\mathbf{x}^k, z^k)] \in \mathbb{R}, \quad \text{e} \\ \nabla \bar{g}_0^k(\mathbf{x}^k, z^k) &= [\nabla g_0^k(\mathbf{x}^k, z^k)] \in \mathbb{R}^{n+1}. \end{aligned}$$

Passo 2: Calculo da Direção de Descida Viável \mathbf{d}_ℓ^k para (PA.1)

i) Calcule $\mathbf{d}_{\alpha,\ell}^k$ e $\boldsymbol{\lambda}_{\alpha,\ell}^k$, resolvendo

$$B^k \mathbf{d}_{\alpha,\ell}^k + \nabla \bar{g}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_{\alpha,\ell}^k = -\nabla F(\mathbf{x}, z) \quad (2.3)$$

$$\Lambda_\ell^k [\nabla \bar{g}_\ell^k(\mathbf{x}^k, z^k)]^T \mathbf{d}_{\alpha,\ell}^k + G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_{\alpha,\ell}^k = 0. \quad (2.4)$$

Calcule $\mathbf{d}_{\beta,\ell}^k$ e $\boldsymbol{\lambda}_{\beta,\ell}^k$, resolvendo

$$B^k \mathbf{d}_{\beta,\ell}^k + \nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_{\beta,\ell}^k = 0 \quad (2.5)$$

$$\Lambda_\ell^k [\nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k)]^T \mathbf{d}_{\beta,\ell}^k + G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_{\beta,\ell}^k = -\boldsymbol{\lambda}_\ell^k, \quad (2.6)$$

onde

$$\begin{aligned} \boldsymbol{\lambda}_{\alpha,\ell}^k &:= (\lambda_{\alpha,1}^k, \dots, \lambda_{\alpha,\ell}^k), & \boldsymbol{\lambda}_{\beta,\ell}^k &:= (\lambda_{\beta,1}^k, \dots, \lambda_{\beta,\ell}^k), & \boldsymbol{\lambda}_\ell^k &:= (\lambda_1^k, \dots, \lambda_\ell^k), \\ \Lambda_\ell^k &:= \text{diag}[\lambda_1^k, \dots, \lambda_\ell^k], & G_\ell^k(\mathbf{x}, z) &:= \text{diag}[g_1^k(\mathbf{x}, z), \dots, g_\ell^k(\mathbf{x}, z)]. \end{aligned}$$

ii) Se $(\mathbf{d}_{\beta,\ell}^k)^T \nabla F(\mathbf{x}, z) > 0$, faça $\rho = \varphi \|\mathbf{d}_{\alpha,\ell}^k\|^2$.

Caso contrário, faça

$$\rho = \min \left\{ \varphi \|\mathbf{d}_{\alpha,\ell}^k\|^2, (\xi - 1) \frac{(\mathbf{d}_{\alpha,\ell}^k)^T \nabla F(\mathbf{x}, z)}{(\mathbf{d}_{\beta,\ell}^k)^T \nabla F(\mathbf{x}, z)} \right\}. \quad (2.7)$$

iii) Calcule a direção de descida viável

$$\mathbf{d}_\ell^k = \mathbf{d}_{\alpha,\ell}^k + \rho \mathbf{d}_{\beta,\ell}^k. \quad (2.8)$$

Passo 3: Calcule um tamanho de passo

$$t_\ell^k = \min \left\{ t_{max}, \max \{ t \mid \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k) + t \mathbf{d}_\ell^k \leq 0 \} \right\}. \quad (2.9)$$

Se

$$\|\mathbf{d}_\ell^k\| \leq \varepsilon$$

então pare com (\mathbf{x}^k, z^k) como sendo a solução. Caso contrário, faça

$$(\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k) = (\mathbf{x}^k, z^k) + \mu t_\ell^k \mathbf{d}_\ell^k,$$

e calcule o valor correspondente $f(\mathbf{y}_\ell^k)$.

Se $f(\mathbf{y}_{\ell+1}^k) \geq w_{\ell+1}^k$, então tem-se um *passo nulo*: vá para o Passo 6). Caso contrário, faça $\mathbf{d}^k = \mathbf{d}_\ell^k$, $\mathbf{d}_\alpha^k = \mathbf{d}_{\alpha\ell}^k$, $\mathbf{d}_\beta^k = \mathbf{d}_{\beta\ell}^k$, $\boldsymbol{\lambda}_\alpha^k = \boldsymbol{\lambda}_{\alpha\ell}^k$, $\boldsymbol{\lambda}_\beta^k = \boldsymbol{\lambda}_{\beta\ell}^k$ e $\ell^k = \ell$.

Se $f(\mathbf{x}^k) \geq f(\mathbf{y}_{\ell+1}^k)$ vá para o Passo 4, senão: vá para o Passo 5.

Passo 4: (Passo sério de descida viável) Faça $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ e $f(\mathbf{x}^{k+1}) = f(\mathbf{y}_{\ell+1}^k)$. Defina $\boldsymbol{\lambda}_0^{k+1} > 0$, a matriz B^{k+1} simétrica e definida positiva, faça $k = k + 1$, $\ell = 0$, $\mathbf{y}_0^k = \mathbf{x}^k$ e vá para o Passo 1.

Passo 5: (Passo s3rio de m3xima descida) Faa

$$(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k) - \mu(z^k - f(\mathbf{x}^k))\nabla f(\mathbf{x}, z) \quad \text{e} \quad f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k). \quad (2.10)$$

Defina $\lambda_0^{k+1} > 0$, a matriz B^{k+1} sim3trica e definida positiva, faa $k = k + 1$, $\ell = 0$, $\mathbf{y}_0^k = \mathbf{x}^k$ e v3 para o Passo 1.

Passo 6: (Passo nulo) Faa $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k)$, calcule $\mathbf{s}_{\ell+1}^k \in \partial f(\mathbf{y}_{\ell+1}^k)$, um novo plano de corte e seu gradiente

$$g_{\ell+1}^k(\mathbf{x}^k, z^k) = f(\mathbf{y}_{\ell+1}^k) + (\mathbf{s}_{\ell+1}^k)^T(\mathbf{x}^k - \mathbf{y}_{\ell+1}^k) - z^k.$$

Considere

$$\nabla g_{\ell+1}^k(\mathbf{x}^k, z^k) = \begin{bmatrix} \mathbf{s}_{\ell+1}^k \\ -1 \end{bmatrix},$$

determine

$$\begin{aligned} \bar{\mathbf{g}}_{\ell+1}^k(\mathbf{x}^k, z^k) &= [g_0^k(\mathbf{x}^k, z^k), \dots, g_\ell^k(\mathbf{x}^k, z^k), g_{\ell+1}^k(\mathbf{x}^k, z^k)]^T \in \mathbb{R}^{\ell+2}, \quad \text{e} \\ \nabla \bar{\mathbf{g}}_{\ell+1}^k(\mathbf{x}^k, z^k) &= [\nabla g_0^k(\mathbf{x}^k, z^k), \dots, \nabla g_\ell^k(\mathbf{x}^k, z^k), \nabla g_{\ell+1}^k(\mathbf{x}^k, z^k)] \in \mathbb{R}^{(n+2) \times (\ell+2)}. \end{aligned}$$

Defina $\ell = \ell + 1$ e v3 para o Passo 2.

Os valores de λ e B devem satisfazer as seguintes hip3teses:

Hip3tese 2.1. *Existem n3meros positivos σ_1 e σ_2 tais que*

$$\sigma_1 \|\mathbf{d}\|^2 \leq \mathbf{d}^T B \mathbf{d} \leq \sigma_2 \|\mathbf{d}\|^2, \quad \text{para} \quad \mathbf{d} \in \mathbb{R}^{n+1}.$$

Hip3tese 2.2. *Existem n3meros positivos λ^I , λ^S , tais que $\lambda^I \leq \lambda_i \leq \lambda^S$, para $i = 0, 1, \dots, \ell$.*

2.2 Análise da Convergência

Nesta seção, realiza-se a demonstração da convergência global do presente algoritmo. Primeiramente, se mostra que a direção de busca \mathbf{d}_ℓ^k é uma direção de descida viável para a z . Então, prova-se que o número de passos nulos em cada iteração é finito. Isto é; Como $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$, depois de um número finito de sub-iterações, obtém-se $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$. Conseqüentemente, a sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ é limitada e está no interior do epígrafo de f . Então, mostra-se que qualquer ponto de acumulação da sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ é uma solução do problema (P.1). Tem-se ainda que $\|\mathbf{d}^k\| = \mathbf{0}$ somente num ponto estacionário e que $\|\mathbf{d}^k\| \rightarrow \mathbf{0}$ quando $k \rightarrow \infty$, fato este que justifica o critério de parada no passo 3).

Finalmente, mostra-se que para pontos de acumulação (\mathbf{x}^*, z^*) da sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$, a condição de otimalidade $\mathbf{0} \in \partial f(\mathbf{x}^*)$ é satisfeita.

Em alguns momentos, alguns índices serão omitidos para simplificar a notação.

Hipótese 2.3. *O conjunto $\{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \leq f(\mathbf{x}^0)\}$ é compacto.*

Hipótese 2.4. *Para todo $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ e todo i tal que $g_i(\mathbf{x}, z) = 0$ os vetores $\nabla g_i(\mathbf{x}, z)$ são linearmente independentes.*

Observa-se que as soluções \mathbf{d}_α , $\boldsymbol{\lambda}_\alpha$, \mathbf{d}_β , e $\boldsymbol{\lambda}_\beta$ dos sistemas lineares (2.3), (2.4), e (2.5), (2.6) são únicas. Este fato é uma consequência do lema provado em [52, 53] e enunciado como segue:

Lema 2.1. *Para qualquer vetor $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ e qualquer matriz positiva definida $B \in \mathbb{R}^{(n+1) \times (n+1)}$, a matriz*

$$\begin{bmatrix} B & \nabla \bar{g}(\mathbf{x}, z) \\ \Lambda[\nabla \bar{g}(\mathbf{x}, z)]^T & G(\mathbf{x}, z) \end{bmatrix},$$

é não singular.

E segue do resultado anterior que \mathbf{d}_α , \mathbf{d}_β , $\boldsymbol{\lambda}_\alpha$ e $\boldsymbol{\lambda}_\beta$ são limitados no conjunto definido na Hipótese 2.3. Como ρ é limitado superiormente, tem-se $\bar{\boldsymbol{\lambda}} = \boldsymbol{\lambda}_\alpha + \rho \boldsymbol{\lambda}_\beta$ é limitado.

Lema 2.2. *A direção \mathbf{d}_α satisfaz $\mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \leq -\mathbf{d}_\alpha^T B \mathbf{d}_\alpha$.*

Demonstração. Veja a demonstração do Lema 4.2. em [30]. □

Como consequência, tem-se que a direção de busca \mathbf{d}_α é de descida para a função objetivo $F(\mathbf{x}, z)$, (i.e., para (PA.1) e (PE.1)).

Lema 2.3. *A direção \mathbf{d} definida no Passo 2, item iii) do algoritmo é uma direção de descida para (PE.1) e (PA.1).*

Demonstração. Por definição $\mathbf{d} = \mathbf{d}_\alpha + \rho \mathbf{d}_\beta$, logo se pode escrever

$$d^T \nabla F(\mathbf{x}, z) = \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) + \rho \mathbf{d}_\beta^T \nabla F(\mathbf{x}, z).$$

No caso em que $\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z) > 0$, tem-se que $\rho \leq (\xi - 1) \frac{\mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z)}{\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z)}$, veja o Passo 2 do algoritmo. E como \mathbf{d}_α é uma direção de descida para F , pelo Lema 2.2, obtém-se

$$\begin{aligned} \mathbf{d}^T \nabla F(\mathbf{x}, z) &\leq \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) + (\xi - 1) \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \\ &= \xi \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \leq 0. \end{aligned}$$

(note que $\mathbf{d}^T \nabla F(\mathbf{x}, z) = 0$ somente se $\mathbf{d}_\alpha = 0$). Agora, supondo que $\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z) \leq 0$, tem-se que a inequação $\mathbf{d}^T \nabla F(\mathbf{x}, z) \leq \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) < 0$ é verdadeira. Portanto, \mathbf{d} é uma direção de descida para F . \square

Embora a direção \mathbf{d} , calculada no Passo 2 do algoritmo, seja de descida para (PE.1) e (PA.1), isso não significa necessariamente que ela seja de descida para (P.1). Na verdade, no próximo Lema mostra-se que o algoritmo é descida em relação a função objetivo f . Em seguida, prova-se que, quando o ponto de iteração corrente está suficientemente próximo da fronteira do epígrafo de f , a direção \mathbf{d} é também de descida para o problema (P.1).

Lema 2.4. *Seja $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$ o ponto na k -ésima iteração do algoritmo. Para $k \geq 1$, tem-se*

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) \quad e \quad z^{k+1} < z^k.$$

Além disso, o próximo iterado $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$.

Demonstração. A atualização do ponto (\mathbf{x}^k, z^k) é feita no Passo 4 ou Passo 5 do algoritmo. No Passo 4, faz-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ somente se,

$$f(\mathbf{y}_{\ell+1}^k) \leq w_{\ell+1}^k \quad e \quad f(\mathbf{x}^k) \geq f(\mathbf{y}_{\ell+1}^k)$$

portanto após a atualização, tem-se que $f(\mathbf{x}^{k+1}) \leq z^{k+1}$ e $f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k)$.

Além disso, tem-se que $(\mathbf{d}^k)^T \nabla F(\mathbf{x}, z) < 0$ pelo Lema 2.3 de onde $d_z^k < 0$ e como o próximo componente z^{k+1} é calculado pela formula

$$z^{k+1} = z^k + \mu t^k d_z^k,$$

com $\mu, t^k > 0$, segue que $z^{k+1} \leq z^k$.

Por outro lado, no Passo 5, usa-se a direção de descida rápida $-\mathbf{e}_z$ como direção de busca. Logo, da equação (2.10) segue $\mathbf{x}^{k+1} = \mathbf{x}^k$ e $z^{k+1} = z^k - \mu(z^k - f(\mathbf{x}^k))$,

para $\mu \in [0, 1]$ e $z^k - f(x^k) > 0$. De onde, novamente tem-se $z^{k+1} \leq z^k$ e $f(\mathbf{x}^{k+1}) = f(x^k) < z^{k+1}$. \square

Lema 2.5. *A sequencia $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ gerada pelo algoritmo é limitada.*

Demonstração. A sequencia $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ é limitada, pois $z^{k+1} < z^k$ para todo k e por ela estar contida no conjunto limitado dado por

$$\text{epi}(f) \cap \{(\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid z < z^0\}. \quad \square$$

Lema 2.6. *Seja $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$ um ponto suficientemente próximo da fronteira do epígrafo de f (i.e. $z^k - f(\mathbf{x}^k) < \mu t^k \mathbf{d}_z^k$). Se (\mathbf{x}^k, z^k) não é um ponto estacionário, então a direção \mathbf{d}^k definida em (2.8) é uma direção de descida para o problema (P.1).*

Demonstração. Como $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$, segue que $z^k = f(\mathbf{x}^k) + \epsilon_1$ para algum $\epsilon_1 \geq 0$. Também tem-se que $d_z^k < 0$ pelo Lema 2.3. O próximo iterado em z é calculado pela fórmula $z^{k+1} = z^k + \mu t^k d_z^k$ com $\mu, t^k > 0$. Logo, se pode escrever $z^{k+1} = z^k - \epsilon_2 = (f(\mathbf{x}^k) + \epsilon_1) - \epsilon_2$, onde $\epsilon_2 = -\mu t^k d_z^k > 0$. Quando ϵ_1 é suficientemente pequeno (i.e. $\epsilon_1 < \epsilon_2$) obtém-se que $z^{k+1} - f(\mathbf{x}^k) < 0$.

Agora $f(\mathbf{x}^{k+1}) \leq z^{k+1}$, pois $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$ pelo Lema 2.4.

Portanto combinando-se os resultados obtém-se que $f(\mathbf{x}^{k+1}) \leq z^{k+1} < f(\mathbf{x}^k)$ e \mathbf{d}^k é uma direção de descida para f por definição. \square

Lema 2.7. *A direção \mathbf{d} definida em (2.8) é limitada.*

Demonstração. A escolha de ρ (veja (2.7)) assegura que se tenha

$$\rho \leq \varphi \|\mathbf{d}_\alpha\|^2 \quad \text{para algum } \varphi > 0. \quad (2.11)$$

Por outro lado, do Lema 2.2 e da Hipótese 2.1, obtém-se que

$$-\mathbf{d}_\alpha^T \nabla f(\mathbf{x}, z) \geq \sigma_1 \|\mathbf{d}_\alpha\|^2,$$

e portanto, devido a (2.7),

$$\rho \geq \min \left\{ \varphi, \frac{(1 - \xi)\sigma_1}{\mathbf{d}_\beta^T \nabla f(\mathbf{x}, z)} \right\} \|\mathbf{d}_\alpha\|^2,$$

se $\mathbf{d}_\beta^T \nabla f(\mathbf{x}, z)$. Devido a (2.7) e como \mathbf{d}_β é limitada, existe um limite inferior $\varphi_{low} > 0$ tal que

$$\rho \geq \varphi_{low} \|\mathbf{d}_\alpha\|^2.$$

Portanto, pela limitação de \mathbf{d}_α , a deflexão dada por ρ é positiva limitada como acima. De (2.8) e (2.11), tem-se

$$\begin{aligned}\|\mathbf{d}\| &= \|\mathbf{d}_\alpha + \rho\mathbf{d}_\beta\| \\ &\leq \|\mathbf{d}_\alpha\| + \|\rho\mathbf{d}_\beta\| \\ &\leq \|\mathbf{d}_\alpha\| + \varphi\|\mathbf{d}_\alpha\|^2\|\mathbf{d}_\beta\| \\ &= (1 + \varphi\|\mathbf{d}_\alpha\| \cdot \|\mathbf{d}_\beta\|)\|\mathbf{d}_\alpha\|.\end{aligned}$$

Portanto, existe $\delta > 1$ tal que $\|\mathbf{d}\| \leq \delta\|\mathbf{d}_\alpha\|$ é válida. E devido a limitação de \mathbf{d}_α , temos que \mathbf{d} é limitada como acima. \square

O Lema seguinte é apenas um exercício que será repetido aqui.

Lema 2.8. *Seja $X \subset \mathbb{R}^n$ um conjunto convexo. Considere $\mathbf{x}_0 \in \text{int } X$ e $\bar{\mathbf{x}} \in X$. Seja $\{\bar{\mathbf{x}}^k\}_{k \in \mathbb{N}} \subset \mathbb{R}^n \setminus X$ uma sequencia, tal que $\bar{\mathbf{x}}^k \rightarrow \bar{\mathbf{x}}$. Seja $\{\mathbf{x}^k\}_{k \in \mathbb{N}} \subset \mathbb{R}^n$ uma sequencia definida por $\mathbf{x}^k = \mathbf{x}_0 + \mu(\bar{\mathbf{x}}^k - \mathbf{x}_0)$ com $\mu \in (0, 1)$. Então existe $k_0 \in \mathbb{N}$ tal que $\mathbf{x}^k \in \text{int } X$, para todo $k > k_0$.*

Demonstração. Observa-se que $\mathbf{x}^k = \mathbf{x}_0 + \mu(\bar{\mathbf{x}}^k - \mathbf{x}_0) \rightarrow \mathbf{x}_0 + \mu(\bar{\mathbf{x}} - \mathbf{x}_0) = \mathbf{x}_\mu$ quando $k \rightarrow \infty$. Como o segmento $[\mathbf{x}_0, \bar{\mathbf{x}}] \subset X$ e $\mu \in (0, 1)$ temos que $\mathbf{x}_\mu \in \text{int } X$ e, como consequência existe $\delta > 0$ tal que $B(\mathbf{x}_\mu, \delta) \subset \text{int } X$. Como $\mathbf{x}^k \rightarrow \mathbf{x}_\mu$ existe $k_0 \in \mathbb{N}$ tal que $\mathbf{x}^k \in B(\mathbf{x}_\mu, \delta) \subset \text{int } X$, para todo $k > k_0$. \square

OBS 2.1: A sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ definida em (2.1) para k fixo, está num conjunto limitado. De fato, pelo Lema 2.5, existe $r > 0$ tal que $\|(\mathbf{x}, z)\| < r$ para todo ponto no conjunto dado por $\text{int}(\text{epi } f) \cap \{(\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid z < z^0\}$. Então para qualquer passo sério tem-se $\|(\mathbf{x}^k, z^k)\| < r$. Para a sequencia de baixo, dada por $(\mathbf{x}_\ell^k, z_\ell^k) = (\mathbf{x}^k, z^k) + t\mathbf{d}$ com t limitado por t_{\max} e $\|\mathbf{d}\|$ limitado por um valor D , segue que, pela desigualdade triangular $\|(\mathbf{x}_\ell^k, z_\ell^k)\| \leq \|(\mathbf{x}^k, z^k)\| + Dt_{\max}$, então a sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ está numa bola centrada na origem e raio $r + Dt_{\max}$.

Proposição 2.9. *Considere a sequencia $\{(\mathbf{x}_i^k, z_i^k)\}_{i \in \mathbb{N}}$ definida em (2.1) para k fixado. Se $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ é um ponto de acumulação desta sequencia, então $\tilde{z}^k = f(\tilde{\mathbf{x}}^k)$.*

Demonstração. Pela construção da sequencia $\{(\mathbf{x}_i^k, z_i^k)\}_{i \in \mathbb{N}}$, segue que, num ponto de acumulação tem-se que $f(\tilde{\mathbf{x}}^k) \geq \tilde{z}^k$. Logo, suponha que $\tilde{z}^k < f(\tilde{\mathbf{x}}^k)$ e considere uma subsequencia convergente $\{(\mathbf{x}_i^k, z_i^k)\}_{i \in \mathbb{N}'} \rightarrow (\tilde{\mathbf{x}}^k, \tilde{z}^k)$ tal que $\{\mathbf{s}_i^k\}_{i \in \mathbb{N}'} \rightarrow \tilde{\mathbf{s}}^k$, onde $\mathbb{N}' \subset \mathbb{N}$. Estas sequencias existem porque, tanto $\{(\mathbf{x}_i^k, z_i^k)\}_{i \in \mathbb{N}'}$ como $\{\mathbf{s}_i^k\}_{i \in \mathbb{N}'}$ estão num conjunto compacto pela Hipótese 2.3. O plano de corte correspondente é representado por $f(\mathbf{x}_i^k) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}_i^k) - z = 0$. Então, $z(\tilde{\mathbf{x}}^k) = f(\mathbf{x}_i^k) + \mathbf{s}_i^T(\tilde{\mathbf{x}}^k - \mathbf{x}_i^k)$ é a projeção vertical de $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ no plano de corte. Tomando o limite quando $i \rightarrow \infty$,

tem-se que $z(\tilde{\mathbf{x}}^k) = f(\tilde{\mathbf{x}}^k)$. Assim, para $i \in \mathbb{N}' > L$ suficientemente grande, $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ está sobre o i -ésimo plano de corte. O que contradiz a suposição inicial. \square

Proposição 2.10. *Seja $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$. O próximo iterado $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$ é obtido após um número finito de sub-iterações.*

Demonstração. A demonstração inicia com a observação de que no passo 4 do algoritmo, tem-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ somente se, $w_{\ell+1}^k > f(\mathbf{y}_{\ell+1}^k)$ (i.e., num *passo sério*), e consequentemente $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$.

Então resta mostrar que este ponto é encontrado após um número finito de iterações. A sequência $\{(\mathbf{x}_i^k, z_i^k)\}_{i \in \mathbb{N}}$ é limitada por construção e, pela Proposição 2.9, ele possui um ponto de acumulação $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ tal que $\tilde{z}^k = f(\tilde{\mathbf{x}}^k)$.

Considerando a sequência definida em (2.2),

$$(\mathbf{y}_i^k, w_i^k) = (\mathbf{x}^k, z^k) + \mu \left\| (\mathbf{x}_i^k, z_i^k) - (\mathbf{x}^k, z^k) \right\|, \quad \mu \in (0, 1)$$

segue do Lema 2.8, que existe $i_0 \in \mathbb{N}$ tal que $(\mathbf{y}_i^k, w_i^k) \in \text{int}(\text{epi } f)$, for $i > i_0$. Mas esta é exatamente a condição para um *passo sério*, o que completa a demonstração. \square

Lema 2.11. *Existe $\tau > 0$ tal que $\bar{\mathbf{g}}_\ell(\mathbf{x}, z) + t\mathbf{d} \leq 0$, $\forall t \in [0, \tau]$ para qualquer $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ e qualquer direção \mathbf{d} dada pelo algoritmo.*

Demonstração. Denote por \mathbf{b} o vetor tal que $b_i = \mathbf{s}_i^T \mathbf{x}_i - f(\mathbf{x}_i)$ for all $i = 0, 1, \dots, \ell$. Então, $\tilde{\mathbf{g}}_\ell(\mathbf{x}, z) + t\mathbf{d} = (\nabla \bar{\mathbf{g}}_\ell(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{b}$, pois

$$\begin{aligned} g_i(\mathbf{x}, z) &= f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i) - z \\ &= f(\mathbf{y}_i) + \mathbf{s}_i^T \mathbf{x} - \mathbf{s}_i^T \mathbf{y}_i - z \\ &= (\mathbf{s}_i^T, -1)^T(\mathbf{x}, z) - \mathbf{s}_i^T \mathbf{y}_i + f(\mathbf{y}_i) \\ &= (\nabla g_i(\mathbf{x}, z))^T(\mathbf{x}, z) - b_i \end{aligned}$$

para todo $i = 0, 1, \dots, \ell$. O tamanho de passo t é definido na equação (2.9) no Passo 3) do algoritmo. Como a restrição do problema auxiliar (PA.1) é linear, para que a condição de busca linear do lema seja satisfeita, a seguinte inequação deve ser verificada:

$$\begin{aligned} g_i((\mathbf{x}, z) + t_i \mathbf{d}) &= (\nabla g_i((\mathbf{x}, z) + t_i \mathbf{d}))^T((\mathbf{x}, z) + t_i \mathbf{d}) - b_i \\ &= (\nabla g_i((\mathbf{x}, z)))^T((\mathbf{x}, z) + t_i \mathbf{d}) - b_i \\ &= (\nabla g_i((\mathbf{x}, z)))^T(\mathbf{x}, z) - b_i + t_i (\nabla g_i((\mathbf{x}, z)))^T \mathbf{d} \\ &= g_i(\mathbf{x}, z) + t_i (\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0, \quad \text{para todo } i = 0, 1, \dots, \ell. \end{aligned} \tag{2.12}$$

Se $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0$ a iniquação é satisfeita para todo $t > 0$. Caso contrário, segue de iii) no Passo 2) que, $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} = (\nabla g_i(\mathbf{x}, z))^T (\mathbf{d}_\alpha + \rho \mathbf{d}_\beta)$.

Mas de (2.4) e (2.6) tem-se que

$$(\nabla g_i(\mathbf{x}, z))^T \mathbf{d}_\alpha = -g_i(\mathbf{x}, z) \frac{\lambda_{\alpha,i}}{\lambda_i},$$

e

$$(\nabla g_i(\mathbf{x}, z))^T \mathbf{d}_\beta = -1 - g_i(\mathbf{x}, z) \frac{\lambda_{\beta,i}}{\lambda_i}.$$

Então (2.12) é equivalente a

$$g_i(\mathbf{x}, z) \left(1 - t_i \frac{\bar{\lambda}_i}{\lambda_i}\right) - \rho t_i \leq 0.$$

Como $\rho t_i > 0$, a última inequação será satisfeita quando $t_i \leq \lambda_i / \bar{\lambda}_i$.

Por construção, $\lambda > 0$ é limitado, $\bar{\lambda}$ é limitado superiormente. Logo, existe $0 < \tau < t_{max}/\mu$ tal que $\tau < \lambda_i / \bar{\lambda}_i$ para todo $i = 0, 1, \dots, \ell$. Portanto, para todo $t \in [0, \tau]$ a condição de busca linear $g_i((\mathbf{x}, z) + t\mathbf{d}) \leq 0$ está satisfeita para todo $i = 0, 1, \dots, \ell$. \square

Lema 2.12. *Seja \mathbf{d}_α^* um ponto de acumulação da sequência $\{\mathbf{d}_\alpha^k\}_{k \in \mathbb{N}}$. Então $\mathbf{d}_\alpha^* = 0$.*

Demonstração. Do Passo 3) do algoritmo, tem-se que $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k) + \mu t^k (\mathbf{d}_x^k, d_z^k)$, logo

$$z^{k+1} = z^k + \mu t^k d_z^k. \quad (2.13)$$

A sequência $\{z^k\}_{k \in \mathbb{N}}$ é decrescente e limitada pela Hipótese 2.3. Agora, denota-se por $z^* = \lim_{k \rightarrow \infty} z^k$ e $\mathbb{N}' \subset \mathbb{N}$ tal que $\{t^k\}_{k \in \mathbb{N}'} \rightarrow t^*$. E segue do Lema 2.11 que $t^* > 0$. Quando $k \rightarrow \infty$, $k \in \mathbb{N}'$ na equação (2.13) tem-se que $z^* = z^* + \mu t^* d_z^*$, e portanto $d_z^* = 0$. Da Proposição 2.3 segue que

$$0 = d_z^* = (\mathbf{d}_\alpha^*)^T \nabla F(\mathbf{x}, z) \leq \xi (\mathbf{d}_\alpha^*)^T \nabla F(\mathbf{x}, z) = \xi d_{\alpha z}^* \leq 0,$$

para algum $\xi \in (0, 1)$ e portanto $d_{\alpha z}^* = 0$. Além disso, pelo Lema 2.2, tem-se que

$$0 = \mathbf{d}_{\alpha z}^* = (\mathbf{d}_\alpha^*)^T \nabla F(\mathbf{x}, z) \leq -(\mathbf{d}_\alpha^*)^T B \mathbf{d}_\alpha^* \leq 0,$$

e como B é definida positiva, tem-se que $\mathbf{d}_\alpha^* = 0$. \square

Segue do resultado anterior que $\mathbf{d}^k = \mathbf{d}_\alpha^k + \rho_k \mathbf{d}_\beta^k \rightarrow 0$ quando $k \rightarrow \infty$, $k \in \mathbb{N}'$, pois $\rho_k \rightarrow 0$ se $\mathbf{d}_\alpha^k \rightarrow 0$.

Lema 2.13. *Seja $(\mathbf{s}^T, -1)^T$ o gradiente de uma restrição ativa num ponto de acumulação (\mathbf{x}^*, z^*) da sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$. Então $\mathbf{s} \in \partial f(\mathbf{x}^*)$.*

Demonstração. Note que no ponto de acumulação (\mathbf{x}^*, z^*) , tem-se que $f(\mathbf{x}^*) = z^*$, logo a primeira restrição $g_0(\mathbf{x}^*, z^*)$ é ativa e $\mathbf{s}_0 \in \partial f(\mathbf{x}^*)$ por construção (veja o Passo 1 do algoritmo).

Suponha que a restrição $g_i(\mathbf{x}, z) = f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) - z$ é ativa em (\mathbf{x}^*, z^*) , para $i > 1$ e $\mathbf{s}_i \in \partial f(\mathbf{y}_i)$. Então $g_i(\mathbf{x}^*, z^*) = f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x}^* - \mathbf{y}_i) - z^* = 0$.

Como f é convexa, se pode escrever

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) \\ &= f(\mathbf{y}_i) - f(\mathbf{x}^*) + f(\mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) - (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*) \\ &= f(\mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*) + [f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x}^* - \mathbf{y}_i) - f(\mathbf{x}^*)] \\ &= f(\mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*) + g_i(\mathbf{x}^*, z^*), \quad \text{pois } f(\mathbf{x}^*) = z^* \\ &= f(\mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*), \quad \text{pois } g_i(\mathbf{x}^*, z^*) = 0. \end{aligned}$$

Logo $f(\mathbf{x}) \geq f(\mathbf{x}^*) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{x}^*)$, e portanto $\mathbf{s}_i \in \partial f(\mathbf{x}^*)$ □

Proposição 2.14. *Para qualquer ponto de acumulação (\mathbf{x}^*, z^*) da sequencia $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$, tem-se que $\mathbf{0} \in \partial f(\mathbf{x}^*)$.*

Demonstração. Considere a sequencia

$$\mathbb{Y} := \{\mathbf{y}_0^0, \mathbf{y}_1^0, \dots, \mathbf{y}_{\ell^0}^0, \mathbf{y}_0^1, \mathbf{y}_1^1, \dots, \mathbf{y}_{\ell^1}^1, \dots, \mathbf{y}_0^k, \mathbf{y}_1^k, \dots, \mathbf{y}_{\ell^k}^k, \dots\},$$

de todos os pontos obtidos nas sub-iterações do algoritmo. Como esta sequencia é limitada, se pode encontrar subsequencias convergentes. Em particular, como $\mathbf{x}^{k+1} = \mathbf{y}_{\ell^k}^k$, pode-se afirmar que existe $\bar{\mathbb{Y}} \subset \mathbb{Y}$ que converge para \mathbf{x}^* .

Considere $\bar{\mathbb{Y}}^k \equiv \bar{\mathbb{Y}} \cap \{\mathbf{y}_0^k, \mathbf{y}_1^k, \dots, \mathbf{y}_{\ell^k}^k\}$. Então segue de (2.3) no Passo 2) do algoritmo, que

$$\lim_{k \rightarrow \infty} \sum_{i=0}^{\ell^k} \lambda_{\alpha i}^k s_i^k = 0 \quad \text{e} \quad \lim_{k \rightarrow \infty} \sum_{i=0}^{\ell^k} \lambda_{\alpha i}^k = 1. \quad (2.14)$$

Defina $\mathbb{I}^k = \{i \mid \mathbf{y}_i^k \in \bar{\mathbb{Y}}^k\}$. Consequentemente $\lambda_{\alpha i}^k > 0$ para $i \in \mathbb{I}^k$ e k suficientemente grande. Então,

$$\lim_{k \rightarrow \infty} \sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k s_i^k = 0 \quad \text{and} \quad \lim_{k \rightarrow \infty} \sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k = 1. \quad (2.15)$$

Considere o ponto auxiliar \mathbf{y}_i^k e o subgradiente $\mathbf{s}_i^k \in \partial f(\mathbf{y}_i^k)$ tal que $i \in \mathbb{I}^k$. Logo, pela definição de subdiferencial [16], pode-se escrever

$$f(\mathbf{x}) \geq f(\mathbf{y}_i^k) + (\mathbf{s}_i^k)^T(\mathbf{x} - \mathbf{y}_i^k) = f(\mathbf{x}^k) + (\mathbf{s}_i^k)^T(\mathbf{x} - \mathbf{x}^k) - \varepsilon_k^i,$$

onde $\varepsilon_k^i = f(\mathbf{x}^k) - f(\mathbf{y}_i^k) - (\mathbf{s}_i^k)^T(\mathbf{x}^k - \mathbf{y}_i^k)$. Como $\mathbf{x}^k \rightarrow \mathbf{x}^*$ e $\mathbf{y}_i^k \rightarrow \mathbf{x}^*$ para $i \in \mathbb{I}^k$, tem-se que $\lim_{k \rightarrow \infty} \varepsilon_k^i = 0$. Portanto, $\mathbf{s}_i^k \in \partial_{\varepsilon_k^i} f(\mathbf{x}^k)$, onde $\partial_{\varepsilon} f(\mathbf{x})$ representa o ε -subdiferencial de f , [16]. Note que se pode escrever

$$\left(\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k \right) f(\mathbf{x}) \geq \left(\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k \right) f(\mathbf{x}^k) + \left(\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k \mathbf{s}_i^k \right)^T (\mathbf{x} - \mathbf{x}^k) - \sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k \varepsilon_k^i,$$

e

$$f(\mathbf{x}) \geq f(\mathbf{x}^k) + \left(\sum_{i \in \mathbb{I}^k} \frac{\lambda_{\alpha i}^k}{\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k} \mathbf{s}_i^k \right)^T (\mathbf{x} - \mathbf{x}^k) - \varepsilon_k,$$

onde $\varepsilon_k = \frac{\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k \varepsilon_k^i}{\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k}$. Como

$$\lim_{k \rightarrow \infty} \varepsilon_k^i = 0 \quad \text{e} \quad \lim_{k \rightarrow \infty} \sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k = 1,$$

tem-se que $\lim_{k \rightarrow \infty} \varepsilon_k = 0$. Agora, considerando $\mathbf{s}^k = \sum_{i \in \mathbb{I}^k} \frac{\lambda_{\alpha i}^k}{\sum_{i \in \mathbb{I}^k} \lambda_{\alpha i}^k} \mathbf{s}_i^k$. Segue da equação (2.15) que $\mathbf{s}^k \in \partial_{\varepsilon_k} f(\mathbf{x}^k)$ e que $\mathbf{0} \in \partial f(\mathbf{x}^*)$.

□

2.3 Resultados Numéricos

O algoritmo NFDA foi implementado em MATLAB e os experimentos numéricos foram realizados num microcomputador com 2GB de RAM para um conjunto fixado de parâmetros.

Na seção 2.3.1 utiliza-se uma coleção bem conhecida de problemas testes que podem ser encontrados em [13] ou [54]. Para realizar comparações de performance, foram considerados alguns solvers clássicos. Por fim, na Seção 2.3.2 realiza-se uma análise dos resultados obtidos, seguindo a metodologia do traçamento do perfil de desempenho de cada método, introduzida por Dolan e Moré [55].

2.3.1 Testes Numéricos

Na Tabela 2.1 estão os problemas teste considerados para analisar a performance do algoritmo. Estes problemas estão descritos em detalhes no Apêndice A, ou nos trabalhos [54] ou [13].

Os parâmetros utilizados nos testes são:

$$B = I, \quad \mu = 0.75, \quad \varphi = 0.1, \quad \xi = 0.7, \quad \text{and} \quad t_{\max} = 1.$$

Tabela 2.1: Tabela de Problemas

No.	Problema	n	f^{opt}
1	CB2	2	1.9522245
2	CB3	2	2
3	DEM	2	-3
4	QL	2	7.20
5	LQ	2	-1.4142136
6	Mifflin1	2	-1
7	Rosen	4	-44
8	Shor	5	22.600162
9	Maxquad	10	-0.8414083
10	Maxq	20	0
11	Maxl	20	0
12	TR48	48	-638565.0
13	Goffin	50	0

Tabela 2.2: Solvers utilizados na comparação

Solver	Autor(s)	Método	Referência
M1FC1	Lemarechal	Steepest Descent	[56]
BTC	Schramm & Zowe	Bundle Trust	[19]
PB	Makela	Proximal Bundle	[13]
SBM	Lukšan & Vlček	Standard Bundle	[42]

O algoritmo para quando $\|d_\alpha^k\| \leq \varepsilon = 10^{-4}$. Este fato está justificado na Proposição 2.12. A atualização para o vetor de multiplicadores λ se da de acordo com a seguinte formulação: Considere, para todo $i = 0, 1, \dots, \ell$,

$$\lambda_i := \max [\lambda_{\alpha i}; \epsilon \|d_\alpha\|^2], \quad \epsilon > 0. \quad (2.16)$$

Com o intuito de melhorar os resultados numéricos em cada iteração k , adiciona-se ao conjunto de planos de corte definido pelo algoritmo os últimos $5n$ planos previamente computados a k -ésima iteração. Ou seja, para a sub-iteração ℓ o algoritmo trabalha com no máximo $\ell + 5n$ planos de corte.

O algoritmo foi comparado com os solvers citados na Tabela 2.2.

Para os resultados apresentados nas Tabelas 2.3, 2.4 e 2.5, considera-se a seguinte notação:

NI - o número de iterações,
 NF - o número de avaliações da função e subgradientes computados,
 f^* - o valor ótimo (conhecido) da função,
 f - o valor encontrado pelo algoritmo.

Tabela 2.3: Resultados para o número de iterações

Solver		SBM	M1FC1	BTC	PB	NFDA
Problema	n	NI	NI	NI	NI	NI
CB2	2	31	11	13	15	13
CB3	2	14	12	13	15	24
DEM	2	17	10	09	07	22
QL	2	13	12	12	17	27
LQ	2	11	16	10	14	15
Mifflin1	2	66	143	49	22	21
Rosen	4	43	22	22	40	49
Shor	5	27	21	29	26	51
Maxquad	10	74	29	45	41	115
Maxq	20	150	144	125	158	272
Maxl	20	39	138	74	34	70
TR48	48	245	163	165	152	317
Goffin	50	52	72	51	51	79

Tabela 2.4: Resultados para o número de avaliações da função

Solver		SBM	M1FC1	BTC	PB	NFDA
Problema	n	NF	NF	NF	NF	NF
CB2	2	33	31	16	16	14
CB3	2	16	44	21	16	25
DEM	2	19	33	13	08	22
QL	2	15	30	17	18	28
LQ	2	12	52	11	15	16
Mifflin1	2	68	281	74	23	22
Rosen	4	45	61	32	41	50
Shor	5	29	71	30	27	52
Maxquad	10	75	69	56	42	116
Maxq	20	151	207	128	159	273
Maxl	20	40	213	84	35	71
TR48	48	251	284	179	153	318
Goffin	50	53	94	53	52	80

Tabela 2.5: Resultados para o valor da função objetivo

Solver		SBM	M1FC1	BTC	PB	NFDA	
Problema	n	f	f	f	f	f	f^*
CB2	2	1.95222	1.95225	1.95222	1.95222	1.9522	1.95222
CB3	2	2.00000	2.00141	2.00000	2.00000	2.0001	2
DEM	2	-3.00000	-3.00000	-3.00000	-3.00000	-2,9863	-3
QL	2	7.20000	7.20001	7.20000	7.20001	7.2000	7.2
LQ	2	-1.41421	-1.14142	-1.41421	-1.41421	-1.4139	-1.41421
Mifflin1	2	-0.99999	-0.99996	-1.00000	-1.00000	-0.9998	-1
Rosen	4	-43.99999	-43.99998	-43.99998	-43.99994	-43.9999	-44
Shor	5	22.60016	22.60018	22.60016	22.60016	22.6002	22.60016
Maxquad	10	-0.84140	-0.84135	-0.84140	-0.84140	-0.8414	-0.84140
Maxq	20	0.16712e-06	0.00000	0.00000	0.00000	3.1780e-7	0
Maxl	20	0.12440e-12	0.00000	0.00000	0.00000	2.8315e-4	0
TR48	48	-638530.48	-633625.5	-638565.0	-63856	-638564.99	-638565
Goffin	50	0.11665e-11	0.00010	0.00000	0.00000	1.3372e-4	0

Pelas tabelas acima, quando se compara os resultados do algoritmo NFDA com os dos solvers da Tabela 2.2 percebe-se que o número de iterações, o número de avaliações da função e o valor final da função objetivo obtido pelo novo método é compatível com os resultados apresentados por estes algoritmos já estabelecidos na literatura.

Contudo, ressaltamos que o novo algoritmo é simples de ser implementado e resolve os problemas de forma robusta e eficiente, no sentido de que foram utilizados os mesmos parâmetros para resolver todos os problemas.

E como alternativa, a nova metodologia apresenta a diferença de não precisar resolver subproblemas de programação quadrática. Este fato, alude a possibilidade de vantagens, como por exemplo, a diminuição do tempo computacional e outros fatos que são abordados na conclusão.

2.3.2 Performance Profile

Para uma boa apresentação visual da comparação entre a performance dos solvers, utilizou-se o *performance profiles*, [55]. Como medidas de performance, foram consideradas o número de iterações e o número de avaliações da função.

As Figuras 2.1 e 2.2 mostram o performance profiles para os solvers considerados utilizando as medidas mencionadas. O performance profile [55], compara a performance de n_s solvers de um conjunto S com respeito a solução de n_p problemas de um conjunto P , utilizando algumas medidas de performance tais como número de iterações, número de avaliações da função, tempo de CPU, etc.

$t_{p,s}$ denota a medida de performance requerida para resolver o problema p com o solver s . Para cada problema p e solver s , define-se o *performance ratio value*

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}$$

se o problema p é resolvido pelo solver s ; caso contrário, $r_{p,s} = r_M$, onde r_M é um parâmetro suficientemente grande.

A função de distribuição $\rho_s : \mathbb{R} \rightarrow [0, 1]$, para o *performance ratio* $r_{p,s}$ representa a performance total do solver s para resolver o conjunto de problemas teste P . Esta função é definida por

$$\rho_s(\tau) = \frac{1}{n_p} |\{p : r_{p,s} \leq \tau\}|,$$

onde $|\{\cdot\}|$ é o número de elementos no conjunto $\{\cdot\}$. O valor de $\rho_s(1)$ indica a probabilidade do solver s ser o melhor método (entre todos os que pertencem ao conjunto S), utilizando $t_{p,s}$ como medida de performance.

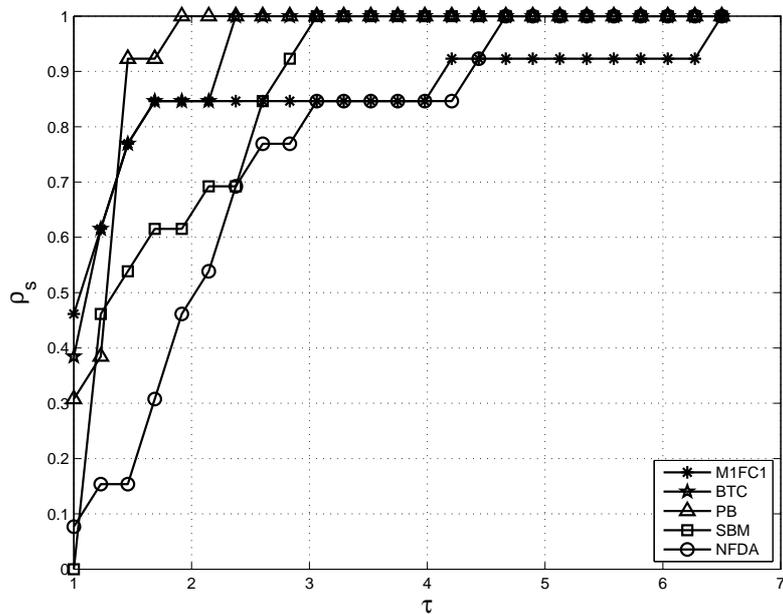


Figura 2.1: Performance Profiles: número de iterações

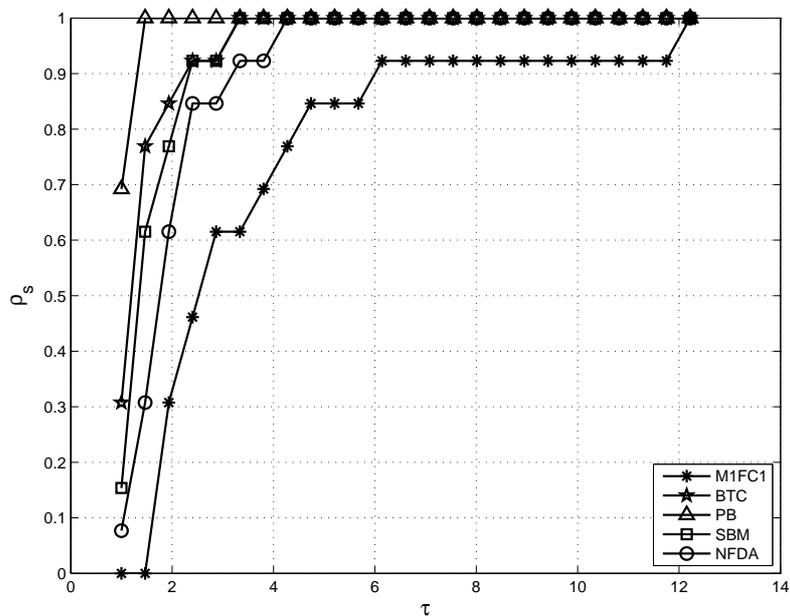


Figura 2.2: Performance Profiles: número de avaliações da função

Capítulo 3

Algoritmo para Otimização Não Convexa e Não Diferenciável

Neste Capítulo propõe-se um algoritmo para resolver problemas de otimização irrestrita com funções não necessariamente diferenciáveis ou convexas. Considera-se o seguinte problema

$$\begin{cases} \text{minimize } f(\mathbf{x}) \\ \text{sujeito a } \mathbf{x} \in \mathbb{R}^n, \end{cases} \quad (\text{P.2})$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função localmente Lipschitz contínua.

O método proposto para resolver (P.2), combina o tradicional método de planos de corte de Kelley [10, 57], com o FDIPA, [51], apresentado na Seção 1.7 do Capítulo 1. E ainda, algumas ideias dos métodos de feixe ([11, 13, 19]) são utilizadas. De fato, usa-se o passo sério e nulo e armazenam-se os planos de corte num feixe. Neste trabalho realiza-se uma extensão para o caso não convexo do algoritmo apresentado no Capítulo 2.

A teoria dos gradientes generalizados com Clarke [16] se mostrou um bom instrumento para ir além e generalizar as ideias quando se está lidando com funções não diferenciáveis e não convexas, (veja [13] ou [11]). Contudo, generalizar o método de planos de corte [10, 57] para o caso não convexo não é uma tarefa simples, mas é aparente que algumas das ideias válidas no caso convexo também sejam úteis no caso não convexo. Por exemplo, em [20], a não convexidade é superada construindo-se aproximações poliedrais inferiores e superiores da função objetivo, e nos métodos de feixe, a forma mais comum de lidar com as dificuldades causadas pela não convexidade é usar o assim chamado subgradiente localmente mensurado ao invés do erro de linearização (veja [11, 13, 19]). No presente trabalho, o emprego direto de um novo plano de corte pode, no caso não convexo, cortar o ponto de iteração atual e, portanto, algumas regras adicionais para a utilização dos planos de corte são necessários.

O problema auxiliar (PA.2) não será resolvido, na verdade é possível que ele nem tenha solução. Então, como no FDIPA (veja Capítulo 1.7), para cada problema auxiliar (PA.2), o algoritmo gera uma direção de descida viável $\mathbf{d}^k = (\mathbf{d}_x^k, \mathbf{d}_z^k)$ da seguinte forma:

Resolve-se os dois sistemas lineares:

$$\begin{cases} B^k \mathbf{d}_\alpha + \nabla \bar{g}_\ell^k(\mathbf{x}, z) \boldsymbol{\lambda}_\alpha = -\mathbf{e}_z \\ \Lambda^k [\nabla \bar{g}_\ell^k(\mathbf{x}, z)]^T \mathbf{d}_\alpha + G^k(\mathbf{x}, z) \boldsymbol{\lambda}_\alpha = 0 \end{cases}$$

e

$$\begin{cases} B^k \mathbf{d}_\beta + \nabla \bar{g}_\ell^k(\mathbf{x}, z) \boldsymbol{\lambda}_\beta = 0 \\ \Lambda^k [\nabla \bar{g}_\ell^k(\mathbf{x}, z)]^T \mathbf{d}_\beta + G^k(\mathbf{x}, z) \boldsymbol{\lambda}_\beta = -\Lambda^k \end{cases}$$

onde

$$\begin{aligned} \mathbf{e}_z &= [0, 0, \dots, 0, 1]^T = \nabla F(\mathbf{x}, z) \in \mathbb{R}^{n+1} \\ \boldsymbol{\lambda}_\alpha^k &:= (\lambda_{\alpha,1}^k, \dots, \lambda_{\alpha,\ell}^k), & \boldsymbol{\lambda}_\beta^k &:= (\lambda_{\beta,1}^k, \dots, \lambda_{\beta,\ell}^k), & \boldsymbol{\lambda}^k &:= (\lambda_1^k, \dots, \lambda_\ell^k), \\ \Lambda_\ell^k &:= \text{diag}[\lambda_1^k, \dots, \lambda_\ell^k], & G_\ell^k(\mathbf{x}, z) &:= \text{diag}[g_1^k(\mathbf{x}, z), \dots, g_\ell^k(\mathbf{x}, z)]. \end{aligned}$$

Feito isso, define-se

$$\mathbf{d}^k = \mathbf{d}_\alpha + \rho \mathbf{d}_\beta \quad \text{e} \quad \boldsymbol{\lambda}^k = \boldsymbol{\lambda}_\alpha + \rho \boldsymbol{\lambda}_\beta,$$

onde

$$0 < \rho < (\xi - 1) \frac{\mathbf{d}_\alpha^T \mathbf{e}_z}{\mathbf{d}_\beta^T \mathbf{e}_z}, \quad \text{para } \xi \in (0, 1).$$

Um tamanho de passo e um novo ponto auxiliar $(\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ viável com respeito a (PA.2) são calculados de acordo com o que segue:

Quando o novo ponto é viável com respeito a (PE.2) e de descida para F , ai atualiza-se a solução, ou seja, faze-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ e se diz que foi realizado um ***passo sério de descida viável***.

Mas se o novo ponto é viável com respeito a (PE.2) e não é de descida para f , considera-se que o ponto de iteração corrente (\mathbf{x}^k, z^k) esta longe da fronteira do epígrafo de f . E neste caso, ao invés de usar a direção calculada pelo FDIPA, usa-se a direção de máxima descida $-\mathbf{e}_z$ ($\mathbf{e}_z = [0, 0, \dots, 0, 1]^T \in \mathbb{R}^{n+1}$) para obter um ponto estritamente viável e que esteja próximo da fronteira do epígrafo. Desta forma, tem-se $f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k)$ no novo ponto e a próxima direção de busca gerada pelo FDIPA será de descida para f , e chama-se este passo de ***passo sério de máxima descida***.

No caso de um passo sério, o algoritmo limpa todas as informações obtidas dos planos de corte armazenadas até então.

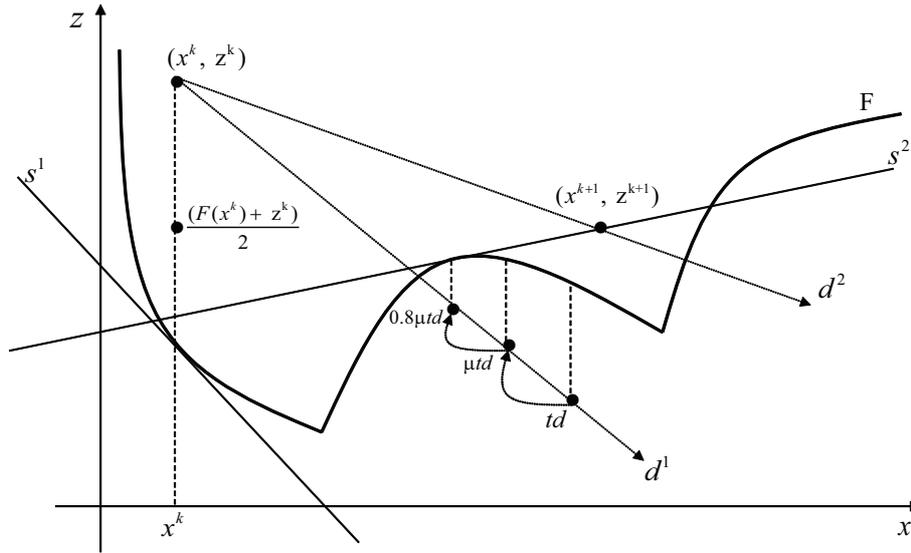


Figura 3.1: Encontrando o próximo iterado do Algoritmo.

Porém, se nenhuma dessas alternativas acontece, realiza-se o equivalente ao **passo nulo** nos métodos de feixe, ou seja, faz-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k)$. No entanto, não se atualiza a solução, um novo plano de corte é calculado e uma nova direção de descida viável com respeito a $(PA_{\ell+1}.2)$ é calculada usando os sistemas do FDIPA. Em seguida, o algoritmo recomeça seu procedimento novamente.

Devido a não convexidade, pode acontecer que o novo plano de corte torne o ponto de iteração corrente (\mathbf{x}^k, z^k) inviável (veja Figura 3.1). Neste caso, ignora-se o plano de corte, retorna-se ao longo da direção de busca e calcula-se um novo plano de corte. Este procedimento de retorno continua até que o ponto de iteração corrente seja viável com respeito ao plano de corte (para fazer com que o método seja mais eficiente, de fato, verifica-se viabilidade do ponto $(\mathbf{x}^k, (f(\mathbf{x}^k) + z^k)/2)$). Devido a função objetivo ser localmente Lipschitz contínua, este tipo de plano pode sempre ser encontrado, veja a Figura 3.2.

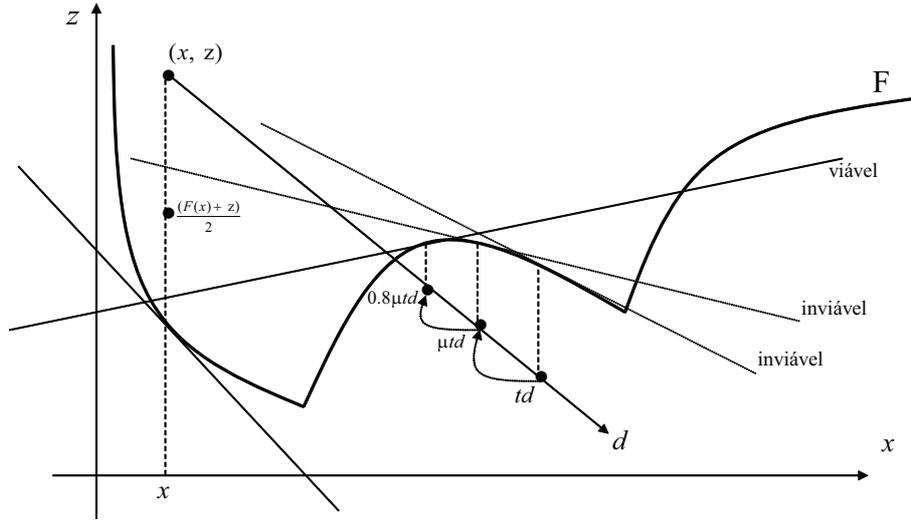


Figura 3.2: Determinação de um plano viável.

3.1.1 Nonsmooth Feasible Direction Nonconvex Algorithm

Apresenta-se o algoritmo NFDNA para resolver problemas de minimização do tipo (P.2). No que segue, assumi-se que em cada ponto $\mathbf{x} \in \mathbb{R}^n$ pode-se avaliar $f(\mathbf{x})$ e um correspondente subgradiente arbitrário $\mathbf{s} \in \partial f(\mathbf{x})$.

ALGORITMO - NFDNA

Parâmetros: Escolha uma tolerância aproximada final $\varepsilon > 0$. Selecione os parâmetros de controle $\rho > 0$ e $\xi \in (0, 1)$ para o limite da deflexão. Selecione os multiplicadores $\mu \in (1/2, 1)$ para o tamanho de passo e o tamanho máximo de passo $t_{\max} > 0$.

Dados: Seja $k = 1$ o contador de iterações e $\ell = 1$ o contador do número de planos de corte. Escolha um ponto inicial estritamente viável $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$, um vetor positivo inicial $\boldsymbol{\lambda}^1 \in \mathbb{R}^\ell$ e uma matriz simétrica definida positiva $B^1 \in \mathbb{R}^{(n+1) \times (n+1)}$. Seja $\mathbf{y}_1^1 = \mathbf{x}^1$. Calcule $f(\mathbf{x})$.

Passo 1: (Plano de Corte inicial)

Calcule $\mathbf{s}_1^k \in \partial f(\mathbf{x}^k)$, e o primeiro plano de corte

$$g_1(\mathbf{x}^k, z^k) = f(\mathbf{x}^k) - z^k, \quad g_1(\mathbf{x}, z) \in \mathbb{R}.$$

Faça

$$\nabla g_1(\mathbf{x}^k, z^k) = \begin{bmatrix} \mathbf{s}_1^k \\ -1 \end{bmatrix} \in \mathbb{R}^{n+1}.$$

Defina $\bar{\mathbf{g}}_1(\mathbf{x}^k, z^k) = [g_1(\mathbf{x}^k, z^k)] \in \mathbb{R}$,

e $\nabla \bar{\mathbf{g}}_1(\mathbf{x}^k, z^k) = [\nabla g_1(\mathbf{x}^k, z^k)] \in \mathbb{R}^{n+1}$.

Passo 2: (Encontrando a Direção) Calcule $\mathbf{d}^k = (\mathbf{d}_x^k, \mathbf{d}_z^k) \in \mathbb{R}^{n+1}$, uma direção de descida viável para (PA.2):

i) (Direção de descida) Determine $\mathbf{d}_\alpha \in \mathbb{R}^{n+1}$ e $\boldsymbol{\lambda}_\alpha \in \mathbb{R}^\ell$ resolvendo o sistema de equações lineares

$$B^k \mathbf{d}_\alpha + \nabla \bar{g}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha = -\mathbf{e}_z \quad (3.1)$$

$$\Lambda^k [\nabla \bar{g}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha + G^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha = 0, \quad (3.2)$$

ii) (Direção viável) Determine os valores $\mathbf{d}_\beta \in \mathbb{R}^{n+1}$ e $\boldsymbol{\lambda}_\beta \in \mathbb{R}^\ell$ resolvendo os sistema linear de equações

$$B^k \mathbf{d}_\beta + \nabla \bar{g}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\beta = 0 \quad (3.3)$$

$$\Lambda^k [\nabla \bar{g}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\beta + G^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\beta = -\boldsymbol{\lambda}^k. \quad (3.4)$$

Se $\mathbf{e}_z^T \mathbf{d}_\beta > 0$, então

$$\rho = \min \left\{ \varrho \|\mathbf{d}_\alpha\|^2, \frac{(\xi - 1) \mathbf{e}_z^T \mathbf{d}_\alpha}{\mathbf{e}_z^T \mathbf{d}_\beta} \right\}. \quad (3.5)$$

Caso contrário,

$$\rho = \varrho \|\mathbf{d}_\alpha\|^2. \quad (3.6)$$

iii) (Direção de descida viável) Calcule a direção de busca

$$\mathbf{d}^k = \mathbf{d}_\alpha + \rho \mathbf{d}_\beta \quad \text{e} \quad \boldsymbol{\lambda}^k = \boldsymbol{\lambda}_\alpha + \rho \boldsymbol{\lambda}_\beta. \quad (3.7)$$

Passo 3: (Tamanho de passo e atualiza solução) Calcule o tamanho de passo

$$t^k = \min\{t_{\max}, \max\{t \mid \bar{g}_\ell((\mathbf{x}^k, z^k) + t\mathbf{d}^k) \leq 0\}\}.$$

Se

$$\|\mathbf{d}_\ell^k\| \leq \varepsilon \quad \text{e} \quad t^k < t_{\max},$$

então pare com (\mathbf{x}^k, z^k) como sendo a solução final. Caso contrário, defina

$$(\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k) = (\mathbf{x}^k, z^k) + \mu t^k \mathbf{d}_\ell^k$$

e calcule o valor correspondente $f(\mathbf{y}_{\ell+1}^k)$.

Se $w_{\ell+1}^k \leq f(\mathbf{y}_{\ell+1}^k)$, o passo não é sério: vá para o Passo 6.

Caso contrário, verifique: Se $f(\mathbf{x}^k) \geq f(\mathbf{y}_{\ell+1}^k)$, vá para o Passo 4.

Senão, vá para o Passo 5.

Passo 4: (Passo sério de descida viável)

Faça $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$.

Apague todos os planos de corte e atualize B^k e λ^k .

Faça $k = k + 1$, $\ell = 1$ e vá para o Passo 1.

Passo 5: (Passo sério de máxima descida)

Defina $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k) - \mu(z^k - f(\mathbf{x}^k))\mathbf{e}_z$ e $f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k)$.

Apague todos os planos de corte e atualize B^k e λ^k .

Faça $k = k + 1$, $\ell = 1$ e vá para o Passo 1.

Passo 6: (Passo Nulo)

i) (Erro de Linearização) Calcule $\mathbf{s}_{\ell+1}^k \in \partial f(\mathbf{y}_{\ell+1}^k)$ e um erro de linearização

$$\alpha = f(\mathbf{x}^k) - f(\mathbf{y}_{\ell+1}^k) - (\mathbf{s}_{\ell+1}^k)^T(\mathbf{x}^k - \mathbf{y}_{\ell+1}^k).$$

ii) (Procedimento de retorno) Se $\alpha < g_1^k(\mathbf{x}^k, z^k)/2$ retorne ao longo do vetor \mathbf{d}^k até que um “ponto viável” seja encontrado: ou seja, faça

$$(\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k) = (\mathbf{x}^k, z^k) + \eta\mu t^k \mathbf{d}^k,$$

$\eta = 0.8\eta$ e vá para o Passo 6(i).

iii) (Planos de corte para passos nulos)

Calcule o novo plano de corte e seus derivados

$$\begin{aligned} g_{\ell+1}^k(\mathbf{x}^k, z^k) &= -\alpha + f(\mathbf{x}^k) - z^k \\ \nabla g_{\ell+1}^k(\mathbf{x}^k, z^k) &= ((\mathbf{s}_{\ell+1}^k)^T, -1)^T. \end{aligned}$$

Defina

$$\bar{\mathbf{g}}_{\ell+1}^k(\mathbf{x}^k, z^k) = [g_1^k(\mathbf{x}^k, z^k), \dots, g_{\ell}^k(\mathbf{x}^k, z^k), g_{\ell+1}^k(\mathbf{x}^k, z^k)]^T \in \mathbb{R}^{\ell+1}$$

e

$$\nabla \bar{\mathbf{g}}_{\ell+1}^k(\mathbf{x}^k, z^k) = [\nabla g_1^k(\mathbf{x}^k, z^k), \dots, \nabla g_{\ell}^k(\mathbf{x}^k, z^k), \nabla g_{\ell+1}^k(\mathbf{x}^k, z^k)]$$

$$\nabla \bar{\mathbf{g}}_{\ell+1}^k(\mathbf{x}, z) \in \mathbb{R}^{(n+1) \times (\ell+1)}$$

Faça $\ell = \ell + 1$ e vá para o Passo 2.

3.2 Análise da convergência

Nesta seção, estuda-se a análise da convergência do algoritmo NFDNA. Mostra-se que a direção de busca \mathbf{d}^k definida em (3.7) é uma direção de descida para F (i.e., para (PA.2) e (PE.2)). Então, prova-se que, quando o ponto de iteração corrente está próximo o suficiente da fronteira do epígrafo de f , \mathbf{d}^k será uma direção de descida também para f (i.e., para (P.2)). Em seguida, mostra-se que o número de passos nulos em cada iteração é finito, e que a sequência $(\mathbf{x}^k, z^k)_{k \in \mathbb{N}}$ é limitada. Finalmente, demonstra-se que todo ponto de acumulação (\mathbf{x}^*, z^*) da sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ é estacionário para f (nota-se que, se a função objetivo é convexa, então o ponto será um mínimo global para o problema (P.2)).

Assumi-se que a função objetivo f é localmente Lipschitz contínua, e as seguintes hipóteses são fabricadas:

Hipótese 3.1. *Existem números positivos ω_1 e ω_2 tais que $\omega_1 \|\mathbf{d}\|^2 \leq \mathbf{d}^T B \mathbf{d} \leq \omega_2 \|\mathbf{d}\|^2$ para todo $\mathbf{d} \in \mathbb{R}^{n+1}$ (veja [53] para menos condições restritivas para B).*

Hipótese 3.2. *Existem números positivos λ^I , λ^S , e g_{\max} tais que $0 < \lambda_i \leq \lambda^S$, $i = 1, \dots, \ell$, e $\lambda_i \geq \lambda^I$ para todo i tal que $\bar{g}_i(x, z) \geq g_{\max}$.*

Hipótese 3.3. *O conjunto $\{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \leq f(\mathbf{x}^1)\}$ é compacto.*

Hipótese 3.4. *Para todo $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ e todo i tal que $g_i(\mathbf{x}, z) = 0$ os vetores $\nabla g_i(\mathbf{x}, z)$ são linearmente independentes.*

Inicia-se a análise teórica do NFDNA observando que as soluções \mathbf{d}_α , $\boldsymbol{\lambda}_\alpha$, \mathbf{d}_β , e $\boldsymbol{\lambda}_\beta$ dos sistemas (3.1), (3.2), e (3.3), (3.4) são únicas. E isto é uma consequência do Lema 3.1 em [52] enunciado como segue:

Lema 3.1. *Dado um ponto $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$, qualquer matriz $B \in \mathbb{R}^{(n+1) \times (n+1)}$ definida positiva qualquer vetor não negativo $\boldsymbol{\lambda} \in \mathbb{R}^\ell$ tal que $\lambda_i > 0$ se $g_i(\mathbf{x}, z) = 0$, a matriz*

$$M = \begin{bmatrix} B & \nabla \bar{g}_\ell(\mathbf{x}, z) \\ \Lambda \nabla \bar{g}_\ell(\mathbf{x}, z)^T & G_\ell(\mathbf{x}, z) \end{bmatrix}$$

é não singular.

Segue do resultado anterior que \mathbf{d}_α , $\boldsymbol{\lambda}_\alpha$, \mathbf{d}_β , e $\boldsymbol{\lambda}_\beta$ são limitados superiormente.

Lema 3.2. *A direção \mathbf{d}_α definida por (3.1) e (3.2) satisfaz*

$$\mathbf{d}_\alpha^T \mathbf{e}_z = \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \leq -\mathbf{d}_\alpha^T B \mathbf{d}_\alpha.$$

Demonstração. Veja a prova do Lema 4.2. em [30]. □

Como consequência do lema anterior, segue que a direção \mathbf{d}_α é uma direção de descida para F (i.e. para (PE.2) e (PA.2)).

Proposição 3.3. *A direção \mathbf{d} definida em (3.7) é uma direção de descida para (PE.2) e (PA.2).*

Demonstração. Como consequência de (3.7), e tendo em mente que $\mathbf{e}_z = \nabla F(\mathbf{x}, z)$,

$$\mathbf{d}^T \nabla F(\mathbf{x}, z) = \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) + \rho \mathbf{d}_\beta^T \nabla F(\mathbf{x}, z).$$

Como $\rho \leq (\xi - 1) \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) / (\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z))$ com $\xi \in (0, 1)$, se $\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z) > 0$ (veja (3.5)), e como \mathbf{d}_α é uma direção de descida para F pelo Lema 3.2, obtém-se

$$\begin{aligned} \mathbf{d}^T \nabla F(\mathbf{x}, z) &\leq \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) + (\xi - 1) \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \\ &= \xi \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) \\ &\leq \mathbf{0}. \end{aligned}$$

(nota-se que $\mathbf{d}^T \nabla F(\mathbf{x}, z) = \mathbf{0}$ somente se $\mathbf{d}_\alpha = \mathbf{0}$).

Por outro lado, se $\mathbf{d}_\beta^T \nabla F(\mathbf{x}, z) \leq 0$ (veja (3.6)), obtém-se a inequação

$$\mathbf{d}^T \nabla F(\mathbf{x}, z) \leq \mathbf{d}_\alpha^T \nabla F(\mathbf{x}, z) < 0.$$

Portanto, \mathbf{d} é uma direção de descida para F . □

Embora \mathbf{d} , calculada no Passo 2 do algoritmo, seja uma direção de descida para (PE.2) e (PA.2), ela não precisa necessariamente ser de descida para (P.2). Na verdade, no próximo lema, mostra-se que \mathbf{d} é de descida também para (P.2), ou seja, os valores da função f em pontos gerados pelo algoritmo não crescem. Pois, prova-se que, quando o ponto corrente está suficientemente próximo da fronteira do epígrafo de f , a direção \mathbf{d} , a partir deste ponto, é uma direção de descida para (P.2).

Lema 3.4. *Seja $(\mathbf{x}_k, z_k) \in \text{int}(\text{epi } f)$ um ponto corrente gerado pelo algoritmo. Para todo $k \geq 1$, segue que*

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) \quad e \quad z^{k+1} < z^k.$$

Além disso, o próximo ponto $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$.

Demonstração. O ponto corrente (\mathbf{x}^k, z^k) é atualizado no Passo 4 ou no Passo 5 do algoritmo. No Passo 4, tem-se que $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ somente se $w_{\ell+1}^k > f(\mathbf{y}_{\ell+1}^k)$ e $f(\mathbf{x}^k) \geq f(\mathbf{y}_{\ell+1}^k)$. Portanto, obviamente, tem-se $z^{k+1} > f(\mathbf{x}^{k+1})$ (i.e. $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$) e $f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k)$. Além disso, $(\mathbf{d}^k)^T \mathbf{e}_z < 0$ pela Proposição 3.3. Portanto, $d_z^k < 0$ e o próximo componente z^{k+1} é calculado pela

fórmula (veja Passo 3 do algoritmo)

$$z^{k+1} = z^k + \mu t^k d_z^k,$$

onde $\mu, t^k > 0$. Portanto $z^{k+1} < z^k$.

Por outro lado, no Passo 5, usa-se a direção de descida máxima $-\mathbf{e}_z$ como direção de busca e portanto, $\mathbf{x}^{k+1} = \mathbf{x}^k$ e, naturalmente, $f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k)$. Tem-se ainda

$$z^{k+1} = z^k - \mu(z^k - f(\mathbf{x}^k)),$$

onde $\mu \in (0, 1)$ e $z^k - f(\mathbf{x}^k) > 0$ pois $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$. Portanto, tem-se novamente que $z^{k+1} < z^k$ e $z^{k+1} > f(\mathbf{x}^k) = f(\mathbf{x}^{k+1})$. \square

Lema 3.5. *Seja $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$ um ponto suficientemente próximo da fronteira do epi f (i.e., $z^k - f(\mathbf{x}^k) < -\mu t^k d_z^k$). Se (\mathbf{x}^k, z^k) não é um ponto estacionário, então a direção \mathbf{d}^k definida por (3.7) é uma direção de descida para o problema (P.2) (i.e. para f).*

Demonstração. Como $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$, tem-se que $z^k = f(\mathbf{x}^k) + \epsilon_1$ para algum $\epsilon_1 \geq 0$. Também tem-se que $(\mathbf{d}^k)^T \mathbf{e}_z < 0$ pela Proposição 3.3, e portanto, $d_z^k < 0$. O próximo iterado é calculado pela fórmula $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k) + \mu t^k (\mathbf{d}_x^k, d_z^k)$ com $\mu, t^k > 0$. Portanto, $z^{k+1} = z^k - \epsilon_2 = f(\mathbf{x}^k) + \epsilon_1 - \epsilon_2$, onde denota-se $\epsilon_2 = -\mu t^k d_z^k > 0$. Quando ϵ_1 é suficientemente pequeno (i.e. $\epsilon_1 < \epsilon_2$) tem-se obviamente que $z^{k+1} - f(\mathbf{x}^k) < 0$. Também segue que $f(\mathbf{x}^{k+1}) \leq z^{k+1}$, pois $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$ pelo Lema 3.4. Agora combinando as duas coisas obtém-se $f(\mathbf{x}^{k+1}) \leq z^{k+1} < f(\mathbf{x}^k)$ e \mathbf{d}^k é uma direção de descida para f por definição. \square

Corolário 3.6. *A seqüência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ gerada pelo algoritmo é limitada.*

Demonstração. Como $z^{k+1} < z^k$ para todo $k \in \mathbb{N}$ e pela Hipótese 3.4 a seqüência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ está limitada no conjunto $(\text{epi } f) \cap \{(\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid z < z^1\}$. \square

Lema 3.7. *A direção \mathbf{d} definida em (3.7) é limitada.*

Demonstração. A escolha de ρ (veja (3.5) e (3.6)) assegura que se tenha

$$\rho \leq \varrho \|\mathbf{d}_\alpha\|^2 \quad \text{para algum } \varrho > 0. \quad (3.8)$$

Por outro lado, do Lema 3.2 e da Hipótese 3.2, obtém-se que

$$-\mathbf{d}_\alpha^T \mathbf{e}_z \geq \omega_1 \|\mathbf{d}_\alpha\|^2,$$

e portanto, devido a (3.5), tem-se

$$\rho \geq \min \left\{ \varrho, \frac{(1-\xi)\omega_1}{\mathbf{d}_\beta^T \mathbf{e}_z} \right\} \|\mathbf{d}_\alpha\|^2,$$

se $\mathbf{d}_\beta^T \mathbf{e}_z > 0$.

Devido a (3.6) e como \mathbf{d}_β é limitada, existe um limite inferior $\varrho_{low} > 0$ tal que

$$\rho \geq \varrho_{low} \|\mathbf{d}_\alpha\|^2.$$

Portanto, pela limitação de \mathbf{d}_α , a deflexão dada por ρ é positiva limitada como acima.

De (3.7) e (3.8), tem-se

$$\begin{aligned} \|\mathbf{d}\| &= \|\mathbf{d}_\alpha + \rho \mathbf{d}_\beta\| \\ &\leq \|\mathbf{d}_\alpha\| + \|\rho \mathbf{d}_\beta\| \\ &\leq \|\mathbf{d}_\alpha\| + \varrho \|\mathbf{d}_\alpha\|^2 \|\mathbf{d}_\beta\| \\ &= (1 + \varrho \|\mathbf{d}_\alpha\| \cdot \|\mathbf{d}_\beta\|) \|\mathbf{d}_\alpha\|. \end{aligned}$$

Portanto, existe $\delta > 1$ tal que $\|\mathbf{d}\| \leq \delta \|\mathbf{d}_\alpha\|$ é válida. E devido a limitação de \mathbf{d}_α , tem-se que \mathbf{d} é limitada como acima. \square

No próximo lema, mostra-se que no Passo 6 do algoritmo, um ponto $(\mathbf{y}_{\ell+1}, w_{\ell+1})$ é encontrado depois de um número finito de loops dentro do passo, de modo que, o ponto corrente $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ seja viável com respeito ao plano de corte calculado no ponto $\mathbf{y}_{\ell+1}$.

Lema 3.8. *Existe $(\mathbf{y}_{\ell+1}, w_{\ell+1}) \in \mathbb{R}^{n+1}$ tal que $f(\mathbf{x}) - z \leq \alpha$, onde $\alpha = f(\mathbf{x}) - f(\mathbf{y}_{\ell+1}) - \mathbf{s}_{\ell+1}^T (\mathbf{x} - \mathbf{y}_{\ell+1})$ e $\mathbf{s}_{\ell+1} \in \partial f(\mathbf{y}_{\ell+1})$. Este ponto é encontrado após um número finito de loops.*

Demonstração. Suponha por contradição que $(\mathbf{y}_{\ell+1}, w_{\ell+1})$ viável não pode ser encontrado. Um ponto $(\mathbf{y}_{\ell+1,i}, w_{\ell+1,i})$ é calculado pela fórmula

$$(\mathbf{y}_{\ell+1,i}, w_{\ell+1,i}) = (\mathbf{x}, z) + \eta_i \mu t \mathbf{d}$$

onde $\mu, t > 0$, $\eta_1 = 1$, $\eta_2 \in (1/2, 1)$, e $\eta_{i+1} = 0.8\eta_i$ ($i = 2, 3, \dots$). Como $(\mathbf{y}_{\ell+1,i}, w_{\ell+1,i})$ viável não é encontrado segue que

$$f(\mathbf{x}) - z > f(\mathbf{x}) - f(\mathbf{y}_{\ell+1,i}) - \mathbf{s}_{\ell+1,i}^T (\mathbf{x} - \mathbf{y}_{\ell+1,i}) = \alpha_i$$

para todo $i \in \mathbb{N}$.

Como $\eta_{i+1} < \eta_i$ para todo i , tem-se $\eta_i \rightarrow 0$. Isto implica que $\mathbf{y}_{\ell+1,i} \rightarrow \mathbf{x}$. Como f é localmente Lipschitz contínua, segue que $|f(\mathbf{y}_{\ell+1,i}) - f(\mathbf{x})| \rightarrow 0$ e, portanto, também $\alpha \rightarrow 0$. Mas $f(\mathbf{x}) - z < 0$ pois $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$. O que é uma contradição. \square

Lema 3.9. *Existe $\tau > 0$ tal que para todo $(\mathbf{x}, z) \in \text{int}(\text{epi } f)$ e para todo $\mathbf{d} \in \mathbb{R}^{n+1}$ gerado pelo algoritmo, tem-se $\bar{\mathbf{g}}_\ell((\mathbf{x}, z) + t\mathbf{d}) \leq \mathbf{0}$ para todo $t \in [0, \tau]$.*

Demonstração. Denota-se por \mathbf{b} um vetor tal que $\mathbf{b}_i = \mathbf{s}_i^T \mathbf{y}_i - f(\mathbf{y}_i)$ para todo $i = 1, \dots, \ell$. Agora $\bar{\mathbf{g}}_\ell(\mathbf{x}, z) = (\nabla \bar{\mathbf{g}}_\ell(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{b}$, pois

$$\begin{aligned} g_i(\mathbf{x}, z) &= f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i) - z \\ &= f(\mathbf{y}_i) + \mathbf{s}_i^T \mathbf{x} - \mathbf{s}_i^T \mathbf{y}_i - z \\ &= (\mathbf{s}_i^T, -1)^T(\mathbf{x}, z) - \mathbf{s}_i^T \mathbf{y}_i + f(\mathbf{y}_i) \\ &= (\nabla g_i(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{b}_i \end{aligned}$$

para todo $i = 1, \dots, \ell$. Por construção, tem-se que

$$t \leq \max\{t_i \mid g_i((\mathbf{x}, z) + t_i \mathbf{d}) \leq 0, \quad i = 1, \dots, \ell\}.$$

Agora combinando estas duas observações, e observando-se que $\nabla g_i(\mathbf{x}, z)$ não depende do ponto (\mathbf{x}, z) mas dos pontos auxiliares \mathbf{y}_i ($i = 1, \dots, \ell$), obtém-se

$$\begin{aligned} g_i((\mathbf{x}, z) + t_i \mathbf{d}) &= (\nabla g_i((\mathbf{x}, z) + t_i \mathbf{d}))^T((\mathbf{x}, z) + t_i \mathbf{d}) - \mathbf{b}_i \\ &= (\nabla g_i(\mathbf{x}, z))^T((\mathbf{x}, z) + t_i \mathbf{d}) - \mathbf{b}_i \\ &= (\nabla g_i(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{b}_i + t_i (\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \\ &= g_i(\mathbf{x}, z) + t_i (\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0 \end{aligned} \tag{3.9}$$

para todo $i = 1, \dots, \ell$. Se $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0$, a equação acima é satisfeita com qualquer $t_i > 0$. Considera-se o caso quando $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} > 0$. Por (3.7), segue que $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} = (\nabla g_i(\mathbf{x}, z))^T(\mathbf{d}_\alpha + \rho \mathbf{d}_\beta)$, e de (3.2) e (3.4) obtém-se

$$\nabla g_i(\mathbf{x}, z)^T \mathbf{d}_\alpha = -g_i(\mathbf{x}, z) \frac{\lambda_{\alpha,i}}{\lambda_i} \quad \text{e} \tag{3.10}$$

$$\nabla g_i(\mathbf{x}, z)^T \mathbf{d}_\beta = -1 - g_i(\mathbf{x}, z) \frac{\lambda_{\beta,i}}{\lambda_i}. \tag{3.11}$$

Agora combinando (3.9), (3.10), e (3.11) obtém-se

$$g_i(\mathbf{x}, z) - t_i g_i(\mathbf{x}, z) \frac{\lambda_{\alpha,i} + \rho \lambda_{\beta,i}}{\lambda_i} - \rho t_i = g_i(\mathbf{x}, z) \left(1 - t_i \frac{\bar{\lambda}_i}{\lambda_i}\right) - \rho t_i \leq 0,$$

onde se tinha denotado $\bar{\lambda}_i = \lambda_{\alpha,i} + \rho\lambda_{\beta,i}$. Obviamente $\rho t_i > 0$ e $g_i(\mathbf{x}, z) < 0$. Portanto, a inequação é satisfeita se

$$t_i \frac{\bar{\lambda}_i}{\lambda_i} \leq 1.$$

Agora, $\boldsymbol{\lambda}$ é limitado pela Hipótese 3.3, e $\boldsymbol{\lambda}_\alpha$, $\boldsymbol{\lambda}_\beta$ e ρ são limitados (veja a prova do Lema 3.7). Portanto, $\bar{\boldsymbol{\lambda}}$ é limitado e existe $\tau > 0$ tal que $\lambda_i/\bar{\lambda}_i > \tau$ para todo $i = 1, \dots, \ell$. Portanto, para todo $t \in [0, \tau]$, segue que $g_i((\mathbf{x}, z) + t\mathbf{d}) \leq 0$. \square

O próximo lema mostra um resultado técnico que será usado posteriormente.

Lema 3.10. *Seja X um conjunto convexo e sejam $\mathbf{x}^0 \in \text{int } X$ e $\bar{\mathbf{x}} \in X$. Seja $\{\bar{\mathbf{x}}^k\} \subset \mathbb{R}^n \setminus X$ uma sequência tal que $\bar{\mathbf{x}}^k \rightarrow \bar{\mathbf{x}}$ e seja \mathbf{x}^k definida por $\mathbf{x}^k = \mathbf{x}^0 + \mu(\bar{\mathbf{x}}^k - \mathbf{x}^0)$ para algum $\mu \in (0, 1)$. Então existe $k_0 \in \mathbb{N}$ tal que $\mathbf{x}^k \in \text{int } X$ para todo $k \geq k_0$.*

Demonstração. Veja o Capítulo 2, Lema 2.8. \square

Lema 3.11. *Seja $(\bar{\mathbf{y}}, \bar{w})$ um ponto de acumulação da sequência $\{(\mathbf{y}_i, w_i)\}_{i \in \mathbb{N}}$ gerada pelo algoritmo. Então $\bar{w} = f(\bar{\mathbf{y}})$.*

Demonstração. Um novo ponto auxiliar (\mathbf{y}_i, w_i) é calculado no Passo 3 do algoritmo. Se $w_i > f(\mathbf{y}_i)$ realiza-se um passo sério (i.e. se vai para ao Passo 4 ou Passo 5). Portanto, no ponto de acumulação tem-se $\bar{w} \leq f(\bar{\mathbf{y}})$. Suponha agora que $\bar{w} < f(\bar{\mathbf{y}})$. Considere o plano de corte $\bar{f}_{\mathbf{s}_i}(\mathbf{x}) = f(\bar{\mathbf{y}}) + \mathbf{s}_i(\mathbf{x} - \bar{\mathbf{y}})$ para algum $\mathbf{s}_i \in \partial f(\bar{\mathbf{y}})$. Seja $\bar{f}_{\mathbf{s}_i}$ uma nova restrição para (PA.2) (i.e. $g_i(\mathbf{x}, z) = \bar{f}_{\mathbf{s}_i}(\mathbf{x}) - z$).

Denota-se por $r = D((\bar{\mathbf{y}}, \bar{w}); \bar{f}_{\mathbf{s}_i})$ a distância entre o ponto $(\bar{\mathbf{y}}, \bar{w})$ e o plano $\bar{f}_{\mathbf{s}_i}$. Como $\bar{w} < f(\bar{\mathbf{y}})$ tem-se que $r > 0$. Seja $\bar{B} = B((\bar{\mathbf{y}}, \bar{w}); \frac{r}{2})$. Obviamente, $\bar{B} \cap \bar{f}_{\mathbf{s}_i} = \emptyset$.

Agora, $(\mathbf{y}_i, w_i) \in \text{epi } \bar{f}_{\mathbf{s}_i}$ para qualquer i e $B \subset (\text{epi } \bar{f}_{\mathbf{s}_i})^c$. Portanto $(\mathbf{y}_i, w_i) \notin \bar{B}$. O que é uma contradição. \square

Lema 3.12. *Seja $(\mathbf{x}^k, z^k) \in \text{int}(\text{epi } f)$. O próximo iterado $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$ é encontrado após um número finito de iterações (i.e. loops do Passo 6 para o Passo 2 do algoritmo).*

Demonstração. O novo ponto $(\mathbf{x}^{k+1}, z^{k+1})$ está no interior do $\text{epi } f$ pelo Lema 3.4. Portanto, necessita-se mostrar somente que o novo ponto é encontrado depois de um número finito de iterações.

Um novo ponto auxiliar (\mathbf{y}_i, w_i) é encontrado após um número finito de loops dentro do Passo 6, pelo Lema 3.8. Se $w_i > f(\mathbf{y}_i)$ tem-se um *passo sério* (Passo 4 ou 5) e, obviamente $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\text{epi } f)$. A sequência $\{(\mathbf{y}_i, w_i)\}_{i \in \mathbb{N}}$ é limitada por construção logo existe um ponto de acumulação $(\bar{\mathbf{y}}, \bar{w})$. Pelo Lema 3.11 este ponto de acumulação esta na fronteira do epígrafo de f .

Denota-se por $\bar{f}_{s_i}(\mathbf{x}) = f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i)$ o plano de corte correspondente a i -ésima restrição e por $\hat{f}(\mathbf{x}) = \max\{\bar{f}_{s_i}(\mathbf{x}) \mid i = 1, \dots, \ell\}$ a função linear por partes definida pelo máximo de todos os planos de corte no ponto de acumulação $(\bar{\mathbf{y}}, \bar{w})$. Pelo Lema 3.10 existe $i_0 \in \mathbb{N}$ tal que $(\mathbf{y}_{i_0}, w_{i_0}) \in \text{int}(\text{epi } \hat{f})$. Mostrar-se-a que $(\mathbf{y}_{i_0}, w_{i_0}) \in \text{int}(\text{epi } f)$ ainda que o $\text{epi } f$ não seja convexo.

Por absurdo, supõe-se que $(\mathbf{y}_{i_0}, w_{i_0}) \notin \text{int}(\text{epi } f)$, ou seja, que $f(\mathbf{y}_{i_0}) \geq w_{i_0}$. Um *passo nulo* ocorre e adiciona-se um novo plano de corte. Agora $(\mathbf{y}_{i_0}, w_{i_0})$ está num segmento de reta que liga o ponto de acumulação $(\bar{\mathbf{y}}, \bar{w})$ e o ponto de iteração corrente (\mathbf{x}^k, z^k) , abaixo do epígrafo de f . Portanto, o novo plano de corte torna o ponto $(\bar{\mathbf{y}}, \bar{w})$ inviável (ele não pode tornar o ponto de iteração corrente inviável). Mas então $(\bar{\mathbf{y}}, \bar{w})$ não pode ser um ponto de acumulação, o que é um contradição.

Portanto, tem-se que $(\mathbf{y}_{i_0}, w_{i_0}) \in \text{int}(\text{epi } f)$ e ainda defini-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{i_0}, w_{i_0})$ (no Passo 4 do algoritmo) ou um *passo sério de máxima descida* ocorre (Passo 5 do algoritmo). \square

Lema 3.13. *Seja \mathbf{d}_α^* um ponto de acumulação da sequência $\{\mathbf{d}_\alpha^k\}_{k \in \mathbb{N}}$. Então $\mathbf{d}_\alpha^* = 0$.*

Demonstração. Por construção tem-se que

$$\begin{aligned} (\mathbf{x}^{k+1}, z^{k+1}) &= (\mathbf{x}^k, z^k) + \mu t^k \mathbf{d}^k \quad \text{ou} \\ (\mathbf{x}^{k+1}, z^{k+1}) &= (\mathbf{x}^k, z^k) - \mu(z^k - f(\mathbf{x}^k))\mathbf{e}_z. \end{aligned}$$

A sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ é limitada pelo Corolário 3.6. Denotando $\mathbf{x}^* = \lim_{k \rightarrow \infty} \mathbf{x}^k$ e $z^* = \lim_{k \rightarrow \infty} z^k$ e seja $K \subset \mathbb{N}$ tal que $\{t^k\}_{k \in K} \rightarrow t^*$. Isto segue do Lema 3.9 onde tem-se $t^* > 0$.

Quando $k \rightarrow \infty$, $k \in K$ segue que

$$\begin{aligned} z^* &= z^* + \mu t^* d_z^* \quad \text{ou} \\ z^* &= (1 - \mu)z^* + \mu f(\mathbf{x}^*). \end{aligned}$$

ou seja, tem-se que $d_z^* = 0$ ou $z^* = f(\mathbf{x}^*)$. Contudo, devido ao Lema 3.5, o último não acontece (\mathbf{d}^* é uma direção de descida para f ou $\mathbf{d}^* = 0$). Portanto $d_z^* = 0$.

Pela Proposição 3.3, segue que

$$0 = d_z^* = (\mathbf{d}^*)^T \mathbf{e}_z = \xi (\mathbf{d}_\alpha^*)^T \mathbf{e}_z = \xi d_{\alpha,z}^* \leq 0$$

com algum $\xi > 0$ e assim $d_{\alpha,z}^* = 0$. Além disso, pelo Lema 3.2 tem-se

$$0 = d_{\alpha,z}^* = (\mathbf{d}_\alpha^*)^T \mathbf{e}_z \leq -(\mathbf{d}_\alpha^*)^T B \mathbf{d}_\alpha^* \leq 0$$

e como B é definida positiva, pode-se concluir que $\mathbf{d}_\alpha^* = 0$. \square

Lema 3.14. *Seja $(\mathbf{s}_i^T, -1)^T$ o gradiente de uma restrição ativa no ponto de acumulação (\mathbf{x}^*, z^*) da sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$. Então $\mathbf{s}_i \in \partial f(\mathbf{x}^*)$.*

Demonstração. Como no ponto de acumulação (\mathbf{x}^*, z^*) tem-se que $f(\mathbf{x}^*) = z^*$ a primeira restrição $g_1(\mathbf{x}^*, z^*)$ é ativa e $\mathbf{s}_1 \in \partial f(\mathbf{x}^*)$ por construção (veja Passo 1 do algoritmo).

Suponha agora que a restrição $g_i(\mathbf{x}^*, z^*)$, $i > 1$, é ativa. Denota-se por $\bar{f}_{\mathbf{s}_i}(\mathbf{x}) = f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i)$ o plano de corte correspondente na restrição ativa. Ou seja, $\bar{f}_{\mathbf{s}_i}(\mathbf{x}^*) = z^*$. Na vizinhança do ponto de acumulação $\mathbf{x} \in B(\mathbf{x}^*; \sigma)$, para algum $\sigma > 0$, tem-se que $\bar{f}_{\mathbf{s}_i}(\mathbf{x})$ é uma aproximação inferior da função objetivo $f(\mathbf{x})$ ou $\mathbf{s}_i = 0$ (no caso do algoritmo já ter parado). Portanto tem-se que para todo $\mathbf{x} \in B(\mathbf{x}^*; \sigma)$, $\sigma > 0$, e $\mathbf{s}_i \in \partial f(\mathbf{y}_i)$

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i) \\ &= f(\mathbf{y}_i) - f(\mathbf{x}^*) + f(\mathbf{x}^*) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i) - \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*) \\ &= f(\mathbf{x}^*) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*) + f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x}^* - \mathbf{y}_i) - f(\mathbf{x}^*) \\ &= f(\mathbf{x}^*) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*) + g_i(\mathbf{x}^*, z^*) \\ &= f(\mathbf{x}^*) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*), \end{aligned}$$

pois $f(\mathbf{x}^*) = z^*$ e $g_i(\mathbf{x}^*, z^*) = 0$.

Agora, se for utilizado $\mathbf{x} = \mathbf{x}^* + t\mathbf{v}$, onde $\mathbf{v} \in \mathbb{R}^n$, $t > 0$ pode-se escrever

$$f(\mathbf{x}) - f(\mathbf{x}^*) \geq \mathbf{s}_i^T(\mathbf{x} - \mathbf{x}^*) = t\mathbf{s}_i^T \mathbf{v}$$

para todo $\mathbf{x} \in B(\mathbf{x}^*; \sigma)$ e obtém-se

$$\begin{aligned} f^\circ(\mathbf{x}^*; \mathbf{v}) &= \limsup_{\substack{\mathbf{x}' \rightarrow \mathbf{x}^* \\ t \downarrow 0}} \frac{f(\mathbf{x}' + t\mathbf{v}) - f(\mathbf{x}')} {t} \\ &\geq \limsup_{t \downarrow 0} \frac{f(\mathbf{x}^* + t\mathbf{v}) - f(\mathbf{x}^*)} {t} \\ &\geq \limsup_{t \downarrow 0} \frac{t\mathbf{s}_i^T \mathbf{v}} {t} = \mathbf{s}_i^T \mathbf{v}, \end{aligned}$$

Portanto, pela definição de subdiferencial $\mathbf{s}_i \in \partial f(\mathbf{x}^*)$. □

No próximo Lema mostra-se que, devido ao problema auxiliar ser convexo, λ_α^k é positivo ou zero na solução.

Lema 3.15. *Para k suficientemente grande, tem-se que $\lambda_\alpha^k \geq 0$.*

Demonstração. Considera-se o seguinte problema de otimização convexa

$$\text{minimize } \Phi(\mathbf{x}, z) \quad \text{s.t. } \bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq \mathbf{0},$$

onde $\Phi(\mathbf{x}, z) = z + \mathbf{d}_\alpha^T B \mathbf{x}$. Um ponto KKT $(\mathbf{x}^\Phi, z^\Phi)$ do problema satisfaz

$$\nabla z + B \mathbf{d}_\alpha + \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\lambda}^\Phi = \mathbf{0} \quad (3.12)$$

$$G_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\lambda}^\Phi = \mathbf{0} \quad (3.13)$$

$$\boldsymbol{\lambda}^\Phi \geq \mathbf{0} \quad (3.14)$$

$$\bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq \mathbf{0}. \quad (3.15)$$

O sistema (3.1)-(3.2) no Passo 2 do algoritmo pode ser reescrito como

$$\nabla z + B \mathbf{d}_\alpha^k + \nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^k = \mathbf{0}$$

$$G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^k = \varphi^k,$$

onde $\varphi^k = -\Lambda_\ell^k [\nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha^k$. Quando $\mathbf{d}_\alpha^k \rightarrow \mathbf{0}$ tem-se que $\varphi^k \rightarrow \mathbf{0}$ e então, para dado $\varepsilon_1 > 0$, existe $K_1 > 0$ tal que

$$\|\boldsymbol{\lambda}_\alpha^k - \boldsymbol{\lambda}^\Phi\| < \varepsilon_1 \quad \text{para } k > K_1.$$

Então como $\boldsymbol{\lambda}^\Phi \geq \mathbf{0}$ por (3.14) deduz-se que $\boldsymbol{\lambda}_\alpha^k \geq \mathbf{0}$ para k suficientemente grande. \square

Teorema 3.16. *Para qualquer ponto de acumulação (\mathbf{x}^*, z^*) da sequencia $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ tem-se que $\mathbf{0} \in \partial f(\mathbf{x}^*)$.*

Demonstração. Considere as equações (3.1) e (3.2). Quando $k \rightarrow \infty$ tem-se que $\mathbf{d}_\alpha^* = \mathbf{0}$ pelo Lema 3.13. Portanto, obtém-se

$$\nabla \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\lambda}_\alpha^* = -\mathbf{e}_z \quad \text{e} \quad \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\lambda}_\alpha^* = \mathbf{0},$$

onde denota-se por $\boldsymbol{\lambda}_\alpha^*$ o vetor de multiplicação de Lagrange correspondente a \mathbf{d}_α^* e por $\bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*)$ a restrição correspondente.

Como

$$\nabla \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_\ell \\ -1 & -1 & \cdots & -1 \end{bmatrix} \quad \text{e} \quad \boldsymbol{\lambda}_\alpha^* = [\lambda_{\alpha,1}^*, \lambda_{\alpha,2}^*, \dots, \lambda_{\alpha,\ell}^*]^T,$$

obtém-se

$$\sum_{i=1}^{\ell} \lambda_{\alpha,i}^* \mathbf{s}_i = \mathbf{0} \quad \text{e} \quad \sum_{i=1}^{\ell} \lambda_{\alpha,i}^* = 1.$$

Agora denota-se por $\mathcal{I} = \{i \mid \mathbf{g}_i^*(\mathbf{x}^*, z^*) = 0\}$ o conjunto dos índices das restrições ativas e por $\mathcal{J} = \{j \mid \mathbf{g}_j^*(\mathbf{x}^*, z^*) < 0\}$ o conjunto das restrições inativas em (\mathbf{x}^*, z^*) . Com isso, segue que

$$\begin{aligned} \mathbf{g}_i^*(\mathbf{x}^*, z^*)\lambda_{\alpha,i}^* &= 0 & \text{para todo } i \in \mathcal{I} \text{ e} \\ \mathbf{g}_j^*(\mathbf{x}^*, z^*)\lambda_{\alpha,j}^* &= 0 & \text{para todo } j \in \mathcal{J}. \end{aligned}$$

Portanto $\lambda_{\alpha,j}^* = 0$ para todo $j \in \mathcal{J}$ e além disso

$$\sum_{i \in \mathcal{I}} \lambda_{\alpha,i}^* \mathbf{s}_i = \mathbf{0} \quad \text{e} \quad \sum_{i \in \mathcal{I}} \lambda_{\alpha,i}^* = 1.$$

Pelo Lema 3.14 tem-se que $\mathbf{s}_i \in \partial f(\mathbf{x}^*)$ para todo $i \in \mathcal{I}$. Pela convexidade do subdiferencial e como $\lambda_{\alpha,i}^* \geq 0$ pelo Lema 3.15 para todo $i \in \mathcal{I}$ obtém-se que

$$\mathbf{0} = \sum_{i \in \mathcal{I}} \lambda_{\alpha,i}^* \mathbf{s}_i \in \partial f(\mathbf{x}^*).$$

□

3.3 Resultados Numéricos

Nesta seção apresenta-se os resultados numéricos obtidos com o método NFDNA desenvolvido na seção 3.1.1.

Assim como no FDIPA, é possível a obtenção de diferentes versões do algoritmo variando as regras para atualização das matrizes $\nabla \bar{\mathbf{g}}^k$, B e do vetor λ .

3.3.1 Implementação

Apagando todos os planos de corte depois de cada passo sério, o algoritmo trabalha bem na teoria. Na prática, isso torna o método menos eficiente. Contudo, quando o método resolve problemas convexos não existe perda de toda a memória, i.e., não é necessário apagar todos os planos de corte, pois no caso convexo os planos são sempre aproximações por baixo do epígrafo da função objetivo e portanto isso não causa qualquer problema.

No caso não convexo, os planos de corte não são necessariamente aproximações inferiores para a função objetivo. E com isso, pode acontecer que um plano deixe de fora o ponto de mínimo. Isto está ilustrado na Figura 3.1. Para preservar a eficiência e evitar que um plano deixe um mínimo de fora da região viável quando se está resolvendo um problema não convexo, o algoritmo limpa a memória somente após 10, 20, ou 40 iterações (dependendo do problema).

O algoritmo foi implementado em MATLAB em um microcomputador Pentium III de 500MHz com 2GB de RAM. Os parâmetros considerados foram os seguintes: Primeiro foram definidos $B = I$, $\xi = 0.1$, $\varphi = 0.1$ para todos os problemas e então foram selecionadas as melhores combinações para os valores $\varepsilon = 10^{-4}$ ou $\varepsilon = 10^{-5}$, $\mu = 0.7, 0.75$ ou 0.8 e $t_{\max} = 1$ ou 10 . individualmente dependendo do problema. O número máximo de planos de corte armazenados foi um conjunto de $5 \times n$ sem qualquer procedimento de agregação (veja [13, 58] para possíveis procedimentos de agregação). A atualização considerada para o vetor λ^k é a mesma do FDIPA [30].

3.3.2 Resultados Obtidos

A performance do algoritmo NFDNA foi testada com um conjunto de problemas teste que são amplamente utilizados para testar novos solvers em otimização não diferenciável. Estes problemas testes podem ser encontrados no trabalho de Lukšan e Vlček [54]. Todos os problemas, exceto o de Rosenbrock, são não diferenciáveis.

Na Tabela 3.3 são apresentados os resultados da comparação com os solvers citados na Tabela 3.2. Como é possível ver, o novo algoritmo resolve todos os problemas de forma robusta e eficiente. E em termos do número de avaliações da função e valor obtido para a função objetivo os algoritmos apresentam resultados similares.

Além disso, como parte da metodologia que está sendo proposta, tem-se a diferença que nos outros solvers é necessária a resolução de um subproblema quadrático em cada iteração. Fato que é evitado com o NFDNA. E como já mencionado, isso sugere que em termos de uso de tempo computacional a eficiência do NFDNA pode ser melhor do que a destes outros solvers. Naturalmente, mais testes devem e vão ser realizados.

Na Tabela 3.1 são utilizadas as seguintes abreviações:

- n - representa a dimensão do problema em questão;
- + - representa problemas convexos;
- - representa problemas não convexos;
- ps - é o número de passos sérios;
- pn - é o número de passos nulos;
- nf - é o número de vezes que $f(x)$ e $s \in \partial f(x)$ foram calculados;
- f^* - valor final da função objetivo obtido com o algoritmo;
- f^{opt} - o valor ótimo do problema como relatado em [54].

Tabela 3.1: Problemas teste.

No.	Problema	n	Convexo	ps	pn	nf	f^*	f^{opt}
1	Rosenbrock	2	–	73	31	146	$7.8129e^{-7}$	0
2	Crescent	2	–	33	1	43	0.0078	0
3	CB2	2	+	14	6	21	1.9522	1.9522
4	CB3	2	+	15	9	25	2.0001	2
5	DEM	2	+	17	2	20	-2.9997	-3
6	QL	2	+	31	2	34	7.2000	7.20
7	LQ	2	+	9	2	12	-1.4139	-1.4142
8	Mifflin1	2	+	7	11	19	-0.9999	-1
9	Mifflin2	2	–	10	9	20	-0.9999	-1
10	Wolfe	2	+	43	10	54	-7.9999	-8
11	Rosen	4	+	45	14	60	-43.9999	-44
12	Shor	5	+	49	23	73	22.6001	22.6001
13	Colville 1	5	–	95	114	210	-32.3484	-32.3486
14	HS78	5	–	851	384	2048	-2.9194	-2.9197
15	El-Attar	6	–	200	287	1028	0.5599	0.5598
16	Maxquad	10	+	12	53	66	-0.8414	-0.8414
17	Gill	10	–	148	649	806	9.7859	9.7857
18	Steiner 2	12	+	26	65	92	16.7038	16.7038
19	Maxq	20	+	84	282	367	$1.4695e^{-8}$	0
20	Maxl	20	+	80	32	113	$2.1196e^{-4}$	0
21	TR48	48	+	22	103	126	-638564.99	-638565.0
22	Goffin	50	+	28	43	72	$5.8786e^{-5}$	0
23	MXHILB	50	+	197	8	206	$2.9024e^{-5}$	0
24	L1HILB	50	+	66	39	106	$1.6129e^{-5}$	0
25	Shell Dual	15	–	536	856	1652	32.3489	32.3486

Tabela 3.2: Solvers utilizados na comparação

Solver	Autor(s)	Método	Referência
NCVX	Fuduli, Gaudioso & Giallombardo	Cutting Plane extension	[20]
NCNS	Gorgone	New Bundle Method	[59]

Tabela 3.3: Resultados da Experiência Numérica.

Solver	NCNS		NCVX		NFDNA		
No.	nf	f^*	nf	f^*	nf	f^*	f^{opt}
1	54	$5.137e^{-6}$	70	$5.009e^{-7}$	146	$7.812e^{-7}$	0
2	53	$5.112e^{-6}$	22	$8.022e^{-6}$	43	0.007	0
3	16	1.952	18	1.952	21	1.952	1.952
4	14	2	15	2	25	2	2
5	13	-3	21	-2.999	20	-2.999	-3
6	15	7.2	28	7.2	34	7.200	7.20
7	15	-1.414	9	-1.414	12	-1.413	-1.414
8	165	-0.999	127	-0.999	19	-0.999	-1
9	14	-0.999	13	-1	20	-0.999	-1
10	43	-7.999	21	-7.999	54	-7.999	-8
11	33	-43.999	29	-44	60	-43.999	-44
12	27	22.600	44	22.600	73	22.600	22.600
13	36	-32.348	47	-32.348	210	-32.348	-32.348
14	237	-2.919	159	-2.919	2048	-2.919	-2.919
15	172	0.559	152	0.559	1028	0.559	0.559
16	90	-0.841	56	-0.841	66	-0.841	-0.841
17	308	9.785	164	9.785	806	9.785	9.785
18	96	16.703	196	16.703	92	16.703	16.703
19	187	$1.561e^{-6}$	293	$1.660e^{-7}$	367	$1.469e^{-8}$	0
20	23	$4.493e^{-15}$	44	$1.110e^{-15}$	113	$2.119e^{-4}$	0
21	1662	-638514.8	1662	-638565	126	-638564.99	-638565
22	56	$1.984e^{-13}$	148	$1.142e^{-13}$	72	$5.878e^{-5}$	0
23	24	$1.764e^{-5}$	33	$1.768e^{-5}$	206	$2.902e^{-5}$	0
24	30	$1.709e^{-5}$	104	$6.978e^{-7}$	106	$1.612e^{-5}$	0
25	642	32.348	1497	32.349	1652	32.348	32.348

Capítulo 4

Uma Técnica para otimização não diferenciável convexa com restrições

Neste capítulo propõe-se um método que utiliza e estende a técnica apresentada no Capítulo 2, para otimização convexa e não diferenciável.

Considera-se o seguinte problema de otimização

$$\begin{cases} \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} & f(\mathbf{x}) \\ \text{sujeito a} & c(\mathbf{x}) \leq 0, \end{cases} \quad (\text{P.3})$$

onde a função objetivo e a restrição de desigualdade $f, c : \mathbb{R}^n \rightarrow \mathbb{R}$ são supostas convexas, e não necessariamente diferenciáveis. Nota-se que não existe perda de generalidade na formulação do problema (P.3) com somente uma restrição escalar, pois se necessário, $c(\mathbf{x})$ pode ser definida como o máximo de uma quantidade finita de funções convexas, e portanto respaldando o caso de múltiplas restrições de desigualdade.

Naturalmente, problemas de otimização não diferenciável com restrições são mais complexos, e poucos métodos práticos de resolução podem ser encontrados na literatura. Problemas convexas com restrições consideradas “fáceis”, tais como restrições lineares, podem ser resolvidos inserindo-se essas restrições diretamente no problema quadrático, como pode ser visto num trabalho de Kiwiel, [60]. Para problemas convexas com restrições mais gerais, como no problema (P.3) acima, uma possibilidade é resolver um problema equivalente sem restrições, definido com uma função de penalidade exata, como pode ser visto em [25]. Um algoritmo que combina as ideias dos métodos de feixe proximal com uma estratégia de filtro para avaliar um ponto candidato é proposto em [27]. Mais recentemente, um método de feixe

proximal por pontos inviáveis que não se utiliza de funções de penalidade ou um filtro, foi apresentado em [61].

Neste trabalho, propõe-se uma técnica, para resolver o problema (P.3), combinando as ideias do tradicional método de planos de corte apresentado por Kelley em [10], algumas ideias do método de feixes [11, 13] com os sistemas internos do FDIPA, [30].

A ideia é substituir o problema convexo restrito e não diferenciável (P.3), por um problema equivalente (PE.3) com duas restrições não diferenciáveis, ou seja, considerar

$$\begin{cases} \underset{(\mathbf{x}, z)}{\text{minimize}} & F(\mathbf{x}, z) = z \\ \text{sujeito a} & f(\mathbf{x}) \leq z \\ & c(\mathbf{x}) \leq 0. \end{cases} \quad (\text{PE.3})$$

O método se baseia na metodologia de aproximar as restrições funcionais do problema equivalente (PE.3) por planos de corte. Esta aproximação é feita com a construção de uma sequência de problemas auxiliares (PA.3). Em cada iteração, uma direção de busca, para o problema auxiliar corrente, é calculada utilizando os sistemas internos do FDIPA.

O algoritmo proposto está descrito na Seção 4.1 e os resultados numéricos para um conjunto de problemas teste são apresentados na Seção 4.3.

4.1 Método

A descrição do método segue com as seguintes ideias: O algoritmo gera uma sequência de pontos $\{(\mathbf{x}^k, z^k)\} \in \text{int}(\Omega)$, onde a região viável Ω é o domínio definido por

$$\Omega = \{(\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid f(\mathbf{x}) \leq z, \text{ e } c(\mathbf{x}) \leq 0\}.$$

A sequência gerada satisfaz,

$$z^{k+1} < z^k, \quad c(\mathbf{x}^{k+1}) \leq 0 \quad \text{para todo } k,$$

e converge para a solução global (\mathbf{x}^*, z^*) do problema (P.3). De fato, o algoritmo gera uma sequência de pontos auxiliares $\{(\mathbf{y}_i, w_i)\}$, que contém $\{(\mathbf{x}^k, z^k)\}$ como uma subsequência. Os pontos pertencentes a $\{(\mathbf{x}^k, z^k)\}$, são chamados *pontos sérios*. Em cada ponto sério (\mathbf{x}^k, z^k) , é criado um modelo aproximado de (PE.3) baseado nos planos de corte no ponto corrente (\mathbf{x}^k, z^k) , e alguns pontos auxiliares (\mathbf{y}_i, w_i) .

São empregadas algumas ideias do método de planos de corte para gerar aproximações lineares das restrições do problema equivalente (PE.3). Sejam \mathbf{y}_i , $i = 0, 1, \dots, \ell$ pontos auxiliares observados até à ℓ -ésima iteração, Com isso, pode-se

formular o seguinte problema:

$$\begin{cases} \underset{(\mathbf{x}, z)}{\text{minimize}} & F(\mathbf{x}, z) = z \\ \text{sujeito a} & z \geq f(\mathbf{y}_i) + \mathbf{s}_i^T(\mathbf{x} - \mathbf{y}_i) \quad i = 0, 1, \dots, \ell \\ & 0 \geq c(\mathbf{y}_i) + \mathbf{r}_i^T(\mathbf{x} - \mathbf{y}_i) \quad i = 0, 1, \dots, \ell \end{cases}$$

onde $\mathbf{s}_i \in \partial f(\mathbf{y}_i)$ e $\mathbf{r}_i \in \partial c(\mathbf{y}_i)$ são subgradientes no ponto \mathbf{y}_i .

A partir disso, é necessário arrumar o problema de modo que seja possível utilizar as ideias do FDIPA. Então define-se os planos de corte da seguinte forma:

$$\begin{aligned} g_i(\mathbf{x}, z) &= f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) - z, \quad i = 0, 1, \dots, \ell, \\ h_i(\mathbf{x}, z) &= c(\mathbf{y}_i) + (\mathbf{r}_i)^T(\mathbf{x} - \mathbf{y}_i), \quad i = 0, 1, \dots, \ell. \end{aligned}$$

É claro que tem-se algumas propriedades interessantes, tais como:

$$f(\mathbf{y}) + \mathbf{s}^T(\mathbf{x} - \mathbf{y}) \leq f(\mathbf{x}) \quad \text{e} \quad c(\mathbf{y}) + \mathbf{r}^T(\mathbf{x} - \mathbf{y}) \leq c(\mathbf{x}), \quad \mathbf{s} \in \partial f(\mathbf{y}), \quad \mathbf{r} \in \partial c(\mathbf{y}),$$

e

$$\begin{aligned} \check{f}(\mathbf{x}) &= \max\{f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x} - \mathbf{y}_i) \mid i = 1, \dots, \ell\} \leq f(\mathbf{x}), \\ \check{c}(\mathbf{x}) &= \max\{c(\mathbf{y}_i) + (\mathbf{r}_i)^T(\mathbf{x} - \mathbf{y}_i) \mid i = 1, \dots, \ell\} \leq c(\mathbf{x}). \end{aligned}$$

Portanto, planos de corte podem ser utilizados para a construção de aproximações para as restrições do problema equivalente. Ou seja, é possível considerar uma aproximação poliedral da região viável do problema equivalente:

$$\Omega^k = \{ (\mathbf{x}^k, z^k) \in \mathbb{R}^{n+1} \mid f(\mathbf{y}_i) + (\mathbf{s}_i)^T(\mathbf{x}^k - \mathbf{y}_i) \leq z^k, \quad i = 1, \dots, \ell \\ c(\mathbf{y}_i) + (\mathbf{r}_i)^T(\mathbf{x}^k - \mathbf{y}_i) \leq 0, \quad i = 1, \dots, \ell \}.$$

No ℓ -th iterado, defini-se o conjunto de restrições (lineares) funcionais por:

$$\begin{aligned} \bar{\mathbf{g}}_\ell(\mathbf{x}, z) &:= [g_0(\mathbf{x}, z), \dots, g_\ell(\mathbf{x}, z)]^T, \quad \bar{\mathbf{g}}_\ell : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{\ell+1} \\ \bar{\mathbf{h}}_\ell(\mathbf{x}, z) &:= [h_0(\mathbf{x}, z), \dots, h_\ell(\mathbf{x}, z)]^T, \quad \bar{\mathbf{h}}_\ell : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{\ell+1}. \end{aligned}$$

Então o problema auxiliar pode ser expressado da seguinte forma,

$$\begin{cases} \underset{(\mathbf{x}, z)}{\text{minimize}} & F(\mathbf{x}, z) = z \\ \text{sujeito a} & \bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq 0 \\ & \bar{\mathbf{h}}_\ell(\mathbf{x}, z) \leq 0. \end{cases} \quad (\text{PA.3})$$

Gerando uma Direção de Descida Viável. Como no FDIPA [30], o algoritmo encontra uma direção de descida $\mathbf{d}_\ell^k = (\mathbf{d}_x^k, d_z^k) \in \mathbb{R}^{n+1}$ para a função objetivo considerando a seguinte condição necessária de otimalidade Karush-Kuhn-Tucker, para o problema (PA.3):

$$\nabla F(\mathbf{x}^k, z^k) + \nabla \bar{g}(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\ell + \nabla \bar{h}(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\ell = \mathbf{0} \quad (4.1)$$

$$G(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\ell = \mathbf{0} \quad (4.2)$$

$$H(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\ell = \mathbf{0} \quad (4.3)$$

$$\bar{g}(\mathbf{x}^k, z^k) \leq 0$$

$$\bar{h}(\mathbf{x}^k, z^k) \leq 0$$

$$\boldsymbol{\lambda}_\ell \geq \mathbf{0}$$

$$\boldsymbol{\eta}_\ell \geq \mathbf{0}.$$

onde $\boldsymbol{\lambda}_\ell$ e $\boldsymbol{\eta}_\ell$ denotam os vetores de variáveis duais

$$\boldsymbol{\lambda}_\ell = (\lambda_0, \dots, \lambda_\ell), \quad \text{e} \quad \boldsymbol{\eta}_\ell = (\eta_0, \dots, \eta_\ell),$$

onde $G(\mathbf{x}, z) = \text{diag}[g_0(\mathbf{x}, z), \dots, g_\ell(\mathbf{x}, z)]$ e $H(\mathbf{x}, z) = \text{diag}[h_0(\mathbf{x}, z), \dots, h_\ell(\mathbf{x}, z)]$ são matrizes diagonais. Uma iteração quasi-Newton para resolver o sistema de equações não-lineares (4.1)-(4.3) pode ser definida por

$$\begin{aligned} & \begin{bmatrix} B^k & \nabla \bar{g}_\ell(\mathbf{x}^k, z^k) & \nabla \bar{h}_\ell(\mathbf{x}^k, z^k) \\ \Lambda_\ell^k [\nabla \bar{g}_\ell(\mathbf{x}^k, z^k)]^T & G^k(\mathbf{x}^k, z^k) & \mathbf{0} \\ \Sigma_\ell^k [\nabla \bar{h}_\ell(\mathbf{x}^k, z^k)]^T & \mathbf{0} & H^k(\mathbf{x}^k, z^k) \end{bmatrix} \begin{bmatrix} \mathbf{x}_\alpha^{k+1} - \mathbf{x}^k \\ \boldsymbol{\lambda}_\alpha^{k+1} - \boldsymbol{\lambda}^k \\ \boldsymbol{\eta}_\alpha^{k+1} - \boldsymbol{\eta}^k \end{bmatrix} = \\ & = - \begin{bmatrix} \nabla F(\mathbf{x}^k, z^k) + \nabla \bar{g}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\lambda}^k + \nabla \bar{h}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\eta}^k \\ G(\mathbf{x}^k, z^k) \boldsymbol{\lambda}^k \\ H(\mathbf{x}^k, z^k) \boldsymbol{\eta}^k \end{bmatrix} \quad (4.4) \end{aligned}$$

onde $(\mathbf{x}^k, \boldsymbol{\lambda}^k, \boldsymbol{\eta}^k)$ é o ponto atual, $(\mathbf{x}_\alpha^{k+1}, \boldsymbol{\lambda}_\alpha^{k+1}, \boldsymbol{\eta}_\alpha^{k+1})$ é a nova estimativa, Λ_ℓ^k e Σ_ℓ^k são as matrizes diagonais $\Lambda_\ell^k = \text{diag}[\lambda_0, \dots, \lambda_\ell]$, $\Sigma_\ell^k = \text{diag}[\eta_0, \dots, \eta_\ell]$, e B^k é uma matriz definida positiva.

Fazendo $\mathbf{d}_\alpha^k = \mathbf{x}_\alpha^{k+1} - \mathbf{x}^k$, pode-se expressar (4.4) como um sistema linear em termos de \mathbf{d}_α^k , $\boldsymbol{\lambda}_\alpha^{k+1}$ e $\boldsymbol{\eta}_\alpha^{k+1}$ dado por

$$B^k \mathbf{d}_\alpha^k + \nabla \bar{g}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^{k+1} + \nabla \bar{h}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\alpha^{k+1} = -\nabla F(\mathbf{x}^k, z^k) \quad (4.5)$$

$$\Lambda_\ell^k [\nabla \bar{g}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha^k + G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^{k+1} = \mathbf{0} \quad (4.6)$$

$$\Sigma_\ell^k [\nabla \bar{h}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha^k + H_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\alpha^{k+1} = \mathbf{0} \quad (4.7)$$

e tem-se o seguinte resultado para \mathbf{d}_α^k :

Lema 4.1. Para a direção \mathbf{d}_α , definida em pelo sistema (4.5)-(4.7), tem-se que $(\mathbf{d}_\alpha)^T \nabla F(\mathbf{x}, z) \leq -(\mathbf{d}_\alpha)^T B \mathbf{d}_\alpha$.

Demonstração. De fato, multiplicando a equação (4.5) por \mathbf{d}_α , obtém-se

$$(\mathbf{d}_\alpha)^T B \mathbf{d}_\alpha + (\mathbf{d}_\alpha)^T \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}, z) \boldsymbol{\lambda}_\alpha + (\mathbf{d}_\alpha)^T \nabla \bar{\mathbf{h}}_\ell(\mathbf{x}, z) \boldsymbol{\eta}_\alpha = -(\mathbf{d}_\alpha)^T \nabla F(\mathbf{x}, z) \quad (4.8)$$

E das equações (4.6) e (4.7) tem-se que

$$(\mathbf{d}_\alpha)^T \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}, z) = -(\boldsymbol{\lambda}_\alpha)^T (\Lambda_\ell)^{-1} G(\mathbf{x}, z)$$

e

$$(\mathbf{d}_\alpha)^T \nabla \bar{\mathbf{h}}_\ell(\mathbf{x}, z) = -(\boldsymbol{\eta}_\alpha)^T (\Sigma_\ell)^{-1} H(\mathbf{x}, z)$$

donde, substituindo estas equações em (4.8), obtém-se

$$\begin{aligned} (\mathbf{d}_\alpha)^T \nabla F(\mathbf{x}, z) &= -(\mathbf{d}_\alpha)^T B \mathbf{d}_\alpha + (\boldsymbol{\lambda}_\alpha)^T (\Lambda_\ell)^{-1} G(\mathbf{x}, z) \boldsymbol{\lambda}_\alpha \\ &\quad + (\boldsymbol{\eta}_\alpha)^T (\Sigma_\ell)^{-1} H(\mathbf{x}, z) \boldsymbol{\eta}_\alpha. \end{aligned}$$

e segue o resultado pelo fato de que $(\Lambda_\ell)^{-1} G$ e $(\Sigma_\ell)^{-1} H$ são negativa definidas devido a viabilidade estrita de (x^k, z^k) . \square

Devido ao Lema anterior e por B^k ser uma matriz definida positiva, tem-se que \mathbf{d}_α^k é uma direção de descida para F (i.e., para (PE.3) e (PA.3)).

Contudo, do sistema (4.6)-(4.7) segue que para qualquer restrição ativa, \mathbf{d}_α^k será ortogonal ao subgradiente associado e, conseqüentemente, pode ser uma direção inviável. Portanto, é necessária a realização de uma deflexão em \mathbf{d}_α^k , para o interior da região viável por meio do vetor \mathbf{d}_β^k definido pelo seguinte sistema de equações lineares:

$$B^k \mathbf{d}_\beta^k + \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\beta^k + \nabla \bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\beta^k = \mathbf{0} \quad (4.9)$$

$$\Lambda_\ell^k [\nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\beta^k + G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\beta^k = -\boldsymbol{\lambda}^k \quad (4.10)$$

$$\Sigma_\ell^k [\nabla \bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}_\beta^k + H_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\beta^k = -\boldsymbol{\eta}^k \quad (4.11)$$

A direção de busca é calculada fazendo,

$$\mathbf{d}^k = \mathbf{d}_\alpha^k + \rho^k \mathbf{d}_\beta^k. \quad (4.12)$$

e também

$$\boldsymbol{\lambda}^k = \boldsymbol{\lambda}_\alpha^k + \rho^k \boldsymbol{\lambda}_\beta^k, \quad (4.13)$$

$$\boldsymbol{\eta}^k = \boldsymbol{\eta}_\alpha^k + \rho^k \boldsymbol{\eta}_\beta^k. \quad (4.14)$$

Aqui, o limite de deflexão $\rho^k > 0$ (veja (4.17)) é selecionado de tal forma que a condição

$$\nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}^k \leq \xi \nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}_\alpha^k, \quad \text{for } \xi \in (0, 1),$$

com $\xi \in (0, 1)$ pré-definido, seja satisfeita (veja Lema 4.3 em [30]).

Portanto, tem-se que $\nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}^k < 0$ e, de (4.6)-(4.7) e (4.10)-(4.12), e com as definições de \mathbf{d}^k , λ^k e η^k , é possível escrever

$$\begin{aligned} \Lambda_\ell^k [\nabla \bar{g}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}^k + G_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}^k &= -\boldsymbol{\lambda}^k \\ \Sigma_\ell^k [\nabla \bar{h}_\ell(\mathbf{x}^k, z^k)]^T \mathbf{d}^k + H_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\eta}^k &= -\boldsymbol{\eta}^k \end{aligned}$$

de onde se obtém que $\nabla g_i(\mathbf{x}^k, z^k)^T \mathbf{d}^k < 0$ e $\nabla h_i(\mathbf{x}^k, z^k)^T \mathbf{d}^k < 0$, $i = 1, \dots, \ell$ para as restrições ativas. Portanto a direção \mathbf{d}^k definida em (4.12), é uma direção de descida viável para (PE.3) e (PA.3).

Gerando um passo viável. O número real $t_\ell^k > 0$ é definido como o maior valor tal que:

$$\bar{\mathbf{g}}_\ell((\mathbf{x}^k, z^k) + t_\ell^k \mathbf{d}_\ell^k) \leq \mathbf{0} \quad \text{e} \quad \bar{\mathbf{h}}_\ell((\mathbf{x}^k, z^k) + t_\ell^k \mathbf{d}_\ell^k) \leq \mathbf{0}.$$

Se o ponto (\mathbf{x}^k, z^k) não é ótimo, defini-se o ponto

$$(\mathbf{x}_{\ell+1}^k, z_{\ell+1}^k) = (\mathbf{x}^k, z^k) + t_\ell^k \mathbf{d}_\ell^k, \quad (4.15)$$

e o novo ponto teste $(\mathbf{y}_{\ell+1}, w_{\ell+1})$ é definido como:

$$(\mathbf{y}_{\ell+1}, w_{\ell+1}) = (\mathbf{x}^k, z^k) + \mu t_\ell^k \mathbf{d}_\ell^k, \quad (4.16)$$

onde $0 < \mu < 1$ é um parâmetro fixado.

Agora, o algoritmo tem os seguintes possíveis cenários:

- a) O ponto auxiliar é viável para (PE.3). Então este passo será chamado de **passo sério**, e o algoritmo define $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}, w_{\ell+1})$, atualiza-se o valor de k e define-se um novo problema auxiliar (PA.3) considerando um plano de corte no ponto $(\mathbf{x}^{k+1}, z^{k+1})$.
- b) O ponto auxiliar é inviável para (PE.3). O passo é chamado de **passo nulo**, o algoritmo faz $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{x}^k, z^k)$, e define-se um novo problema auxiliar (PA.3) considerando um plano de corte no ponto $(\mathbf{y}_{\ell+1}, w_{\ell+1})$.

4.1.1 Nonsmooth Feasible Direction Constraint Algorithm

ALGORITMO - NFDCA

Parâmetros: Escolha a tolerância aproximada final $\varepsilon > 0$. Selecione os parâmetros de controle $\varphi > 0$ e $\nu \in (0, 1)$ para o limite de deflexão. Selecione os multiplicadores $\mu \in (1/2, 1)$ para o tamanho de passo e o tamanho máximo de passo $t_{\max} > 0$.

Dados: Escolha um ponto estritamente viável $(\mathbf{x}^0, z^0) \in \text{int}(\Omega)$, vetores positivos iniciais $\boldsymbol{\lambda}^0$ e $\boldsymbol{\eta}^0 \in \mathbb{R}^\ell$ e uma matriz simétrica definida positiva $B^0 \in \mathbb{R}^{(n+1) \times (n+1)}$.

Seja $\mathbf{y}^0 = \mathbf{x}^0$, $k = 0$ e $\ell = 0$. Calcule $f(\mathbf{x}^0)$ e $c(\mathbf{x}^0)$.

Passo 1: Calcule $\mathbf{s}_\ell \in \partial f(\mathbf{y}_\ell)$, $\mathbf{r}_\ell \in \partial c(\mathbf{y}_\ell)$ e os planos de corte

$$g_\ell(\mathbf{x}^k, z^k) = f(\mathbf{y}_\ell) + (\mathbf{s}_\ell)^T (\mathbf{x}^k - \mathbf{y}_\ell) - z^k \quad \text{e} \quad h_\ell(\mathbf{x}^k, z^k) = c(\mathbf{y}_\ell) + (\mathbf{r}_\ell)^T (\mathbf{x}^k - \mathbf{y}_\ell).$$

Defina

$$\nabla g_\ell(\mathbf{x}^k, z^k) = \begin{bmatrix} \mathbf{s}_\ell \\ -1 \end{bmatrix} \quad \text{e} \quad \nabla h_\ell(\mathbf{x}^k, z^k) = \begin{bmatrix} \mathbf{r}_\ell \\ 0 \end{bmatrix}.$$

Defina

$$\bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k) = [g_0(\mathbf{x}^k, z^k), \dots, g_\ell(\mathbf{x}^k, z^k)]^T \in \mathbb{R}^{\ell+1},$$

$$\nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k) = [\nabla g_0(\mathbf{x}^k, z^k), \dots, \nabla g_\ell(\mathbf{x}^k, z^k)] \in \mathbb{R}^{(n+1) \times (\ell+1)},$$

e

$$\bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k) = [h_0(\mathbf{x}^k, z^k), \dots, h_\ell(\mathbf{x}^k, z^k)]^T \in \mathbb{R}^{\ell+1},$$

$$\nabla \bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k) = [\nabla h_0(\mathbf{x}^k, z^k), \dots, \nabla h_\ell(\mathbf{x}^k, z^k)] \in \mathbb{R}^{(n+1) \times (\ell+1)}.$$

Passo 2: (Direção de Busca) Calcule $\mathbf{d}_\ell^k = (\mathbf{d}_x^k, d_z^k) \in \mathbb{R}^{n+1}$, uma direção viável para (PA.3):

i) (Direção de Descida)

Determine \mathbf{d}_α^k , $\boldsymbol{\lambda}_\alpha^k$ e $\boldsymbol{\eta}_\alpha^k$ através do sistema de equações lineares (4.5)-(4.7).

ii) (Direção Viável)

Determine \mathbf{d}_β^k , $\boldsymbol{\lambda}_\beta^k$ e $\boldsymbol{\eta}_\beta^k$ através do sistema de equações lineares (4.9)-(4.11).

iii) Faça

$$\rho = \begin{cases} \min \left[\varphi \|\mathbf{d}_\alpha^k\|^2, (\xi - 1) \frac{\nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}_\alpha^k}{\nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}_\beta^k} \right], & \text{if } \nabla F(\mathbf{x}^k, z^k)^T \mathbf{d}_\beta^k > 0 \\ \varphi \|\mathbf{d}_\alpha^k\|^2, & \text{Caso contrário.} \end{cases} \quad (4.17)$$

iv) (Direção de Descida Viável)

Calcule a direção de descida viável \mathbf{d}_ℓ^k , $\boldsymbol{\lambda}_\ell^k$ e $\boldsymbol{\eta}_\ell^k$

$$\begin{aligned} \mathbf{d}_\ell^k &= \mathbf{d}_\alpha^k + \rho^k \mathbf{d}_\beta^k, \\ \boldsymbol{\lambda}_\ell^k &= \boldsymbol{\lambda}_\alpha^k + \rho^k \boldsymbol{\lambda}_\beta^k, \\ \boldsymbol{\eta}_\ell^k &= \boldsymbol{\eta}_\alpha^k + \rho^k \boldsymbol{\eta}_\beta^k. \end{aligned}$$

Passo 3: Determine o tamanho do passo t^k fazendo

$$\begin{aligned} \bar{t} &= \max \left\{ t \mid \bar{\mathbf{g}}_\ell((\mathbf{x}^k, z^k) + t\mathbf{d}_\ell^k) \leq \mathbf{0} \text{ e } \bar{\mathbf{h}}_\ell(\mathbf{x}^k + t\mathbf{d}_\ell^k) \leq \mathbf{0} \right\}, \\ t^k &= \min\{\bar{t}, t_{\max}\}. \end{aligned}$$

Se

$$\|\mathbf{d}_\alpha^k\| \leq \varepsilon$$

então pare com (\mathbf{x}^k, z^k) como solução final. Caso contrário, faça

$$(\mathbf{y}_{\ell+1}, w_{\ell+1}) = (\mathbf{x}^k, z^k) + \mu t^k \mathbf{d}_\ell^k,$$

e calcule os correspondentes valores $f(\mathbf{y}_\ell)$ e $c(\mathbf{y}_\ell)$.

Se

$$f(\mathbf{y}_{\ell+1}) < w_{\ell+1} \text{ e } c(\mathbf{y}_{\ell+1}) < 0,$$

vá para o Passo 4, caso contrário vá para o Passo 5.

Passo 4: (Passo Séri) Defina $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}, w_{\ell+1})$.

Atualize B^k e $\boldsymbol{\lambda}^k$.

Faça $k = k + 1$, $\ell = \ell + 1$, e vá para o Passo 1.

Passo 5: (Passo Nulo) Compute $\mathbf{s}_{\ell+1} \in \partial f(\mathbf{y}_{\ell+1})$, $\mathbf{r}_{\ell+1} \in \partial c(\mathbf{y}_{\ell+1})$ e um novo plano de corte em $(\mathbf{y}_{\ell+1}, w_{\ell+1})$,

$$\begin{aligned} g_\ell(\mathbf{x}^k, z^k) &= f(\mathbf{y}_{\ell+1}) + (\mathbf{s}_{\ell+1})^T(\mathbf{x}^k - \mathbf{y}_{\ell+1}) - z^k \quad \text{e} \\ h_\ell(\mathbf{x}^k, z^k) &= c(\mathbf{y}_{\ell+1}) + (\mathbf{r}_{\ell+1})^T(\mathbf{x}^k - \mathbf{y}_{\ell+1}). \end{aligned}$$

Atualize o vetor de restrições:

$$\begin{aligned} \bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k) &= [g_0(\mathbf{x}^k, z^k), \dots, g_\ell(\mathbf{x}^k, z^k), g_{\ell+1}(\mathbf{x}^k, z^k)]^T, \\ \bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k) &= [h_0(\mathbf{x}^k, z^k), \dots, h_\ell(\mathbf{x}^k, z^k), h_{\ell+1}(\mathbf{x}^k, z^k)]^T, \quad \text{e} \end{aligned}$$

$$\begin{aligned} \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^k, z^k) &= [\nabla g_0(\mathbf{x}^k, z^k), \dots, \nabla g_\ell(\mathbf{x}^k, z^k), \nabla g_{\ell+1}(\mathbf{x}^k, z^k)], \\ \nabla \bar{\mathbf{h}}_\ell(\mathbf{x}^k, z^k) &= [\nabla h_0(\mathbf{x}^k, z^k), \dots, \nabla h_\ell(\mathbf{x}^k, z^k), \nabla h_{\ell+1}(\mathbf{x}^k, z^k)]. \end{aligned}$$

Faça $\ell = \ell + 1$ e vá para o Passo 2.

4.2 Análise da Convergência

Nesta seção demonstra-se a convergência global do algoritmo proposto na Seção 4.1.1. Para isso, são consideradas as seguintes hipóteses:

Hipótese 4.1. *O conjunto $\Omega \cap \{(\mathbf{x}, z) \in \mathbb{R}^{n+1} \mid z < z^0\}$ é compacto.*

Hipótese 4.2. *Para o problema (P.3) vale a condição de qualificação de Slater [62].*

Hipótese 4.3. *Existem números positivos σ_1 e σ_2 tais que $\sigma_1 \|\mathbf{d}\|^2 \leq \mathbf{d}^T \mathbf{B} \mathbf{d} \leq \sigma_2 \|\mathbf{d}\|^2$ para quaisquer $\mathbf{d} \in \mathbb{R}^{n+1}$.*

Hipótese 4.4. *Existem números positivos λ^I , λ^S , e η^I , η^S tais que $\lambda^I \leq \lambda_i \leq \lambda^S$ e $\eta^I \leq \eta_i \leq \eta^S$, para $i = 0, 1, \dots, \ell$.*

Hipótese 4.5. *Para todo $(\mathbf{x}, z) \in \text{int}(\Omega)$, o conjunto*

$$\{\nabla h_i(\mathbf{x}, z), i = 0, 1, \dots, \ell\} \cup \{\nabla g_i(\mathbf{x}, z), \text{ para todo } i \text{ tal que } g_i(\mathbf{x}, z) = 0\}$$

é linearmente independente.

Inicia-se a análise teórica do método observando que as soluções $\mathbf{d}_\alpha^k, \boldsymbol{\lambda}_\alpha^k, \boldsymbol{\eta}_\alpha^k$, e $\mathbf{d}_\beta^k, \boldsymbol{\lambda}_\beta^k, \boldsymbol{\eta}_\beta^k$ dos sistemas (4.5)-(4.7) e (4.9)-(4.11) são únicas. Este fato é consequência direta do Lema 3.1 em [52] enunciado como segue, usando a notação do paper:

Lema 4.2. *Para qualquer $(\mathbf{x}, z) \in \text{int}(\Omega)$, e $B \in \mathbb{R}^{(n+1) \times (n+1)}$ definida positiva, a matriz*

$$M(\mathbf{x}, B, \boldsymbol{\lambda}, \boldsymbol{\eta}) = \begin{bmatrix} B & \nabla \bar{\mathbf{g}}(\mathbf{x}, z) & \nabla \bar{\mathbf{h}}(\mathbf{x}, z) \\ \Lambda[\nabla \bar{\mathbf{g}}(\mathbf{x}, z)]^T & G(\mathbf{x}, z) & \mathbf{0} \\ \Sigma[\nabla \bar{\mathbf{h}}(\mathbf{x}, z)]^T & \mathbf{0} & H(\mathbf{x}, z) \end{bmatrix},$$

é não singular.

Observa-se que, como a matriz B é limitada por hipótese, \mathbf{x}^k está num conjunto compacto, $\boldsymbol{\lambda}^k$ e $\boldsymbol{\eta}^k$ são limitados (veja [30]), a matriz $M(\mathbf{x}, B, \boldsymbol{\lambda}, \boldsymbol{\eta})$ é limitada fora do zero e portanto, \mathbf{d}_α^k , $\boldsymbol{\lambda}_\alpha^k$, $\boldsymbol{\eta}_\alpha^k$, \mathbf{d}_β^k , $\boldsymbol{\lambda}_\beta^k$, $\boldsymbol{\eta}_\beta^k$ são limitados superiormente.

Lema 4.3. *Seja $(\mathbf{x}^k, z^k) \in \text{int } \Omega$ um ponto de iteração gerado pelo algoritmo. Para todo $k \geq 1$, tem-se que*

$$z^{k+1} < z^k \quad \text{e} \quad c(\mathbf{x}^{k+1}) < 0.$$

Além disso, o próximo ponto iterado $(\mathbf{x}^{k+1}, z^{k+1})$ está no interior de Ω .

Demonstração. Quando o algoritmo gera a direção de descida (veja seção 4.1), tem-se a propriedade de que $(\mathbf{d}^k)^T \nabla F(\mathbf{x}, z) < 0$, de onde segue que $d_z^k < 0$. Pela construção do novo ponto em (4.16), tem-se que

$$z^{k+1} = z^k + \mu t^k d_z^k, \quad \text{onde} \quad \mu, t^k > 0,$$

logo, $z^{k+1} < z^k$.

O ponto corrente (\mathbf{x}^k, z^k) é atualizado no Passo 4 do algoritmo. Ao atualizar o ponto corrente com um “passo sério”, faz-se $(\mathbf{x}^{k+1}, z^{k+1}) = (\mathbf{y}_{\ell+1}^k, w_{\ell+1}^k)$ somente se, $w_{\ell+1}^k > f(\mathbf{y}_{\ell+1}^k)$ e $c(\mathbf{y}_{\ell+1}^k) < 0$. Donde obtém-se de imediato que

$$f(\mathbf{x}^{k+1}) = f(\mathbf{y}_{\ell+1}^k) < w_{\ell+1}^k = z^{k+1} \quad \text{e} \quad c(\mathbf{x}^{k+1}) = c(\mathbf{y}_{\ell+1}^k) < 0.$$

Portanto, $(\mathbf{x}^{k+1}, z^{k+1}) \in \text{int}(\Omega)$. □

OBS 4.1: Note que a sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ definida em (4.15) para k fixo, está num conjunto limitado. De fato, pela Hipótese 4.1 e Lema 4.3, o conjunto $\Omega_0 := \{(\mathbf{x}, z) \in \Omega \mid z \leq z^0\}$ é limitado. Logo, existe $r > 0$ tal que $\|(\mathbf{x}, z)\| < r$ para todo ponto em Ω (que é o mesmo que dizer que Ω_0 está numa bola centrada na origem e de raio r). Então para qualquer passo sério tem-se $\|(\mathbf{x}^k, z^k)\| < r$. Para a sequencia de baixo, dada por $(\mathbf{x}_\ell^k, z_\ell^k) = (\mathbf{x}^k, z^k) + t\mathbf{d}$ com t limitado por t_{\max} e $\|\mathbf{d}\|$ limitado por um valor D , segue que, pela desigualdade triangular $\|(\mathbf{x}_\ell^k, z_\ell^k)\| \leq \|(\mathbf{x}^k, z^k)\| + Dt_{\max}$, então a sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ está numa bola centrada na origem e raio $r + Dt_{\max}$.

Lema 4.4. *Seja $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ um ponto de acumulação da sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ definida em (4.15) para k fixo. Então $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ é um ponto pertencente a fronteira de Ω .*

Demonstração. Sem perda de generalidade, considere que a sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ converge para $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ (caso necessário, considera-se uma subsequencia apropriada). Pela definição da sequencia $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$ tem-se que o ponto $(\mathbf{x}_{\ell+1}^k, z_{\ell+1}^k)$ é viável com respeito ao planos de corte 2ℓ do problema auxiliar número ℓ . Logo se pode escrever que

$$\left. \begin{aligned} z_m^k &> f(\mathbf{x}_\ell^k) + \mathbf{s}_\ell^T (\mathbf{x}_m^k - \mathbf{x}_\ell^k) \\ 0 &\geq c(\mathbf{x}_\ell^k) + \mathbf{r}_\ell^T (\mathbf{x}_m^k - \mathbf{x}_\ell^k) \end{aligned} \right\} \quad \forall m \geq \ell + 1. \quad (4.18)$$

Tomando o limite sobre esta expressão, quando $m \rightarrow \infty$, obtém-se:

$$\begin{aligned}\tilde{z}^k &\geq f(\mathbf{x}_\ell^k) + \mathbf{s}_\ell^T(\tilde{\mathbf{x}}^k - \mathbf{x}_\ell^k), \\ 0 &\geq c(\mathbf{x}_\ell^k) + \mathbf{r}_\ell^T(\tilde{\mathbf{x}}^k - \mathbf{x}_\ell^k).\end{aligned}\tag{4.19}$$

Agora, considerando o limite quando $\ell \rightarrow \infty$, e levando em consideração que os subgradients são limitados, tem-se:

$$\begin{aligned}\tilde{z}^k &\geq f(\tilde{\mathbf{x}}^k), \\ 0 &\geq c(\tilde{\mathbf{x}}^k).\end{aligned}\tag{4.20}$$

Portanto $(\tilde{\mathbf{x}}^k, \tilde{z}^k) \in \Omega$. Pela definição da sequência $\{(\mathbf{x}_\ell^k, z_\ell^k)\}_{\ell \in \mathbb{N}}$, nenhum dos pontos $(\mathbf{x}_\ell^k, z_\ell^k)$ estão no interior de Ω . Então, o limite $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$ não pode estar no interior de Ω donde segue o resultado. \square

Lema 4.5. *Seja $(\mathbf{x}^k, z^k) \in \text{int } \Omega$. O próximo ponto de iteração $(\mathbf{x}^{k+1}, z^{k+1})$ está no interior de Ω e é encontrado após um número finito de iterações.*

Demonstração. Pelo Lema 4.3, um novo iterado $(\mathbf{x}^{k+1}, z^{k+1})$ está no interior de Ω . Então, basta mostrar que ele é encontrado após um número finito de iterações. Considera-se a sequência $\{(\tilde{\mathbf{x}}_\ell^k, \tilde{z}_\ell^k)\}_{\ell \in \mathbb{N}}$ definida em (4.15) e tal que, $\{(\tilde{\mathbf{x}}_\ell^k, \tilde{z}_\ell^k)\}_{\ell \in \mathbb{N}} \subset \mathbb{R}^n \setminus \Omega$. Por construção, esta sequência é limitada e portanto possui um ponto de acumulação, denotado por $(\tilde{\mathbf{x}}^k, \tilde{z}^k)$. Como $(\tilde{\mathbf{x}}^k, \tilde{z}^k) \in \Omega$ (veja o Lema 4.4), e $(\mathbf{x}^k, z^k) \in \text{int } \Omega$, pode-se construir uma sequência $\{(\mathbf{x}_\ell^k, z_\ell^k)\}$ tal que:

$$(\mathbf{x}_\ell^k, z_\ell^k) = (\mathbf{x}^k, z^k) + \mu[(\tilde{\mathbf{x}}_\ell^k, \tilde{z}_\ell^k) - (\mathbf{x}^k, z^k)],$$

para a qual (pelo Lema 2.8), tem-se que existe $\ell_0 \in \mathbb{N}$ tal que $(\mathbf{x}_\ell^k, z_\ell^k) \in \text{int}(\Omega)$, para todo $\ell > \ell_0$. Portanto, pode-se fazer $\mathbf{x}^{k+1} = \mathbf{x}_\ell^k$ e $z^{k+1} = z_\ell^k$ para $\ell > \ell_0$, o que completa a demonstração. \square

OBS 4.2: Como consequência do Lema 4.5, a sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ está inteiramente contida no interior de Ω . E pelo Lema 4.3, tem-se que $z^{k+1} < z^k < z^0$, Portanto, a sequência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ está contida no conjunto limitado Ω_0 definido na **OBS 4.1**.

Lema 4.6. *Existe $\tau > 0$ tal que para todo $(\mathbf{x}, z) \in \text{int}(\Omega)$ e para toda direção $\mathbf{d} \in \mathbb{R}^{n+1}$ gerada pelo algoritmo, tem-se que $\bar{\mathbf{g}}((\mathbf{x}, z) + t\mathbf{d}) \leq \mathbf{0}$ e $\bar{\mathbf{h}}((\mathbf{x}, z) + t\mathbf{d}) \leq \mathbf{0}$ para todo $t \in [0, \tau]$.*

Demonstração. Como no lema 2.11, tem-se que $\bar{\mathbf{g}}_\ell(\mathbf{x}, z) = (\nabla \bar{\mathbf{g}}_\ell(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{b}$, onde $\mathbf{b}_i = \mathbf{s}_i^T \mathbf{y}_i - f(\mathbf{y}_i)$ e $\mathbf{s}_i \in \partial f(\mathbf{y}_i)$, para todo $i = 0, 1, \dots, \ell$.

E de modo análogo, faz-se,

$$\begin{aligned}
h_i(\mathbf{x}, z) &= c(\mathbf{y}_i) + \mathbf{r}_i^T(\mathbf{x} - \mathbf{y}_i), \quad \text{para } \mathbf{r}_i \in \partial c(\mathbf{y}_i); \\
&= c(\mathbf{y}_i) + \mathbf{r}_i^T \mathbf{x} - \mathbf{r}_i^T \mathbf{y}_i \\
&= ((\mathbf{r}_i)^T, 0)^T(\mathbf{x}, z) - \mathbf{r}_i^T \mathbf{y}_i + c(\mathbf{y}_i) \\
&= (\nabla h_i(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{c}_i, \quad \text{onde } \mathbf{c}_i = \mathbf{r}_i^T \mathbf{y}_i - c(\mathbf{y}_i), \quad \forall i = 0, 1, \dots, \ell,
\end{aligned}$$

pode-se escrever: $\bar{\mathbf{h}}_\ell(\mathbf{x}, z) = (\nabla \bar{\mathbf{h}}_\ell(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{c}$.

Por construção, o algoritmo define o tamanho de passo t fazendo:

$$t = \min \left\{ t_{\max}, \max \{ t \mid \bar{\mathbf{g}}_\ell((\mathbf{x}, z) + t\mathbf{d}_\ell) \leq \mathbf{0} \text{ e } \bar{\mathbf{h}}_\ell(\mathbf{x} + t\mathbf{d}_\ell^k) \leq \mathbf{0} \} \right\}.$$

Agora combinando estas duas observações, e nota-se que $\nabla g_i(\mathbf{x}, z)$ e $\nabla h_i(\mathbf{x}, z)$ não dependem do ponto (\mathbf{x}, z) , mas dos pontos auxiliares \mathbf{y}_i ($i = 0, 1, \dots, \ell$), obtém-se

$$\begin{aligned}
g_i((\mathbf{x}, z) + t_i \mathbf{d}) &= [\nabla g_i((\mathbf{x}, z) + t_i \mathbf{d})]^T((\mathbf{x}, z) + t_i \mathbf{d}) - \mathbf{b}_i \\
&= g_i(\mathbf{x}, z) + t_i (\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0
\end{aligned} \tag{4.21}$$

e

$$\begin{aligned}
h_i((\mathbf{x}, z) + t_i \mathbf{d}) &= [\nabla h_i((\mathbf{x}, z) + t_i \mathbf{d})]^T((\mathbf{x}, z) + t_i \mathbf{d}) - \mathbf{c}_i \\
&= (\nabla h_i(\mathbf{x}, z))^T((\mathbf{x}, z) + t_i \mathbf{d}) - \mathbf{c}_i \\
&= (\nabla h_i(\mathbf{x}, z))^T(\mathbf{x}, z) - \mathbf{c}_i + t_i (\nabla h_i(\mathbf{x}, z))^T \mathbf{d} \\
&= h_i(\mathbf{x}, z) + t_i (\nabla h_i(\mathbf{x}, z))^T \mathbf{d} \leq 0, \quad \forall i = 0, 1, \dots, \ell.
\end{aligned} \tag{4.22}$$

Se $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} \leq 0$ e $(\nabla h_i(\mathbf{x}, z))^T \mathbf{d} \leq 0$, as equações do lema são satisfeitas para qualquer $t_i > 0$.

Portanto, considera-se que $(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} > 0$ e $(\nabla h_i(\mathbf{x}, z))^T \mathbf{d} > 0$. Por (4.12), tem-se que

$$(\nabla g_i(\mathbf{x}, z))^T \mathbf{d} = (\nabla g_i(\mathbf{x}, z))^T(\mathbf{d}_\alpha + \rho \mathbf{d}_\beta) \text{ e } (\nabla h_i(\mathbf{x}, z))^T \mathbf{d} = (\nabla h_i(\mathbf{x}, z))^T(\mathbf{d}_\alpha + \rho \mathbf{d}_\beta).$$

Mas

$$\nabla g_i(\mathbf{x}, z)^T \mathbf{d}_\alpha = -g_i(\mathbf{x}, z) \frac{\lambda_{\alpha,i}}{\lambda_i} \text{ e } \nabla g_i(\mathbf{x}, z)^T \mathbf{d}_\beta = -1 - g_i(\mathbf{x}, z) \frac{\lambda_{\beta,i}}{\lambda_i}, \tag{4.23}$$

e

$$\nabla h_i(\mathbf{x}, z)^T \mathbf{d}_\alpha = -h_i(\mathbf{x}, z) \frac{\eta_{\alpha,i}}{\eta_i} \text{ e } \nabla h_i(\mathbf{x}, z)^T \mathbf{d}_\beta = -1 - h_i(\mathbf{x}, z) \frac{\eta_{\beta,i}}{\eta_i}. \tag{4.24}$$

Agora, combinando as equações (4.23) com (4.21) e (4.24) com (4.22), obtém-se que

$$g_i(\mathbf{x}, z) - t_i g_i(\mathbf{x}, z) \frac{\lambda_{\alpha,i} + \rho \lambda_{\beta,i}}{\lambda_i} - \rho t_i = g_i(\mathbf{x}, z) \left(1 - t_i \frac{\bar{\lambda}_i}{\lambda_i}\right) - \rho t_i \leq 0,$$

e

$$h_i(\mathbf{x}, z) - t_i h_i(\mathbf{x}, z) \frac{\eta_{\alpha,i} + \rho \eta_{\beta,i}}{\lambda_i} - \rho t_i = h_i(\mathbf{x}, z) \left(1 - t_i \frac{\bar{\eta}_i}{\eta_i}\right) - \rho t_i \leq 0.$$

onde $\bar{\lambda}_i = \lambda_{\alpha,i} + \rho \lambda_{\beta,i}$ e $\bar{\eta}_i = \eta_{\alpha,i} + \rho \eta_{\beta,i}$. Como $\rho t_i > 0$, $g_i(\mathbf{x}, z) < 0$ e $h_i(\mathbf{x}, z) < 0$, tem-se

$$t_i \frac{\bar{\lambda}_i}{\lambda_i} \leq 1 \quad \text{e} \quad t_i \frac{\bar{\eta}_i}{\eta_i} \leq 1.$$

Agora, $\boldsymbol{\lambda}$ e $\boldsymbol{\eta}$ são limitados pela Hipótese 4.4, e $\boldsymbol{\lambda}_\alpha$, $\boldsymbol{\lambda}_\beta$, $\boldsymbol{\eta}_\alpha$, $\boldsymbol{\eta}_\beta$ e ρ são limitados. Portanto, $\bar{\boldsymbol{\lambda}}$ e $\bar{\boldsymbol{\eta}}$ são limitados e existe $\tau > 0$ de modo que $\lambda_i/\bar{\lambda}_i > \tau$ e $\eta_i/\bar{\eta}_i > \tau$ para todo $i = 1, \dots, \ell$.

Portanto, para todo $t \in [0, \tau]$, tem-se $g_i((\mathbf{x}, z) + t\mathbf{d}) \leq 0$ e $h_i((\mathbf{x}, z) + t\mathbf{d}) \leq 0$. \square

Lema 4.7. *Seja \mathbf{d}_α^* um ponto de acumulação da sequência $\{\mathbf{d}_\alpha^k\}_{k \in \mathbb{N}}$. Então $\mathbf{d}_\alpha^* = \mathbf{0}$.*

Demonstração. A demonstração é análoga a do Lema 2.12. \square

Lema 4.8. *Sejam $(\mathbf{s}^T, -1)^T$ e $(\mathbf{r}^T, 0)^T$ gradientes de restrições ativas no ponto de acumulação (\mathbf{x}^*, z^*) da sequência gerada pelo algoritmo, então $\mathbf{s} \in \partial f(\mathbf{x}^*)$ e $\mathbf{r} \in \partial c(\mathbf{x}^*)$.*

Demonstração. A demonstração de que $\mathbf{s} \in \partial f(\mathbf{x}^*)$ já foi feita no Lema 2.13. Resta mostrar que $\mathbf{r} \in \partial c(\mathbf{x}^*)$. Seja $h(\mathbf{x}, z) = c(\mathbf{y}) + \mathbf{r}^T(\mathbf{x} - \mathbf{y})$ uma restrição ativa em (\mathbf{x}^*, z^*) , onde $\mathbf{r} \in \partial c(\mathbf{y})$. Logo, $h(\mathbf{x}^*, z^*) = c(\mathbf{y}) + \mathbf{r}^T(\mathbf{x}^* - \mathbf{y}) = 0$.

Como a restrição c é convexa, para todo $\mathbf{x} \in \Omega$

$$\begin{aligned} c(\mathbf{x}) &\geq c(\mathbf{y}) + \mathbf{r}^T(\mathbf{x} - \mathbf{y}) \\ &= c(\mathbf{y}) - c(\mathbf{x}^*) + c(\mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{y}) - \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) \\ &= c(\mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) + [c(\mathbf{y}) + \mathbf{s}^T(\mathbf{x}^* - \mathbf{y})] - c(\mathbf{x}^*) \\ &= c(\mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) + h(\mathbf{x}^*, z^*) - c(\mathbf{x}^*) \\ &= c(\mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) - c(\mathbf{x}^*) \end{aligned}$$

Como $c(\mathbf{x}^*) \leq 0$ segue que $c(\mathbf{x}) \geq c(\mathbf{x}^*) + \mathbf{r}^T(\mathbf{x} - \mathbf{x}^*)$, para todo $\mathbf{x} \in \Omega$ e portanto $\mathbf{r} \in \partial c(\mathbf{x}^*)$. \square

No lema seguinte mostra-se que, devido o problema auxiliar ser convexo, $\boldsymbol{\lambda}_\alpha^k$ e $\boldsymbol{\eta}_\alpha^k$ são não-negativos na solução.

Lema 4.9. *Para k suficientemente grande, tem-se que $\boldsymbol{\lambda}_\alpha^k \geq \mathbf{0}$, e $\boldsymbol{\eta}_\alpha^k \geq \mathbf{0}$.*

Demonstração. Considere o seguinte problema de otimização convexa

$$\text{minimize } \Phi(\mathbf{x}, z) \text{ sujeito a } \bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq \mathbf{0}, \quad \bar{\mathbf{h}}_\ell(\mathbf{x}, z) \leq \mathbf{0},$$

onde $\Phi(\mathbf{x}, z) = F(\mathbf{x}, z) + \mathbf{d}_\alpha^T B \mathbf{x}$. Um ponto KKT $(\mathbf{x}^\Phi, z^\Phi)$ deste problema satisfaz

$$\begin{aligned} \nabla F(\mathbf{x}, z) + B \mathbf{d}_\alpha + \nabla \bar{\mathbf{g}}_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\lambda}^\Phi + \nabla \bar{\mathbf{h}}_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\eta}^\Phi &= \mathbf{0} \\ G_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\lambda}^\Phi &= \mathbf{0} \\ H_\ell(\mathbf{x}^\Phi, z^\Phi) \boldsymbol{\eta}^\Phi &= \mathbf{0} \\ \boldsymbol{\lambda}^\Phi &\geq \mathbf{0} \end{aligned} \quad (4.25)$$

$$\boldsymbol{\eta}^\Phi \geq \mathbf{0} \quad (4.26)$$

$$\bar{\mathbf{g}}_\ell(\mathbf{x}, z) \leq \mathbf{0},$$

$$\bar{\mathbf{h}}_\ell(\mathbf{x}, z) \leq \mathbf{0}.$$

Nota-se que os sistemas (4.5), (4.6) e (4.7) do algoritmo podem ser reescritos como

$$\begin{aligned} \nabla F(\mathbf{x}, z) + B \mathbf{d}_\alpha^k + \nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^k + \nabla \bar{\mathbf{h}}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\alpha^k &= \mathbf{0} \\ \bar{G}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\lambda}_\alpha^k &= \boldsymbol{\varphi}_1^k, \\ \bar{H}_\ell^k(\mathbf{x}^k, z^k) \boldsymbol{\eta}_\alpha^k &= \boldsymbol{\varphi}_2^k, \end{aligned}$$

onde $\boldsymbol{\varphi}_1^k = -\Lambda_\ell^k [\nabla \bar{\mathbf{g}}_\ell^k(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha^k$, e $\boldsymbol{\varphi}_2^k = -\Sigma_\ell^k [\nabla \bar{\mathbf{h}}_\ell^k(\mathbf{x}^k, z^k)]^T \mathbf{d}_\alpha^k$.

Quando $\mathbf{d}_\alpha^k \rightarrow \mathbf{0}$ tem-se que $\boldsymbol{\varphi}_i^k \rightarrow \mathbf{0}$ for $i = 1, 2$ e então, para dados $\varepsilon_1, \varepsilon_2 > 0$, existem $K_1, K_2 > 0$ tais que

$$\|\boldsymbol{\lambda}_\alpha^k - \boldsymbol{\lambda}^\Phi\| < \varepsilon_1 \quad \text{for } k > K_1 \quad \text{e} \quad \|\boldsymbol{\eta}_\alpha^k - \boldsymbol{\eta}^\Phi\| < \varepsilon_2 \quad \text{for } k > K_2.$$

Então como $\boldsymbol{\lambda}^\Phi, \boldsymbol{\eta}^\Phi \geq \mathbf{0}$ por (4.25) e (4.26) deduz-se que $\boldsymbol{\lambda}_\alpha^k, \boldsymbol{\eta}_\alpha^k \geq \mathbf{0}$ para k suficientemente grande. \square

No próximo resultado mostra-se que todo ponto de acumulação (\mathbf{x}^*, z^*) da sequencia $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$ satisfaz a condição de otimalidade para minimizar uma função convexa num conjunto convexo, dada no Teorema 1.27.

Teorema 4.10. *Seja (\mathbf{x}^*, z^*) um ponto de acumulação da seqüência $\{(\mathbf{x}^k, z^k)\}_{k \in \mathbb{N}}$. Então, $0 \in \partial f(\mathbf{x}^*) + N_\Omega(\mathbf{x}^*)$, onde $N_\Omega(\mathbf{x}^*)$ é o cone normal de Ω em \mathbf{x}^* . E portanto \mathbf{x}^* é o minimizador de f em Ω . (veja Teorema 1.27)*

Demonstração. Considere a sequencia

$$\mathbb{Y} := \{\mathbf{y}_0^0, \mathbf{y}_1^0, \dots, \mathbf{y}_{\ell^0}^0, \mathbf{y}_0^1, \mathbf{y}_1^1, \dots, \mathbf{y}_{\ell^1}^1, \dots, \mathbf{y}_0^k, \mathbf{y}_1^k, \dots, \mathbf{y}_{\ell^k}^k, \dots\},$$

de todos os pontos obtidos nas sub-iterações do algoritmo. Como esta sequência é limitada, se pode encontrar uma subsequência convergente. Em particular, como $\mathbf{x}^{k+1} = \mathbf{y}_{\ell^k}^k$, pode-se afirmar que existe uma subsequência $\bar{\mathbb{Y}} \subset \mathbb{Y}$ que converge para \mathbf{x}^* . Considere $\bar{\mathbb{Y}}^k \equiv \bar{\mathbb{Y}} \cap \{\mathbf{y}_0^k, \mathbf{y}_1^k, \dots, \mathbf{y}_{\ell^k}^k\}$. Agora, Defina $\mathbb{I}^k = \{i \mid \mathbf{y}_i^k \in \bar{\mathbb{Y}}^k\}$.

Considerando as equações (4.5), (4.6) e (4.7). Quando $k \rightarrow \infty$ tem-se que $\mathbf{d}_\alpha^* = \mathbf{0}$, pelo Lema 4.7. Logo obtém-se,

$$\begin{aligned}\nabla \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\lambda}_\alpha^* + \nabla \bar{\mathbf{h}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\eta}_\alpha^* &= -\nabla F(\mathbf{x}^*, z^*), \\ \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\lambda}_\alpha^* &= \mathbf{0}, \\ \bar{\mathbf{h}}_\ell^*(\mathbf{x}^*, z^*) \boldsymbol{\eta}_\alpha^* &= \mathbf{0},\end{aligned}$$

onde denota-se por $\boldsymbol{\lambda}_\alpha^*$, $\boldsymbol{\eta}_\alpha^*$ os vetores multiplicadores de Lagrange correspondentes a \mathbf{d}_α^* e por $\bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*)$ e $\bar{\mathbf{h}}_\ell^*(\mathbf{x}^*, z^*)$ as correspondentes restrições.

Como

$$\nabla \bar{\mathbf{g}}_\ell^*(\mathbf{x}^*, z^*) = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \dots & \mathbf{s}_\ell \\ -1 & -1 & \dots & -1 \end{bmatrix}, \quad \text{e} \quad \nabla \bar{\mathbf{h}}_\ell^*(\mathbf{x}^*, z^*) = \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \dots & \mathbf{r}_\ell \\ 0 & 0 & \dots & 0 \end{bmatrix}$$

$$\boldsymbol{\lambda}_\alpha^* = [\lambda_{\alpha,1}^*, \lambda_{\alpha,2}^*, \dots, \lambda_{\alpha,\ell}^*]^T \quad \text{e} \quad \boldsymbol{\eta}_\alpha^* = [\eta_{\alpha,1}^*, \eta_{\alpha,2}^*, \dots, \eta_{\alpha,\ell}^*]^T$$

obtém-se

$$\sum_{i=1}^{\ell} \lambda_{\alpha,i}^* \mathbf{s}_i = -\sum_{i=1}^{\ell} \eta_{\alpha,i}^* \mathbf{r}_i \quad \text{e} \quad \sum_{i=1}^{\ell} \lambda_{\alpha,i}^* = 1.$$

E considerando apenas os índices $i \in \mathbb{I}^*$, e definindo

$$\mu^* = \sum_{i \in \mathbb{I}^*} \eta_{\alpha,i}^*, \quad \text{e} \quad \bar{\eta}_{\alpha,i}^* = \eta_{\alpha,i}^* / \mu^*$$

tem-se

$$\sum_{i \in \mathbb{I}^*} \lambda_{\alpha,i}^* \mathbf{s}_i + \mu^* \sum_{i \in \mathbb{I}^*} \bar{\eta}_{\alpha,i}^* \mathbf{r}_i = \mathbf{0} \quad \text{e} \quad \sum_{i \in \mathbb{I}^*} \lambda_{\alpha,i}^* = \sum_{i \in \mathbb{I}^*} \bar{\eta}_{\alpha,i}^* = 1. \quad (4.27)$$

Pelo Lema 3.14 tem-se que $\mathbf{s}_i \in \partial f(\mathbf{x}^*)$ e $\mathbf{r}_i \in \partial c(\mathbf{x}^*)$ para todo $i \in \mathbb{I}^*$ e pela convexidade do subdiferencial pode-se escrever

$$\mathbf{s} := \sum_{i \in \mathbb{I}^*} \lambda_{\alpha,i}^* \mathbf{s}_i \quad \text{onde} \quad \mathbf{s} \in \partial f(\mathbf{x}^*) \quad \text{e} \quad \mathbf{r} := \sum_{i \in \mathbb{I}^*} \bar{\eta}_{\alpha,i}^* \mathbf{r}_i \quad \text{onde} \quad \mathbf{r} \in \partial c(\mathbf{x}^*).$$

Com esta notação é possível notar que, pela equação (4.27), $-\mathbf{r} \in \partial f(\mathbf{x}^*)$.

Como $\mathbf{r} \in \partial c(\mathbf{x}^*)$, pela definição de subdiferencial, tem-se que

$$c(\mathbf{x}) \geq c(\mathbf{x}^*) + \mathbf{r}^T (\mathbf{x} - \mathbf{x}^*), \quad \text{para todo} \quad \mathbf{x} \in \Omega. \quad (4.28)$$

Se $c(\mathbf{x}^*) < 0$, então $\mathbf{x}^* \in \text{int}(\Omega)$ e com isso, $N_\Omega(\mathbf{x}^*) = \{0\}$ e portanto tem-se que $0 \in \partial f(\mathbf{x}^*)$ de forma análoga ao que foi feito no Capítulo 2.

Agora, supondo que $c(\mathbf{x}^*) = 0$, como $c(\mathbf{x}) \leq 0$, pela equação (4.28), tem-se que $-\mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) \geq 0$ para todo $\mathbf{x} \in \Omega$ o que significa que $-\mathbf{r} \in N_\Omega(\mathbf{x}^*)$ (Veja definição de cone normal no Capítulo 1).

Portanto mostrou-se que existe $-\mathbf{r} \in \partial f(\mathbf{x}^*)$ tal que $-\mathbf{r}^T(\mathbf{x} - \mathbf{x}^*) \geq 0$ para todo $\mathbf{x} \in \Omega$, o que pelo Teorema 1.27 é equivalente a $0 \in \partial f(\mathbf{x}^*) + N_\Omega(\mathbf{x}^*)$. \square

E com este resultado a prova da convergência está completa.

4.3 Resultados Numéricos

Nesta seção são apresentados os resultados preliminares obtidos com o método NFDCA descrito na Seção 4.1.1. Os detalhes dos problemas teste são apresentados na Tabela 4.1, os resultados preliminares são apresentados na Tabela 4.2 e o resultado da comparação com outros solvers está na Tabela 4.3.

4.3.1 Experiência Computacional Preliminar

Para validar a experiência computacional inicial do algoritmo NFDCA, realizou-se uma implementação em MATLAB, num micro computador com 500MHz e 2 GB de RAM. Para avaliar sua performance, foi utilizada uma bateria de problemas testes que podem ser encontrados no trabalho de Schittkowski [63], Lukšan e Vlček [54], Kiwiel [34], e estão descritos em detalhes no Apêndice B. Esta coleção de problemas é considerada “padrão” em otimização convexa não diferenciável. Foram adicionados alguns problemas do livro de Schittkowski [63] que foram identificados com convexos. Note que estes problemas são originalmente diferenciáveis. E foram transformados em problemas não diferenciáveis ao se considerar o máximo das restrições.

Os parâmetros utilizados nos testes foram os seguintes:

$$B = I, \quad \varepsilon = 10^{-4}, \quad \mu = 0.8, \quad \varrho = 1, \quad \rho = 0.1.$$

A performance do algoritmo é avaliada com base no número de avaliações das restrições e função objetivo como em [27]. E para isso, está sendo utilizada a seguinte expressão

$$\text{RELACC} := \frac{-\log_{10}\left(\left|\frac{f(\mathbf{x}^{best}) - f(\bar{\mathbf{x}})}{f(\bar{\mathbf{x}})}\right|\right)}{\#\text{OR}_{calls}},$$

onde \mathbf{x}^{best} é o melhor ponto encontrado pelo algoritmo e $\#OR_{calls}$ é o número de avaliações necessárias para determinar o ponto \mathbf{x}^{best} . E quando $f(\bar{\mathbf{x}}) = 0$, troca-se o denominador no logaritmo por 1.

O valor da expressão RELACC pode ser interpretado como o número exato de dígitos adquiridos por chamada ao oráculo utilizada.

Por exemplo, se o RELACC for 2 então você ganha 2 dígitos exatos a mais por cada avaliação da função. Quer dizer, se a solução exata for: 1.23456789 e a solução atual for 1.2XXXXX (os X são dígitos não exatos da solução aproximada) então chamando 2 vezes o oráculo o algoritmo é capaz de melhorar a solução aproximada, que terá então 2x2 dígitos exatos a mais, ou seja, será 1.23456XX.

Os seguintes solvers foram considerados para a comparação da performance do algoritmo:

CLEVEL e NLEVEL - são duas variantes do *Level Bundle Method* em [64].

CPBFM - *Constraint Proximal Bundle Filter Method* em [27].

Todos os parâmetros utilizados foram os parâmetros sugeridos pelos respectivos códigos. E neste caso, também foram utilizados um conjunto máximo de 100 chamadas para o oráculo, ou seja, avaliações das funções objetivo, restrições e subgradientes. E uma tolerância para a determinação do ótimo igual a 10^{-4} .

Os resultados obtidos mostram que para uma comparação inicial, o algoritmo NFDCA obteve uma boa performance em relação aos três solvers acima citados. Em todo caso, o algoritmo resolveu todos os problemas testes da Tabela 4.1, onde foram utilizados os mesmos parâmetros. É claro que mais testes devem ser realizados na tentativa de melhorar o desempenho do algoritmo.

Para o NLEVEL observou-se algumas falhas, que aparecem como 0.000 na Tabela 4.3.

Tabela 4.1: Problemas Teste

No.	Problema	n	$c(x)$	Ponto Inicial	$f(x^*)$
1	Maxquad	10	lin	(0, 0, 0, 0, 0, 0, 0, 0, 0, 0)	-0.36816644
2	Rosen	4	cvx	(0, 0, 0, 0)	-44
3	Wong2	20	cvx	(2, 3, 5, 5, 1, 2, 7, 3, 6, 10)	24.306209
4	Wong3	20	cvx	(*)	133.72828
5	Prob12	2	dif	(0, 0)	-30
6	Prob29	3	dif	(1, 1, 1)	-22
7	Prob35	3	cvx	(0.5, 0.5, 0.5)	1/9
8	Prob76	4	cvx	(0.5, 0.5, 0.5, 0.5)	-4.6818
9	Prob100	7	cvx	(1, 2, 0, 4, 0, 1, 1)	680.630057
10	Prob113	10	cvx	(2, 3, 5, 5, 1, 2, 7, 3, 6, 10)	24.306209
11	Prob227	2	cvx	(0.5, 0.5)	1
12	Prob228	2	cvx	(0.0, 0.0)	-3
13	Prob264	4	cvx	(0.0, 0.0, 0.0, 0.0)	-44
14	Prob284	15	cvx	0	-1840
15	Prob285	15	cvx	0	-8252
16	MiniMax	4	cvx	(15, 22, 27, 11)	23.886767
17	MiniSum	6	cvx	(0, 0, 0, 0, 0, 0)	68.82856
18	Streit	4	cvx	(0, 0, 0, 0)	0.7071068
19	Colville	6	cvx	(0, 0, 0, 0, 0, 0)	-23.0448869
20	Ill-Conditioned LP	30	lin	0	0

(*) - (2, 3, 5, 5, 1, 2, 7, 3, 6, 10, 2, 2, 6, 15, 1, 2, 1, 2, 1, 3)

Na Tabela 4.1 utilizou-se a seguinte notação:

n - representa a dimensão do problema em questão;

lin - abreviação para linear;

cvx - abreviação para convexo;

dif - abreviação para diferenciável;

f^{opt} - o valor ótimo conhecido do problema.

Tabela 4.2: Resultados Preliminares do NFDCA

Problema	ps	pn	nf	f^*	RELACC
Maxquad	19	29	48	-0.368	0.079
Rosen	23	27	30	-44.000	0.252
Wong2	106	112	218	24.306	0.028
Wong3	175	191	366	133.728	0.018
prob12	22	4	26	-30.000	0.360
prob35	23	27	50	0.111	0.087
prob76	22	13	35	-4.682	0.155
prob100	91	127	218	680.741	0.017
prob113	65	59	124	24.282	0.024
prob227	12	4	16	1.0000	0.393
prob228	14	4	18	-3.000	0.355
prob264	32	19	51	-44.000	0.162
prob284	190	248	438	-1840.000	0.019
prob285	200	218	418	-8252.000	0.019

Abreviações: ps - o número de passos sérios;
 pn - o número de passos nulos;
 nf - o número de avaliações da restrições e função objetivo;
 f^* - o valor final da função objetivo.

Tabela 4.3: Valores do RELACC

	CLEVEL	NLEVEL	CPBFM	NFDCA
Maxquad	0.09	0.09	0.15	0.079
Rosen	0.15	0.19	0.12	0.252
prob12	0.08	0.00	0.33	0.360
prob100	0.08	0.00	0.06	0.017
prob113	0.03	0.09	0.08	0.024
prob227	0.18	0.20	0.38	0.393
prob228	0.10	0.17	1.88	0.355
prob264	0.03	0.10	0.14	0.162
prob284	0.03	0.00	0.22	0.019
prob285	0.07	0.00	0.15	0.019

Capítulo 5

Otimização Topológica Robusta de Estruturas Reticuladas

Nesta seção apresenta-se uma aplicação do algoritmo NFDA, apresentado no Capítulo 2, para o problema de otimização topológica robusta.

5.1 Introdução

A otimização estrutural pode ser aplicada a todos os tipos de estruturas, desde estruturas bem simples até uma barra engastada ou até estruturas bem mais complexas, como uma estação espacial. Essas estruturas podem ter uma, duas ou três dimensões. A princípio tratam-se apenas de estruturas reticuladas, que por sua simplicidade são mais didáticas e permitem um estudo sistemático.

Uma estrutura reticulada, ou treliça, é um dos principais tipos de estruturas de engenharia. Ela oferece, ao mesmo tempo, uma solução prática e econômica a muitas situações, especialmente no projeto de pontes e edifícios. Uma *treliça* é uma construção 2D ou 3D composta por finas barras elásticas ligadas umas as outras por *nós* - pontos de um conjunto finito, do plano ou do espaço, inicialmente considerado. Quando estas estruturas são sujeitadas a uma dada *carga* - distribuição de forças externas aplicadas sobre os *nós* - a estrutura se deforma até que as forças internas (axiais), compensem as cargas externas. A treliça deformada capacita certa energia potencial, e esta energia - a *complacência* - mede a rigidez da treliça, sua capacidade de suportar a carga.

A otimização topológica de estruturas reticuladas busca selecionar uma configuração ótima de um sistema de estruturas (mecânica, engenharia civil, aeroespacial). Este é um tema clássico que vem crescendo rapidamente ao longo dos últimos anos. O problema tradicional consiste em encontrar a configuração ótima que minimiza a energia de deformação máxima da estrutura sujeita a um

conjunto de estados de carga. As variáveis de projeto, em geral, são: o tamanho da barra, o conjunto de *coordenadas nodais* da estrutura e o volume total das barras consideradas inicialmente.

O projeto ótimo assegura o melhor comportamento possível da estrutura sobre determinadas cargas (ou forças). Contudo, pode acontecer que pequenas forças não considerados na estrutura, causem uma deformação inapropriada da estrutura. Portanto, um projeto robusto deve assegurar um comportamento razoável sobre todos as possíveis cargas de pequena magnitude.

5.2 Modelos em Otimização Topológica

No modelo supõe-se n pontos nodais (m graus de liberdade), ou seja, pontos do plano ou do espaço tridimensional. São consideradas b barras ou arestas possíveis, ou seja, conexões entre os *nós* que podem ser barras da estrutura. Estas barras podem ser todas as possíveis entre os *nós* num subconjunto delas. São dados também os apoios, isto é, as restrições ao deslocamento para cada um dos *nós*, e um ou vários estados de carga que a estrutura deve suportar. As variáveis fundamentais do problema são os volumes do material que se atribuem a cada aresta, podendo dispor também de um dado volume máximo de material. O objetivo é encontrar a distribuição ótima de material, i.e., a distribuição de volume que corresponde a melhor estrutura reticulada de acordo com um determinado critério. Na solução do problema, algumas das possíveis arestas (ou barras) podem ficar com volume de material nulo, por isso, se dizem que os modelos utilizados são de Otimização Topológica.

Denota-se a_j , l_j a área da seção transversal e o comprimento da barra de número j , respectivamente, e assume-se que todas as barras são feitas de material elástico linear com coeficiente de elasticidade (modulo de Young) E_j . Desse modo, o volume da estrutura é dado por $V = \sum_{j=1}^b a_j l_j$. Contudo, para simplificar a notação, introduz-se o volume da barra, $x_j = a_j l_j$, $j = 1, \dots, b$, como as variáveis de projeto.

Então a *condição de equilíbrio* ou *equilíbrio estático* é expressada por

$$\mathbf{B}q = p.$$

onde q o vetor de forças e p é o vetor de força nodal. A estrutura é escolhida de modo que a matriz de equilíbrio \mathbf{B} tenha posto completo e então que $b \geq m$. A *matriz de rigidez* da estrutura é escrita da seguinte forma

$$\mathbf{K}(x) = \sum_{j=1}^b x_j \mathbf{K}_j$$

onde $x_j \mathbf{K}_j$ é o elemento da matriz de rigidez para a barra de número j , escrita em coordenadas globais. Nota-se que $\mathbf{K}_j = \frac{E_j}{l_j^3} \mathbf{b}_j \mathbf{b}_j^T$ onde \mathbf{b}_j é a j -ésima coluna de \mathbf{B} .

O problema de determinar a menor complacência da estrutura para um dado volume de material tem uma formulação bem conhecida, Bendsøe [65]

$$\begin{aligned} & \underset{u,x}{\text{minimize}} && p^T u \\ & \text{sujeito a} && \sum_{j=1}^b x_j \mathbf{K}_j u = p \\ & && \sum_{j=1}^b x_j = V \\ & && x_j \geq 0, \quad j = 1, \dots, b. \end{aligned} \tag{5.1}$$

onde p representa o vetor global de cargas nodais atuando sobre a estrutura reticulada, u é o vetor global de deslocamentos nodais, \mathbf{K}_i é a contribuição da matriz de rigidez da estrutura da barra de número i calculada para um volume unitário.

Neste modelo as variáveis são os deslocamentos u e os volumes das barras expressados pela variável vetorial $x \in \mathbb{R}^b$. Como solução, temos que x obtido, é a distribuição de volume que maximiza a rigidez da estrutura relativa a carga p . A equação $\sum_{j=1}^b x_j \mathbf{K}_j u = p$ é a equação de equilíbrio do problema. A equação $\sum_{j=1}^b x_j = V$ é a uma restrição para a distribuição de volume, a qual expressa, que a soma dos volumes é igual a quantidade de volume de material que se deseja repartir entre as barras. Essa equação pode ser substituída pela seguinte equação:

$$\sum_{j=1}^b x_j \leq V, \tag{5.2}$$

e esta versão é a utilizada neste trabalho e conduz sempre a mesma solução, quando se considera a do caso anterior.

Um problema é considerado bem formulado se para áreas unitárias a matriz de rigidez é não singular, ou seja, quando se tem que

$$\sum_{j=1}^b \mathbf{K}_j \succ 0. \tag{5.3}$$

No que segue se utilizará esta notação para indicar que uma matriz é definida positiva e se indicará com o símbolo “ \succeq ” quando a matriz for semidefinida positiva.

Na referência [65], Bendsøe define o seguinte problema e mostra sua equivalência com o problema (5.1):

$$\inf_{\substack{x > 0 \\ \sum_{j=1}^b x_j = V}} \left\{ \phi(x) \equiv p^T u = \max_u \left[2p^T u - u^T \sum_{j=1}^b x_j \mathbf{K}_j u \right] \right\}. \quad (5.4)$$

A função $\phi(x)$ é a energia de deformação da estrutura quando está na configuração de equilíbrio estando aplicada a carga p . Aqui, $\phi(x)$ valerá infinito quando não existir uma configuração de equilíbrio, ou seja, a estrutura não poderá suportar a carga p (para este caso é necessário que a matriz $\mathbf{K}(x)$ seja singular, isto só pode acontecer se um ou vários dos volumes x_j for zero).

É importante notar que para um valor da variável x tal que a função $\phi(x)$ seja limitada superiormente pode ser encontrado o argumento u (não necessariamente único) para o qual se obtém o máximo de $\phi(x)$ em (5.4), por outro lado, esse argumento u verificará necessariamente a equação de equilíbrio $\sum_{j=1}^b x_j \mathbf{K}_j u = p$. Neste trabalho, se optará por um procedimento que itere na variável x (com x estritamente positivo em cada iteração) e de modo a diminuir em cada iteração o valor da função $\phi(x)$ até chegar a solução com uma certa tolerância. Em cada passo se deverá encontrar um valor de u que seja argumento máximo na equação de ϕ em (5.4) (isto sempre acontecerá, dado que o valor da função $\phi(x)$ é limitado pelo valor inicial) e portanto defina a função $\phi(x)$. Este valor de u é fácil de se achar quando x é estritamente positivo, pois segue de resolver o sistema dado pela equação (5.5) onde a matriz $\mathbf{K}(x)$ é definida positiva

$$\mathbf{K}(x)u = p. \quad (5.5)$$

Estas considerações nos permitem formular o seguinte problema:

$$\underset{x}{\text{minimize}} \phi(x) = p^T u(x) \quad (5.6)$$

$$\text{sujeito a } \sum_{j=1}^b x_j \leq V \quad (5.7)$$

$$x_j \geq 0, \quad j = 1, \dots, b. \quad (5.8)$$

Salienta-se que o problema (5.4) é convexo. Isto segue do fato de que a função complacência $\phi(x)$, é expressada como o supremo (máximo) sobre uma família de funções (lineares neste caso) convexas.

Resta agora, achar uma expressão que proporcione um subgradiente da função $\phi(x)$. Contudo para valores onde $x > 0$, tem-se que a função $\phi(x)$ é diferenciável.

Então considerando que $\phi(x) \equiv p^T u(x)$ e $\mathbf{K}(x)u(x) = p$, segue que

$$\frac{\partial \phi(x)}{\partial x_i} = p^T \frac{\partial u(x)}{\partial x_i} = u(x)^T \mathbf{K}(x) \frac{\partial u(x)}{\partial x_i} \quad (5.9)$$

e

$$\frac{\partial \mathbf{K}(x)}{\partial x_i} u(x) + \mathbf{K}(x) \frac{\partial u(x)}{\partial x_i} = 0 \quad (5.10)$$

Agora, substituindo a equação (5.10) na equação (5.9), obtém-se

$$\frac{\partial \phi(x)}{\partial x_i} = -u(x)^T \frac{\partial \mathbf{K}(x)}{\partial x_i} u(x) \quad (5.11)$$

Além disso, como $\mathbf{K}(x) = \sum_{i=1}^b x_i \mathbf{K}_i$, tem-se que $\frac{\partial \mathbf{K}(x)}{\partial x_i} = \mathbf{K}_i$.

Portanto, obtém-se que

$$\frac{\partial \phi(x)}{\partial x_i} = -u(x)^T \mathbf{K}_i u(x). \quad (5.12)$$

5.3 Modelo de Otimização Topológica Robusta

O modelo considerado aqui foi proposto inicialmente por Ben-Tal e Nemirovski em [66], para um tratamento com programação semidefinida. Nesta seção apresenta-se uma aplicação do Algoritmo NFDA (Capítulo 2) no projeto de estruturas reticuladas robustas.

Um modelo utilizado para otimização topológica considera estruturas submetidas a um conjunto de carregamentos, que chamaremos de cargas “primárias”, e olha para o volume de cada barra que minimiza a complacência estrutural. Neste trabalho é considerada a *robustez* da estrutura. Uma estrutura pode ser considerada “robusta” se é razoavelmente rígida, para os estados de carga dados (cargas primárias) e também para os estados de carga que compreendem todas as possíveis cargas de pequena magnitude (cargas secundárias) que podem eventualmente atuar sobre a mesma, Ben-tal e Nemirovski [66].

A complacência a ser minimizada é a pior possível produzida para uma carga em um dos dois conjuntos. Nesta seção realiza-se uma formulação alternativa que lida com o problema de otimização convexa não diferenciável e é resolvido com o Algoritmo NFDA.

O modelo considera estruturas com duas ou três dimensões com n nós e m graus de liberdade, submetido a um conjunto finito de condições de carregamento $P \equiv \{p^1, p^2, \dots, p^s\}$ tais que $p^i \in \mathbb{R}^m$ para $i = 1, 2, \dots, s$, e seja b o número de barras iniciais (ou possíveis). As variáveis de projeto são os volumes das barras,

denotados por x_j , $j = 1, 2, \dots, b$. A matriz de rigidez é

$$\mathbf{K}(x) = \sum_{j=1}^b x_j \mathbf{K}_j, \quad (5.13)$$

onde $\mathbf{K}_j \in \mathbb{R}^{m \times m}$, $j = 1, 2, \dots, b$, são as matrizes de rigidez reduzidas correspondentes as barras de volume unitário. Para obter um problema bem posto, a matriz $\sum_{j=1}^b \mathbf{K}_j$ deve ser definida positiva [66]. A complacência relatada para a condição de carregamento $p^i \in P$ pode ser definida como em [65]:

$$\phi(x, p^i) = \sup_u \{2u^T p^i - u^T \mathbf{K}(x)u, \quad u \in \mathbb{R}^m\}. \quad (5.14)$$

Seja $\hat{\phi}(x) = \sup_{p^i} \{\phi(x, p^i), \quad p^i \in P\}$ a pior complacência possível para o conjunto P . Um modelo de energia para otimização topológica com várias condições de carregamento pode ser colocada na seguinte forma:

$$\begin{cases} \min_{x \in \mathbb{R}^b} \hat{\phi}(x) \\ \text{s.t.} \quad \sum_{j=1}^b x_j \leq V, \\ x_j \geq 0, \quad j = 1, \dots, b \end{cases} \quad (5.15)$$

O valor $V > 0$ é a quantidade máxima de material a ser distribuída na estrutura.

Ao invés de maximizar ϕ no domínio finito P , considera-se o modelo proposto por Ben-Tal e Nemirovski [66] que maximiza ϕ no elipsoide M de condições de carregamento, definido abaixo:

$$M = \{\mathbf{Q}e \mid e \in \mathbb{R}^q, \quad e^T e \leq 1\}, \quad (5.16)$$

onde

$$[\mathbf{Q}] = [p^1, \dots, p^s, r f^1, \dots, r f^{q-s}]. \quad (5.17)$$

Os vetores $\{p^1, \dots, p^s\}$ devem ser linearmente independentes e $r f^i$, representam o i -ésimo carregamento secundário. O valor r é a magnitude do carregamento secundário e o conjunto $\{f^1, \dots, f^{q-s}\}$ deve ser escolhido como sendo uma base ortogonal de um subespaço linear ortogonal ao subespaço gerado por P . O procedimento para escolher uma base conveniente é explicado com mais detalhes no Exemplo 1.

O procedimento para uma escolha base $\{f^1, \dots, f^{q-s}\}$ conveniente será explicado depois. Para este modelo tem-se que

$$\hat{\phi}(x) = \sup_p \{\phi(x, p), \quad p \in M\}.$$

Um projeto robusto é então obtido resolvendo-se (5.15) com $\hat{\phi}(x)$ previamente definida. Em [66], está demonstrada a equivalência entre as duas seguintes expressões:

$$\tau \geq \hat{\phi}(x) \quad (5.18)$$

e

$$\mathbf{A}(\tau, x) = \begin{pmatrix} \tau \mathbf{I}^q & \mathbf{Q}^T \\ \mathbf{Q} & \mathbf{K}(x) \end{pmatrix} \succeq 0, \quad (5.19)$$

onde \mathbf{I}^q é uma matriz identidade de tamanho $\mathbb{R}^q \times \mathbb{R}^q$, $\tau \in \mathbb{R}$ e $\mathbf{A} \succeq 0$ é a notação utilizada quando a matriz \mathbf{A} é semidefinida positiva.

Como o epígrafo de $\hat{\phi}$ coincide com o conjunto $\{(\tau, x) \mid \mathbf{A}(\tau, x) \succeq 0\}$, e este último é convexo, Vandenberghe and Boyd, [67], tem-se que $\hat{\phi}$ é uma função convexa.

Desse modo para Otimização Robusta é possível utilizar o modelo proposto por Ben-Tal e Nemirovski [66], e resolver o problema (5.15) usando o algoritmo NFDA para Otimização Não Diferenciável. Note que este problema tem restrições de desigualdade lineares. Logo, para resolver este problema é necessário incluir todo o conjunto de restrições do problema inicial no problema auxiliar (PA.1). E ainda, o ponto inicial x^0 deve ser interior para a região viável definida pelas restrições de desigualdade lineares. Resta mostrar como calcular a função $\hat{\phi}$ num ponto interior x .

Segue que o problema (5.19) é equivalente a:

$$\tau \mathbf{K}(x) - \mathbf{Q}\mathbf{Q}^T \succeq 0, \quad (\text{veja [66]}). \quad (5.20)$$

Então, a equivalência entre (5.18) e (5.20) mostra que $\hat{\phi}(x)$ é igual a τ que resolve o problema:

$$\begin{cases} \min_{\tau} \tau \\ \text{s.t. } \tau \mathbf{K}(x) - \mathbf{Q}\mathbf{Q}^T \succeq 0. \end{cases} \quad (5.21)$$

Como está sendo considerado $\mathbf{K}(x) \succ 0$, tem-se um chamado Problema de autovalor generalizado. Consequentemente, $\hat{\phi}(x)$ é o maior autovalor generalizado do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$, [67]. Existem rotinas eficientes para avaliar os autovalores generalizados, [68].

Se o maior autovalor λ é simples, a função $\hat{\phi}$ é diferenciável. Contudo, se este for múltiplo, a função $\hat{\phi}$ geralmente é não diferenciável. Em ambos os casos, é possível calcular os subgradientes requeridos, veja [69–71].

São considerados quatro problemas teste. Em todos os quais o módulo de Young do material é $E = 1.0$ e o volume máximo é $V = 1.0$.

Exemplos

Exemplo 1

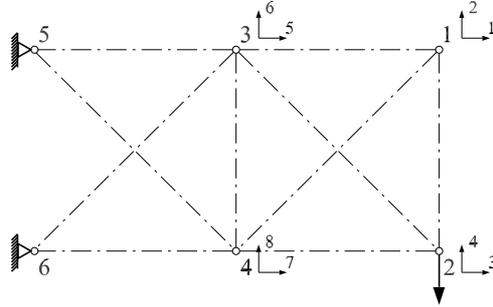


Figura 5.1: Treliça do Exemplo 1.

O primeiro exemplo considera a estrutura reticulada da Figura 5.1. O tamanho de cada barra horizontal e vertical é igual a 1.0 e a magnitude das cargas é 1.0. Os carregamentos secundários tem magnitude $r = 0.3$ e define-se uma base do complemento ortogonal do subespaço gerado por P (denotado por $L(P)$), no subespaço linear F gerado pelos graus de liberdade 2 e 4. De acordo com a numeração dos graus de liberdade da Figura 5.1, o carregamento primário e a matriz $A = [e^1, e^2, e^3, e^4]$ de vetores da base de F são:

$$p^1 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Determina-se uma base ortonormal $\{f_1, f_2, f_3\}$ do complemento ortogonal de $L(P)$ em F . Como cada um dos vetores f^i está em F , eles satisfazem $f^i = Av^i$ para algum $v^i \in \mathbb{R}^4$. Como são normais a p^1 , os vetores v^i satisfazem $(p^1)^T Av^i = 0$. Então, se pode encontrar $\{v^1, v^2, v^3\}$ como sendo uma base ortonormal do núcleo de $(p^1)^T A$. Esta base pode ser encontrada utilizando a *Decomposição por Valores*

Singulares, [72], de $(p^1)^T A$. O resultado obtido para v^i é $[v^1, v^2, v^3] = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$.

O resultado final é:

$$\mathbf{Q} = [p^1, rf^1, rf^2, rf^3] = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3 & 0 \\ 0 & 0 & 0 & 0.3 \end{pmatrix}.$$

A Figura 5.2 mostra a estrutura ótima obtida para o Exemplo 1 e a Figura 5.3 mostra a evolução dos quatro maiores autovalores dos sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$.

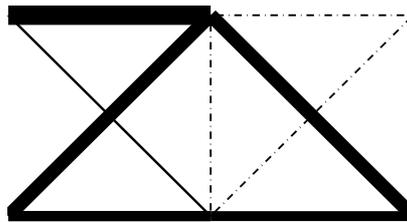


Figura 5.2: Estrutura ótima obtida no Exemplo 1.

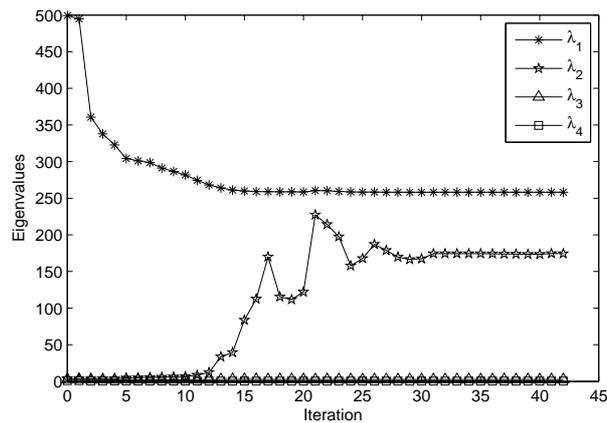


Figura 5.3: Exemplo 1 - Evolução dos quatro maiores auto-valores.

Exemplo 2

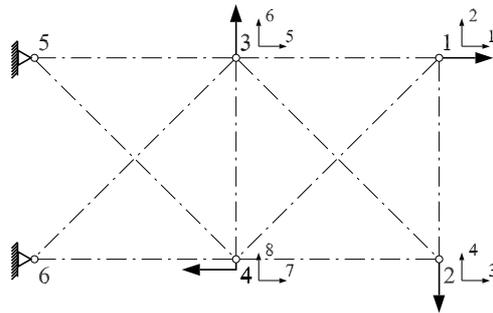


Figura 5.4: Treliça do Exemplo 2.

Este exemplo considera a mesma estrutura reticulada do Exemplo 1, e uma condição de carregamento como mostra a Figura 5.4, A magnitude das cargas primárias é 2.0 e as cargas secundárias tem magnitude $r = 0.4$ e define-se uma base do complemento ortogonal de $L(P)$ no subespaço linear de F de todos os graus de liberdade da estrutura.

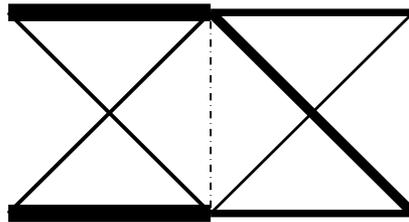


Figura 5.5: Estrutura ótima obtida no Exemplo 2.

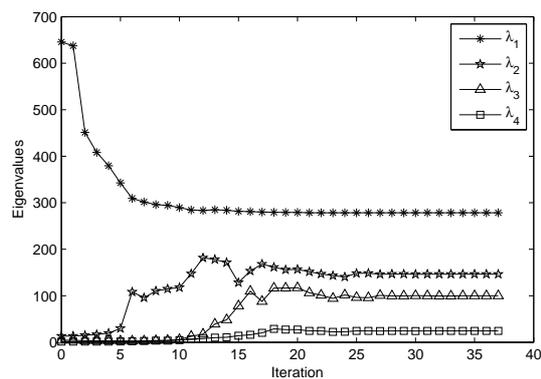


Figura 5.6: Evolução dos quatro maiores auto-valores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$.

Exemplo 3

Este exemplo consiste de uma estrutura tri-dimensional com nós fixados no plano horizontal $z = 0$ e nós livres no plano horizontal $z = 2$. A estrutura tem 8 nós de coordenadas com $N = 4$.

$$\begin{aligned} x &= \cos(2\pi i/N), & y &= \sin(2\pi i/N), & z &= 0, & i &\in \{1, \dots, N\}, \\ x &= \frac{1}{2} \cos(2\pi i/N), & y &= \frac{1}{2} \sin(2\pi i/N), & z &= 2, & i &\in \{N+1, \dots, 2N\}, \end{aligned} \quad (5.22)$$

Todas as possíveis barras entre os nós livre-livre ou livre-fixo são considerados. A condição de carregamento consiste de quatro forças atuando simultaneamente e aplicadas nos nós no plano $z = 2$. A força no nó i é com $\rho = 0.001$.

$$p_i = \left(\frac{1}{\sqrt{N(1+\rho^2)}} \right) [\sin(2\pi i/N), -\cos(2\pi i/N), -\rho]^T, \quad i \in \{N+1, \dots, 2N\}, \quad (5.23)$$

As cargas secundárias possuem uma magnitude $r = 0.3$ e definem uma base do complemento ortogonal de $L(P)$ no espaço linear F de todos os graus de liberdade da estrutura.

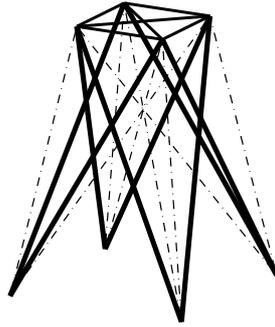


Figura 5.7: Estrutura ótima obtida no Exemplo 3

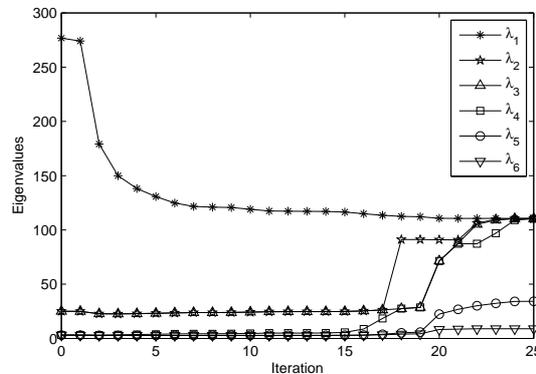


Figura 5.8: Evolução dos seis maiores autovalores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$.

Exemplo 4

Este exemplo é similar ao anterior. As coordenadas e forças nodais são dadas pelas equações (5.22) e (5.23), respectivamente, mas com $N = 5$ e $\rho = 0.01$. As cargas secundárias tem magnitude $r = 0.3$ e definem uma base para o complemento ortogonal de $L(P)$ no espaço linear F de todos os graus de liberdade da estrutura.

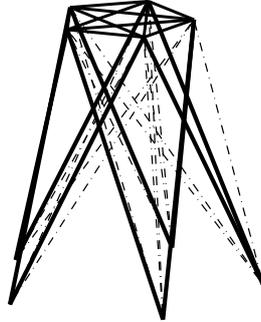


Figura 5.9: Estrutura ótima obtida no Exemplo 4.

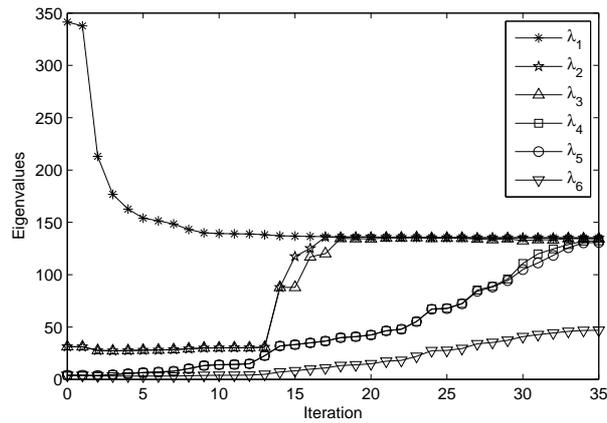


Figura 5.10: Evolução dos seis maiores auto-valores do sistema $(\mathbf{Q}\mathbf{Q}^T, \mathbf{K}(x))$.

Tabela 5.1: Resultados dos exemplos de otimização.

	NB	NI	NF	F
Exemplo 1	10	36	90	258.24
Exemplo 2	10	37	161	278.40
Exemplo 3	22	25	35	110.56
Exemplo 4	35	35	112	135.27

Notação: NB: número de barras;

NI: número de iterações;

NF: número de avaliações da função;

F: valor ótimo da função objetivo.

As Tabelas 5.1 e 5.2 mostram os resultados numéricos obtidos.

Tabela 5.2: Volumes das barras de estrutura ótima.

Exemplo 1		Exemplo 2		Exemplo 3		Exemplo 4	
barra	volume	barra	volume	barra	volume	barra	volume
5-3	2.478e-1	5-3	2.448e-1	1-6	1.247e-1	1-7	1.000e-1
6-4	1.276e-1	3-1	1.195e-1	1-8	1.246e-1	1-10	9.926e-2
4-2	1.251e-1	6-4	2.448e-1	2-5	1.246e-1	2-6	9.947e-2
5-4	3.715e-3	4-2	1.195e-1	2-7	1.247e-1	2-8	1.003e-1
6-3	2.478e-1	5-4	1.265e-2	3-6	1.246e-1	3-7	9.922e-2
3-2	2.478e-1	6-3	1.265e-2	3-8	1.247e-1	3-9	1.002e-1
		3-2	2.368e-1	4-5	1.247e-1	4-8	9.943e-2
		4-1	9.195e-3	4-7	1.246e-1	4-10	1.001e-1
				5-6	4.842e-4	5-6	1.003e-1
				5-7	4.343e-4	5-9	9.935e-2
				5-8	4.847e-4	6-7	2.573e-4
				6-7	4.847e-4	6-8	2.169e-4
				6-8	4.343e-4	6-9	2.366e-4
				7-8	4.842e-4	6-10	2.530e-4
						7-8	2.548e-4
						7-9	2.345e-4
						7-10	2.162e-4
						8-9	2.728e-4
						8-10	2.137e-4
						9-10	2.669e-4

Capítulo 6

Conclusões

Este trabalho tem como principal objetivo apresentar uma alternativa metodológica, em forma de novos algoritmos, para resolver problemas de otimização não diferenciáveis. Inicialmente, procurou-se dar continuidade ao trabalho apresentado por Freire em sua tese de doutoramento [29], onde o mesmo apresenta um algoritmo de direções viáveis e planos de corte para otimização convexa não diferenciável. Neste sentido, introduz-se um algoritmo para o problema de otimização não convexa e não diferenciável, e em seguida, apresenta-se um algoritmo para otimização convexa não diferenciável, com restrições convexas não diferenciáveis. Finalmente, considera-se o problema de Otimização Topológica de estruturas reticuladas robustas, como exemplo de aplicação desta nova metodologia por hora proposta.

6.1 Contribuições deste trabalho

Com relação ao trabalho de Freire [29], como pode ser visto no Capítulo 2, realizou-se um novo estudo com relação as hipóteses inicialmente utilizadas, e conseqüentemente, uma nova prova da convergência global do algoritmo foi necessária. Com isso, novos testes numéricos e uma comparação com alguns métodos bem conhecidos da literatura foi apresentada. Este Capítulo resultou no artigo intitulado: *A Feasible Directions Method for Nonsmooth Convex Optimization*, feito em parceria com os professores Wilhelm P. Freire (Universidade Federal de Juiz de Fora - UFJF) e Alfredo Canelas (Universidad de la República - UDELAR), que foi aceito para publicação na revista: *Structural and Multidisciplinary Optimization*, Springer.

No Capítulo 3, introduziu-se um novo algoritmo para otimização não diferenciável não convexa, chamado NFDNA - *Nonsmooth Feasible Direction Non Convex Algoritmo*. Mostou-se a convergência global para funções localmente Lipschitz contínuas. Este Capítulo resultou no artigo intitulado: *Globally Convergent Cutting Plane Method for Nonconvex Nonsmooth Minimization*, realizado em parceria com a professora Napsu Karmita (Universidade de Turku, Finlândia), que foi aceito e será publicado na revista: *Journal of Optimization Theory and Applications*: Volume 148, Issue 3 (2011), Page 528. O artigo já está disponível eletronicamente no Link: <http://www.springerlink.com/content/v680w60k616120x7>.

Um novo algoritmo para a otimização convexa não diferenciável com restrições de desigualdade convexas não diferenciáveis foi apresentado no Capítulo 4. Intitulado NFDCA - *Nonsmooth Feasible Direction Constrained Algoritmo*, o algoritmo teve sua convergência global realizada, bem como, testes numéricos e sua comparação com outros algoritmos já estabelecidos.

Esta nova metodologia tem aspectos similares ao do tradicional método de planos de corte de Kelley [10]. Contudo, no método de Kelley um conjunto compacto S contendo o ponto de mínimo de f precisa ser determinado, para garantir a existência de uma direção que minimize o problema linear auxiliar. Por outro lado, ao minimizar o problema linear auxiliar, este pode ser ilimitado inferiormente na iteração inicial. Nos métodos aqui apresentados, a direção de busca para o problema auxiliar é calculada usando o FDIPA. Com o FDIPA a direção de busca pode ser computada ainda se o problema auxiliar não tiver mínimo. Portanto, não é necessário determinar qualquer restrição artificial.

Os algoritmos: NFDA, NFDNA e NFDCA são simples de implementar. Na experiência computacional preliminar verificou-se que todos os problemas propostos foram resolvidos de forma robusta e eficiente, para cada algoritmo. A robustez, no sentido de que foram utilizados o mesmo conjunto de parâmetros, por cada algoritmo, na resolução de uma bateria de problemas teste, respectivamente proposta. Fato preponderante para a resolução de problemas de otimização voltados à Engenharia.

Ressalta-se que, a principal característica desta nova metodologia, é a não necessidade de resolver um subproblema de programação quadrática em cada iteração. Como no FDIPA, são considerados dois sistemas de equações lineares, que utilizam a mesma matriz, para determinar uma direção de descida viável, e mais, não se faz necessária a utilização de qualquer tipo de função penalidade, barreira ou filtros.

Conclui-se que o estudo realizado neste trabalho, serve de base para a introdução de uma nova metodologia para abordar problemas de otimização não diferenciáveis,

ou pelo menos de uma alternativa, através de planos de corte, pontos interiores e direções viáveis. Lembrando que hoje, a maioria dos métodos existentes se baseiam nas ideias dos métodos de feixe. Ou que, pelo menos, precisam lidar com subproblemas de programação quadrática em cada iteração.

6.2 Trabalhos Futuros

Como parte dos estudos futuros estão os seguintes tópicos:

- Estudar possíveis regras de atualização da matriz B ;
- Nenhum estudo sobre a velocidade da convergência foi realizado e este é um assunto que merece mais atenção;
- Desenvolver um algoritmo para otimização não diferenciável com restrições de desigualdade para funções localmente Lipschitz contínuas.
- A ausência do subproblema quadrático e a simples resolução de sistemas lineares internos (como no FDIPA) com a mesma matriz, alude a possibilidade de lidarmos com problemas de larga escala. Neste contexto, já existem alguns trabalhos, como [73] onde os autores descrevem um método de feixe com pontos interiores com memória limitada. Em 2008, Overton e Lewis [74] utilizaram uma formulação BFGS para lidar com problemas não diferenciáveis, e mais recentemente, [75] propôs uma formulação BFGS com uma busca linear inexata e memória limitada para problemas de larga escala.
- O FDIPA se comporta bem na solução de problemas não lineares, e portanto pode ser interessante adicionar um termo quadrático estabilizante, similar aos usados por métodos de feixe tradicionais, aos modelos aqui apresentados;
- Aplicações na Engenharia: Explorar mais problemas que possam ser resolvidos com os algoritmos aqui apresentados, como por exemplo:

Minimização do maior autovalor. Muitas aplicações requerem a solução do problema

$$\text{minimize } f(\mathbf{x}) := \lambda_{\max}(\mathbf{A}(\mathbf{x})); \quad (\text{P}_\lambda)$$

onde $\mathbf{A}(\cdot)$ é uma matriz real simétrica $m \times m$, que depende linearmente de $\mathbf{x} \in \mathbb{R}^n$, e $\lambda_{\max}(\mathbf{A}(\mathbf{x}))$ denota o maior autovalor de $\mathbf{A}(\mathbf{x})$. As seguintes propriedades são válidas (veja [19]):

- f é convexa;

- f é não diferenciável em \mathbf{x} , se o maior autovalor $f(\mathbf{x})$ tem multiplicidade maior que 1;
- se \mathbf{v} é um autovetor de $\mathbf{A}(\mathbf{x})$ para o autovalor $f(\mathbf{x})$ e $\|\mathbf{v}\|_2 = 1$, então um subgradiente de f em \mathbf{x} pode ser calculado utilizando o produto interno $\mathbf{v}\mathbf{v}^T$.

Otimização estrutural com autovalores. A maximização do autovalor fundamental de uma estrutura é um problema clássico da engenharia estrutural. Contudo, o problema de autovalores generalizado é tipicamente na forma:

$$\mathbf{K}(\mathbf{x})\mathbf{w} = \lambda\mathbf{M}(\mathbf{x})\mathbf{w},$$

onde $\mathbf{K}(\mathbf{x})$ e $\mathbf{M}(\mathbf{x})$ são matrizes simétricas, semidefinida positivas, que dependem continuamente do parâmetro \mathbf{x} . Este problema vem sendo extensivamente tratado na literatura em engenharia desde os anos 60, veja [76]. A principal dificuldade é a dependência não diferenciável do autovalores como função da variável de projeto \mathbf{x} . Em Otimização Topológica, existe ainda uma outra dificuldade que é o fato das componentes de \mathbf{x} poderem ser zero. Fato que pode tornar as matrizes correspondentes $\mathbf{K}(\mathbf{x})$ e $\mathbf{M}(\mathbf{x})$ singulares. O problema de otimização de autovalores generalizados também é um problema clássico de Álgebra Linear, quando a matriz $\mathbf{M}(\mathbf{x})$ é positiva definida para todo \mathbf{x} , veja [77].

- Otimização multi-objetivo: Estender as ideias aqui apresentadas para considerar problemas da forma:

$$\begin{cases} \text{minimize} & f(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_s(\mathbf{x}))^T \\ \text{sujeito a} & g_i(\mathbf{x}) \leq 0 \text{ para } i = 1, \dots, m. \end{cases} \quad (\text{PMO})$$

onde a função objetivo $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ e as funções de restrição $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ são supostas convexas.

Referências Bibliográficas

- [1] FLETCHER, R. *Practical Methods of Optimization*. Jhon Wiley and Sons, 1987.
- [2] MOREAU, J., PANAGIOTOPOULOS, P., STRANG, G. *Topics in Nonsmooth Mechanics*. Birkhäuser Verlag, Basel, 1988.
- [3] OUTRATA, J., KOČVARA, M., ZOWE, J. *Nonsmooth Approach to Optimization Problems With Equilibrium Constraints. Theory, Applications and Numerical Results*. Kluwer Academic Publishers, 1998.
- [4] CLARKE, F., LEDYAEV, Y. S., STERN, R., et al. *Nonsmooth Analysis and Control Theory*. Springer, 1998.
- [5] MÄKELÄ, M. M. “Survey of Bundle Methods for Nonsmooth Optimization”, *Optimization Methods and Software*, v. 17, pp. 1–29, 2001.
- [6] K. MIETTINEN, MÄKELÄ, M., MÄNNIKKÖ, T. “Optimal control of continuous casting by nondifferentiable multiobjective optimization”, *Computational Optimization and Applications*, v. 11, pp. 177 – 194, 1998.
- [7] LEMARÉCHAL, C. “Optimization”. cap. VII - Nondifferentiable optimization, pp. 529–572, Elsevier North-Holland, inc., New York, 1989.
- [8] HASLINGER, J., NEITTAANMÄKI, P. *Finite Element Approximation for Optimal Shape, Material and Topology Design*. J. Wiley & Sons, 1996.
- [9] ROCKAFELLAR, R. T. *Convex Analysis*. Princeton University Press, 1970.
- [10] KELLEY, J. J. E. “The cutting-plane method for solving convex programs”, *Journal of the Society for Industrial and Applied Mathematics*, v. 8, pp. 703–712, 1960.
- [11] KIWIEL, K. C. *Methods of Descent for Nondifferentiable Optimization*. Springer-Verlag, 1985.
- [12] LEMARÉCHAL, C. *Nonsmooth Optimization and Descent Methods*. Relatório técnico, IIASA - report, 1978.

- [13] MÄKELÄ, M., NEITTAANMÄKI, P. *Nonsmooth Optimization, Analysis and Algorithms with Applications to Optimal Control*. Word Scientific Publishing, 1992.
- [14] IZMAILOV, A. F., SOLODOV, M. V. *Otimização Volume 2 - Métodos Computacionais*. IMPA, 2007.
- [15] BONNANS, J. F., GILBERT, J. C., LEMARÉCHAL, C., et al. *Numerical Optimization: Theoretical and Practical Aspects*. Springer Verlag, 2003.
- [16] CLARKE, F. H. *Optimization and Nonsmooth Analysis*. John Wiley and Sons, 1983.
- [17] LEMARÉCHAL, C. “Combining Kelley’s and conjugate gradient method”. In: *Abstract of IX International Symposium on Mathematical Programming, Budapest, Hungary*, 1976.
- [18] HIRIART URRUTY, J. B., LEMARÉCHAL, C. *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*. Springer-Verlag, 1993.
- [19] SCHRAMM, H., ZOWE, J. “A version of the bundle idea for minimizing a nonsmooth functions: conceptual idea, convergence analysis, numerical results”, *SIAM Journal on Optimization*, v. 2, pp. 121–152, 1992.
- [20] FUDULI, A., GAUDIOSO, M., GIALLOMBARDO, G. “Minimizing nonconvex nonsmooth functions via cutting planes and proximity control”, *SIAM Journal Optimization*, v. 14, pp. 743–756, 2004.
- [21] FRANGIONI, A. “Generalized bundle methods”, *SIAM Journal on Optimization*, v. 13(1), pp. 117–156, 2002.
- [22] KIWIEL, K. “A constraint linearization method for nondifferentiable convex minimization”, *Numerische Mathematik*, v. 51, pp. 395–414, 1987.
- [23] AUSLENDER, A. “Numerical methods for nondifferentiable convex optimization”, *Mathematical Programming Study*, v. 30, pp. 102–126, 1987.
- [24] MIFFLIN, R. “A Modification and an extension of Lemaréchal’s algorithm for nonsmooth minimization”, *Mathematical Programming Study*, v. 17, pp. 77–90., 1982.

- [25] KIWIEL, K. C. “Exact penalty functions in proximal bundle methods for constrained convex nondifferentiable minimization”, *Mathematical Programming*, v. 52, pp. 285–302, 1991.
- [26] FLETCHER, R., LEYFFER, S. *A bundle filter method for nonsmooth nonlinear optimization*. Relatório técnico, Numerical Analysis Report NA/195, Dundee University, 1999.
- [27] KARAS, E., RIBEIRO, A., SAGASTIZÁBAL, C., et al. “A Bundle-Filter Method for Nonsmooth Convex Constrained Optimization”, *Mathematical Programming, série B*, v. 116, pp. 297–320, 2009.
- [28] FLETCHER, R., LEYFFER, S. “Nonlinear programming without a penalty function”, *Mathematical Programming*, v. 91, pp. 239–269, 2002.
- [29] FREIRE, P. W. *Um Algoritmo de Direções Viáveis para Otimização Convexa Não Diferenciável*. Tese de Doutorado, COPPE/PEM - Universidade Federal do Rio de Janeiro, 2005.
- [30] HERSKOVITS, J. “Feasible direction interior-point technique for nonlinear optimization”, *JOTA Journal of Optimization Theory and Applications*, v. 99, pp. 121–146, 1998.
- [31] HIRIART URRUTY, J. B., LEMARÉCHAL, C. *Convex Analysis and Minimization Algorithms I: Fundamentals*. Springer-Verlag, 1993.
- [32] IZMAILOV, A. F., SOLODOV, M. *Otimização - Volume 1. Condições de Otimalidade, Elementos de Análise Convexa e de Dualidade*. IMPA, 2005.
- [33] ROCKAFELLAR, R. “Monotone operators and the proximal point algorithm”, *SIAM Journal on Optimal Control and Optimization*, v. 14, pp. 877–898, 1976.
- [34] KIWIEL, K. “Proximity control in bundle methods for convex nondifferentiable optimization”, *Mathematical Programming*, v. 46, pp. 105–122, 1990.
- [35] LEVENBERG, K. “A method for the solution of certain nonlinear problems in least squares”, *Quart. Appl. Math*, v. 2, pp. 164–166, 1944.
- [36] MARQUARDT, D. “An algorithm for least-squares estimation of nonlinear parameters”, *SIAM Journal on Applied Mathematics*, v. 11, pp. 431–441, 1963.

- [37] LEMARÉCHAL, C., SAGASTIZÁBAL, C. “Variable metric bundle methods: from conceptual to implementable forms”, *Mathematical Programming*, v. 76, pp. 393–410, 1997.
- [38] J.R. BIRGE, L. Q., WEI, Z. “Convergence analysis of some methods for minimizing a nonsmooth convex function”, *Journal of Optimization Theory and Applications*, v. 97, pp. 357–383, 1998.
- [39] BERTSEKAS, D. *Nonlinear Programming*. Athena Scientific, 1995.
- [40] LEMARÉCHAL, C. “Numerical experiments in nonsmooth optimization. In: E.A. Nurminski (Ed.)”, *Progress in Nondifferentiable Optimization*, v. IIASA-report, Laxenburg, Austria, pp. 61–84, 1982.
- [41] GAUDISO, M., MONACO, M. F. “Variants to the cutting plane approach for convex nondifferentiable optimization”, *Optimization*, v. 25, pp. 65–75, 1992.
- [42] LUKSĀN, L., VLCĚK, J. “Globally convergent variable metric method for convex nonsmooth unconstrained minimization”, *Journal of Optimization Theory and Applications*, v. 102, pp. 593–613, 1999.
- [43] LUKSĀN, L., VLCĚK, J. “A Bundle-Newton Method for Nonsmooth Unconstrained Minimization”, *Mathematical Programming*, v. 83, pp. 373–391, 1998.
- [44] KIWIEL, K. “An algorithm for nonsmooth convex minimization with errors”, *Mathematics of Computation*, v. 45, pp. 173–180, 1985.
- [45] KIWIEL, K. “Approximations in proximal bundle methods and decomposition of convex programs”, *Journal of Optimization Theory and Applications*, v. 84, pp. 529–548, 1995.
- [46] WOLFE, P. “A method of conjugate subgradients for minimizing nondifferentiable functions”, *Mathematical Programming Study*, v. 3, pp. 145–173, 1975.
- [47] GAUDIOSO, M., MONACO, M. F. “A bundle type approach to the unconstrained minimization of convex nonsmooth functions”, *Mathematical Programming*, v. 23, pp. 216–226, 1982.
- [48] LEMARECHAL, C., STRODIOT, J.-J., BIHAIN, A. “Nonlinear Programming”. cap. On a bundle algorithm for nonsmooth optimization, pp. 245–281, Academic Press, 1981.

- [49] PSHENICHNYI, B., DANILIN, Y. M. *Numerical Methods for Extremal Problems*. Mir, Moscow, 1978.
- [50] HERSKOVITS, J. “A two-stage feasible directions algorithm for nonlinear constrained optimization”, *Mathematical Programming*, v. 36, pp. 19–38, 1986.
- [51] HERSKOVITS, J. “Advances in Structural Optimization”. cap. A View on Nonlinear Optimization, pp. 71–116, Kluwer Academic Publishers, 1995.
- [52] PANIER, E., TITS, A., HERSKOVITS, J. “A QP-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization”, *SIAM Journal on Control and Optimization*, v. 26, pp. 788–811, 1988.
- [53] TITS, A., WÄCHTER, A., BAKHTIARI, S., et al. “A primal-dual interior-point method for nonlinear programming with strong global and local convergence properties”, *SIAM Journal on Optimization*, v. 14, pp. 173–199, 2003.
- [54] LUKŠAN, L., VLČEK, J. *Test Problems for Nonsmooth Unconstrained and Linearly Constrained Optimization, N-798*. Relatório técnico, Institute of Computer Science, Academy of Sciences of the Czech Republic, 2000.
- [55] DOLAN, E. D., MOREÉ, J. J. “Benchmarking optimization software with performance profile”, *Mathematical Programming*, v. 91, pp. 201–213, 2002.
- [56] LEMARÉCHAL, C., IMBERT, M. B. *Le module M1FC1*. Relatório técnico, Institut de Recherche d’Informatique, Le Chesnay, 1985.
- [57] CHENEY, E. W., GOLDSTEIN, A. A. “Newton’s method for convex programming and Tchebycheff approximation”, *Numeriche Mathematic*, v. 1, pp. 253–268, 1959.
- [58] FUDULI, A., GAUDIOSO, M., GIALLOMBARDO, G. “A DC piecewise affine model and a bundling technique in nonconvex nonsmooth minimization”, *Optimization Methods and Software*, v. 25, pp. 89–102, 2004.
- [59] GORGONE, E. *Nonsmooth optimization: theory and algorithms*. Tese de Doutorado, Università della Calabria, 2008.
- [60] KIWIEL, K. C. “A subgradient selection method for minimizing convex functions subject to linear constraints”, *Computing*, v. 39, pp. 293–305, 1987.

- [61] SAGASTIZÁBAL, C., SOLODOV, M. “An Infeasible Bundle Method for Nonsmooth Convex Constrained Optimization without a Penalty Function or Filter”, *SIAM Journal on Optimization*, v. 16, pp. 146–169, 2005.
- [62] MANGASARIAN, O. L. *Nonlinear Programming*. McGraw-Hill, 1969.
- [63] HOCK, W., SCHITTKOWSKI, K. *Test Examples for Nonlinear Programming Codes (Lecture Notes in Economics and Mathematical Systems, N 187)*. Springer-Verlag, 1981.
- [64] LEMARÉCHAL, C., NEMIROVSKI, A. “New variants of bundle methods”, *Mathematical Programming*, v. 69, pp. 111–148, 1995.
- [65] BENDSØE, M. P. *Optimization of structural topology, shape, and material*. Berlin, Springer-Verlag, 1995. ISBN: 3-540-59057-9.
- [66] BEN-TAL, A., NEMIROVSKI, A. “Robust truss topology design via semidefinite programming”, *SIAM Journal on Optimization*, v. 7, n. 4, pp. 991–1016, 1997. ISSN: 1052-6234.
- [67] VANDENBERGHE, L., BOYD, S. “Semidefinite Programming”, *SIAM Review*, v. 38, n. 1, pp. 49–95, 1996.
- [68] Bai, Z., Demmel, J., Dongarra, J., et al. (Eds.). *Templates for the solution of algebraic eigenvalue problems*, v. 11, *Software, Environments, and Tools*. Philadelphia, PA, Society for Industrial and Applied Mathematics (SIAM), 2000. ISBN: 0-89871-471-0. A practical guide.
- [69] SEYRANIAN, A. P., LUND, E., OLHOFF, N. “Multiple eigenvalues in structural optimization problems”, *Structural Optimization*, v. 8, n. 4, pp. 207–227, 1994.
- [70] RODRIGUES, H. C., GUEDES, J. M., BENDSØE, M. P. “Necessary conditions for optimal design of structures with a nonsmooth eigenvalue based criterion”, *Structural Optimization*, v. 9, n. 1, pp. 52–56, 1995.
- [71] CHOI, K. K., KIM, N. H. *Structural sensitivity analysis and optimization 1: linear systems*. Springer, 2004.
- [72] GOLUB, G. H., VAN LOAN, C. F. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Third ed. Baltimore, MD, Johns Hopkins University Press, 1996. ISBN: 0-8018-5413-X; 0-8018-5414-8.

- [73] KARMITSA, N., MÄKELÄ, M. M., ALI, M. M. “Limited memory interior point bundle method for large inequality constrained nonsmooth minimization”, *Applied Mathematics and Computation*, v. 198, Issue 1, pp. 382–400, 2008.
- [74] LEWIS, A. S., OVERTON, M. L. “Nonsmooth Optimization via BFGS”, *optimization online*, 2008.
- [75] SKAJAA, A. *Limited Memory BFGS for Nonsmooth Optimization*. Tese de Mestrado, Courant Institute of Mathematical Science, 2010.
- [76] ACHTZIGER, W., KOČVARA, M. “Structural topology optimization with eigenvalues”, *SIAM J. Optim.*, v. 18, n. 4, pp. 1129–1164, 2007. ISSN: 1052-6234. doi: 10.1137/060651446. Disponível em: <<http://dx.doi.org/10.1137/060651446>>.
- [77] BOYD, S., GHAOUI, L., FERON, E., et al. *Linear matrix inequalities in system and control theory*. SIAM, 1994.
- [78] LEMARÉCHAL, C., MIFFLIN, R. *Nonsmooth Optimization*. Pergamon Press, 1978.

Apêndice A

Problemas Irrestritos

Neste Apêndice apresenta-se uma bateria de problemas testes que podem ser encontrados nos trabalhos de [13, 54].

Tabela A.1: Tabela de Problemas

No.	Problema	n	Convexo	f^{opt}
1	Rosenbrock	2	–	0
2	Crescent	2	–	0
3	CB2	2	+	1.9522245
4	CB3	2	+	2
5	DEM	2	+	-3
6	QL	2	+	7.20
7	LQ	2	+	-1.4142136
8	Mifflin1	2	+	-1
9	Mifflin2	2	–	-1
10	Wolfe	2	+	-8
11	Rosen	4	+	-44
12	Shor	5	+	22.600162
13	Colville 1	5	–	-32.348679
14	HS78	5	–	-2.9197004
15	El-Attar	6	–	0.5598131
16	Maxquad	10	+	-0.8414083
17	Gill	10	–	9.7857
18	Steiner 2	12	+	16.703838
19	Maxq	20	+	0
20	Maxl	20	+	0
21	TR48	48	+	-638565.0
22	Goffin	50	+	0
23	MXHILB	50	+	0
24	L1HILB	50	+	0
25	Shell Dual	15	–	32.348679

A.1 Problemas Teste Convexos

Abaixo estão descritos os problema testes **convexos** que foram utilizados, para verificar a performance do algoritmo. Testou-se a confiabilidade do código através de um conjunto de problemas acadêmicos amplamente utilizados para testar novos algoritmos.

CB2

Dimensão	2
Função objetivo	$f(x) = \max\{x_1^2 + x_2^4, (2 - x_1)^2 + (2 - x_2)^2, 2e^{-x_1+x_2}\}$
Ponto ótimo	$x^* = (1.139286, 0.899365)$
Valor ótimo	$f(x^*) = 1.9522245$
Ponto inicial	$x_0 = (1, -0.1), f(x_0) = 5.41.$

CB3

Dimensão	2
Função objetivo	$f(x) = \max\{x_1^4 + x_2^2, (2 - x_1)^2 + (2 - x_2)^2, 2e^{-x_1+x_2}\}$
Ponto ótimo	$x^* = (1, 1)$
Valor ótimo	$f(x^*) = 2$
Ponto inicial	$x_0 = (2, 2).$

DEM

Dimensão	2
Função objetivo	$f(x) = \max\{5x_1 + x_2, -5x_1 + x_2, x_1^2 + x_2^2 + 4x_2\}$
Ponto ótimo	$x^* = (0, -3)$
Valor ótimo	$f(x^*) = -3$
Ponto inicial	$x_0 = (1, 1), f(x_0) = 6.$

QL

Dimensão	2
Função objetivo	$f(x) = \max\{f_1(x), f_2(x), f_3(x)\}$
onde	$f_1(x) = x_1^2 + x_2^2$ $f_2(x) = x_1^2 + x_2^2 + 10(-4x_1 - x_2 + 4)$ $f_3(x) = x_1^2 + x_2^2 + 10(-x_1 - 2x_2 + 6)$
Ponto ótimo	$x^* = (1.2, 2.4)$
Valor ótimo	$f(x^*) = 7.2$
Ponto inicial	$x_0 = (-1, 5), f(x_0) = 56.$

LQ

Dimensão	2
Função objetivo	$f(x) = \max\{-x_1 - x_2, -x_1 - x_2 + (x_1^2 + x_2^2 - 1)\}$
Ponto ótimo	$x^* = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$
Valor ótimo	$f(x^*) = -\sqrt{2}$
Ponto inicial	$x_0 = (-0.5, -0.5)$.

Mifflin1

Dimensão	2
Função objetivo	$f(x) = -x_1 + 20 \max\{x_1^2 + x_2^2 - 1, 0\}$
Ponto ótimo	$x^* = (1, 0)$
Valor ótimo	$f(x^*) = -1$
Ponto inicial	$x_0 = (0.8, 0.6), f(x_0) = -0.8$.

Rosen-Suzuki

Dimensão	4
Função objetivo	$f(x) = \max\{f_1, f_1 + 10f_2, f_1 + 10f_3, f_1 + 10f_4\}$
onde	$f_1(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$ $f_2(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8$ $f_3(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10$ $f_4(x) = x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5$
Ponto ótimo	$x^* = (0, 1, 2, -1)$
Valor ótimo	$f(x^*) = -44$
Ponto inicial	$x_0 = (0, 0, 0, 0), f(x_0) = 0..$

Shor

Dimensão	5
Função objetivo	$f(x) = \max_{1 \leq i \leq 10} \{b_i \sum_{j=1}^5 (x_j - a_{ij})^2\}$
Valor ótimo	$f(x^*) = 22.60016$
Ponto inicial	$x_0 = (0, 0, 0, 0, 1), f(x_0) = 80$

onde

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 1 & 1 & 3 \\ 1 & 2 & 1 & 1 & 2 \\ 1 & 4 & 1 & 2 & 2 \\ 3 & 2 & 1 & 0 & 1 \\ 0 & 2 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 & 1 \\ 0 & 0 & 2 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 \end{bmatrix} \quad \text{e} \quad b = \begin{bmatrix} 1 \\ 5 \\ 10 \\ 2 \\ 4 \\ 3 \\ 1.7 \\ 2.5 \\ 6 \\ 3.5 \end{bmatrix}.$$

Maxquad

Dimensão 10

Função objetivo $f(x) = \max_{1 \leq i \leq 5} \{x^T A^i x - x^T b^i\}$

$$A_{kj}^i = A_{jk}^i = e^{j/k} \cos(jk) \sin(i), \quad j < k,$$

$$A_{jj}^i = \frac{j}{10} |\sin(i)| + \sum_{k \neq j} |A_{jk}^i|,$$

$$b_j^i = e^{j/i} \sin(ij),$$

Valor ótimo $f(x^*) = -0.8414084$

Ponto inicial 1 $x_0(i) = 1, \quad i = 1, \dots, 10, \quad f(x_0) = 5337.1$

Ponto inicial 2 $x_0(i) = 0, \quad i = 1, \dots, 10, \quad f(x_0) = 0$

Maxq

Dimensão 20

Função objetivo $f(x) = \max_{1 \leq i \leq 20} x_i^2$

Valor ótimo $f(x^*) = 0$

Ponto inicial $x_0(i) = i, \quad i = 1, \dots, 10, \quad x_0(i) = -i, \quad i = 11, \dots, 20$
 $f(x_0) = 400.$

Maxl

Dimensão 20

Função objetivo $f(x) = \max_{1 \leq i \leq 20} |x_i|$

Valor ótimo $f(x^*) = 0$

Ponto inicial $x_0(i) = i, \quad i = 1, \dots, 10, \quad x_0(i) = -i, \quad i = 11, \dots, 20$
 $f(x_0) = 20.$

TR48

Dimensão	48
Função objetivo	$f(x) = \sum_{j=1}^{48} d_j \max_{1 \leq i \leq 48} \{x_i - a_{ij}\} - \sum_{i=1}^{48} s_i x_i$
Valor ótimo	$f(x^*) = -638565$
Ponto inicial	$x_0(i) = 0, \quad i = 1, \dots, 48, \quad f(x_0) = -464816.$
Os coeficientes a_{ij}, s_i, d_j são dados em [78].	

Goffin

Dimensão	50
Função objetivo	$f(x) = 50 \max_{1 \leq i \leq 50} x_i - \sum_{j=1}^{50} x_j$
Valor ótimo	$f(x^*) = 0$
Ponto inicial	$x_0(i) = i - 25.5, \quad i = 1, \dots, 50, \quad f(x_0) = 1225.$

Wolfe

Dimensão	2
Função objetivo	$f(x) = \begin{cases} 5\sqrt{(9x_1^2 + 16x_2^2)}, & x_1 \geq x_2 \\ 9x_1 + 16 x_2 , & 0 < x_1 < x_2 \\ 9x_1 + 16 x_2 - x_1^9, & x_1 \leq 0. \end{cases}$
Valor ótimo	$f(x^*) = -8$
Ponto inicial	$x_0 = (3, 2).$

Steiner 2

Dimensão	12
Função objetivo	$f(x) = \sqrt{x_1^2 + x_{1+m}^2} + \sqrt{(\bar{a}_{21} - x_m)^2 + (\bar{a}_{22} - x_{2m})^2} +$ $+ \sum_{j=1}^m p_j \sqrt{(a_{j1} - x_j)^2 + (a_{j2} - x_{j+m})^2} +$ $+ \sum_{j=1}^{m-1} \tilde{p}_j \sqrt{(x_j - x_{j+1})^2 + (x_{j+m} - x_{j+m+1})^2}, \quad m = 6$

onde

$$\begin{array}{llll}
 \bar{a}_{21} = 5.5, & \bar{a}_{22} = -1.0 & & \\
 a_{11} = 0.0, & a_{12} = 2.0, & p_1 = 2, & \tilde{p}_1 = 1 \\
 a_{21} = 2.0, & a_{22} = 3.0, & p_2 = 1, & \tilde{p}_2 = 1 \\
 a_{31} = 3.0, & a_{32} = -1.0, & p_3 = 1, & \tilde{p}_3 = 2 \\
 a_{41} = 4.0, & a_{42} = -0.5, & p_4 = 5, & \tilde{p}_4 = 3 \\
 a_{51} = 5.0, & a_{52} = 2.0, & p_5 = 1, & \tilde{p}_5 = 2 \\
 a_{61} = 6.0, & a_{62} = 2.0, & p_6 = 1 & .
 \end{array}$$

Valor ótimo $f(x^*) = 16.703838$
 Ponto inicial x_0 é tal que:

$$\begin{array}{ll}
 x_{0,1} = (a_{11} + a_{21})/3 & x_{0,(1+m)} = (a_{12} + a_{22})/3 \\
 x_{0,j} = (\bar{x}_{j-1} + a_{j1} + a_{(j+1)1})/3 & x_{0,(j+m)} = (\bar{x}_{j-1+m} + a_{j2} + a_{(j+1)2})/3, \quad 2 \leq j \leq m-1 \\
 x_{0,m} = (\bar{x}_{m-1} + a_{m1} + \bar{a}_{21})/3 & x_{0,(2m)} = (\bar{x}_{2m-1} + a_{m2} + \bar{a}_{22})/3
 \end{array}$$

MXHILB

Dimensão 50
 Função objetivo $f(x) = \max_{1 \leq i \leq 50} \left| \sum_{j=1}^{50} \frac{x_j}{i+j-1} \right|$
 Valor ótimo $f(x^*) = 0$
 Ponto inicial $x_0(i) = 1, i = 1, \dots, 50.$

L1HILB

Dimensão 50
 Função objetivo $f(x) = \sum_{i=1}^{50} \left| \sum_{j=1}^{50} \frac{x_j}{i+j-1} \right|$
 Valor ótimo $f(x^*) = 0$
 Ponto inicial $x_0(i) = 1, i = 1, \dots, 50.$

A.2 Problemas Teste Não Convexos

Nesta seção são apresentados os problema testes **Não Convexos**. Todos os problemas, exceto o Rosenbrock, são não diferenciáveis.

Rosenbrock

Dimensão	2
Função objetivo	$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$
Ponto ótimo	$x^* = (1, 1)$
Valor ótimo	$f(x^*) = 0$
Ponto inicial	$x^0 = (-1.2, 1)$.

Crescente

Dimensão	2
Função objetivo	$f(x) = \max\{f_1(x), f_2(x)\}$
onde	$f_1(x) = x_1^2 + (x_2 - 1)^2 + x_2 - 1$ $f_2(x) = -x_1^2 - (x_2 - 1)^2 + x_2 + 1$
Ponto ótimo	$x^* = (0, 0)$
Valor ótimo	$f(x^*) = 0$
Ponto inicial	$x^0 = (-1.5, 2)$.

Mifflin2

Dimensão	2
Função objetivo	$f(x) = -x_1 + 2(x_1^2 + x_2^2 - 1) + 1.75 x_1^2 + x_2^2 - 1 $
Ponto ótimo	$x^* = (1, 0)$
Valor ótimo	$f(x^*) = -1$
Ponto inicial	$x^0 = (-1, -1)$.

El-Attar

Dimensão	6
Função objetivo	$f(x) = \sum_{i=1}^{51} x_1 e^{-x_2 t_i} \cos(x_3 t_i + x_4) + x_5 e^{-x_6 t_i} - y_i $ $y_i = 0.5e^{-t_i} - e^{-2t_i} + 0.5e^{-3t_i} + 1.5e^{-1.5t_i} \sin(7t_i) + e^{-2.5t_i} \sin(5t_i),$ $t_i = 0.1(i - 1), \quad 1 \leq i \leq 51$
Valor ótimo	$f(x^*) = 0.5598131$
Ponto inicial	$x^0 = (2, 2, 7, 0, -2, 1)$

Colville 1

Dimensão	5
Função objetivo	$f(x) = f_1 + f_2 + f_3 + f_4 + f_5$ $f_1(x) = \sum_{j=1}^5 d_j x_j^3, \quad f_2(x) = \sum_{i=1}^5 \sum_{j=1}^5 c_{ij} x_i x_j, \quad f_3(x) = \sum_{j=1}^5 e_j x_j$ $f_4(x) = 50 \max \left[0, \max_{1 \leq i \leq 10} \left(b_i - \sum_{j=1}^5 a_{ij} x_j \right) \right]$

onde

$$A = \begin{bmatrix} -16 & 2 & 0 & 1 & 0 \\ 0 & -2 & 0 & 4 & 2 \\ -3.5 & 0 & 2 & 0 & 0 \\ 0 & -2 & 0 & -4 & -1 \\ 0 & -9 & -2 & 1 & -2.8 \\ 2 & 0 & -4 & 0 & 0 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -2 & -3 & -2 & -1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} -40 \\ -2 \\ -0.25 \\ -4 \\ -4 \\ -1 \\ -40 \\ -60 \\ 5 \\ 1 \end{bmatrix},$$

$$C = \begin{bmatrix} 30 & -20 & -10 & 32 & -10 \\ -20 & 39 & -6 & -31 & 32 \\ -10 & -6 & 10 & -6 & -10 \\ 32 & -31 & -6 & 39 & -20 \\ -10 & 32 & -10 & -20 & 30 \end{bmatrix}, \quad d = \begin{bmatrix} 4 \\ 8 \\ 105 \\ 6 \\ 2 \end{bmatrix},$$

$$e^T = [-15 \quad -27 \quad -36 \quad -18 \quad -12],$$

Valor ótimo $f(x^*) = -32.348679$

Ponto inicial $x^0 = (0, 0, 0, 0, 1)$.

Gill

Dimensão 10

Função objetivo $f(x) = \max\{f_1(x), f_2(x), f_3(x)\}$,

$$f_1(x) = \sum_{i=1}^{10} (x_i - 1)^2 + 10^{-3} \sum_{i=1}^{10} \left(x_i^2 - \frac{1}{4}\right)^2,$$

$$f_2(x) = \sum_{i=2}^{30} \left[\sum_{j=2}^{10} x_j (j-1) \left(\frac{i-1}{29}\right)^{j-2} - \left(\sum_{j=1}^{10} x_j \left(\frac{i-1}{29}\right)^{j-1} \right)^2 - 1 \right]^2 + x_1^2 + (x_2 - x_1^2 - 1)^2,$$

$$f_3(x) = \sum_{i=2}^{10} [(x_i - x_{i-1}^2)^2 + (1 - x_i)^2],$$

Valor ótimo $f(x^*) = 9.7857721$

Ponto inicial $x_0(i) = -0.1, 1 \leq i \leq 10$.

HS78

Dimensão 5

Função objetivo $f(x) = x_1 x_2 x_3 x_4 x_5 + 10 \sum_{i=1}^3 |f_i(x)|$

onde

$$f_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10,$$

$$f_1(x) = x_2x_3 - 5x_4x_5,$$

$$f_1(x) = x_1^3 + x_2^3 + 1,$$

Valor ótimo $f(x^*) = -2.9197004$
 Ponto inicial $x^0 = (-2, 1.5, 2, -1, -1).$

Shell Dual

Dimensão 15

Função objetivo

$$f(x) = 2 \left| \sum_{i=1}^5 x_i \right| + \sum_{i=1}^5 \sum_{j=1}^5 c_{ij} x_{i+10} x_{j+10} -$$

$$- \sum_{i=1}^{10} b_i x_i + 100 \left(\sum_{i=1}^5 \max(0, P_i(x)) - Q(x) \right)$$

$$P_i(x) = \sum_{j=1}^{10} a_{ji} x_j - 2 \sum_{i=1}^5 c_{ij} x_{j+10} - 3d_i x_{i+10}^2 - e_i, \quad 1 \leq i \leq 5$$

$$Q(x) = \sum_{i=1}^{15} \min(0, x_i),$$

Valor ótimo $f(x^*) = 0$
 Ponto inicial $x_0(i) = 10^{-4}, \quad 1 \leq i \leq 15, \quad i \neq 7, \quad x_7 = 60.$

As Matrizes A, B, C, d, e são as mesmas como no Problema Colville 1.

Apêndice B

Problemas Com Restrição

Para uma validação preliminar do código desenvolvido na seção 4 utilizou-se o seguinte conjunto de problemas testes. Estes problemas podem ser encontrados nos trabalhos de Schittkowski [63], Luksan [54] e Kiwiel [34].

Tabela B.1: Problemas Teste

No.	Problema	n	$c(x)$	Ponto Inicial	$f(x^*)$
1	Maxquad	10	lin	(0, 0, 0, 0, 0, 0, 0, 0, 0, 0)	-0.36816644
2	Rosen	4	cvx	(0, 0, 0, 0)	-44
3	Wong2	20	cvx	(2, 3, 5, 5, 1, 2, 7, 3, 6, 10)	24.306209
4	Wong3	20	cvx	(*)	133.72828
5	Prob12	2	dif	(0, 0)	-30
6	Prob29	3	dif	(1, 1, 1)	-22
7	Prob35	3	cvx	(0.5, 0.5, 0.5)	1/9
8	Prob76	4	cvx	(0.5, 0.5, 0.5, 0.5)	-4.6818
9	Prob100	7	cvx	(1, 2, 0, 4, 0, 1, 1)	680.630057
10	Prob113	10	cvx	(2, 3, 5, 5, 1, 2, 7, 3, 6, 10)	24.306209
11	Prob227	2	cvx	(0.5, 0.5)	1
12	Prob228	2	cvx	(0.0, 0.0)	-3
13	Prob264	4	cvx	(0.0, 0.0, 0.0, 0.0)	-44
14	Prob284	15	cvx	0	-1840
15	Prob285	15	cvx	0	-8252
16	MiniMax	4	cvx	(15, 22, 27, 11)	23.886767
17	MiniSum	6	cvx	(0, 0, 0, 0, 0, 0)	68.82856
18	Streit	4	cvx	(0, 0, 0, 0)	0.7071068
19	Colville	6	cvx	(0, 0, 0, 0, 0, 0)	-23.0448869
20	Ill-Conditioned LP	30	lin	0	0

(*) - (2, 3, 5, 5, 1, 2, 7, 3, 6, 10, 2, 2, 6, 15, 1, 2, 1, 2, 1, 3)

B.1 Problemas com restrições

MAXQUAD

Dimensão 10

Função objetivo $f(x) = \max_{1 \leq i \leq 20} \{x A_i x^t - b_i x^t\}$

Restrição

$$c_i(x) = |x_i| - 0.05, \text{ para } i = 1, \dots, 10,$$

$$c_{11}(x) = \sum_{i=1}^{10} x_i - 0.05,$$

Valor ótimo $f(x^*) = -0.36816644$

Ponto inicial $x_0(i) = 0, i = 1, \dots, 10, f(x_0) = 0.$

ROSEN - O problema Rosen-Suzuki encontrado em [63], p.66.

Dimensão 4

Função Objetivo $f(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4.$

Restrição

$$c(x) = \max \left\{ \begin{array}{l} x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8 \\ x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10 \\ 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5 \end{array} \right\}.$$

Valor ótimo $f(x^*) = -44.$

Ponto inicial $x_0 = (0, 0, 0, 0).$

Os problemas Wong2 e wong3 foram retirados do trabalho de Lukšan e Vlček [54], p.24.

Wong2

Dimensão 10

Função Objetivo $f(x) = \max_{1 \leq i \leq 6} f_i(x)$

$$f_1(x) = x_1^2 + x_2^2 + x_1 x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45,$$

$$f_2(x) = f_1(x) + 10(3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120),$$

$$f_3(x) = f_1(x) + 10(5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40),$$

$$f_4(x) = f_1(x) + 10(0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30),$$

$$f_5(x) = f_1(x) + 10(x_1^2 + 2(x_2 - 2)^2 - 2x_1 x_2 + 14x_5 - 6x_6),$$

$$f_6(x) = f_1(x) + 10(-3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10}).$$

Restrição

$$c(x) = \max_{1 \leq i \leq 3} c_i(x),$$

$$c_1(x) = 4x_1 + 5x_2 - 3x_7 + 9x_8 - 105 \leq 0,$$

$$c_2(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0,$$

$$c_3(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0.$$

Ponto inicial $x_0 = (2, 3, 5, 5, 1, 2, 7, 3, 6, 10)$.

Valor ótimo $f(x^*) = 24.306209$.

Wong3

Dimensão 20

Função Objetivo $f(x) = \max_{1 \leq i \leq 14} f_i(x)$,

$$\begin{aligned} f_1(x) = & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + \\ & + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + (x_{11} - 9)^2 + \\ & + 10(x_{12} - 1)^2 + 5(x_{13} - 7)^2 + 4(x_{14} - 14)^2 + 27(x_{15} - 1)^2 + x_{16}^4 + (x_{17} - 2)^2 + \\ & + 13(x_{18} - 2)^2 + (x_{19} - 3)^2 + x_{20}^2 + 95, \end{aligned}$$

$$f_2(x) = f_1(x) + 10(3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120),$$

$$f_3(x) = f_1(x) + 10(5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40),$$

$$f_4(x) = f_1(x) + 10(0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30),$$

$$f_5(x) = f_1(x) + 10(x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6),$$

$$f_6(x) = f_1(x) + 10(-3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10}),$$

$$f_7(x) = f_1(x) + 10(x_1^2 + 15x_{11} - 8x_{12} - 28),$$

$$f_8(x) = f_1(x) + 10(4x_1 + 9x_2 + 5x_{13}^2 - 9x_{14} - 87),$$

$$f_9(x) = f_1(x) + 10(3x_1 + 4x_2 + 3(x_{13} - 6)^2 - 14x_{14} - 10),$$

$$f_{10}(x) = f_1(x) + 10(14x_1^2 + 35x_{15} - 79x_{16} - 92),$$

$$f_{11}(x) = f_1(x) + 10(15x_2^2 + 11x_{15} - 61x_{16} - 54),$$

$$f_{12}(x) = f_1(x) + 10(5x_1^2 + 2x_2 + 9x_{17}^2 - x_{18} - 68),$$

$$f_{13}(x) = f_1(x) + 10(x_1^2 - x_2 + 19x_{19} - 20x_{20} + 19),$$

$$f_{14}(x) = f_1(x) + 10(7x_1^2 + 5x_2^2 + x_{19}^2 - 30x_{20}).$$

Restrição

$$c(x) = \max_{1 \leq i \leq 3} c_i(x),$$

$$c_1(x) = 4x_1 + 5x_2 - 3x_7 + 9x_8 - 105 \leq 0,$$

$$c_2(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0,$$

$$c_3(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0.$$

Ponto inicial $x_0 = (2, 3, 5, 5, 1, 2, 7, 3, 6, 10, 2, 2, 6, 15, 1, 2, 1, 2, 1, 3)$.
 Valor ótimo $f(x^*) = 133.72828$.

Os problemas 12, 29, 35, 76, 100, 113, 227, 228, 264, 284 e 285 foram retirados do livro de Schittkowski [63].

Problema 12

Dimensão 2
 Função Objetivo $f(x) = 0.5x_1^2 + x_2^2 - x_1x_2 - 7x_1 - 7x_2$.
 Restrição

$$c(x) = -25 + 4x_1^2 + x_2^2.$$

Valor ótimo $f(x^*) = -30$
 Ponto inicial $x_0 = (0.0, 0.5)$.

Problema 29

Dimensão 3
 Função Objetivo $f(x) = -x_1x_2x_3$.
 Restrição

$$c(x) = x_1^2 - 2x_2^2 - 4x_3^2 + 48.$$

Valor ótimo $f(x^*) = -22.627417$
 Ponto inicial $x_0 = (1, 1, 1)$.

Problema 35

Dimensão 3
 Função Objetivo $f(x) = 9 - 8x_1 - 6x_2 - 4x_3 + 2x_1^2 + 2x_2^2 + x_3^2 + 2x_1x_2 + 2x_1x_3$.
 Restrição

$$c(x) = \max \left\{ \begin{array}{l} x_1 + x_2 + 2x_3 - 3 \\ -x_1, -x_2, -x_3 \end{array} \right\}.$$

Valor ótimo $f(x^*) = 0.11111111...$
 Ponto inicial $x_0 = (0.5, 0.5, 0.5)$.

Problema 76

Dimensão 4

Função Objetivo $f(x) = x_1^2 + .5x_2^2 + x_3^2 + .5x_4^2 - x_1x_3 + x_3x_4 - x_1 - 3x_2 + x_3 - x_4.$

Restrição

$$c(x) = \max \left\{ \begin{array}{l} x_1 + 2x_2 + x_3 + x_4 - 5 \\ 3x_1 + x_2 + 2x_3 - x_4 - 4 \\ -x_2 - 4x_3 + 1.5 \leq 0 \\ -x_1, -x_2, -x_3, -x_4 \end{array} \right\}.$$

Valor ótimo $f(x^*) = -4.681818.$ Ponto inicial $x_0 = (0.5, 0.5, 0.5, 0.5).$ **Problema 100**

Dimensão 7

Função Objetivo

$$f(x) = (x_1 - 10)^2 + 5(x_1 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7.$$

Restrição

$$c(x) = \max \left\{ \begin{array}{l} 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 - 127 \\ 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 - 282 \\ 23x_1 - x_2^2 + 6x_6^2 - 8x_7 \\ 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \end{array} \right\}.$$

Valor ótimo $f(x^*) = -4.681818.$ Ponto inicial $x_0 = (1.0, 2.0, 0.0, 4.0, 0.0, 1.0, 1.0).$ **Problema 113**

Dimensão 10

Função Objetivo

$$f(x) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45.$$

Restrição

$$c(x) = \max \left\{ \begin{array}{l} 4x_1 + 5x_2 - 3x_7 + 9x_8 - 105 \\ 10x_1 - 8x_2 + 17x_7 + 2x_8 \\ 8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \\ 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \\ 5x_1^2 + 8x_2 + (x_3 - 6)^2 + 2x_4 - 40 \\ 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \\ x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \\ 3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \end{array} \right\}.$$

Valor ótimo $f(x^*) = 24.3062091$.
 Ponto inicial $x_0 = (2, 3, 5, 5, 1, 2, 7, 3, 6, 10)$

Problema 227

Dimensão 2
 Função Objetivo $f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$
 Restrição

$$c(x) = \max \left\{ \begin{array}{l} x_1^2 - x_2 \\ -x_1 + x_2^2 \end{array} \right\}.$$

Valor ótimo $f(x^*) = 1.000000$
 Ponto inicial $x_0 = (0.5, 0.5)$.

Problema 228

Dimensão 2
 Função Objetivo $f(x) = x_1^2 + x_2$
 Restrição

$$c(x) = \max \left\{ \begin{array}{l} x_1 + x_2 - 1 \\ x_1^2 + x_2^2 - 9 \end{array} \right\}.$$

Valor ótimo $f(x^*) = -3.000000$
 Ponto inicial $x_0 = (0.0, 0.0)$.

Problema 264

Dimensão 4
 Função Objetivo $f(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$
 Restrição

$$c(x) = \max \left\{ \begin{array}{l} -8 + x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 \\ -9 + x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 \\ -5 + 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 \end{array} \right\}.$$

Valor ótimo $f(x^*) = -44.000000$
 Ponto inicial $x_0 = (0.0, 0.0, 0.0, 0.0)$.

Problema 284

Dimensão 15
 Função Objetivo $f(x) = -C * x(1 : n - 1)$
 Restrição

$$c(x) = \max \{h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}\}.$$

onde

$$x_2 = x(1 : n - 1)^2,$$

$$\begin{aligned}
h_1 &= A(1, :) * x_2 - B(1), \\
h_2 &= A(2, :) * x_2 - B(2), \\
h_3 &= A(3, :) * x_2 - B(3), \\
h_4 &= A(4, :) * x_2 - B(4), \\
h_5 &= A(5, :) * x_2 - B(5), \\
h_6 &= A(6, :) * x_2 - B(6), \\
h_7 &= A(7, :) * x_2 - B(7), \\
h_8 &= A(8, :) * x_2 - B(8), \\
h_9 &= A(9, :) * x_2 - B(9), \\
h_{10} &= A(10, :) * x_2 - B(10).
\end{aligned}$$

$$C = [20 \ 40 \ 400 \ 20 \ 80 \ 20 \ 40 \ 140 \ 380 \ 280 \ 80 \ 40 \ 140 \ 40 \ 120],$$

e

$$B = [385 \ 470 \ 560 \ 565 \ 6457 \ 430 \ 485 \ 455 \ 390 \ 460],$$

e

$$A = \begin{bmatrix} 100 & 100 & 10 & 5 & 10 & 0 & 0 & 25 & 0 & 10 & 55 & 5 & 45 & 20 & 0 \\ 90 & 100 & 10 & 35 & 20 & 5 & 0 & 35 & 55 & 25 & 20 & 0 & 40 & 25 & 10 \\ 70 & 50 & 0 & 55 & 25 & 100 & 40 & 50 & 0 & 30 & 60 & 10 & 30 & 0 & 40 \\ 50 & 0 & 0 & 65 & 35 & 100 & 35 & 60 & 0 & 15 & 0 & 75 & 35 & 30 & 65 \\ 50 & 10 & 70 & 60 & 45 & 45 & 0 & 35 & 65 & 5 & 75 & 100 & 75 & 10 & 0 \\ 40 & 0 & 50 & 95 & 50 & 35 & 10 & 60 & 0 & 45 & 15 & 20 & 0 & 5 & 5 \\ 30 & 60 & 30 & 90 & 0 & 30 & 5 & 25 & 0 & 70 & 20 & 25 & 70 & 15 & 15 \\ 20 & 30 & 40 & 25 & 40 & 25 & 15 & 10 & 80 & 20 & 30 & 30 & 5 & 65 & 20 \\ 10 & 70 & 10 & 35 & 25 & 65 & 0 & 30 & 0 & 0 & 25 & 0 & 15 & 50 & 55 \\ 5 & 10 & 100 & 5 & 20 & 5 & 10 & 35 & 95 & 70 & 20 & 10 & 35 & 10 & 30 \end{bmatrix}.$$

Valor ótimo $f(x^*) = -1840$

Ponto inicial $x_0(i) = 0.0$ para $1 \leq i \leq 15$.

Problema 285

Dimensão 15

Função Objetivo $f(x) = -C * x(1 : n - 1)$

Restrição

$$c(x) = \max \{h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}\}.$$

onde h_i , para $1 \leq i \leq 10$, B e A são as mesmas do Problema 284 e C como segue

$$C = [486 \ 640 \ 758 \ 776 \ 477 \ 707 \ 175 \ 619 \ 627 \ 614 \ 475 \ 377 \ 524 \ 468 \ 529].$$

Valor ótimo $f(x^*) = -8252$

Ponto inicial $x_0(i) = 0.0$ para $1 \leq i \leq 15$.

Problema Minimax Location

Dimensão 4

Função Objetivo

$$f(x) = \max_{1 \leq i \leq 9} \{w_{i1} \|a^i - (x_1, x_2)\|_{p_{i1}}; w_{i2} \|a^i - (x_3, x_4)\|_{p_{i2}}; \|(x_1, x_2) - (x_3, x_4)\|\}$$

Restrições

$$c_1(x) = \|(x_1, x_2) - a^1\|^2 - 144,$$

$$c_2(x) = \|(x_1, x_2) - a^4\|^2 - 121,$$

$$c_3(x) = \|(x_3, x_4) - a^1\|^2 - 225,$$

$$c_4(x) = \|(x_3, x_4) - a^2\|^2 - 144$$

onde

Tabela B.2: parâmetros

i	a_i	w_{i1}	p_{i1}	w_{i2}	p_{i2}
1	(11.4, 11.6)	2.0	2.0	1.0	2.0
2	(35.3, 13.5)	1.0	2.0	2.0	2.0
3	(8.8, 37.2)	1.3	1.1	0.85	1.4
4	(20.9, 30.6)	1.1	1.5	1.0	1.9
5	(25.5, 28)	1.5	1.4	1.5	1.2
6	(29.7, 27.7)	1.0	2.0	1.5	2.0
7	(36.2, 27.8)	0.5	1.8	1.0	1.7
8	(45.5, 21.3)	0.5	2.0	0.5	2.0
9	(15.8, 28.2)	0.5	1.1	0.5	1.8

Valor ótimo $f(x^*) = 23.886767$,

Ponto inicial $x_0 = (15, 22, 27, 11)$.

Problema Minisum Location

Dimensão 6

Função Objetivo

$$f(x) = \sum_{i=1}^3 \sum_{j=1}^5 w_{ij} \|(x_{2i-1}, x_{2i}) - a^j\|_p + \sum_{1 \leq i < j \leq 3} \|(x_{2i-1}, x_{2i}) - (x_{2j-1}, x_{2j})\|_p;$$

Restrição

$$c(x) = x_5 + x_6 - 3,$$

onde $p = 1.78$, $a_1 = (2, 3)$, $a_2 = (4, 2)$, $a_3 = (5, 4)$, $a_4 = (3, 5)$, $a_5 = (6, 7)$,
 $w_{ij} = 1$, exceto para $w_{13} = w_{15} = 6$ e $w_{21} = 2$.

Valor ótimo $f(x^*) = 68.82856$,

Ponto inicial $x_0 = (0, 0, 0, 0, 0, 0)$.

Colville

Dimensão 5

Função objetivo $f(x) = \sum_{i=1}^5 \sum_{j=1}^5 c_{ij} x_i x_j + \sum_{i=1}^5 d_i x_i + \sum_{i=1}^5 e_i x_i$

Restrição

$$c_i(x) = b_i - \sum_{j=1}^5 a_{ij} x_j, \text{ onde } i = 1, \dots, 10,$$

$$c_{10+i}(x) = -x_i, \text{ onde } i = 1, \dots, 10,$$

onde a_{ij} , b_i , c_{ij} , d_i e e_i são os mesmos do Problema Colville dado no Apêndice A.

Valor ótimo $f(x^*) = -23.0448869$,

Ponto inicial $x_0 = (0, 0, 0, 0, 1)$.

Ill conditioned LP

Dimensão 30

Função Objetivo $f(x) = \sum_{i=1}^{30} c_i(x_i - 1)$

Restrição $c(x) = (a^i)^T x - b_i$ onde $i = 1, \dots, 30$,

onde $a_j^i = \frac{1}{i+j}$, $b_i = \sum_{j=1}^{30} a_j^i$, $c_i = -b_i - \frac{1}{1+i}$.

Valor ótimo $f(x^*) = 0$,

Ponto inicial $x_0 = 0$.