



UNICAMP

UNIVERSIDADE ESTADUAL DE
CAMPINAS

Instituto de Matemática, Estatística e
Computação Científica

JEAN CARLOS APARECIDO MEDEIROS

**Métodos de feixes baseados em conjuntos de
nível para otimização não suave aplicados a
problemas de Steiner**

Campinas

2021

Jean Carlos Aparecido Medeiros

Métodos de feixes baseados em conjuntos de nível para otimização não suave aplicados a problemas de Steiner

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestre em Matemática Aplicada.

Orientadora: Sandra Augusta Santos

Este trabalho corresponde à versão final da Dissertação defendida pelo aluno Jean Carlos Aparecido Medeiros e orientada pela Profa. Dra. Sandra Augusta Santos.

Campinas

2021

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

M467m Medeiros, Jean Carlos Aparecido, 1995-
Métodos de feixes baseados em conjuntos de nível para otimização não suave aplicados a problemas de Steiner / Jean Carlos Aparecido Medeiros. – Campinas, SP : [s.n.], 2021.

Orientador: Sandra Augusta Santos.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Otimização não diferenciável. 2. Otimização irrestrita. 3. Método de feixes. 4. Diferenciação automática. 5. Problemas de Steiner. 6. Experimentos numéricos. I. Santos, Sandra Augusta, 1964-. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Bundle methods based on level sets for non-smooth optimization applied to Steiner problems

Palavras-chave em inglês:

Nondifferentiable optimization

Unrestricted optimization

Bundle methods

Automatic differentiation

Steiner problems

Numerical experiments

Área de concentração: Matemática Aplicada

Titulação: Mestre em Matemática Aplicada

Banca examinadora:

Sandra Augusta Santos [Orientador]

Paulo José da Silva e Silva

Wellington Luis de Oliveira

Data de defesa: 29-03-2021

Programa de Pós-Graduação: Matemática Aplicada

Identificação e informações acadêmicas do(a) aluno(a)

- ORCID do autor: <https://orcid.org/0000-0003-1414-2424>

- Currículo Lattes do autor: <http://lattes.cnpq.br/8749446095144706>

**Dissertação de Mestrado defendida em 29 de março de 2021 e aprovada
pela banca examinadora composta pelos Profs. Drs.**

Prof(a). Dr(a). SANDRA AUGUSTA SANTOS

Prof(a). Dr(a). PAULO JOSÉ DA SILVA E SILVA

Prof(a). Dr(a). WELINGTON LUIS DE OLIVEIRA

A Ata da Defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria de Pós-Graduação do Instituto de Matemática, Estatística e Computação Científica.

Dedico o presente trabalho às minhas avós Rita (in memoriam) e Araci (in memoriam) e à minha mãe, que não mediram esforços para me ajudar em toda a minha jornada.

Agradecimentos

À Sandra, minha orientadora, pela oportunidade de realizar esse trabalho tão enriquecedor e desafiador para a minha trajetória acadêmica, pelas inúmeras horas carinhosamente despendidas a este projeto, pela imensa paciência e excelente orientação. Muitíssimo obrigado!

Ao Welington, ao Lúcio e ao Paulo, por aceitarem compor a banca da defesa da minha dissertação de Mestrado e pela enorme contribuição com valiosas sugestões, orais e escritas, para a versão final desta dissertação.

À minha mãe, Ana Paula Costa Medeiros, pelo apoio incondicional não apenas neste período, mas em todos os momentos de minha existência, pelo constante incentivo e torcida e à minha irmã, Jackeline Costa Medeiros pelo zelo, carinho, atenção para comigo e amizade que tanto me alegra. Este tem sido um sonho concretizado a seis mãos.

Aos meus familiares e amigos que torcem pelo meu êxito, por compreenderem minha ausência nos períodos de estudo. Em especial aos meus primos Tayná e Guilherme e aos meus amigos Raíssa, Mariana, Heliton, Isabela, Deborah, Nati, Larissa, Mayara, Alfredo, Laranja, Casalate, Thiago e Luana.

Aos docentes de quem fui aluno, pelo incentivo e confiança em meu potencial. Em especial aos docentes, Claudina, Pulino, Sandra, Cheti, Márcia, Kelly, Thelma, Valéria, Fábio, Conceição, Silvana e Maria Inês.

Aos meus colegas da graduação e pós-graduação, em especial aos integrantes da sala 01-B e seus agregados, pelos grupos de estudo que contribuíram para com a minha formação acadêmica, por todo apoio e amizade.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo

O método de feixes em níveis tem se mostrado um algoritmo robusto para problemas de otimização não suave. Uma das vantagens da variante em níveis está no parâmetro de estabilização, que possui uma forma bem definida de atualização. Inspirados em um trabalho recente, nosso objetivo foi estudar tal método aplicado à resolução de instâncias de problemas de Steiner. Consideramos a resolução desses problemas do ponto de vista da otimização não suave, de modo a explorar suas características geométricas.

A partir de uma técnica de diferenciação automática baseada em números duais, implementada em Julia, utilizamos um oráculo adequado ao cálculo dos subgradientes do problema de interesse. Propusemos quatro formas distintas de implementação do método de feixes em níveis, explorando a formulação primal, ou a dual equivalente, bem como o uso, ou não, da técnica de compressão do feixe. Essas quatro versões foram o principal objeto de estudo dos nossos experimentos numéricos. Com base na análise dos resultados numéricos, tivemos fortes indícios de que tanto o oráculo é uma ferramenta bastante robusta para o problema estudado, quanto as quatro versões implementadas do método de feixes em níveis são capazes de obter resultados satisfatórios, com uma pequena vantagem para as versões baseadas no problema primal, utilizando, ou não, a técnica de compressão.

Palavras-chave: Otimização não diferenciável. Otimização irrestrita. Método de feixes. Diferenciação automática. Problema de Steiner. Experimentos numéricos.

Abstract

Level bundle methods have proven to be robust algorithms for non-smooth optimization problems. One of the advantages of the level variant is the stabilization parameter, which has a well-defined form of updating. Inspired by a recent work, our goal was to study such a method applied to address instances of the Steiner problem. We have considered solving these problems from the non-smooth optimization perspective, in order to take advantage of their geometric features.

Using an automatic differentiation technique based on dual numbers, implemented in Julia, we rested upon an oracle that turned out to be adequate for computing subgradients of the problem of interest. We proposed four different ways of implementing the level bundle method, exploring the primal, or the dual equivalent formulation, as well as the use, or not, of the bundle compression technique. These four versions were the main object of study of our numerical experiments. Based on the analysis of the numerical results, we have noticed that, for the family of problems under consideration, not only the oracle is a very robust tool, but also the four implemented versions of the level bundle method are capable of obtaining satisfactory results, with a small advantage for the versions based on the primal problem, using or not the compression technique.

Keywords: Nondifferentiable optimization. Unrestricted optimization. Bundle methods. Automatic differentiation. Steiner problem. Numerical experiments.

Lista de ilustrações

Figura 1 – Em 1(a) e 1(b) podemos observar que os pontos verdes nos dois gráficos são não suaves.	15
Figura 2 – Fluxograma com a organização dos temas abordados no trabalho.	16
Figura 3 – Em 3(a) temos exemplos de conjuntos convexos enquanto que em 3(b) temos exemplos de conjuntos não-convexos.	17
Figura 4 – Exemplo de uma função $f : \Omega \mapsto \mathbb{R}$ convexa.	18
Figura 5 – Exemplo de uma função convexa f , não-diferenciável em x_0 ; e de duas aproximações lineares passando por x_0 e ambas abaixo do gráfico da f	20
Figura 6 – Exemplo de uma função de uma única variável, convexa e suave. Observe que para todo ponto x da combinação convexa entre x_0 e x_1 temos que $\bar{l}(x) \geq l(x)$	20
Figura 7 – Ilustração com três aproximações lineares em $x = 0$ para a função $f(x) = x $	21
Figura 8 – Ilustração da interpretação geométrica do problema dual.	25
Figura 9 – Janela do Julia com a definição e o cálculo de um subgradiente da função f em A	29
Figura 10 – Janela do Julia com a definição e indeterminação no cálculo do subgradiente da g em A	30
Figura 11 – Diagrama com os métodos apresentados neste capítulo.	31
Figura 12 – Algumas iterações do método de Planos Cortantes	34
Figura 13 – Note que x^2 seria um bom candidato para minimizador, porém ao adicionar o plano cortante associado a ele, o minimizador do modelo acaba se afastando do minimizador da f , ou seja, o valor funcional f_3 seria maior que f_2	36
Figura 14 – Visualização dos principais elementos envolvidos no Método de Feixes em Níveis. A função objetivo original f está em destaque (azul), assim como o segmento que define o conjunto viável do subproblema na instância ilustrada.	60
Figura 15 – Visualização dos principais elementos envolvidos no Método de Feixes em Níveis, com ênfase para o modelo agregado $\varphi_a(y) = f(x^k) - \varepsilon_k + \langle s_N^k, y - x^k \rangle$, e elementos do subproblema corrente (função objetivo e conjunto viável) para a instância ilustrada.	66
Figura 16 – Resolução proposta por Torricelli e, em vermelho, o ponto de Torricelli.	71
Figura 17 – Exemplos de triângulos e respectivos pontos de Fermat: solução no interior do triângulo em 17(a), e solução no vértice do triângulo obtusângulo cujo ângulo interno é igual ou superior a 120° , em 17(b).	71

Figura 18 – Um exemplo de problema de Steiner com $N = 4$	72
Figura 19 – Exemplos de quadriláteros com os respectivos pontos de Steiner: em 19(a) os dois pontos estão no interior, e em 19(b), um dos pontos coincide com um vértice do quadrilátero.	73
Figura 20 – Ilustração com o grafo representando o problema e o grafo representando a solução do problema resolvido por Donno e Pesamosca.	74
Figura 21 – Ilustração com o grafo representando o problema e o grafo representando a solução do problema resolvido por Facchinei e Lucidi.	75
Figura 22 – Domínio dos vértices A , B e C , separados por cor.	77
Figura 23 – Domínio dos vértices A , B , C e D , separados por cor.	78
Figura 24 – Comparando a acurácia das versões utilizando o ponto gerado aleatoriamente e o baricentro como ponto inicial.	80
Figura 25 – Comparando o custo computacional dos métodos alterando o ponto inicial.	81
Figura 26 – Comparando a acurácia das versões sem compressão.	82
Figura 27 – Comparando o custo computacional das versões sem compressão para cada valor do parâmetro α	83
Figura 28 – Distribuição dos passos de cada versão sem rotina de compressão, para cada valor do parâmetro α	84
Figura 29 – Comparando a acurácia das versões com compressão para cada valor do parâmetro α	84
Figura 30 – Comparando o custo computacional das versões com compressão para cada valor do parâmetro α	85
Figura 31 – Distribuição dos passos de cada versão com rotina de compressão, para cada valor do parâmetro α	86
Figura 32 – Comparando o custo computacional das versões aplicadas no conjunto de quadriláteros.	87
Figura 33 – Acurácia e Custo Computacional das versões na resolução do Problema 5	88
Figura 34 – Acompanhando a evolução do <i>gap</i> , do nível e atualização de centro de estabilidade na versão CP para o Problema 5.	89
Figura 35 – Acompanhando a evolução do <i>gap</i> , do nível e da atualização de centro de estabilidade na versão SP para o Problema 5.	89
Figura 36 – Acurácia e Custo Computacional das versões na resolução do Problema 6	90
Figura 37 – Acompanhando a evolução do <i>gap</i> , do nível e da atualização de centro de estabilidade na versão CP para o Problema 6.	91
Figura 38 – Acompanhando a evolução do <i>gap</i> , do nível e atualização de centro de estabilidade na versão SP para o Problema 6.	91

Lista de tabelas

Tabela 1 – Tempo e Alocação de memória das versões com ponto inicial sendo o baricentro.	80
Tabela 2 – Tempo e Alocação de memória das versões com ponto inicial aleatório.	81
Tabela 3 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.2$	82
Tabela 4 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.5$	83
Tabela 5 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.8$	83
Tabela 6 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.2$	85
Tabela 7 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.5$	85
Tabela 8 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.8$	85
Tabela 9 – Tempo e Alocação de memória das versões aplicadas no conjunto de quadriláteros.	87
Tabela 10 – Tempo e Alocação de memória das versões aplicadas no Problema 5.	88
Tabela 11 – Tempo e Alocação de memória das versões aplicado no Problema 6.	90

Lista de Algoritmos

Algoritmo 1 – MÉTODOS DE SUBGRADIENTES	33
Algoritmo 2 – MÉTODO DE PLANOS CORTANTES	35
Algoritmo 3 – MÉTODO DE FEIXES	40
Algoritmo 4 – MÉTODO DE FEIXES EM NÍVEIS	67
Algoritmo 5 – ROTINA DE COMPRESSÃO	68

Sumário

	Introdução	15
1	CONCEITOS FUNDAMENTAIS	17
1.1	Elementos de convexidade	17
1.2	A ideia de subdiferenciabilidade	19
1.3	Elementos da dualidade	22
1.3.1	Caso Não-Linear	22
1.3.2	Caso Linear	25
1.3.3	O Caso Quadrático	26
1.4	Números duais e a diferenciação automática	27
2	MÉTODOS COMPUTACIONAIS PARA OTIMIZAÇÃO NÃO SUAVE	31
2.1	Métodos de Subgradiente	32
2.2	Método de Planos Cortantes	34
2.3	Método de Feixes	37
2.3.1	Estabilização	37
2.3.2	Algumas variantes do Método de Feixes	42
2.3.3	Compressão do Feixe	48
2.3.4	Análise de Convergência da versão Proximal	51
2.3.5	Método de Feixes em Níveis	59
2.4	Métodos de Segunda Ordem	69
3	PROBLEMA: ÁRVORE EUCLIDIANA DE STEINER	70
3.1	O problema de Steiner	70
3.2	O problema de Steiner Generalizado	73
4	EXPERIMENTOS NUMÉRICOS	77
4.1	Metodologia	77
4.1.1	Geração dos problemas teste	77
4.1.2	Implementação	78
4.1.3	Indicadores analisados	79
4.2	Experimento 1: Influência do Ponto Inicial	80
4.2.1	Conclusão do Experimento 1	81
4.3	Experimento 2: Influência do parâmetro α	82
4.3.1	Versões sem rotina de compressão: SD e SP	82
4.3.2	Versões com rotina de compressão: CD e CP	84

4.3.3	Conclusão do Experimento 2	86
4.4	Experimento 3: Quadriláteros	86
4.4.1	Conclusão do Experimento 3	87
4.5	Experimento 4: Problemas de dimensão maiores	88
4.5.1	Problema 5	88
4.5.2	Problema 6	90
4.5.3	Conclusão do Experimento 4	91
5	CONSIDERAÇÕES FINAIS	92
	Referências	93
	APÊNDICE A – GLOSSÁRIO	96

Introdução

Este trabalho aborda a minimização irrestrita de funções convexas, não necessariamente suaves. Para começarmos o estudo da otimização não suave, ilustramos na Figura 1 o gráfico de duas funções que são contínuas em todo o domínio, porém não são diferenciáveis em todo domínio.

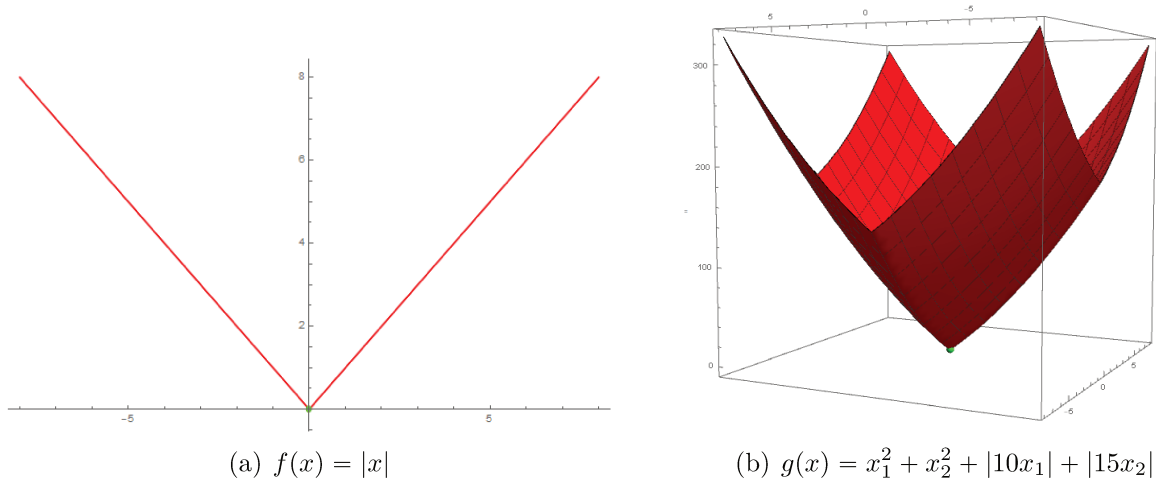


Figura 1 – Em 1(a) e 1(b) podemos observar que os pontos verdes nos dois gráficos são não suaves.

Relembrando de cálculo diferencial, sabemos que nesses pontos sem suavidade a função não possui plano tangente. Dessa forma, mesmo que a não suavidade ocorra em um único ponto, se tal ponto for o minimizador, como acontece nas situações ilustradas na Figura 1, esse minimizador não estará associado ao vetor gradiente nulo. Isso justifica a necessidade de um tratamento próprio para problemas não suaves, com métodos específicos, conforme veremos nesse trabalho.

Nossa escolha pela variante do método de feixes baseada em níveis foi inspirada no recente trabalho de [Iutzeler et al.](#) em [17], em que os autores revisitam a proposta clássica de [Lemaréchal et al.](#) em [20], sob um ponto de vista mais prático e computacional. Almejamos compreender e implementar um algoritmo de feixes baseados em conjuntos de nível, em Julia¹. Particularmente para a análise de convergência, também estudamos a variante Proximal, seguindo a apresentação de [Bonnans et al.](#)[5].

Para a implementação computacional, inicialmente, e de maneira a acompanhar todo o funcionamento do algoritmo, trabalhamos com problemas de baixa dimensão, nos quais a interpretação geométrica dos principais elementos e das propriedades de

¹ Julia é uma linguagem razoavelmente nova e que está chamando atenção de muitos usuários, uma boa referência para começar a programar em Julia é o livro de [Balbaert](#) [1].

convergência pudesse ser explorada em toda a sua extensão. Nossos experimentos numéricos contemplam instâncias do problema de Fermat-Torricelli, e instâncias do problema de Steiner, com topologia pré-estabelecida.

Para complementar o trabalho, também resolvemos problemas com maior número de variáveis, de maneira a detectarmos as forças e fraquezas do método. Nesse sentido, trabalhamos com duas instâncias do problema de Steiner generalizado, também resolvidas por [Facchinei e Lucidi](#) em [12].

Na Figura 2 apresentamos a organização dos capítulos, para auxiliar o leitor a acompanhar a construção das ideias utilizadas nesse trabalho.

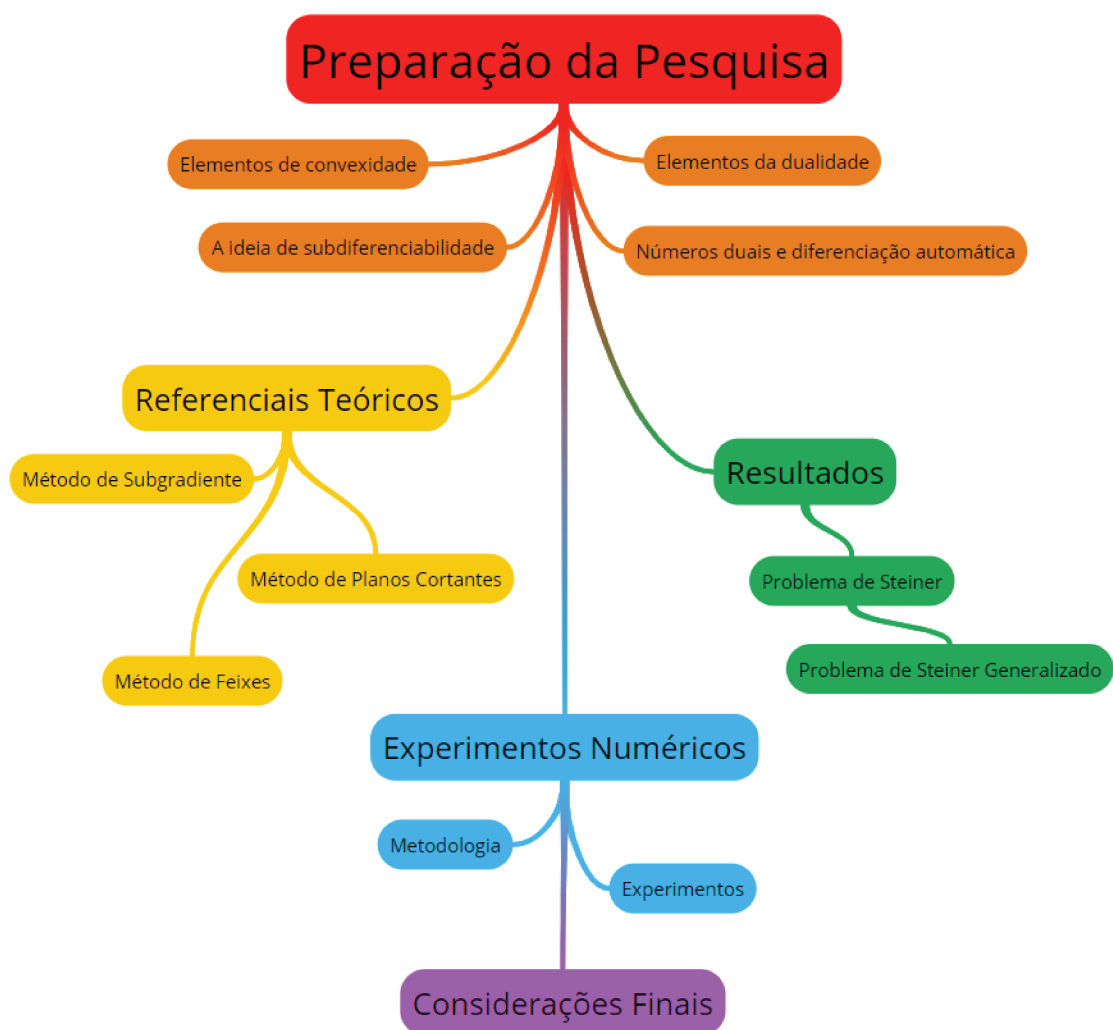


Figura 2 – Fluxograma com a organização dos temas abordados no trabalho.

1 Conceitos fundamentais

Nesta primeira seção vamos apresentar alguns conceitos primordiais, tanto para a compreensão dos problemas não diferenciáveis e suas possíveis aplicações bem como os métodos de otimização que serão apresentados nas próximas seções para resolver essa classe de problemas.

1.1 Elementos de convexidade

Antes de mais nada, vamos apresentar as definições de um conjunto convexo e de uma função convexa, pois os problemas nos quais estamos interessados em resolver têm como função objetivo uma função convexa e o seu respectivo conjunto viável é um conjunto convexo.

Definição 1. Um subconjunto $\Omega \subset \mathbb{R}^n$ é dito convexo se

$$\alpha x + (1 - \alpha)y \in \Omega, \quad \forall x, y \in \Omega, \forall \alpha \in [0, 1]. \quad (1.1)$$

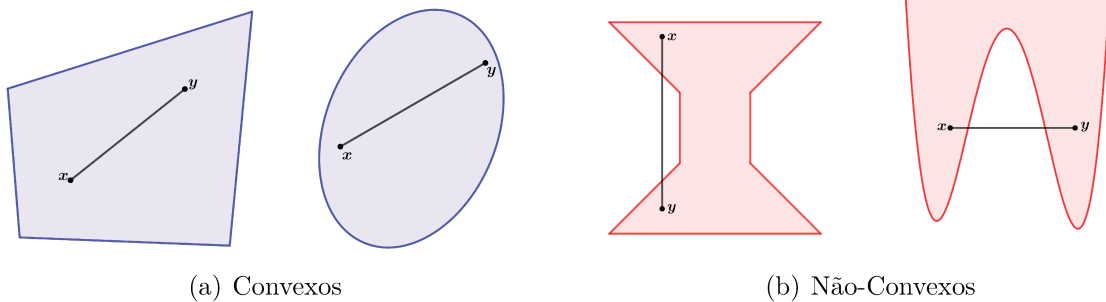


Figura 3 – Em 3(a) temos exemplos de conjuntos convexos enquanto que em 3(b) temos exemplos de conjuntos não-convexos.

Uma propriedade muito interessante sobre conjuntos convexos e que iremos utilizar no futuro é a seguinte:

Proposição 1. O conjunto formado pela intersecção $\bigcap_{i \in I} \Omega_i$ de qualquer coleção de conjuntos convexos $\{\Omega_i \mid i \in I\}$ é convexo.

Demonstração. A demonstração sai diretamente da Definição 1. Tome $x, y \in \bigcap_{i \in I} \Omega_i$, ou seja, x e y pertencentes a cada um dos conjuntos Ω_i . Como cada Ω_i é um conjunto convexo, então o segmento que conecta x e y também está contido em cada um dos conjuntos Ω_i , logo ele pertence à intersecção de todos eles. \square

Outras propriedades interessantes podem ser consultadas em [4, Seção 1.1]. Vamos voltar nossa atenção para dois tipos de conjuntos convexos, frequentemente encontrados nos problemas de otimização: hiperplanos e semi-espacos. Quando começarmos a estudar os algoritmos na seção 2 ficará muito clara a importância de compreendermos bem esses dois tipos de conjunto. Pensando primeiramente em \mathbb{R}^2 , qualquer reta é um hiperplano e geometricamente é fácil ver que dados quaisquer dois pontos pertencentes à reta, o segmento de reta formado por eles também está na reta. Ainda recorrendo à Geometria Euclidiana, o Postulado de Separação nos garante que uma reta divide o plano em dois semi-espacos disjuntos e convexos (ver [25, Postulado 7, Capítulo 1]) e assim como no \mathbb{R}^2 essa ideia pode ser estendida para o \mathbb{R}^n , ou seja, todo hiperplano divide o espaco em dois semi-espacos disjuntos e convexos.

Definidos esses conjuntos convexos e munidos da Proposição 1, temos que a intersecção de semi-espacos convexos é também um conjunto convexo.

Definição 2. *Seja Ω um subconjunto convexo de \mathbb{R}^n . Dizemos que uma função $f : \Omega \mapsto \mathbb{R}$ é convexa se*

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y), \quad \forall x, y \in \Omega, \forall \alpha \in [0, 1]. \quad (1.2)$$

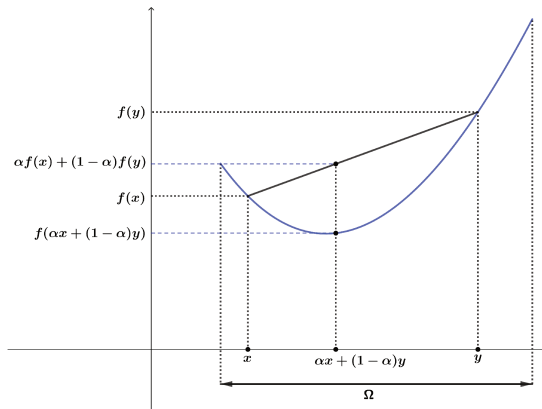


Figura 4 – Exemplo de uma função $f : \Omega \mapsto \mathbb{R}$ convexa.

Na Figura 4, para $\alpha \in [0, 1]$, a combinação linear $\alpha x + (1 - \alpha)y$ é chamada de *combinação convexa*, mas vale ressaltar que em \mathbb{R}^n podemos fazer combinações convexas envolvendo mais de dois pontos. Do ponto de vista da otimização, a convexidade é uma característica muito valiosa, pois se a função objetivo é convexa podemos concluir facilmente que os seus pontos críticos são minimizadores locais da função e mais ainda, os

minimizadores locais são globais. Considerando a classe das funções convexas, podemos dividi-la em duas subclasses distintas: as funções suaves e as funções não suaves. Existem técnicas de otimização suficientemente simples, que aplicadas aos problemas de minimização com funções da primeira subclasse conseguem obter resultados com ótima [acurácia](#) e muito rapidamente. Esse é o caso, por exemplo, do método de Newton, aplicado a problemas com derivadas segundas contínuas. No entanto, o mesmo já não pode ser dito para a segunda subclasse. Isso porque, além do critério de parada dependente da norma do gradiente avaliado no [ponto corrente](#) não poder ser utilizado em pontos de não diferenciabilidade, temos que utilizar [condições de otimalidade](#) específicas para essa subclasse. Pensando nisso, iremos apresentar um conceito muito importante na Secção 1.2, que nos permitirá lidar com funções convexas, porém não suaves.

1.2 A ideia de subdiferenciabilidade

Na otimização suave, uma classe de métodos bastante utilizada é a dos métodos de busca linear, que basicamente consistem em determinar uma direção de descida d a partir de um determinado ponto x^k e calcular um tamanho de passo t que estabelecerá o quanto se andará ao longo dessa direção, e então produzir o próximo [iterado](#) $x^{k+1} = x^k + td$. Para verificar se d é de fato uma direção de descida em x^k testamos se $\nabla^T f(x^k)d < 0$, ou seja, necessitamos da informação do [gradiente](#) da função no ponto para tal verificação. Será que no caso não suave conseguimos fazer algo semelhante? Nesta seção vamos apresentar algumas ferramentas utilizadas em substituição ao gradiente, e que nos darão alternativas para lidar com a otimização não suave.

A primeira ferramenta necessária que vamos apresentar é o *subgradiente*:

Definição 3. *Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa. Um subgradiente de f no ponto $x \in \mathbb{R}^n$ é um vetor $s \in \mathbb{R}^n$, tal que*

$$f(y) \geq f(x) + s^T(y - x), \quad \forall y \in \mathbb{R}^n. \quad (1.3)$$

A partir da Definição 3, podemos pensar que queremos construir uma aproximação linear l para a função f no ponto x , em que $l(y) := f(x) + s^T(y - x)$, de tal forma que o valor $l(y)$ esteja sempre abaixo do valor da $f(y)$ para todo $y \in \mathbb{R}^n$. Claro que se f for uma função suave, podemos substituir s por $\nabla f(x)$ e vamos obter a aproximação de Taylor de primeira de ordem de f , que já nos fornece de forma única a função l . Agora, se f tiver algum ponto de não-diferenciabilidade não seremos capazes de gerar uma l a partir da aproximação de Taylor de primeira ordem nesse ponto, já que tal aproximação não está definida nesse ponto, e é justamente para contornar essa dificuldade que vamos utilizar *um* subgradiente de f em x . Note aqui que enfatizamos o artigo indefinido *um* e

isso é porque vale ressaltar que há liberdade para se escolher o vetor s nos pontos de não diferenciabilidade de f e isso fica bem claro na Figura 5.

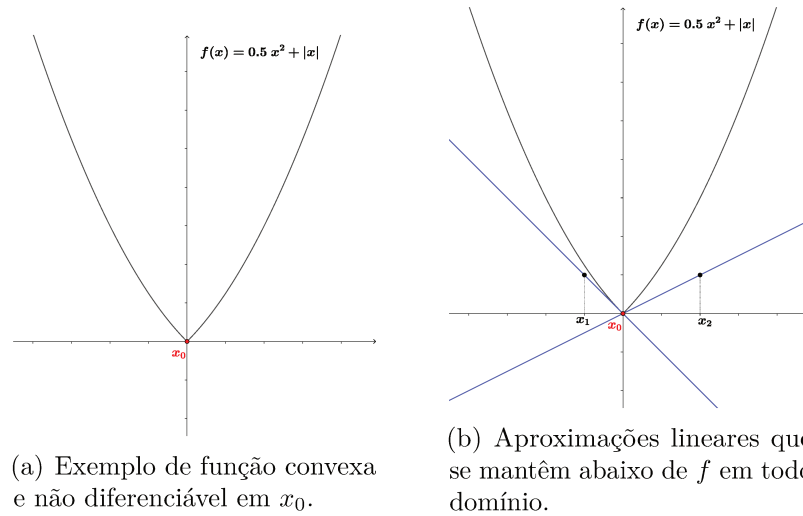


Figura 5 – Exemplo de uma função convexa f , não-diferenciável em x_0 ; e de duas aproximações lineares passando por x_0 e ambas abaixo do gráfico da f .

Note que se a função for não-diferenciável em x_0 conseguimos encontrar mais de uma reta que passa por $(x_0, f(x_0))$ e um outro ponto sem que essas retas sejam *secantes* à curva gráfico de f , ou seja, sem que seus valores sejam superiores ao da função em nenhum ponto, como podemos ver na Figura 5.

Geometricamente falando, se f é uma função de uma única variável, convexa e diferenciável em \bar{x} , a reta tangente l à curva gráfico de f passando pelo ponto $(\bar{x}, f(\bar{x}))$ é a única reta tal que $f(\bar{x}) = l(\bar{x})$ e $f(x) \geq l(x)$ para todo $x \in \mathbb{R}$. Qualquer outra reta que passe também por $(\bar{x}, f(\bar{x}))$ vai obrigatoriamente ser uma reta secante à curva de f , logo vai existir um intervalo em que o valor de l será superior ao da função f . Na Figura 6 ilustramos essas ideias graficamente.

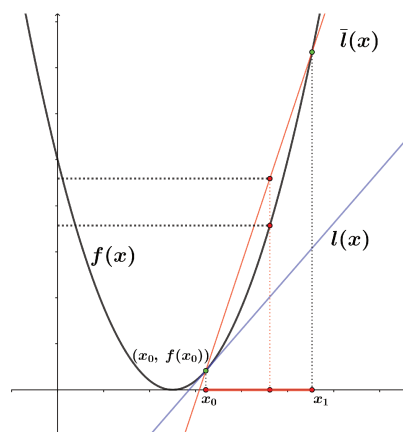


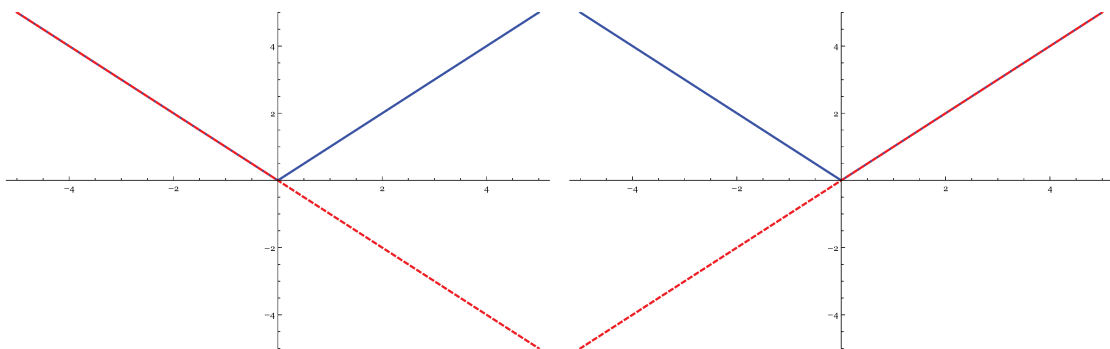
Figura 6 – Exemplo de uma função de uma única variável, convexa e suave. Observe que para todo ponto x da combinação convexa entre x_0 e x_1 temos que $\bar{l}(x) \geq l(x)$.

Com isso em mente, vamos definir os conjuntos *subdiferencial* e ε -*subdiferencial*:

Definição 4. O conjunto de todos os subgradientes de $f : \mathbb{R}^n \rightarrow \mathbb{R}$ em x é chamado de subdiferencial de f em x e é denotado por $\partial f(x)$:

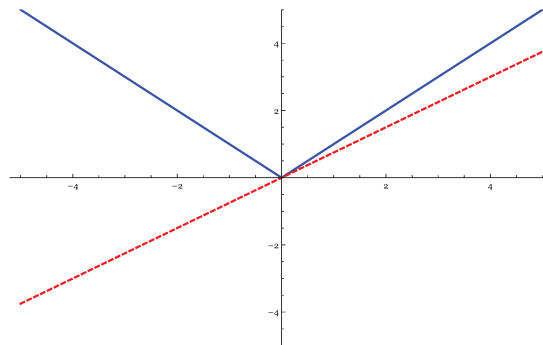
$$\partial f(x) := \{s \in \mathbb{R}^n : f(y) \geq f(x) + s^T(y - x), \forall y \in \mathbb{R}^n\}.$$

Por exemplo, a função valor absoluto, $f(x) = |x|$, possui como subdiferencial em $x = 0$ o intervalo fechado $[-1, 1]$, uma vez que qualquer aproximação linear em 0 que tenha inclinação entre -1 e 1 subaproxima a função $f(x)$. Na Figura 7 ilustramos esse exemplo:



(a) Aproximação linear com inclinação -1 .

(b) Aproximação linear com inclinação 1 .



(c) Aproximação linear com inclinação 0.75 .

Figura 7 – Ilustração com três aproximações lineares em $x = 0$ para a função $f(x) = |x|$.

Definição 5. Podemos adicionar uma tolerância $\varepsilon > 0$ na Definição 4 (também chamado de parâmetro de viscosidade [5, Sec. 9.2.2]), definindo o ε -subdiferencial da seguinte maneira:

$$\partial_\varepsilon f(x) := \{s \in \mathbb{R}^n : f(y) \geq f(x) + s^T(y - x) - \varepsilon, \forall y \in \mathbb{R}^n\}.$$

Veremos no próximo capítulo que no contexto de otimização não suave estamos interessados em obter \bar{x} tal que $0 \in \partial f(\bar{x})$. O conjunto ε -subdiferencial vai nos permitir um

fôlego adicional para os métodos, pois iremos nos contentar em obter \bar{x} tal que $0 \in \partial_\varepsilon f(\bar{x})$. Ou seja, aumentamos o nosso conjunto de busca, uma vez que $\partial f(x) \subset \partial_\varepsilon f(x)$ para todo $\varepsilon > 0$. Do ponto de vista geométrico, é evidente que se a função f for diferenciável em x então $\partial f(x) = \{\nabla f(x)\}$. A prova algébrica desse fato pode ser vista em [18, Prop. 3.4.53].

1.3 Elementos da dualidade

A teoria da dualidade estuda as relações e características entre um problema de otimização original, que recebe o nome de *Problema Primal*, e um problema de otimização associado a ele, que recebe o nome de *Problema Dual*. Nesta seção iremos estudar algumas relações entre o par primal dual e veremos que em alguns casos é mais fácil de se resolver o problema dual do que o problema primal. Um bom exemplo disso é quando o problema primal é linear com um número muito grande de variáveis e um número pequeno de restrições, pois como veremos, as variáveis do problema primal estarão associadas às restrições do problema dual e as restrições do problema primal gerarão as variáveis do problema dual.

Para compreendermos bem as relações entre os problemas primal e dual vamos estudar o caso não-linear separadamente do caso linear, como fazem [Izmailov e Solodov](#) em [18].

1.3.1 Caso Não-Linear

Um problema de otimização não-linear com restrições de igualdade e desigualdade é escrito da seguinte forma:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.a} \quad & h(x) = 0 \\ & g(x) \leq 0 \end{aligned} \tag{1.4}$$

em que $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ e $g : \mathbb{R}^n \rightarrow \mathbb{R}^l$. Tomando $\lambda \in \mathbb{R}^m$ e $\mu \in \mathbb{R}_+^l$, podemos definir a função Lagrangiana $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_+^l \rightarrow \mathbb{R}$:

$$\mathcal{L}(x, \lambda, \mu) := f(x) + \lambda^T h(x) + \mu^T g(x). \tag{1.5}$$

Note que para todo x viável temos que $\lambda^T h(x) = 0$, pois $h(x) = 0$ e como μ é não-negativo então $\mu^T g(x) \leq 0$ já que $g(x) \leq 0$. Logo $\mathcal{L}(x, \lambda, \mu) \leq f(x)$, para todos os pontos viáveis do problema (1.4) e veja que se $\mu = 0$, então

$$\mathcal{L}(x, \lambda, 0) = f(x) + 0 + 0 = f(x),$$

logo o $\sup_{(\lambda, \mu) \in \mathbb{R}^m \times \mathbb{R}_+^l} \mathcal{L}(x, \lambda, \mu)$ coincide com $f(x)$ para todo x viável.

Agora, se x é **inviável** então $h_i(x) \neq 0$ para algum $i \in \{1, \dots, m\}$ e/ou $g_j(x) > 0$ para algum $j \in \{1, \dots, l\}$. Supondo que $g_j(x) \leq 0$ para todo $j \in \{1, \dots, l\}$ e $h_i(x) \neq 0$ para algum $i \in \{1, \dots, m\}$, vamos tomar

$$\lambda_j = \begin{cases} kh_i(x), & \text{se } j = i, \\ 0, & \text{se } j \in \{1, \dots, m\} \setminus \{i\}, \end{cases} \quad (1.6)$$

com $k > 0$. Com $\mu = 0$ e fazendo $k \rightarrow \infty$ obtemos

$$\mathcal{L}(x, \lambda, \mu) = f(x) + kh_i(x)^2 \rightarrow \infty.$$

Agora, supondo que $g_j(x) > 0$ para algum $j \in \{1, \dots, l\}$ e $h_i(x) = 0$ para todo $i \in \{1, \dots, m\}$, vamos tomar $\lambda = 0$ e

$$\mu_i = \begin{cases} k, & \text{se } i = j, \\ 0, & \text{se } i \in \{1, \dots, l\} \setminus \{j\}. \end{cases} \quad (1.7)$$

com $k > 0$. Novamente, fazendo $k \rightarrow \infty$ obtemos

$$\mathcal{L}(x, 0, \mu) = f(x) + kg_j(x) \rightarrow \infty.$$

Com isso, mostramos que

$$\sup_{(\lambda, \mu)} \mathcal{L}(x, \lambda, \mu) = \begin{cases} f(x) & \text{se } x \text{ é viável} \\ \infty & \text{se } x \text{ é inviável.} \end{cases}$$

Então, reescrevendo o problema temos:

$$\min_{x \in \Omega} \left\{ \sup_{(\lambda, \mu) \in \mathbb{R}^m \times \mathbb{R}_+^l} \mathcal{L}(x, \lambda, \mu) \right\}, \quad (1.8)$$

com $\Omega := \{x \in \mathbb{R}^n \mid \sup_{(\lambda, \mu) \in \mathbb{R}^m \times \mathbb{R}_+^l} \mathcal{L}(x, \lambda, \mu) < \infty\}$. Pela definição do conjunto Ω , podemos trocar a ordem entre a maximização e a minimização sem alterar o valor ótimo, bem como o **supremo** pelo **ínfimo**, desde que este último também seja finito. Dessa forma, obtemos o problema dual

$$\max_{(\lambda, \mu) \in \Delta} \left\{ \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \mu) \right\}, \quad (1.9)$$

em que $\Delta := \{(\lambda, \mu) \in \mathbb{R}^m \times \mathbb{R}_+^l : \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \mu) > -\infty\}$, que podemos escrever como

$$\max_{(\lambda, \mu) \in \Delta} \phi(\lambda, \mu) \quad (1.10)$$

em que $-\phi$ é uma função convexa não-diferenciável, com ϕ definida da seguinte maneira:

$$\phi(\lambda, \mu) := \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \mu).$$

Em [2, Sec. 6.1] os autores apresentam a interpretação geométrica do problema dual, iremos reproduzir aqui essa interpretação de maneira mais ilustrativa na Figura 8 e assim como os autores, vamos assumir que não há restrições de igualdade ($m = 0$), temos apenas uma restrição de desigualdade ($l = 1$) e o domínio da variável x é um conjunto convexo e limitado X , apenas para simplificar a nossa visualização.

Vamos trabalhar no plano y - z e por meio das relações $y = g(x)$ e $z = f(x)$ para todo $x \in X$, construímos a região G como a imagem do mapeamento de X no plano y - z . O teorema de Weierstrass (conforme [18, Teorema 1.2.1]) vai nos garantir que f assume valores máximo e mínimo em X , indicados na Figura 8(a) pelas ordenadas dos pontos A e B , respectivamente. Definimos também a função

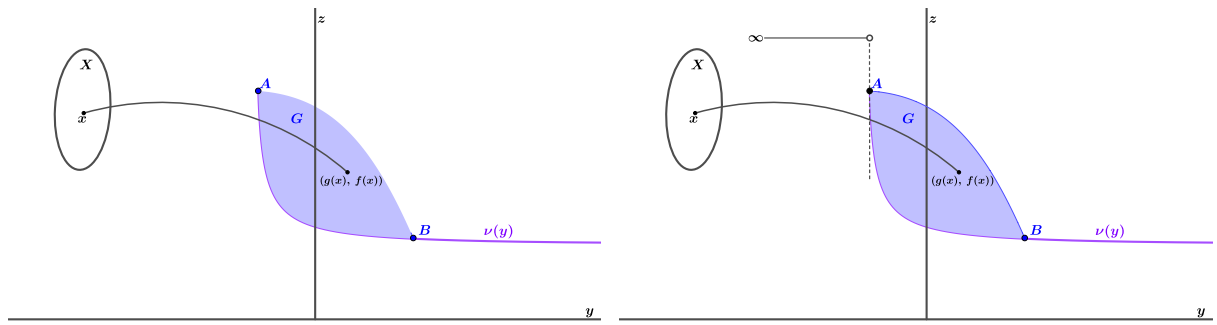
$$\nu(y) := \min\{f(x) : g(x) \leq y, \forall x \in X\}.$$

Podemos observar na Figura 8(b) que para os pontos à esquerda de A , temos $\nu(y) = \infty$, isso devido à **inviabilidade** dual, ou seja, a reta tangente a ν no ponto A é paralela ao eixo z ; e para pontos à direita de B , temos que a função ν permanece constante, pois a ordenada de B é o valor mínimo que a função f assume em X .

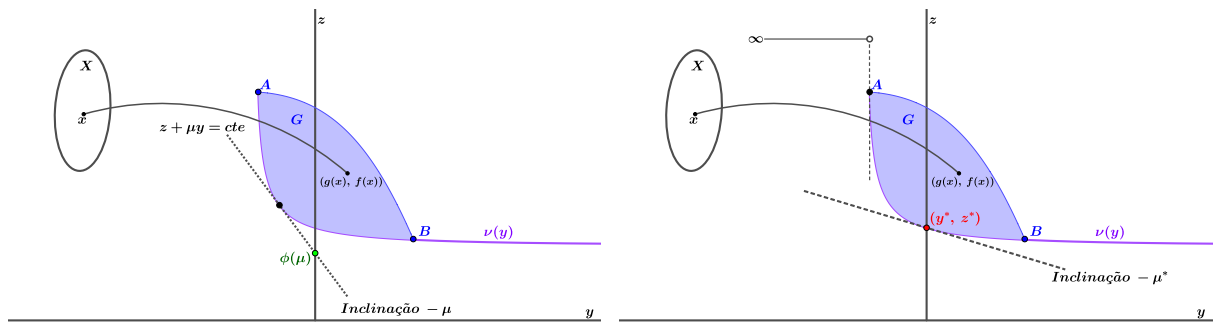
Perceba que a **compacidade** de X não é essencial. Essa hipótese foi adicionada para que a ilustração fique mais simples, pois se X fosse ilimitado, por exemplo, teríamos que as ordenadas dos pontos A e B poderiam tender a mais e menos infinito, respectivamente.

Suponha agora que μ seja dado. Para determinar $\phi(\mu)$, precisamos minimizar $f + \mu g$, para todo $x \in X$, ou seja, queremos minimizar $z + \mu y$ sobre os pontos em G . Observe que $z + \mu y = cte$ é a equação de uma linha reta com inclinação $-\mu$ e intercepta o eixo z em cte . Para minimizar $z + \mu y$ sobre G , precisamos mover a reta $z + \mu y = cte$ paralelamente a si mesma o mais abaixo possível (ou seja, na direção oposta de seu gradiente) de forma que ela ainda permaneça em contato com G . Em outras palavras, movemos essa linha paralelamente até que o conjunto G fique acima da reta e o tangencie. A intersecção dessa reta com o eixo z fornece $\phi(\mu)$, como mostra a Figura 8(c).

Note que o problema primal consiste em encontrar um ponto em G , tal que a **abscissa** $y \leq 0$ possua **ordenada** z mínima. Claramente podemos observar que esse ponto é (y^*, z^*) na Figura 8(d). O problema dual é, portanto, equivalente a encontrar a inclinação do hiperplano suporte de G , de modo que sua intersecção com o eixo z seja máxima. Na Figura 8(d), esse hiperplano possui a inclinação $-\mu^*$ e tangencia o conjunto G no ponto (y^*, z^*) . Assim, a solução dual ótima é μ^* e o valor ótimo da função objetivo do problema dual é z^* . Além disso, os valores objetivo primal e dual são iguais neste caso.



(a) Mapeamento de X no plano y - z e curva $\nu(y)$. (b) Fronteira de inviabilidade do problema dual.



(c) Restrição perturbada $g(x) \leq y$ e valor dual $\phi(\mu)$. (d) Solução ótima do problema dual, quando $y = 0$.

Figura 8 – Ilustração da interpretação geométrica do problema dual.

1.3.2 Caso Linear

Dado o problema de otimização linear da forma:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.a} \quad & Ax \geq b, \end{aligned} \tag{1.11}$$

em que $c, x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ e $A \in \mathbb{R}^{m \times n}$; podemos definir a função Lagrangiana: $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}_+^m \rightarrow \mathbb{R}$

$$\mathcal{L}(x, \mu) := c^T x + \mu^T (b - Ax), \tag{1.12}$$

e a partir daí, conseguimos escrever uma função ϕ associada ao problema (1.11):

$$\begin{aligned} \phi(\mu) &= \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \mu) \\ &= \inf_{x \in \mathbb{R}^n} \{c^T x + \mu^T (b - Ax)\} \\ &= \inf_{x \in \mathbb{R}^n} \{(c - A^T \mu)^T x + \mu^T b\}. \end{aligned} \tag{1.13}$$

Para que o ínfimo do problema (1.13) exista para qualquer $x \in \mathbb{R}^n$, isto é, que $\phi(\mu) > -\infty$, precisamos tomar $c - A^T \mu = 0$, daí temos que $\inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \mu) = \mu^T b$. O problema dual será

então dado pela maximização de $\phi(\mu)$, ou seja

$$\begin{aligned} \max \quad & b^T \mu \\ \text{s.a} \quad & A^T \mu = c \\ & \mu \geq 0. \end{aligned} \tag{1.14}$$

Note que o dual de um problema linear também é um problema linear, só que enquanto o problema primal é um problema de minimização o dual é um problema de maximização. Também podemos perceber que o vetor de custos do problema primal (1.11) c se tornou o vetor dos termos independentes nas restrições do problema dual (1.14) e vice-versa.

1.3.3 O Caso Quadrático

Um problema de otimização quadrático estritamente convexo é escrito da seguinte forma:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Qx + p^T x \\ \text{s.a} \quad & Ax = b \\ & Cx \leq d, \end{aligned} \tag{1.15}$$

em que $x, p \in \mathbb{R}^n$; $b \in \mathbb{R}^m$; $d \in \mathbb{R}^l$; $A \in \mathbb{R}^{m \times n}$; $C \in \mathbb{R}^{l \times n}$ e $Q \in \mathbb{R}^{n \times n}$ é simétrica e definida positiva. Colocando esse problema na forma padrão, temos:

$$\begin{aligned} \min \quad & \frac{1}{2}x^T Qx + p^T x \\ \text{s.a} \quad & Ax - b = 0 \\ & Cx - d \leq 0, \end{aligned} \tag{1.16}$$

Seu Lagrangiano é dado por:

$$\begin{aligned} \mathcal{L}(x, \lambda, \mu) &= \frac{1}{2}x^T Qx + p^T x + \lambda^T(b - Ax) + \mu^T(Cx - d) \\ &= \frac{1}{2}x^T Qx + p^T x - \lambda^T Ax + \lambda^T b + \mu^T Cx - \mu^T d \\ &= \frac{1}{2}x^T Qx + x^T p - x^T A^T \lambda + b^T \lambda + x^T C^T \mu - d^T \mu \\ &= \frac{1}{2}x^T Qx + x^T (p - A^T \lambda + C^T \mu) - d^T \mu + b^T \lambda. \end{aligned} \tag{1.17}$$

A partir daí, conseguimos escrever uma função Φ associada ao problema (1.15):

$$\begin{aligned} \Phi(\lambda, \mu) &= \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \mu) \\ &= \inf_{x \in \mathbb{R}^n} \left\{ \frac{1}{2}x^T Qx + x^T (p - A^T \lambda + C^T \mu) - d^T \mu + b^T \lambda \right\}. \end{aligned} \tag{1.18}$$

Vamos verificar se o ínfimo dessa quadrática é atingido e para isso calculamos o gradiente da Lagrangiana em relação a x , obtendo:

$$\nabla_x \mathcal{L}(x, \lambda, \mu) = Qx + (p - A^T \lambda + C^T \mu).$$

Queremos encontrar os pontos estacionários da função Lagrangiana, logo, igualando esse gradiente a zero temos

$$\begin{aligned} \nabla_x \mathcal{L}(x, \lambda, \mu) &= 0 \\ \Rightarrow Qx + (p - A^T \lambda + C^T \mu) &= 0 \\ \Rightarrow Qx &= A^T \lambda - p - C^T \mu. \end{aligned}$$

Por hipótese, Q é definida positiva, então ela é invertível, o que nos garante que existe um minimizador \bar{x} que é escrito da seguinte forma:

$$\bar{x} = Q^{-1}(A^T \lambda - p - C^T \mu). \quad (1.19)$$

Então, o ínfimo da Lagrangiana é atingido em \bar{x} e partir dessa solução podemos substituir x por \bar{x} em (1.18):

$$\begin{aligned} \Phi(\lambda, \mu) &= \min_x \mathcal{L}(x, \lambda, \mu) \\ &= \min_x \left\{ \frac{1}{2} x^T Q x + p^T x + \lambda^T (b - Ax) + \mu^T (Cx - d) \right\} \\ &= -\frac{1}{2} (A^T \lambda - p - C^T \mu)^T Q^{-1} (A^T \lambda - p - C^T \mu) + \lambda^T b + \mu^T d. \end{aligned} \quad (1.20)$$

O problema dual será então dado pela maximização de $\Phi(\lambda, \mu)$ sujeito a $\mu \geq 0$, ou seja, o dual de um problema quadrático também é um problema quadrático, cuja única restrição é a não negatividade do multiplicador μ , o que torna esse problema muito mais simples que o problema primal (1.15). No capítulo 4 iremos explorar essa vantagem.

Cabe observar que o caso mais interessante de se trabalhar com o problema dual ocorre quando ele pode ser decomposto, mas o primal não. Por exemplo, quando o primal possui função objetivo separável, e restrições de acoplamento, bem como restrições lineares-afins separáveis, descritas por conjuntos X_i , isto é,

$$\begin{aligned} \min \quad & \sum_i f_i(x_i) \\ \text{s.a} \quad & Ax = b \\ & x_i \in X_i, \end{aligned}$$

com cada $x_i \in \mathbb{R}^{n_i}$, o problema dual tira proveito da separabilidade.

1.4 Números duais e a diferenciação automática

A diferenciação automática é uma técnica para calcular derivadas numéricas de forma exata, ou seja, sem erros advindos de arredondamento ou aproximações.

A ideia é definir um símbolo algébrico ε tal que $\varepsilon \neq 0$, mas que $\varepsilon^2 = 0$. Assim como os números complexos que envolvem o símbolo algébrico i , podemos definir os *números duais*, que envolvem o símbolo algébrico ε , por exemplo, $1 + 2\varepsilon$ ou $\pi - 3\varepsilon$.

Uma vez definida a aritmética de números duais (mais detalhes em [3]), similar à dos números complexos, com as devidas adaptações decorrentes de $\varepsilon^2 = 0$, para encontrarmos a derivada de alguma função f em um ponto x de forma exata, precisamos apenas calcular $f(x + \varepsilon)$, pois

$$f(x + \varepsilon) = f(x) + f'(x)\varepsilon.$$

Isso fica claro quando escrevemos a expansão em série de Taylor de $f(x + \varepsilon)$:

$$f(x + \varepsilon) = f(x) + f'(x)\varepsilon + \frac{f''(x)\varepsilon^2}{2!} + \frac{f'''(x)\varepsilon^3}{3!} + \dots$$

Como $\varepsilon^2 = 0$, apenas os primeiros dois termos não se anulam, logo, o valor que multiplica ε é exatamente o valor da primeira derivada. Então, suponha que $f(x) = x^2$ e queremos calcular o valor de $f'(1)$, então basta calcularmos o valor de $f(1 + \varepsilon)$:

$$\begin{aligned} f(1 + \varepsilon) &= (1 + \varepsilon)^2 \\ &= 1 + 2\varepsilon + \underbrace{\varepsilon^2}_0 \\ &= 1 + 2\varepsilon, \end{aligned}$$

daí temos que $f(1) = 1$ e $f'(1) = 2$, o que sabemos que é verdade. E isso vale para qualquer função analítica, pois os números duais respeitam as regras de derivação (regra da cadeia, regra do produto, regra do quociente e a regra do tombo).

Esse recurso será muito útil, pois como veremos na próxima seção, os métodos de otimização não suave, em geral, baseiam-se na existência de um *oráculo*, isto é, dados uma função f e um ponto x , o oráculo deve ser capaz de calcular o valor funcional e pelo menos um subgradiente de f no ponto x . Com a diferenciação automática não precisaremos nos preocupar em construir um oráculo para cada função, pois nosso oráculo vai usar o pacote *ForwardDiff*¹, que contém rotinas capazes de obter derivadas, gradientes, jacobianas, *hessianas* e derivadas de ordem superior de funções no *Julia*, usando o modo progressivo de diferenciação automática (*AD*). Em [24] os autores apresentam com mais detalhes a representação multidimensional de números duais em *Julia* e advogam o desempenho competitivo do pacote, bem como as vantagens da compilação *just-in-time* (*JIT*) para recompilar de forma transparente o código de um usuário não habituado com *AD*.

Cabe ressaltar que a diferenciação automática e particularmente o pacote *ForwardDiff* não são uma panaceia, e podem falhar. De fato, o funcionamento depende não só de como as funções intrínsecas estão implementadas internamente, como também da forma com que as definimos em nossa programação.

¹ Toda a documentação desse pacote pode ser encontrada no *GitHub*.

Por exemplo, a função $f(S) = \|S - A\|_2$ tem apenas um ponto de não suavidade, dado por $S = A$. Para $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ e $A = (4, 9)^T$, quando tentamos calcular um subgradiente de f no ponto A no Julia utilizando a função nativa para a norma obtemos:

```
julia> A=[4.; 9.];
julia> f(S)=norm(S-A);
julia> ForwardDiff.gradient(f, A)
2-element Array{Float64,1}:
 0.0
 1.0
```

Figura 9 – Janela do Julia com a definição e o cálculo de um subgradiente da função f em A .

E de fato, podemos facilmente verificar que o vetor obtido é um subgradiente da f em A :

$$\begin{aligned}
 f(Y) &= \|Y - A\|_2 \\
 &= \sqrt{(A_1 - Y_1)^2 + (A_2 - Y_2)^2} \\
 &\geq \sqrt{(A_2 - Y_2)^2} \\
 &= |A_2 - Y_2| \\
 &\geq 9 - Y_2,
 \end{aligned} \tag{1.21}$$

logo $f(Y) \geq 9 - Y_2$. Agora, avaliando a linearização obtida com o vetor $S = (0, 1)^T$ no ponto $A = (4, 9)^T$ temos

$$\begin{aligned}
 f(A) + S^T(A - Y) &= 0 + S^T(A - Y) \\
 &= S^T A - S^T Y \\
 &= \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} 4 \\ 9 \end{pmatrix} - \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \\
 &= 9 - Y_2.
 \end{aligned} \tag{1.22}$$

De (1.21) e (1.22), temos

$$f(Y) \geq f(A) + S^T(A - Y), \quad \forall Y \in \mathbb{R}^2,$$

então, pela Definição 3, $S = (0, 1)^T$ é subgradiente de f em A .

No entanto, se reescrevermos a função f de outra maneira, por exemplo, $g(S) = \sqrt{(S - A)^T(S - A)}$, para $g : \mathbb{R}^2 \rightarrow \mathbb{R}$ e $A = (4, 9)^T$, quando tentamos calcular um subgradiente de f no ponto A no Julia obtemos:

```
julia> A=[4.; 9.];  
  
julia> g(S)=sqrt(dot(S-A, S-A));  
  
julia> ForwardDiff.gradient(g, A)  
2-element Array{Float64,1}:  
 NaN  
 NaN
```

Figura 10 – Janela do Julia com a definição e indeterminação no cálculo do subgradiente da g em A .

Nos experimentos deste trabalho, tiramos proveito dessa característica, pois nos problemas considerados, a função objetivo é constituída pela soma de normas da diferença entre dois pontos, de modo que conhecemos os pontos em que a função é não suave. Como vimos acima, usando a função nativa para a norma, podemos assegurar que o pacote é capaz de calcular um subgradiente nesses pontos.

2 Métodos Computacionais para Otimização Não Suave

A fim de compreendermos melhor o *Método de Feixes em Níveis*, que será o foco do nosso trabalho, nesta seção iremos nos debruçar sobre os métodos computacionais mais comumente conhecidos para Otimização Não Suave e também sobre os seus principais aspectos. Iremos nos apoiar em [5] e [19] para fazer essa introdução.

Assim como na Otimização Suave, podemos classificar os métodos computacionais em duas classes: os de primeira ordem e os de segunda ordem, ou seja, métodos que utilizam apenas a primeira derivada e métodos que também trabalham com a segunda derivada. Ambas as classes tratam de métodos iterativos que têm como objetivo gerar uma sequência $\{x^k\}$ cujos pontos de acumulação são minimizadores do problema, sempre que $\{x^k\}$ é convergente. No diagrama abaixo podemos ter uma ideia de onde se encaixa cada um dos métodos que iremos apresentar.



Figura 11 – Diagrama com os métodos apresentados neste capítulo.

Tanto os métodos de primeira ordem quanto os de segunda ordem buscam determinar uma direção d^k , em que no melhor caso será uma direção de descida, e um tamanho de passo $\lambda_k > 0$ e com isso temos que $x^{k+1} = x^k + \lambda_k d^k$.

Queremos chamar a atenção aqui para o fato da direção d^k ser uma direção de descida da função f num ponto x nos melhores casos. Note que em Otimização Suave, uma direção d era dita de descida para f num ponto x se $d^T \nabla f(x) < 0$. Estendendo essa ideia para Otimização Não Suave, podemos definir uma direção de descida da seguinte maneira:

Definição 6. Se a direção $d \in \mathbb{R}^n$ satisfaz

$$d^T y < 0 \quad \forall y \in \partial f(x),$$

então ela é dita direção de descida.

Apenas com a Definição 6 já vemos que para caracterizar uma direção de descida precisamos conhecer todo o conjunto subdiferencial, o que nem sempre é possível. Então, do ponto de vista prático, estamos interessados em métodos capazes de funcionar com informações incompletas do subdiferencial, ou seja, métodos eficientes para resolver o problema:

$$\min_{x \in \mathbb{R}^n} f(x) \tag{2.1}$$

em que $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função convexa e não diferenciável.

Tendo isso em vista, todos os métodos que apresentaremos nas próximas seções conterão uma rotina denominada *oráculo*. Essa rotina é capaz de fornecer o valor da função num ponto bem como um subgradiente da função no ponto. Uma outra nomenclatura conhecida pra essa rotina é *caixa-preta* ou *simulador*, pois tais informações podem provir experimentalmente, ou por meio de simulações.

2.1 Métodos de Subgradiente

O primeiro método que vamos apresentar é o *Método de Subgradientes*. É o método mais simples (e também o mais rústico e ineficiente que existe), mas a partir do entendimento dele poderemos compreender a evolução e criação dos métodos mais sofisticados. Sua descrição é dada no Algoritmo 1, que segue a apresentação de [5, Algorithm 9.2].

Aqui nos deparamos com a primeira problemática da Otimização Não Suave: *o critério de parada*. Note que no caso suave, bastava verificarmos se $\|\nabla f(x^k)\| < \epsilon_{\text{tol}}$, ou seja, se o gradiente da k -ésima iteração está próximo do vetor nulo. Já no caso não suave, precisamos verificar se o vetor nulo pertence ao subdiferencial da função na k -ésima

iteração, ou seja, queremos testar se $\|s^k\| < \epsilon_{\text{tol}}$ com $s^k \in \partial f(x^k)$, o que pode nunca acontecer. Isso porque o oráculo não é capaz de calcular todo subdiferencial. Tomemos a função valor absoluto como exemplo, ou seja $f(x) = |x|$ e suponha que o oráculo retorne sempre um subgradiente no fecho do subdiferencial. Sabemos que $x^k = 0$ é o minimizador de f , porém nesse caso $|s^k| = 1$, então se $\epsilon_{\text{tol}} < 1$ o critério de parada nunca seria satisfeito.

Algoritmo 1 – MÉTODOS DE SUBGRADIENTES

Entrada: $x^1 \in \mathbb{R}^n$ e $itmax \in \mathbb{N}$.

$k = 1$

repita

Passo 1: Receber informação do oráculo

Chamamos o oráculo para obter $f(x^k)$ e um subgradiente s^k

Passo 2: Definir a candidata a direção de descida

$$d^k = -\frac{s^k}{\|s^k\|}$$

Passo 3: Efetuar busca linear

Encontramos, se possível, o passo $\lambda_k > 0$ satisfazendo $f(x^k + \lambda_k d^k) < f(x^k)$.

Passo 4: Atualizar o iterado

$$x^{k+1} = x^k + \lambda_k d^k$$

$k \leftarrow k + 1$

até $k \geq itmax$;

Saída: x

Note também que no Passo 3, temos uma busca linear relaxada, pois como o oráculo nos dá apenas um subgradiente e não o subdiferencial todo, não é possível dizer se d^k é uma direção de descida. Diferentemente do caso suave, em que a direção oposta ao gradiente é sempre de descida, podemos ter vetores opostos aos subgradientes que não sejam de descida. Consequentemente, é difícil estabelecer regras claras para o cálculo de λ_k e as mais utilizadas têm uma taxa de convergência *sublinear*. Quando o valor ótimo da função é conhecido, é possível fazer algumas melhorias para acelerar um pouco o método, mas conhecer o valor ótimo é, em geral, uma hipótese forte.

Tendo isso em vista, começa a ficar claro que as implementações desses métodos podem ser ineficientes devido aos problemas inerentes do algoritmo, por exemplo critérios de parada nunca atingido, geração de direções que não são de descida, taxa de convergência lenta, entre outros.

Contudo, já era de se esperar que não fosse possível ter um método completamente eficiente utilizando um único vetor do subdiferencial a cada iteração. Pensando

nisso podemos pensar numa outra abordagem para atacar os problemas não suaves, em que cada subgradiente calculado é salvo na memória e utilizado na iteração seguinte.

2.2 Método de Planos Cortantes

Se nós guardarmos o subgradiente de cada iteração, podemos construir um modelo com planos que subaproximam o **epígrafo** da função objetivo, isto é,

$$\text{epi}(f) := \{(y, z) \in \mathbb{R}^n \times \mathbb{R} \mid f(y) \leq z\}. \quad (2.2)$$

Então, dados

$$f_i := f(x^i) \quad \text{e} \quad s^i := s(x^i), \quad \forall i = 1, 2, \dots, k, \quad (2.3)$$

podemos construir um modelo, \tilde{f} da seguinte maneira:

$$\tilde{f}_k(y) := \max_{i=1, \dots, k} \{f_i + \langle s^i, (y - x^i) \rangle\}.$$

Cabe uma observação referente à notação para o produto interno: a partir desse ponto, passaremos a utilizar $\langle u, v \rangle$ ao invés de $u^T v$, para melhor destaque e apresentação dos elementos que envolverão índices e expressões.

Para encontrar a solução do problema (2.1), a cada iteração iremos resolver o subproblema de minimizar o modelo sobre um conjunto convexo e compacto \mathbb{X} . No Algoritmo 2 vai ficar claro que se \mathbb{X} não for compacto, nas primeiras iterações o modelo poderia ser ilimitado inferiormente. Na Figura 12 podemos visualizar a filosofia do método.

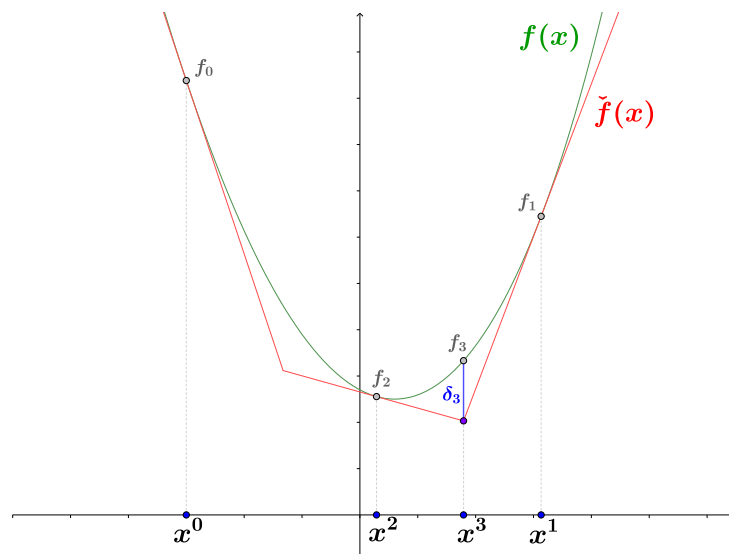


Figura 12 – Algumas iterações do método de Planos Cortantes

Pela definição de subgradiente (Definição 3) e pela construção do modelo, temos que

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

e como f é convexa, o epígrafo dos planos cortantes \check{f}_k vai se aproximando por baixo do epígrafo da f conforme k vai aumentando. Então, é possível obtermos um critério de parada bem definido, que seria o *decréscimo nominal* do modelo \check{f} com relação à função f , que iremos denotar por δ . Se δ for menor que uma certa tolerância, decretamos a parada do método. Perceba que apesar de termos um critério de parada bem definido não significa que já estamos no melhor dos mundos. Note que a sequência de valores funcionais geradas por esse método não é necessariamente *monótona*. Um bom exemplo disso pode ser observado na Figura 12, pois a sequência $\{f_i\}$ começa decrescente, porém $f_3 > f_2$. Mais adiante veremos que esse aspecto não-monótono da sequência de valores funcionais será uma das motivação para a criação dos Métodos de Feixes.

Algoritmo 2 – MÉTODO DE PLANOS CORTANTES

Entrada: $x^1 \in \mathbb{X}$, $\epsilon_{\text{tol}} > 0$ e $itmax \in \mathbb{N}$.

$k = 1$

$\check{f}_0 = -\infty$

repita

Passo 1: Receber informação do oráculo e verificar critério de parada

Chame o oráculo para obter $f(x^k)$ e um subgradiente s^k

Calcule $\delta_k = f(x^k) - \check{f}_{k-1}(x^k)$

se $\delta_k < \epsilon_{\text{tol}}$ **então**

| **retorna** x^k

fim

Passo 2: Definir a candidata a direção de descida

$d^k \in \arg \min_{x^k+d \in \mathbb{X}} \check{f}_k(x^k + d)$

Passo 3: Atualizar iterado

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

$k \leftarrow k + 1$

até $\delta_k \leq \epsilon_{\text{tol}}$ ou $k \geq itmax$;

Saída: x

Se o epígrafo de f tiver um formato *poliedral* com poucas faces ou um formato de V , este método pode ter um bom desempenho, no sentido de que para qualquer tolerância ϵ_{tol} , em particular para $\epsilon_{\text{tol}} = 0$, o método converge em número finito de passos, pois funções lineares por partes vão coincidir localmente com as aproximações do método de planos cortantes. Mas em geral, não é esse o caso. E em um cenário mais caótico, se a

função poliedral tiver muitas faces perto da solução, então o método vai precisar de muitas iterações para convergir.

Além da geração de uma sequência, possivelmente não monótona de valores funcionais, ainda temos alguns outros empecilhos. O principal deles é a instabilidade numérica. De certa forma, o fato do método não ter nenhum controle sobre os valores funcionais está relacionado com a instabilidade numérica. Na Figura 13 podemos ver que ao se aproximar do minimizador de f , os planos de corte vão ficando quase horizontais e conseqüentemente podemos gerar aproximações que acabam se afastando do minimizador.

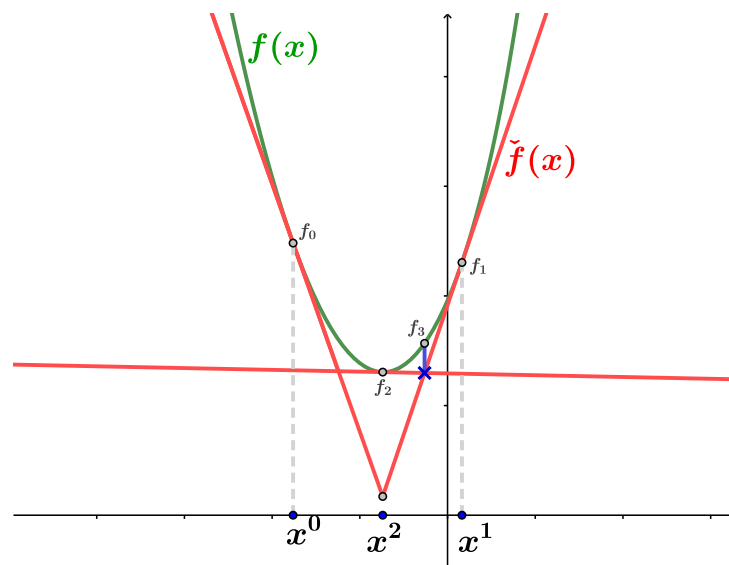


Figura 13 – Note que x^2 seria um bom candidato para minimizador, porém ao adicionar o plano cortante associado a ele, o minimizador do modelo acaba se afastando do minimizador da f , ou seja, o valor funcional f_3 seria maior que f_2 .

Uma outra dificuldade está na quantidade crescente de planos cortantes conforme as iterações aumentam, o que amplia o número de restrições do subproblema. Em geral, o excesso de planos de corte dificulta a resolução do subproblema de forma eficiente. Visando estabilizar o processo e também ganhar eficiência na obtenção de um minimizador, surgiram os métodos de feixes, que veremos a seguir.

2.3 Método de Feixes

Nesta seção iremos apresentar a filosofia do Método de Feixes. Primeiramente, vejamos o que seria um *feixe*. Assim como o método de planos cortantes, este método também mantém uma memória dos pontos que são avaliados pelo oráculo até a iteração corrente, bem como seus respectivos subgradientes; esse conjunto de dados, pontos e subgradientes, é denominado *feixe*¹ e é daí que vem a nomenclatura. Antes de apresentarmos a variante de conjuntos de níveis, que é o foco deste trabalho, iremos explicar como funciona essa estabilização numérica do método, que é um denominador comum entre suas variantes, e verificar a equivalência do Método de Feixes em Níveis com algumas variantes. Aqui já podemos notar um contraste entre o método de feixes e o de planos cortantes: os subproblemas dos planos cortantes são problemas lineares, que são em geral os mais simples de se resolver. Já no método de feixes, embora existam variantes com subproblemas lineares, não lineares e até de programação inteira mista (conforme Frangioni [15]), usualmente os subproblemas são quadráticos. Apesar de parecer que complicamos os subproblemas, veremos mais adiante que na verdade eles têm uma estrutura bastante especial e é justamente essa estrutura que vai nos permitir reduzir o tamanho do subproblema. Então, o custo benefício de se trocar um subproblema linear por, por exemplo, um quadrático é muito bom, pois além de conseguirmos reduzir o tamanho do subproblema (que era uma das desvantagens do método de plano cortantes), ainda ganhamos a estabilidade numérica como bônus.

2.3.1 Estabilização

A ideia da estabilização é impedir que a sequência se afaste do minimizador, como na Figura 13. Para isso, seria interessante que o algoritmo guardasse o ponto com o *melhor* valor obtido até a iteração corrente. Logo, notamos que esse algoritmo irá gerar duas sequências: uma sequência de pontos y^k , a partir dos quais iremos construir o modelo, e uma sequência de pontos x^k , que consiste dos pontos y^k que produziram decréscimo suficiente na função objetivo, ou seja, $\{x^k\}$ é uma sequência cujos elementos pertencem à sequência $\{y^k\}$. Iremos denominar os pontos y^k como *candidatos* e os pontos x^k como *centros de estabilidade*. Para gerar essas duas sequências iremos formular um subproblema de acordo com os seguintes princípios de estabilização:

- (a) A escolha de um modelo φ_k que seja convexo e conforme apontado por Frangioni em [14], em geral seja uma subaproximação para a função objetivo, isto é, para todo

¹ Em inglês: *bundle*.

k temos $\varphi_k(y) \leq f(y)$ para qualquer $y \in \mathbb{R}^n$. Isso é o que vai nos permitir calcular um *decréscimo nominal*. É importante trocarmos nossa notação, pois mais adiante veremos que podemos construir um modelo que agrega todas as informações dos planos de corte, porém sem necessariamente guardarmos todos os subgradientes que os geraram, isso vai nos permitir fazer uma redução no feixe. A priori, podemos continuar utilizando os modelos dos planos de corte, ou seja, $\varphi_k = \check{f}_k$, mas para isso precisamos fazer uma correção ao modelo, uma vez que agora estamos interessados em um modelo em relação ao centro de estabilidade x^k . Para isso, precisamos definir o erro que estamos cometendo ao assumir a linearização como uma aproximação para a função objetivo em torno do centro de estabilidade. Relembrando da notação introduzida em (2.3), vamos definir

$$\xi_i := f(x^k) - [f_i + \langle s^i, (x^k - y^i) \rangle], \quad i = 1, 2, \dots, k.$$

Note que o termo entre colchetes é justamente a subaproximação de f , logo $\xi_i > 0$ para todo i . E se relembrarmos da Definição 5, temos que $s^i \in \partial_{\xi_i} f(x^k)$. Com isso, podemos finalmente definir o feixe:

$$\mathcal{B} := \{(s^i, \xi_i) : s^i \in \partial_{\xi_i} f(x^k), i = 1, 2, \dots, k\},$$

associado ao centro de estabilidade, seu valor funcional e seu subgradiente, ou seja, a tripla $(x^k, f(x^k), s(x^k))$. Com isso, podemos escrever nosso modelo como sendo

$$\varphi_k(y) = f(x^k) + \max_{i=1, \dots, k} \{-\xi_i + \langle s^i, (y - x^k) \rangle\}. \quad (2.4)$$

Vale destacar que essa descrição para o feixe em que trocamos os pontos pelos erros da aproximação nesses pontos permite economizar memória e ganhar eficiência.

- (b) A escolha do centro de estabilidade x^k , que será o ponto com o melhor valor obtido até a iteração corrente. Existem diversas maneiras de se determinar o centro de estabilidade, em geral essas diferentes maneiras é que são responsáveis pela existência das variantes do método de feixes. Em outras palavras, cada variante vai definir uma forma diferente de se obter o centro de estabilidade. Neste trabalho iremos apresentar três variantes, mas o leitor pode encontrar diversas outras na referência [20]. O centro de estabilidade é o que vai de fato impedir que a sequência se afaste do minimizador, pois x^k será atualizado apenas se o novo candidato y^{k+1} produzir um *decréscimo* suficiente na função objetivo. Quando isso ocorre dizemos que tivemos um *passo sério*²; já quando y^{k+1} não gera um *decréscimo* suficiente na função objetivo o centro de estabilidade da próxima iteração permanece igual, ou seja, $x^{k+1} = x^k$ e quando isso ocorre dizemos que tivemos um *passo nulo*³.

² Em inglês: *serious step*.

³ Em inglês: *null step*.

- (c) A escolha de uma norma $|\cdot|_k$ adequada, com a qual iremos controlar a distância dos candidatos ao melhor ponto obtido até o momento, ou seja, a norma da diferença entre o candidato e o centro de estabilidade. Então, seja M_k uma matriz quadrada, simétrica e definida positiva; vamos tomar a norma induzida pelo produto interno:

$$|\cdot|_k^2 := \langle M_k \cdot, \cdot \rangle \quad (2.5)$$

e sua norma dual

$$\|\cdot\|_k^2 := \langle \cdot, M_k^{-1} \cdot \rangle. \quad (2.6)$$

O papel dessa norma é impedir grandes oscilações, então mesmo que a sequência $\{f(y^k)\}$ esteja tendo oscilações, a sequência $\{f(x^k)\}$ vai ser sempre monótona não crescente. E note que o subíndice k implica que essa norma pode variar ao longo das iterações. Com isso, no início podemos escolher uma norma mais tolerante, para permitir passos maiores enquanto estamos longe da solução e conforme k vai aumentando, podemos aumentar a exigência por meio da norma, o que também é uma forma de prevenir a instabilidade numérica que ocorreria próximo da solução. Mas é importante destacarmos aqui que essa atualização da norma só é conveniente quando temos passos sérios, uma vez que os passos nulos não geram nenhuma atualização do centro de estabilidade.

Vamos apresentar o algoritmo genérico para o Método de Feixes, porém algumas observações são pertinentes. No Passo 2, o subproblema estabilizado foi propositalmente escrito de maneira abstrata, apenas deixando explícita a dependência da função que descreve o modelo φ_k , do centro de estabilidade corrente x^k e da norma primal. Escolhas apropriadas serão apresentadas mais adiante.

Note que no Passo 3 temos um teste similar à [Condição de Armijo](#), e que é muito parecido com a ideia de métodos de região de confiança (ver [22]), pois estamos verificando se o houve um decréscimo da f que seja ao menos uma fração α da redução predita pelo modelo. Na próxima seção iremos apresentar uma variante que usa a filosofia de região de confiança para resolver o subproblema (2.7). Outro ponto que merece destaque é que, como iremos atualizar o centro de estabilidade apenas quando essa condição é satisfeita, é possível que tenhamos redução simples $f(y^{k+1}) < f(x^k)$, mas o decréscimo não satisfaça o teste do Passo 3, baseado no decréscimo nominal. Por esta razão é que dizemos que o centro de estabilidade é o ponto que obteve o *melhor* valor objetivo até o momento, e não o *menor*.

Algoritmo 3 – MÉTODO DE FEIXES**Entrada:** $y^1 \in \mathbb{X}$, $\epsilon_{\text{tol}} \geq 0$, e $\alpha \in (0, 1)$.

Inicializa os parâmetros:

 $\delta_1 = \infty$, $|\cdot|_k$, $itmax \in \mathbb{N}$, $x^1 = y^1$, $M_1 \in \mathbb{R}^{n \times n}$ positiva definida e $k = 1$ Envia y^1 para o oráculo.**repita****Passo 1: Receber informação do oráculo e verificar critério de parada**Constrói modelo φ_k **se** $\delta_k < \epsilon_{\text{tol}}$ **então**| $x^* = x^k$ **fim****Passo 2: Determinar um novo candidato**Resolve o subproblema estabilizado (*spe*):

$$y^{k+1} = \operatorname{argmin}\{spe(\varphi_k, x^k, |\cdot|_k)\} \quad (2.7)$$

Define:

$$\delta_{k+1} = f(y^{k+1}) - \varphi_k(y^{k+1})$$

Passo 3: Efetuar o teste de descidaEnvia y^{k+1} para o oráculo**se** $f(x^k) - f(y^{k+1}) \geq \alpha \delta_{k+1}$ **então**| $x^{k+1} := y^{k+1}$

#Passo Sério

senão| $x^{k+1} := x^k$

#Passo Nulo

fim**Passo 4: Incrementar o modelo**Inclui y^{k+1} no modelo

Atualiza parâmetros para a próxima iteração

Define M_{k+1} $k \leftarrow k + 1$ Volte ao **Passo 1**.**até** $\delta_k < \epsilon_{\text{tol}}$ ou $k \geq itmax$;**Saída:** x^*

Vamos denotar por K_s o conjunto de índices k associados aos passos sérios. Em particular, quando temos infinitos passos sérios, a sequência $\{\delta_k\}_{k \in K_s}$ é convergente sempre que (2.1) tiver minimizadores.

Lema 1 (Boa definição do Algoritmo 3). *Considere o Algoritmo 3 e suponha que ele execute infinitas iterações. Seja $f_* := \lim_{k \in K_s} f(x^k)$ e assumamos que $f_* > -\infty$, então*

$$0 \leq \sum_{k \in K_s} \delta_k \leq \frac{f_1 - f_*}{\alpha}.$$

Demonstração. Veja que como a tolerância de parada ϵ_{tol} é não negativa, para que o algoritmo faça infinitas iterações, o decréscimo nominal precisa ser sempre maior que zero, ou seja, $\delta_k > 0$ para todo $k \in K_s$. Tomando um índice arbitrário $k \in K_s$, e como em K_s estão indexados todos os passos sérios, então $x^{k+1} = y^{k+1}$ e

$$f(x^k) - f(x^{k+1}) = f(x^k) - f(y^{k+1}) \geq \alpha \delta_{k+1}.$$

Agora tomemos o índice subsequente a k em K_s , vamos denotá-lo por k' . Então, entre k e k' o algoritmo produz apenas passos nulos, mantendo assim seu centro de estabilidade, ou seja, $x^{k+1} = x^{k+j}$ para todo $j = 2, \dots, k' - k$. O teste de descida em $k' - 1$ nos fornece

$$f(x^k) - f(x^{k'}) \geq \alpha \delta_{k'}, \quad (2.8)$$

o que nos permite definir o novo centro de estabilidade $x^{k''}$. Para um índice qualquer $k'' \in K_s$, temos

$$\begin{aligned} \alpha \sum_{\substack{k_i \in K_s \\ i \in \mathbb{N}}}^{k_i = k''} \delta_{k_i} &\stackrel{(2.8)}{\leq} \sum_{\substack{k_i \in K_s \\ i \in \mathbb{N}}}^{k_i = k''} (f(x^{k_{i-1}}) - f(x^{k_i})) \\ &= f_1 - f_{k''} \\ &\leq f_1 - f_*, \end{aligned}$$

em que a igualdade vem da [soma telescópica](#). Então dividindo por α ambos os lados da desigualdade vem

$$\sum_{\substack{k_i \in K_s \\ i \in \mathbb{N}}}^{k_i = k''} \delta_{k_i} \leq \frac{f_1 - f_*}{\alpha}.$$

Fazendo $k'' \rightarrow \infty$ concluímos a demonstração. \square

Este resultado nos garante a convergência do método sempre que o algoritmo gera uma sequência infinita de passos sérios. Isso porque a sequência de valores funcionais $\{f(x^k)\}_{k \in K_s}$ é estritamente decrescente e então ou $\{f(x^k)\} \rightarrow -\infty$, quando f é ilimitada inferiormente e $\{x^k\}$ é trivialmente uma sequência minimizante; ou $\{f(x^k)\} \rightarrow f_*$ e, principalmente, $\{\delta_k\} \rightarrow 0$; essa relação é a chave para mostrarmos adiante a convergência do método nos itens (ii) e (iii) do Lema 2. Por fim, note que se $\epsilon_{\text{tol}} > 0$, sob a hipótese de limitação para f no Lema 1, então o Algoritmo 3 fará um número finito de iterações. Na Seção 2.3.4 iremos nos apoiar neste resultado para a análise de convergência.

2.3.2 Algumas variantes do Método de Feixes

Iremos apresentar três variantes do método de feixes: Níveis, Região de Confiança e Proximal. Qualquer uma dessas variantes produzirá o subproblema estabilizado (2.7) mencionado no Passo 2 do Algoritmo 3.

A variante via Região de Confiança, como o próprio nome já diz, se baseia no método de região de confiança (ver [22] para mais detalhes sobre esse método em específico) e estamos interessados em minimizar o modelo φ_k sujeito a uma região em que se confia que o modelo representa bem a função objetivo. Essa região vai estar centrada no centro de estabilidade x^k e terá um raio κ_k pré-definido. Matematicamente, podemos escrever o subproblema da seguinte maneira:

$$\begin{aligned} \min \quad & \varphi_k(y) \\ \text{s.a} \quad & |y - x^k|_k^2 \leq \kappa_k. \end{aligned} \quad (2.9)$$

O decréscimo nominal é definido como $\delta_{k+1} := f(x^k) - \varphi_k(y^{k+1})$. Um dos critérios de parada nessa variante é o tamanho do raio da região de confiança, κ_k , pois conforme $k \rightarrow \infty$ temos que $\kappa_k \rightarrow 0$.

A variante que estamos interessados em explorar neste trabalho é a variante com níveis, na qual queremos um ponto que minimiza a distância ao centro de estabilidade, com a restrição de que o valor funcional do modelo nesse ponto não ultrapasse o valor pré-fixado ℓ_k . Aqui podemos estabelecer um paralelo com a variante de região de confiança. Enquanto na região de confiança estamos minimizando o modelo sujeito a uma restrição no domínio, nesta variante vamos minimizar a norma da diferença entre o ponto e o centro de estabilidade sujeito a uma restrição na imagem do modelo. Matematicamente, neste caso podemos escrever o subproblema da seguinte maneira:

$$\begin{aligned} \min \quad & \frac{1}{2}|y - x^k|_k^2 \\ \text{s.a} \quad & \varphi_k(y) \leq \ell_k. \end{aligned} \quad (2.10)$$

E assim como na variante anterior, o decréscimo nominal também é definido como $\delta_{k+1} := f(x^k) - \varphi_k(y^{k+1})$. Iremos explorar mais características dessa variante na próxima seção.

Por último, temos a variante Proximal, em que adicionamos uma parcela positiva ao modelo φ_k , relacionada com a distância do ponto até o centro de estabilidade. Quanto mais longe do centro de estabilidade, maior é o valor dessa parcela, penalizando assim o valor da função objetivo. Isso faz com que o método dê preferência para os pontos próximos ao centro de estabilidade x^k . Matematicamente, escrevemos o subproblema da seguinte maneira:

$$\min_{y \in \mathbb{R}^n} \varphi_k(y) + \frac{1}{2}\mu_k|y - x^k|_k^2 \quad (2.11)$$

em que $\mu_k > 0$ é uma fator de regularização responsável por intensificar o valor da penalização. Essa variante tem duas características: a primeira, e melhor delas, é de que temos um subproblema irrestrito, e a segunda é um pouco menos restritiva com relação ao decréscimo nominal, que é definido como

$$\delta_{k+1} := f(x^k) - \left(\varphi_k(y^{k+1}) + \frac{1}{2} \mu_k |y^{k+1} - x^k|_k^2 \right). \quad (2.12)$$

Note que podemos rearranjar os termos (2.12) da seguinte maneira:

$$\begin{aligned} \delta_{k+1} &= \underbrace{f(x^k) - \varphi_k(y^{k+1})}_{\delta_{k+1}^{restrito}} - \left(\frac{1}{2} \mu_k |y^{k+1} - x^k|_k^2 \right) \\ &= \delta_{k+1}^{restrito} - \left(\frac{1}{2} \mu_k |y^{k+1} - x^k|_k^2 \right), \end{aligned} \quad (2.13)$$

em que $\delta_{k+1}^{restrito}$ é o decréscimo nominal dos problemas (2.9) e (2.10). Logo o decréscimo nominal da variante proximal é menor que o decréscimo nominal das variantes via região de confiança e via níveis.

Vale ressaltar que existem outras variantes do método de feixes, inclusive uma que combina a variante Proximal com a variante em Níveis, chamada de *método de feixes duplamente estabilizado*⁴, em que o algoritmo escolhe automaticamente qual método usará para definir o próximo iterado. Dessa forma, é possível combinar a simplicidade de atualização do parâmetro de nível para produzir uma regra simples e eficiente para atualizar o parâmetro proximal, acelerando assim o processo de otimização. Para mais detalhes ver [10] e [6].

O próximo resultado, é muito importante, pois estabelece formalmente a equivalência das três variantes do método que foram apresentadas anteriormente.

Teorema 1 (Equivalência dos subproblemas quadráticos). *Considere os problemas (2.9), (2.11) e (2.10), com a mesma norma $|\cdot|_k$. As três condições abaixo são satisfeitas:*

- (i) *Dado $\kappa_k > 0$, seja y^{k+1} solução do problema (2.9). Então existe $\mu_k > 0$ tal que y^{k+1} também resolve o problema (2.11).*
- (ii) *Dado $\mu_k > 0$, seja y^{k+1} solução do problema (2.11). Então existe $\ell_k \in \mathbb{R}$ tal que y^{k+1} também resolve o problema (2.10).*
- (iii) *Dado $\ell_k \in \mathbb{R}$, seja y^{k+1} solução do problema (2.10). Então existe $\kappa_k \geq 0$ tal que y^{k+1} também resolve o problema (2.9).*

⁴ Em inglês: *doubly stabilized bundle method*

Demonstração. A demonstração abaixo é uma versão um pouco mais detalhada da demonstração encontrada em [5, Sec. 10.3].

- (i) Seja y^{k+1} solução do problema (2.9), que é convexo e satisfaz a [condição de Slater](#) [4, Proposição 5.3.1], pois $\kappa_k > 0$ e x^k é interior à bola centrada em x^k , de raio $\sqrt{\kappa_k}$. Portanto, existe um multiplicador associado $\lambda \geq 0$, $\lambda \in \mathbb{R}$, para o qual a solução para (2.11) é obtida minimizando a Lagrangiana, ou seja, $y^{k+1} = \operatorname{argmin}_{y \in \mathbb{R}^n} \mathcal{L}(y, \lambda)$ em que

$$\mathcal{L}(y, \lambda) := \varphi_k(y) + \frac{1}{2}\lambda (|y - x^k|_k^2 - \kappa_k).$$

Daí temos que y^{k+1} resolve o problema (2.11) com $\mu_k := \lambda$.

- (ii) Seja y^{k+1} solução do problema (2.11). Para todo $y \in \mathbb{R}^n$ temos

$$\varphi_k(y^{k+1}) + \frac{1}{2}\mu_k |y^{k+1} - x^k|_k^2 \leq \varphi_k(y) + \frac{1}{2}\mu_k |y - x^k|_k^2.$$

Em particular, se $\ell_k := \varphi_k(y^{k+1})$, então para todo y tal que $\varphi_k(y) \leq \ell_k$, temos

$$\frac{1}{2}\mu_k |y^{k+1} - x^k|_k^2 \leq \frac{1}{2}\mu_k |y - x^k|_k^2.$$

Como μ_k é sempre positivo, então temos que y^{k+1} resolve o problema (2.10) com $\ell_k := \varphi_k(y^{k+1})$.

- (iii) Seja y^{k+1} solução do problema (2.10). Como o conjunto viável de (2.10) é poliedral e a função objetivo é quadrática convexa, se $\varphi_k(y^{k+1}) < \ell_k$ então $y^{k+1} = x^k$ e o resultado vale com $\kappa_k = 0$. Suponhamos então que $\varphi_k(y^{k+1}) = \ell_k$ e $\varphi_k(x^k) \geq \ell_k$, ou seja, o centro de estabilidade não é estritamente interior ao conjunto viável. Dessa forma, tomando $\kappa_k := |y^{k+1} - x^k|_k^2$, segue que y^{k+1} resolve o problema (2.9), pois y^{k+1} minimiza $\varphi_k(y)$ no conjunto $\{y \in \mathbb{R}^n \mid |y - x^k|_k^2 \leq \kappa_k\}$.

□

Tendo em vista a equivalência estabelecida no Teorema 1 entre as três variantes que apresentamos, assim como em [5] vamos utilizar a variante proximal do método de feixes para apresentar a seguir um resultado que irá nos auxiliar a caracterizar a solução do subproblema quadrático e estabelecer algumas relações fundamentais entre as medidas nas normas primal e dual:

Lema 2 (Caracterização da solução do subproblema proximal). *Sejam y^{k+1} a solução única do problema (2.11) e $\mu_k > 0$. Então*

$$y^{k+1} = x^k - \frac{1}{\mu_k} M_k^{-1} \hat{s}^k \quad \text{em que} \quad \hat{s}^k := \sum_{i=1}^{m_k} \bar{\gamma}_i s^i, \quad (2.14)$$

e $\bar{\gamma} \in \mathbb{R}_+^{m_k}$ é a solução de

$$\begin{aligned} \min_{\gamma \in [0, 1]^{m_k}} \quad & \frac{1}{2} \left\| \sum_{i=1}^{m_k} \gamma_i s^i \right\|_k^2 + \mu_k \sum_{i=1}^{m_k} \gamma_i \xi_i \\ \text{s.a} \quad & \sum_{i=1}^{m_k} \gamma_i = 1, \quad \gamma \geq 0. \end{aligned} \quad (2.15)$$

Além disso, as seguintes relações são satisfeitas:

(i) $\hat{s}^k \in \partial \varphi_k(y^{k+1})$;

(ii) $\delta_{k+1} = \varepsilon_k + \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2$, em que definimos $\varepsilon_k := \sum_{i=1}^{m_k} \bar{\gamma}_i \xi_i$ e

(iii) $\hat{s}^k \in \partial_{\varepsilon_k} f(x^k)$.

Demonstração. Reescrevendo o problema (2.11) com uma variável extra

$$\begin{aligned} \min_{(y, r) \in \mathbb{R}^n \times \mathbb{R}} \quad & r + \frac{1}{2} \mu_k |y - x^k|_k^2 \\ \text{s.a} \quad & r \geq f(x^k) - \xi_i + \langle s^i, (y - x^k) \rangle, \quad i = 1, \dots, k, \end{aligned} \quad (2.16)$$

a Lagrangiana correspondente, para $\gamma \in \mathbb{R}_+^{m_k}$, será

$$\begin{aligned} \mathcal{L}(y, r, \gamma) &= r + \frac{1}{2} \mu_k |y - x^k|_k^2 + \sum_{i=1}^{m_k} \gamma_i (f(x^k) - \xi_i + \langle s^i, (y - x^k) \rangle - r) \\ &= \left(1 - \sum_{i=1}^{m_k} \gamma_i \right) r + \frac{1}{2} \mu_k |y - x^k|_k^2 + \sum_{i=1}^{m_k} \gamma_i (f(x^k) - \xi_i + \langle s^i, (y - x^k) \rangle). \end{aligned}$$

Note que como o problema (2.11) possui função objetivo fortemente convexa, possui uma única solução y^{k+1} . O problema equivalente (2.16) tem restrições lineares, então existe um multiplicador ótimo $\bar{\gamma}$ associado à solução y^{k+1} . E como ambos são problemas convexos, não há folga de dualidade, logo $(y^{k+1}, \bar{\gamma})$ pode ser obtido tanto pelo problema primal (2.16), que também pode ser expresso como

$$\min_{(y, r) \in \mathbb{R}^n \times \mathbb{R}} \max_{\gamma \in \mathbb{R}_+^{m_k}} \mathcal{L}(y, r, \gamma), \quad (2.17)$$

quanto por seu problema dual

$$\max_{\gamma \in \mathbb{R}_+^{m_k}} \min_{(y, r) \in \mathbb{R}^n \times \mathbb{R}} \mathcal{L}(y, r, \gamma). \quad (2.18)$$

Tanto o problema (2.17) quanto (2.18) têm o mesmo valor ótimo, no entanto, o problema (2.18) envolve a minimização irrestrita de \mathcal{L} com respeito a r . Perceba que para que o valor ótimo dual seja finito, o termo r precisa ser eliminado, isso implica que $\sum_{i=1}^{m_k} \gamma_i = 1$.

Em outras palavras, γ pertence ao **simplex** unitário $\Delta_k := \left\{ u \in [0, 1]^{m_k} \mid \sum_{i=1}^{m_k} u_i = 1 \right\}$.

Com isso, podemos reescrever os problemas (2.17) e (2.18), respectivamente, como:

$$\min_{y \in \mathbb{R}^n} \max_{\gamma \in \mathbb{R}_+^{m_k}} \mathcal{L}(y, \gamma) \equiv \max_{\gamma \in \mathbb{R}_+^{m_k}} \min_{y \in \mathbb{R}^n} \mathcal{L}(y, \gamma), \quad (2.19)$$

em que $\mathcal{L}(y, \gamma) := f(x^k) + \frac{1}{2} \mu_k |y - x^k|_k^2 + \sum_{i=1}^{m_k} \gamma_i (\xi_i + \langle s^i, (y - x^k) \rangle)$.

Note que para cada $\gamma \in \Delta_k$ fixo, podemos definir $y(\gamma) := \underset{y}{\operatorname{argmin}} \mathcal{L}(y, \gamma)$ e o minimizador y satisfaz $\nabla_y \mathcal{L}(\gamma, y(\gamma)) = 0_{\mathbb{R}^n}$, ou seja,

$$0_{\mathbb{R}^n} = \mu_k M_k (y(\gamma) - x^k) + \sum_{i=1}^{m_k} \gamma_i s^i. \quad (2.20)$$

Em particular, quando $\gamma = \bar{\gamma}$ temos que $y(\bar{\gamma}) = y^{k+1}$, então (2.14) é satisfeita.

Perceba que a partir da caracterização de y^{k+1} dada em (2.14), conseguimos escrever \hat{s}^k da seguinte maneira:

$$\hat{s}^k = -\mu_k M_k (y^{k+1} - x^k). \quad (2.21)$$

Agora, vamos expandir o quadrado da norma dual da segunda parcela em (2.20):

$$\begin{aligned} \left\| \sum_{i=1}^{m_k} \gamma_i s^i \right\|_k^2 &= \left\langle \sum_{i=1}^{m_k} \gamma_i s^i, M_k^{-1} \sum_{i=1}^{m_k} \gamma_i s^i \right\rangle \\ &\stackrel{(2.14)}{=} \langle \hat{s}^k, M_k^{-1} \hat{s}^k \rangle \\ &\stackrel{(2.21)}{=} \langle -\mu_k M_k (y^{k+1} - x^k), M_k^{-1} (-\mu_k M_k (y^{k+1} - x^k)) \rangle \\ &= \mu_k^2 \langle M_k (y^{k+1} - x^k), (y^{k+1} - x^k) \rangle \\ &= \mu_k^2 |y^{k+1} - x^k|_k^2. \end{aligned}$$

Daí, dividindo por μ_k de ambos os lados temos:

$$\frac{1}{\mu_k} \left\| \sum_{i=1}^{m_k} \gamma_i s^i \right\|_k^2 = \mu_k |y^{k+1} - x^k|_k^2, \quad (2.22)$$

e então

$$\mathcal{L}(y(\gamma), \gamma) = f(x^k) - \frac{1}{2\mu_k} \left\| \sum_{i=1}^{m_k} \gamma_i s^i \right\|_k^2 - \sum_{i=1}^{m_k} \gamma_i \xi_i.$$

Como $f(x^k)$ é fixo,

$$\max_{\gamma \in [0, 1]^{m_k}} \mathcal{L}(y(\gamma), \gamma) = f(x^k) - \min_{\gamma \in [0, 1]^{m_k}} \left\{ \frac{1}{2\mu_k} \left\| \sum_{i=1}^{m_k} \gamma_i s^i \right\|_k^2 - \sum_{i=1}^{m_k} \gamma_i \xi_i \right\}.$$

Com isso, além de verificarmos que $\bar{\gamma}$ é a solução do problema (2.15), ainda podemos mostrar as três relações adicionais:

- (i) Escrevendo a condição de otimalidade de (2.11) e usando a definição de \hat{s}_k e a sua expressão em (2.21) vem:

$$0 \in \partial\varphi_k(y^{k+1}) + \mu_k M_k(y^{k+1} - x^k) = \partial\varphi_k(y^{k+1}) - \sum_{i=1}^{m_k} \bar{\gamma}_i s^i = \partial\varphi_k(y^{k+1}) - \hat{s}^k.$$

Logo $\hat{s}^k \in \partial\varphi_k(y^{k+1})$.

- (ii) Como não temos folga de dualidade, o valor ótimo primal e dual são iguais, ou seja

$$\varphi_k(y^{k+1}) + \frac{1}{2}\mu_k \|y^{k+1} - x^k\|_k^2 = f(x^k) - \frac{1}{2\mu_k} \left\| \sum_{i=1}^{m_k} \bar{\gamma}_i s^i \right\|_k^2 - \sum_{i=1}^{m_k} \bar{\gamma}_i \xi_i,$$

subtraindo $f(x^k)$ e multiplicando por (-1) de ambos os lados, obtemos:

$$\underbrace{f(x^k) - \left(\varphi_k(y^{k+1}) + \frac{1}{2}\mu_k \|y^{k+1} - x^k\|_k^2 \right)}_{\delta_{k+1}} = \frac{1}{2\mu_k} \underbrace{\left\| \sum_{i=1}^{m_k} \bar{\gamma}_i s^i \right\|_k^2}_{\hat{s}^k} + \underbrace{\sum_{i=1}^{m_k} \bar{\gamma}_i \xi_i}_{\varepsilon_k}.$$

Note que o lado esquerdo é exatamente o decréscimo nominal definido em (2.12), daí temos

$$\delta_{k+1} = \varepsilon_k + \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2.$$

- (iii) Usando o fato de que $f \geq \varphi_k$ e o item (i), temos que para todo $y \in \mathbb{R}^n$

$$f(y) \geq \varphi_k(y) \geq \varphi_k(y) + \langle \hat{s}^k, (y - y^{k+1}) \rangle,$$

daí temos

$$\begin{aligned} f(y) &\geq \varphi_k(y) + \langle \hat{s}^k, (y - y^{k+1}) \rangle \\ &= \varphi_k(y) + \langle \hat{s}^k, (y \pm x^k - y^{k+1}) \rangle \\ &= \varphi_k(y) + \langle \hat{s}^k, (y - x^k) \rangle - \langle \hat{s}^k, (y^{k+1} - x^k) \rangle \\ &\stackrel{(2.49)}{=} \varphi_k(y) + \langle \hat{s}^k, (y - x^k) \rangle - \langle \hat{s}^k, \frac{1}{\mu_k} M_k^{-1} \hat{s}^k \rangle \\ &= \varphi_k(y) + \langle \hat{s}^k, (y - x^k) \rangle + \frac{1}{\mu_k} \|\hat{s}^k\|_k^2 \\ &= f(x^k) + \langle \hat{s}^k, (y - x^k) \rangle - \left(f(x^k) - \varphi_k(y) - \frac{1}{\mu_k} \|\hat{s}^k\|_k^2 \right). \end{aligned}$$

Utilizando a relação do item (ii) temos

$$f(y) \geq f(x^k) + \langle \hat{s}^k, (y - x^k) \rangle - \varepsilon_k.$$

Logo, pela Definição 5, concluímos que $\hat{s}^k \in \partial_{\varepsilon_k} f(x^k)$. \square

Veja que, assim como no métodos de planos cortantes, a cada iteração estamos adicionando uma restrição ao subproblema e, naturalmente, muitas dessas restrições vão se tornando redundantes, ou seja, não alteram em nada a solução do modelo quadrático. E isso encarece a resolução do problema quadrático, sem contar que estamos gastando memória armazenando todas essas informações. Pensando nisso, nesta seção vamos apresentar estratégias para contornar essa dificuldade e dessa forma aperfeiçoar o funcionamento do Algoritmo 3.

2.3.3 Compressão do Feixe

Conforme o número de iterações vai aumentando, o tamanho do feixe \mathcal{B} também vai aumentando, então para controlarmos o tamanho do feixe iremos introduzir um mecanismo de *seleção e compressão* no Algoritmo 3.

Suponha que, na iteração k , o feixe possua m_k \mathcal{B} -pares da forma (s^i, ξ_i) , com $s^i \in \partial_{\xi_i} f(x^k)$, que definem o modelo φ_k . Como vimos no Lema 2, conseguimos caracterizar a solução do subproblema utilizando o vetor \hat{s}_k , que é uma combinação convexa de todos os subgradientes do feixe, então perceba que se algum dos coeficientes dessa combinação for nulo, ou seja $\bar{\gamma}_i = 0$, então o subgradiente e o erro ξ_i correspondentes não são necessários para caracterização da solução. Por isso, os subgradientes correspondentes aos coeficientes $\bar{\gamma}_i = 0$ são ditos *dispensáveis* enquanto os subgradientes correspondentes aos coeficientes $\bar{\gamma}_i \neq 0$ são ditos *indispensáveis*.

Quando a *cardinalidade* de \mathcal{B} atingir um número muito grande e pré-estabelecido m_{max} , vamos aplicar as seguintes técnicas:

1. *seleção* dos \mathcal{B} -pares dispensáveis, que poderão ser excluídos do feixe;
2. caso ainda restem muitos \mathcal{B} -pares indispensáveis, iremos fazer uma *compressão* desses \mathcal{B} -pares num único \mathcal{B} -par, denominado *\mathcal{B} -par agregado*.

A agregação é uma técnica que vai nos permitir condensar todas as informações essenciais do feixe em um único \mathcal{B} -par, $(\hat{s}^k, \varepsilon_k)$ (em que \hat{s}^k e ε_k foram definidos no Lema 2). No lugar de construirmos uma estabilização a partir dos planos cortantes, \check{f}_k ,

quando fizermos a compressão iremos utilizar um modelo linear, denominado *linearização agregadora*, que é definido da seguinte maneira

$$f_a(y) := f(x^k) - \varepsilon_k + \langle \hat{s}^k, y - x^k \rangle. \quad (2.23)$$

Note que a função acima tem a mesma estrutura da função (2.4), porém ela condensa todas as informações armazenadas no feixe até a iteração k , com a única exceção de que nesse modelo não há um elemento y^i tal que $\hat{s}^k \in \partial f(y^i)$.

O resultado seguinte vai nos garantir algumas propriedades importantes que são satisfeitas pela linearização agregadora e conseqüentemente vai nos permitir fazer a limpeza do feixe \mathcal{B} .

Lema 3 (Propriedades da Linearização Agregadora). *Dada a linearização agregadora definida em (2.23), as seguintes relações são satisfeitas:*

(i) $f_a(y) = \varphi_k(y^{k+1}) + \langle \hat{s}^k, y - y^{k+1} \rangle$, para todo $y \in \mathbb{R}^n$.

(ii) $f_a(y) \leq \varphi_k(y)$, para todo $y \in \mathbb{R}^n$.

(iii) Seja $\psi : \mathbb{R}^n \rightarrow \mathbb{R}$ uma função convexa tal que $\psi(y) \geq f_a(y)$, para todo $y \in \mathbb{R}^n$ e $\psi(y^{k+1}) = f_a(y^{k+1}) = \varphi_k(y^{k+1})$. Então

$$y^{k+1} = \underset{y \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \psi(y) + \frac{1}{2} \mu_k |y - x^k|_k^2 \right\}. \quad (2.24)$$

Demonstração. A demonstração sai quase que imediatamente da definição (2.23).

(i) Vamos reescrever e rearranjar os termos da seguinte expressão:

$$\begin{aligned} f(x^k) - \varepsilon_k + \langle \hat{s}^k, y^{k+1} - x^k \rangle &\stackrel{(2.14)}{=} f(x^k) - \varepsilon_k + \langle \hat{s}^k, -\frac{1}{\mu_k} M_k^{-1} \hat{s}^k \rangle \\ &\stackrel{\text{Lema 2 (ii)}}{=} f(x^k) - \delta_{k+1} + \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2 - \frac{1}{\mu_k} \|\hat{s}^k\|_k^2 \\ &= f(x^k) - \delta_{k+1} - \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2 \\ &\stackrel{(2.12)}{=} \varphi_k(y^{k+1}) + \underbrace{\frac{\mu_k}{2} |y^{k+1} - x^k|_k^2 - \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2}_{\stackrel{(2.22)}{=} 0} \\ &= \varphi_k(y^{k+1}). \end{aligned}$$

Somando $\langle \hat{s}^k, y - y^{k+1} \rangle$ de ambos os lados obtemos:

$$\begin{aligned} \varphi_k(y^{k+1}) + \langle \hat{s}^k, y - y^{k+1} \rangle &= f(x^k) - \varepsilon_k + \langle \hat{s}^k, y^{k+1} - x^k \rangle + \langle \hat{s}^k, y - y^{k+1} \rangle \\ &= f(x^k) - \varepsilon_k + \langle \hat{s}^k, y^{k+1} - x^k + y - y^{k+1} \rangle \\ &= f(x^k) - \varepsilon_k + \langle \hat{s}^k, y - x^k \rangle \\ &\stackrel{(2.23)}{=} f_a(y). \end{aligned}$$

(ii) Da relação (i) do Lema 2, temos que $\hat{s}^k \in \partial\varphi_k(y^{k+1})$, isso significa que

$$\varphi_k(y) \geq \varphi_k(y^{k+1}) + \langle \hat{s}^k, y - y^{k+1} \rangle, \forall y \in \mathbb{R}^n.$$

Mas pelo item (i) que acabamos de provar, o lado direito da desigualdade acima é justamente $f_a(y)$. Logo

$$f_a(y) \leq \varphi_k(y), \forall y \in \mathbb{R}^n.$$

(iii) Por último, temos que a condição de otimalidade do problema (2.24) é que 0 pertença ao subdiferencial da função em y^{k+1} , ou seja

$$0 \in \partial\psi(y^{k+1}) + \mu_k M_k(y^{k+1} - x^k) \stackrel{(2.21)}{=} \partial\psi(y^{k+1}) - \hat{s}^k.$$

□

Esse resultado nos mostra que podemos fazer a redução do feixe sintetizando todas as informações indispensáveis com a linearização agregadora. Note que poderíamos também escrever o nosso modelo de uma forma desagregada, ou seja, da seguinte maneira:

$$\psi(y) = f(x^k) + \max_{i: \bar{\gamma}_i > 0} \{-\xi_i + \langle s^i, (y - x^k) \rangle\}.$$

Além disso, do Lema 3, podemos perceber que f_a é **minimal** na classe de funções que estão delimitadas por f_a e φ_k e que deixam y^{k+1} invariante.

É importante compreendermos com afinco o papel que cada um desses elementos têm na compressão do feixe. O primeiro deles é o valor m_k , que é a cardinalidade do feixe na iteração k , que iremos comparar com um parâmetro m_{max} , que será o máximo de elementos que vamos permitir no feixe \mathcal{B} . No início do algoritmo, ou seja, antes que $m_k \geq m_{max}$ pela primeira vez, o algoritmo é exatamente igual ao Algoritmo 3, já que o nosso modelo $\varphi_k = \check{f}$ e $m_k = k$. No entanto, devemos enfatizar que a necessidade de substituir os \mathcal{B} -pares pelo \mathcal{B} -par agregado vai depender de quantos \mathcal{B} -pares foram descartados na etapa de seleção. Veja que nós poderíamos comprimir os \mathcal{B} -pares restantes, independentemente de sua quantidade, porém não estaríamos ganhando nada no sentido de informação, já que todas as informações indispensáveis já estariam sendo utilizadas de forma desagregada, então nesse sentido vamos fazer a compressão apenas quando a cardinalidade dos \mathcal{B} -pares

indispensáveis também for maior que m_{max} . Em outras palavras, esse número é diretamente relacionado com a necessidade de realizarmos ou não a compressão do feixe. E além disso, m_{max} vai determinar a dimensão de cada subproblema (2.15), ou seja, a dimensão da variável γ .

O segundo aspecto muito importante que devemos considerar é quais os \mathcal{B} -pares que serão selecionados, isto é, quais deles iremos descartar. A seleção que apresentamos até o momento sempre descarta os \mathcal{B} -pares dispensáveis, ou seja, os que estão associados com $\gamma_i = 0$. Porém existem diversas formas de fazer essa seleção, por exemplo, poderíamos descartar aqueles que tem os menores ξ_i , ou os mais antigos, ou ainda uma combinação dessas opções. Contanto que o \mathcal{B} -par agregado seja inserido toda vez que um \mathcal{B} -par indispensável for descartado, o algoritmo vai continuar convergindo. Obviamente, diferentes formas de seleção vão ter diferentes impactos práticos na convergência.

O terceiro e último aspecto a ser considerado é o modelo. Suponha que tenhamos descartado os p primeiros \mathcal{B} -pares, então o modelo seguinte será

$$\begin{aligned} \varphi_{k+1}(y) &= \max \left\{ \max_{p \leq i \leq k+1} \{f_i + \langle s^i, y - y^i \rangle\}, \varphi_k(y^{k+1}) + \langle \hat{s}^k, y - y^{k+1} \rangle \right\} \\ &= \max \left\{ f(x^{k+1}) + \max_{p \leq i \leq k+1} \{-\xi_i + \langle s^i, y - x^{k+1} \rangle\}, f_a(y) \right\}. \end{aligned}$$

Note que pelo Lema 3, a última função linear pode ser substituída por qualquer função ψ , tal que $\psi \geq f_a$. E veja que neste caso, por construção, para todo k e para todo $y \in \mathbb{R}^n$ temos

$$f_a(y) \leq \varphi_{k+1}(y) \leq f(y) \quad \text{e} \quad \varphi_{k+1}(y) \geq f_{k+1} + \langle s^{k+1}, y - y^{k+1} \rangle. \quad (2.25)$$

Com isso em mãos, já podemos analisar a convergência do método e posteriormente apresentar o algoritmo da variante de Feixes em Níveis, bem como suas propriedades. Na Seção 2.3.5 apresentaremos tanto a versão com o mecanismo de compressão quanto sem, e então todas essas características ficarão mais claras.

2.3.4 Análise de Convergência da versão Proximal

Como vimos anteriormente no Lema 1, quando $\{\delta_k\} \rightarrow 0$ se $\epsilon_{\text{tol}} > 0$ então o Algoritmo 3 executa um número finito de iterações, ou seja, ele vai parar em alguma iteração k_{last} . Quando $\delta_{k_{last}} \leq \epsilon_{\text{tol}}$, pelo item (ii) do Lema 2, tanto $\epsilon_{k_{last}}$ quanto $\frac{\|\hat{s}^{k_{last}}\|_{k_{last}}}{\mu_{k_{last}}}$ são pequenos, logo pelo (iii) do Lema 2, o último passo sério antes de parar satisfaz a seguinte relação de otimalidade aproximada

$$f(y) \geq f(x^{k_{last}}) - \|\hat{s}^{k_{last}}\|_{k_{last}} |y - x^{k_{last}}|_{k_{last}} - \epsilon_{k_{last}}, \quad \forall y \in \mathbb{R}^n. \quad (2.26)$$

Da relação do ε -subdiferencial, como $\hat{s}^k \in \partial_{\varepsilon_k} f(x^k)$ temos

$$f(y) \geq f(x^k) + \langle \hat{s}^k, y - x^k \rangle - \varepsilon_k, \quad \forall y \in \mathbb{R}^n.$$

Na relação (2.26) o produto interno acima foi cotado por $\langle \hat{s}^k, y - x^k \rangle \geq \|\hat{s}^k\|_k |y - x^k|_k$.

Quando $\epsilon_{\text{tol}} = 0$, ou o algoritmo para por ter encontrado uma solução, com a inclusão $0 \in \partial f(x^{k_{\text{last}}})$ válida, ou itera infinitas vezes. Com base no resultado de equivalência entre os subproblemas estabelecido no Teorema 1, a análise da convergência global dessas duas possibilidades será feita com base na variante proximal com agregação, com as seguintes características:

- As matrizes M_k são múltiplos escalares de uma matriz simétrica definida positiva M escolhida no passo inicial: $M_k = \eta_k M$ com η_k variando apenas para $k \in K_s$ (índices associados aos passos sérios). Dessa forma, as normas primal e dual estão limitadas pela norma Euclidiana usando λ e Λ autovalores extremos de M :

$$\eta_k \lambda \|\cdot\|^2 \leq |\cdot|_k^2 \leq \eta_k \Lambda \|\cdot\|^2 \quad \text{e} \quad \frac{1}{\eta_k \Lambda} \|\cdot\|^2 \leq \|\cdot\|_k^2 \leq \frac{1}{\eta_k \lambda} \|\cdot\|^2. \quad (2.27)$$

- A solução y^{k+1} do problema regularizado (2.11) é computada via (2.14) pelo dual (2.15), com decréscimo nominal δ_{k+1} dado por (2.12) e ε_k como no item (ii) do Lema 2.
- O modelo φ_k de (2.4) é construído de acordo com as técnicas de compressão.
- A tolerância de parada é $\epsilon_{\text{tol}} = 0$.

Teorema 2 (Convergência dos Passos Sérios). *Assumindo que o Algoritmo 3 com compressão gera uma sequência infinita de passos sérios x^k , $k \in K_s$, então ou o conjunto solução do problema é vazio e $\{f(x^k)\} \rightarrow -\infty$ ou vale o seguinte:*

(i) $\{\delta_k\}_{k \in K_s} \rightarrow 0$ e $\{\varepsilon_k\}_{k \in K_s} \rightarrow 0$

(ii) Se para todo $k \in K_s$ temos

$$\eta_{k+1} \leq \eta_k \quad \text{e} \quad \sum_{k \in K_s} \frac{\eta_{k+1}}{\mu_k \eta_k} = \infty \quad (2.28)$$

então $\{x^k\}$ é sequência minimizante para o problema.

(iii) Se além de (2.28), existem constantes positivas η_{\min} e B tais que para todo $k \in K_s$

$$\eta_{\min} \leq \eta_k \quad \text{e} \quad \frac{\eta_{k+1}}{\mu_k \eta_k} \leq B \quad (2.29)$$

então a sequência $\{x^k\}$ é limitada e converge para um minimizador do problema.

Demonstração. (i) Observe inicialmente que, como $\epsilon_{\text{tol}} = 0$ e o algoritmo não pára, vale que $\delta_{k+1} > 0$ para todo $k \in K_s$. Consequentemente, a sequência infinita $\{f(x^k)\}$ é estritamente decrescente. Se o problema original não possui solução, esta sequência vai para $-\infty$. Caso contrário, pelo Lema 1, a série $\sum_{k \in K_s} \delta_{k+1}$ converge. Logo o termo geral vai para zero para todo $k \in K_s$ e pelo item (ii) do Lema 2, $\epsilon_k \rightarrow 0$ para todo $k \in K_s$, pois δ_{k+1} é a soma de dois termos não negativos.

(ii) Para mostrar (ii) vamos usar a definição de M_k e x^{k+1} ($= y^{k+1}$ em (2.14)) para expandir o quadrado abaixo:

$$\begin{aligned} |x^{k+1} - x|_{k+1}^2 &= \langle x^{k+1} - x, \eta_{k+1} M(x^{k+1} - x) \rangle \\ &= \eta_{k+1} \left\langle x^k - x - \frac{1}{\mu_k \eta_k} M^{-1} \hat{s}^k, M(x^k - x) - \frac{1}{\mu_k \eta_k} \hat{s}^k \right\rangle \\ &= \frac{\eta_{k+1}}{\eta_k} \left\langle x^k - x, \underbrace{\eta_k M}_{M_k} (x^k - x) \right\rangle - \frac{2\eta_{k+1}}{\mu_k \eta_k} \langle (x^k - x), \hat{s}^k \rangle \\ &\quad + \frac{\eta_{k+1}}{\eta_k} \frac{1}{\mu_k^2} \left\langle \underbrace{\frac{1}{\eta_k} M^{-1} \hat{s}^k}_{M_k^{-1}}, \hat{s}^k \right\rangle \\ &= \frac{\eta_{k+1}}{\eta_k} |x^k - x|_k^2 + \frac{\eta_{k+1}}{\mu_k \eta_k} \left(2 \langle x - x^k, \hat{s}^k \rangle + \frac{1}{\mu_k} \|\hat{s}^k\|_k^2 \right). \end{aligned}$$

Usando o item (iii) do Lema 2:

$$\hat{s}^k \in \partial_{\epsilon_k} f(x^k) \iff f(x) \geq f(x^k) + \langle \hat{s}^k, x - x^k \rangle - \epsilon_k,$$

temos a limitação

$$2 \langle x - x^k, \hat{s}^k \rangle \leq 2(f(x) - f(x^k) + \epsilon_k)$$

e pelo item (ii) do Lema 2 podemos cotar o terceiro termo:

$$\frac{1}{\mu_k} \|\hat{s}^k\|_k^2 = \delta_{k+1} - \epsilon_k \leq 2(\delta_{k+1} - \epsilon_k).$$

Assim, usando a relação $\eta_{k+1} \leq \eta_k$ de (2.28) obtemos

$$\begin{aligned} |x^{k+1} - x|_k^2 &\leq |x^k - x|_k^2 + \frac{2\eta_{k+1}}{\mu_k \eta_k} (f(x) - f(x^k) + \epsilon_k + \delta_{k+1} - \epsilon_k) \\ &= |x^k - x|_k^2 + \frac{2\eta_{k+1}}{\mu_k \eta_k} (f(x) - f(x^k) + \delta_{k+1}). \end{aligned} \quad (2.30)$$

Para obter (ii), suponha por absurdo que existem $\tilde{x} \in \mathbb{R}^n$ e um escalar $\rho > 0$ tais que $f(\tilde{x}) < f(x^k) - \rho$ para todo $k \in K_s$. Como, pelo item (i), $\{\delta_k\} \rightarrow 0$, então existe k_ρ tal que $\delta_k \leq \frac{\rho}{2}$ para todo $k \in K_s$, $k \geq k_\rho$. Logo escrevendo a relação (2.30) para

$x = \tilde{x}$, vem

$$\begin{aligned}
0 &\leq |x^{k+1} - \tilde{x}|_k^2 \\
&\leq |x^k - \tilde{x}|_k^2 + \frac{2\eta_{k+1}}{\mu_k \eta_k} \underbrace{(f(\tilde{x}) - f(x^k))}_{< -\rho} + \underbrace{\delta_{k+1}}_{\leq \frac{\rho}{2}} \\
&\leq |x^k - \tilde{x}|_k^2 + \frac{2\eta_{k+1}}{\mu_k \eta_k} \left(-\rho + \frac{\rho}{2}\right) \\
&= |x^k - \tilde{x}|_k^2 - \frac{\eta_{k+1}\rho}{\mu_k \eta_k},
\end{aligned}$$

para todo $k \in K_s$, $k \geq k_\rho$. Somando essas desigualdades para $k_\rho \leq k \in K_s$, vem

$$0 \leq |x^{k_\rho} - \tilde{x}|_{k_\rho}^2 - \rho \sum_{\substack{k \in K_s \\ k \geq k_\rho}} \frac{\eta_{k+1}}{\mu_k \eta_k}.$$

Fazendo $k_\rho \rightarrow \infty$, obtemos o absurdo pois o lado direito da desigualdade vai para $-\infty$, ou seja, a expressão acima contradiz a ilimitação de (ii).

- (iii) Tome $x = \tilde{x}$ em (2.30) uma solução para o problema (2.1) e some para todo $k \in K_s$. Como $f(\tilde{x}) \leq f(x^k)$ então

$$\begin{aligned}
|x^k - \tilde{x}|_k^2 &< 2 \sum_{k \in K_s} \frac{\eta_{k+1}}{\mu_k \eta_{k+1}} \underbrace{(f(\tilde{x}) - f(x^k))}_{\leq 0} + \delta_{k+1} \\
&\leq 2 \underbrace{\sum_{k \in K_s} \frac{\eta_{k+1}}{\mu_k \eta_{k+1}}}_{\substack{(2.29) \\ \leq B}} + \underbrace{\delta_{k+1}}_{> 0} < \infty
\end{aligned}$$

Por (2.27) e (2.29), $\eta_{\min} \lambda \|x^k - \tilde{x}\|^2 \leq |x^k - \tilde{x}|_k^2$.

Logo a sequência $\{x^k\}$ é limitada. Extraia uma subsequência $\{x^{k_i}\}_{k_i \in K_s}$, convergente para \tilde{x} para $i \rightarrow \infty$. Para ver que a sequência toda converge para \tilde{x} , dado $\rho > 0$, tome i suficientemente grande tal que

$$\|x^{k_i} - \tilde{x}\|^2 \leq \frac{\rho \eta_{\min} \lambda}{2 \eta_{k_i} \Lambda} \quad \text{e} \quad \sum_{\substack{k \in K_s \\ k \geq k_i}} \delta_{k+1} \leq \frac{\rho \eta_{\min} \lambda}{2}. \quad (2.31)$$

Escrevendo a soma de (2.30) com $x = \tilde{x}$, para $k \in K_s$ e indo de k_i a um \bar{k} arbitrário, $\bar{k} > k_i$, vem

$$|x^{\bar{k}+1} - \tilde{x}|_{\bar{k}+1}^2 \leq |x^{k_i} - \tilde{x}|_{k_i}^2 + \sum_{\substack{k \in K_s \\ k \geq k_i}}^{\bar{k}} \delta_{k+1}.$$

Por (2.27) e (2.29), seque que

$$\begin{aligned}
 \eta_{\min} \lambda \|x^{\bar{k}+1} - \tilde{x}\|^2 &\leq |x^{\bar{k}+1} - \tilde{x}|_{\bar{k}+1}^2 \\
 &\leq |x^{k_i} - \tilde{x}|_{k_i}^2 + \sum_{\substack{k \in K_s \\ k \geq k_i}}^{\bar{k}} \delta_{k+1} \\
 &\leq \eta_{k_i} \Lambda \|x^{k_i} - \tilde{x}\|^2 + \sum_{\substack{k \in K_s \\ k \geq k_i}}^{\bar{k}} \delta_{k+1} \\
 &\stackrel{(2.31)}{\leq} \frac{\rho \eta_{\min} \lambda}{2} + \frac{\rho \eta_{\min} \lambda}{2} = \rho \eta_{\min} \lambda.
 \end{aligned}$$

Logo $\|x^{\bar{k}+1} - \tilde{x}\|^2 \leq \rho$. Sendo $\rho > 0$ arbitrário, vemos que $x^{\bar{k}+1}$ está arbitrariamente próximo de \tilde{x} , concluindo assim a prova. \square

Teorema 3 (Convergência dos Passos Nulos). *Assuma que o Algoritmo 3 com compressão gera um último iterado $x^{k_{last}}$, seguido de uma sequência infinita de passos nulos. Se para todo $k \geq k_{last}$,*

$$\mu_{k+1} \geq \mu_k \quad e \quad \mu_k \leq \mu_{max} \quad (2.32)$$

para alguma constante positiva μ_{max} então a sequência $\{y^k\}$ converge para $x^{k_{last}}$ e $x^{k_{last}}$ minimiza f .

Demonstração. Para não carregar a notação, na prova vamos suprimir os índices k_{last} e denotar $x^{k_{last}}$ e $\eta_{k_{last}}$ por x e η , respectivamente.

Como as matrizes M_k só variam em passos sérios, para todo $k \geq k_{last}$,

$$|\cdot|_k^2 = \eta \langle \cdot, M \cdot \rangle =: \eta |\cdot|_M^2 \quad e \quad \|\cdot\|_k^2 = \frac{1}{\eta} \eta \langle \cdot, M^{-1} \cdot \rangle =: \frac{1}{\eta} |\cdot|_{M^{-1}}^2.$$

Para $y \in \mathbb{R}^n$ arbitrário, considere a função

$$L_k(y) := \varphi_k(y^{k+1}) + \frac{1}{2} \mu_k \eta |y^{k+1} - x|_M^2 + \frac{1}{2} \mu_k \eta |y^{k+1} - y|_M^2.$$

Sendo y^{k+1} , solução do problema (2.10), então

$$\varphi_k(y^{k+1}) + \frac{\mu_k \eta}{2} |y^{k+1} - x|_M^2 \leq \varphi_k(x)$$

e por (2.25) $\varphi_k(x) \leq f(x)$, logo

$$L_k(y^{k+1}) \leq f(x), \quad \forall k \geq k_{last}. \quad (2.33)$$

Além disso, a identidade $\mu_k \eta M(x - y^{k+1}) = \hat{s}^k$ e o lado direito da desigualdade em (2.25), isto é, $\varphi_{k+1}(x) \geq f_{k+1} + \langle s^{k+1}, y - y^{k+1} \rangle$, nos dão

$$\begin{aligned}
\varphi_{k+1}(y) &\geq \underbrace{f_{k+1}}_{\geq \varphi_k(y^{k+1})} + \langle \hat{s}^k, y - y^{k+1} \rangle \\
&\geq \varphi_k(y^{k+1}) \mu_k \eta \langle M(x - y^{k+1}), y - y^{k+1} \rangle \\
&\stackrel{(i) \text{ Lema 3}}{=} f_a(y).
\end{aligned} \tag{2.34}$$

De fato,

$$\begin{aligned}
f_a(y) &\stackrel{(2.23)}{=} f(x) - \varepsilon_k + \langle \hat{s}^k, y - x \rangle \\
&= f(x) + \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2 - \delta_{k+1} + \langle \hat{s}^k, y - x \rangle \\
&= f(x) + \frac{1}{2\mu_k} \|\hat{s}^k\|_k^2 - f(x) + \varphi_k(y^{k+1}) + \frac{\mu_k}{2} |y^{k+1} - x|_k^2 + \langle \hat{s}^k, y - x \rangle \\
&= \varphi_k(y^{k+1}) + \frac{1}{2\mu_k} \left\langle \hat{s}^k, \frac{1}{\eta} M^{-1} \hat{s}^k \right\rangle + \frac{\mu_k}{2} \langle \eta M(y^{k+1} - x), y^{k+1} - x \rangle \\
&\quad + \langle \hat{s}^k, y - x \rangle \\
&= \varphi_k(y^{k+1}) + \frac{1}{2\mu_k \eta} \langle \mu_k \eta M(x - y^{k+1}), \mu_k \eta (x - y^{k+1}) \rangle \\
&\quad + \frac{\mu_k \eta}{2} \langle M(y^{k+1} - x), y^{k+1} - x \rangle + \langle \mu_k \eta M(x - y^{k+1}), x - y^{k+1} \rangle \\
&= \varphi_k(y^{k+1}) + \mu_k \eta \langle M(x - y^{k+1}), x - y^{k+1} \rangle + \mu_k \eta \langle M(x - y^{k+1}), y - x \rangle \\
&= \varphi_k(y^{k+1}) + \mu_k \eta \langle M(x - y^{k+1}), y - y^{k+1} \rangle.
\end{aligned}$$

Usando a hipótese (2.32), a desigualdade (2.34) escrita para $y = y^{k+2}$, e a definição de L_k , vem

$$\begin{aligned}
L_{k+1}(y^{k+2}) &= \varphi_k(y^{k+2}) + \frac{1}{2} \mu_{k+1} \eta |y^{k+2} - x|_M^2 + \frac{1}{2} \mu_{k+1} \eta |y^{k+2} - y^{k+1}|_M^2 \\
&\geq \varphi_k(y^{k+2}) + \frac{1}{2} \mu_k \eta |y^{k+2} - x|_M^2 \\
&\geq \varphi_k(y^{k+1}) + \mu_k \eta \langle M(x - y^{k+1}), y^{k+2} - y^{k+1} \rangle + \frac{1}{2} \mu_k \eta |y^{k+2} - x|_M^2 \\
&= L_k(y^{k+1}) - \frac{1}{2} \mu_k \eta |y^{k+2} - x|_M^2 + \mu_k \eta \langle M(x - y^{k+1}), y^{k+2} - y^{k+1} \rangle \\
&\quad + \frac{1}{2} \mu_k \eta |y^{k+1} - x|_M^2.
\end{aligned}$$

Expandindo a diferença de quadrados

$$\begin{aligned}
|y^{k+2} - x|_M^2 - |y^{k+1} - x|_M^2 &= \langle y^{k+2} - x + y^{k+1} - x, M(y^{k+2} - y^{k+1}) \rangle \\
&= \langle y^{k+2} - y^{k+1} + 2(y^{k+1} - x), M(y^{k+2} - y^{k+1}) \rangle \\
&= \langle y^{k+2} - y^{k+1}, M(y^{k+2} - y^{k+1}) \rangle \\
&\quad + 2 \langle y^{k+1} - x, M(y^{k+2} - y^{k+1}) \rangle \\
&= |y^{k+2} - y^{k+1}|_M^2 + 2 \langle M(y^{k+1} - x), y^{k+2} - y^{k+1} \rangle
\end{aligned}$$

obtemos a desigualdade

$$L_{k+1}(y^{k+2}) \geq L_k(y^{k+1}) + \mu_k \eta |y^{k+2} - y^{k+1}|_M^2. \quad (2.35)$$

Como a sequência crescente $\{L_k(y^{k+1})\}$ é limitada superiormente por (2.33), então essa sequência converge. Vamos mostrar agora que a sequência $\{y^{k+1}\}$ é limitada, com $\{y^{k+1} - y^k\} \rightarrow 0$. Usando novamente a identidade $\mu_k \eta M(x - y^{k+1}) = \hat{s}^k$ e a relação $f_a \leq f$ de (2.25), temos que

$$\begin{aligned} L_k(y^{k+1}) + \frac{1}{2} \mu_k \eta |y^{k+1} - x|_M^2 &= \varphi_k(y^{k+1}) + \mu_k \eta |y^{k+1} - x|_M^2 \\ &= \varphi_k(y^{k+1}) + \langle x - y^{k+1}, \mu_k \eta M(x - y^{k+1}) \rangle \\ &= \varphi_k(y^{k+1}) + \langle x - y^{k+1}, \hat{s}^k \rangle \\ &\stackrel{(i) \text{ Lema 3}}{=} f_a(x) \leq f(x). \end{aligned}$$

Pela relação (2.33), $L_k(y^{k+1}) \leq f(x)$, $\forall k \geq k_{last}$ e como $\mu_k \geq \mu_{k_{last}}$

$$\begin{aligned} L_k(y^{k+1}) + \frac{\mu_{k_{last}}}{2} \eta |y^{k+1} - x|_M^2 &\leq L_k(y^{k+1}) + \frac{\mu_k}{2} \eta |y^{k+1} - x|_M^2 \leq f(x) \\ \Rightarrow \frac{\mu_{k_{last}}}{2} \eta |y^{k+1} - x|_M^2 &\leq \underbrace{f(x) - L_k(y^{k+1})}_{=: A} (\geq 0) \\ \Rightarrow |y^{k+1} - x|_M^2 &\leq \frac{2A}{\mu_{k_{last}} \eta}, \end{aligned}$$

ou seja, $\{y^{k+1}\}$ é uma sequência limitada. Além disso, usando novamente que $\mu_k \geq \mu_{k_{last}}$, por (2.32),

$$\begin{aligned} L_{k+1}(y^{k+2}) &\geq L_k(y^{k+1}) + \frac{1}{2} \mu_k \eta |y^{k+2} - y^{k+1}|_M^2 \\ &\geq L_k(y^{k+1}) + \frac{1}{2} \mu_{k_{last}} \eta |y^{k+2} - y^{k+1}|_M^2 \end{aligned}$$

e passando ao limite obtemos que $\{y^{k+2} - y^{k+1}\} \rightarrow 0$.

Como f é convexa, então é uma função **localmente Lipschitz contínua** com subdiferencial localmente limitado. Sejam L a **constante de Lipschitz** e T o limitante para o subdiferencial em uma região limitada contendo a sequência $\{y^k\}$. Usando a desigualdade do lado direito em (2.25), isto é, $\varphi_k(y) \geq f_{k+1} + \langle \hat{s}^{k+1}, y - y^{k+1} \rangle$

$$- \underbrace{T}_{\|\hat{s}^k\| \leq T} \|y^{k+1} - y^k\| \leq \langle \hat{s}^k, y^{k+1} - y^k \rangle \leq \varphi_k(y^{k+1}) - f(y^k).$$

Por outro lado, usando a desigualdade $\varphi_k \leq f$ de (2.25) vem

$$\varphi_k(y^{k+1}) - f(y^k) \leq f(y^{k+1}) - f(y^k) \leq L \|y^{k+1} - y^k\|.$$

Portanto, como $\{y^{k+1} - y^k\} \rightarrow 0$ quando $k \rightarrow \infty$, segue que $\{\varphi_k(y^{k+1}) - f(y^k)\} \rightarrow 0$. Da sequência limitada $\{y^k\}$ extraímos uma subsequência convergente $\{y^{k_i}\} \rightarrow \bar{y}$ quando $i \rightarrow \infty$ e note que $\{y^{k_i+1}\} \rightarrow \bar{y}$, porque $\{y^{k+1} - y^k\} \rightarrow 0$.

Como f é Lipschitz e $\{y^{k_i+1} - y^{k_i}\} \rightarrow 0$, escrevemos

$$f(y^{k_i+1}) - \varphi_{k_i}(y^{k_i+1}) = f(y^{k_i+1}) - f(y^{k_i}) + \underbrace{f(y^{k_i}) - \varphi_{k_i}(y^{k_i+1})}_{\rightarrow 0},$$

para ver que quando $i \rightarrow \infty$

$$f(y^{k_i}) - \varphi_{k_i}(y^{k_i+1}) \rightarrow 0 \quad \text{e} \quad \varphi_{k_i}(y^{k_i+1}) \rightarrow f(\bar{y}). \quad (2.36)$$

Para mostrar que x minimiza f , lembramos que para $k \geq k_{last}$, o teste de decréscimo suficiente $(f(x^k) - f(y^{k+1})) \geq \alpha \delta_{k+1}$ nunca mais é satisfeito. Portanto, multiplicando por (-1)

$$f(y^{k_i+1}) - f(x) > -\alpha \delta_{k_i+1}.$$

Adicionando δ_{k_i+1} de ambos os lados dessa desigualdade e usando a definição de δ_{k+1} em (2.12) vem

$$\begin{aligned} 0 &\leq (1 - \alpha) \delta_{k_i+1} \leq f(y^{k_i+1}) - f(x) + \delta_{k_i+1} \\ &= f(y^{k_i+1}) - \varphi_{k_i}(y^{k_i+1}) - \frac{1}{2} \mu_{k_i} \eta |x - y^{k_i+1}|_M^2 \\ &\leq f(y^{k_i+1}) - \varphi_{k_i}(y^{k_i+1}). \end{aligned}$$

Passando ao limite com $i \rightarrow \infty$ e usando (2.36) concluímos que $\delta_{k_i+1} \rightarrow 0$. Pelo item (ii) do Lema 2 segue que $\varepsilon_{k_i} + \frac{1}{2\mu_{k_i}} |\hat{s}^{k_i}|_{M-1}^2$ converge para 0. Portanto, $\varepsilon_{k_i} \rightarrow 0$ e $\hat{s}^{k_i} \rightarrow 0$ com $i \rightarrow \infty$, já que $\mu_k \leq \mu_{max}$ conforme (2.32).

Agora pelo item (iii) do Lema 2, $\hat{s}^{k_i} \in \partial_{\varepsilon_{k_i}} f(x)$, ou seja,

$$f(y) \geq f(x) + \langle \hat{s}^{k_i}, y - x \rangle - \varepsilon_{k_i}, \quad \forall y \in \mathbb{R}^n.$$

Passando ao limite com $i \rightarrow \infty$, a desigualdade mostra que x minimiza f em \mathbb{R}^n .

Finalmente, vamos mostrar que um ponto limite arbitrário \bar{y} é igual a x . Usando que x minimiza f e $f \geq \varphi_k$ conforme (2.25), juntamente com o fato de que y^{k+1} é solução do problema (2.11) e a desigualdade $\mu_k \leq \mu_{max}$ conforme (2.32), temos as seguintes relações:

$$\begin{aligned} f(\bar{y}) &\geq f(x) \geq \varphi_{k_i}(x) \geq \varphi_{k_i}(y^{k_i+1}) + \frac{1}{2} \mu_{k_i} \eta |x - y^{k_i+1}|_M^2 \\ &\geq \varphi_{k_i}(y^{k_i+1}) + \frac{1}{2} \mu_{k_{last}} \eta |x - y^{k_i+1}|_M^2. \end{aligned}$$

Por (2.36), obtemos no limite que

$$\begin{aligned} f(\bar{y}) &\geq \lim_{i \rightarrow \infty} \left(\varphi_{k_i}(y^{k_i+1}) + \frac{1}{2} \mu_{k_{last}} \eta |x - y^{k_i+1}|_M^2 \right) \\ &= f(\bar{y}) + \frac{1}{2} \mu_{k_{last}} \eta |\bar{y} - x|_M^2 \end{aligned}$$

e tal relação só se cumpre quando $\bar{y} = x$. Como o desenvolvimento foi feito para um ponto limite arbitrário da sequência $\{y^{k_i+1}\}$, o limite é único: x . \square

Embora o Teorema 1 garanta que as três variantes são equivalentes, como as relações estabelecidas em tal resultado são de existência, não é possível relacionar, pelo menos diretamente, a convergência para a variante proximal com a convergência para as outras variantes. De qualquer forma, desse ponto de vista teórico, ficaremos apenas com a análise apresentada nesta seção. Cabe destacar que os Teoremas 2 e 3, e suas demonstrações, foram incluídos para registro de nossos estudos, com passagens e detalhamentos adicionais aos apresentados em [5]. Analisaremos a seguir a variante em níveis, com ênfase nos aspectos computacionais.

2.3.5 Método de Feixes em Níveis

Como é apontado por Frangioni em [15], geralmente o modelo $\varphi_k = \check{f}_k$ é muito otimista, ou seja, ele subestima demais o valor exato de f em grande parte do domínio, o que muitas vezes ocasiona de $\varphi_k(y^k) \ll f(x^k)$ enquanto $f(y^k) \gg f(x^k)$. Uma das vantagens da variante de níveis está no parâmetro de estabilização ℓ_k . Uma vez que estabelecemos uma maneira clara de se atualizar o parâmetro ℓ_k (diferentemente das outras variantes que fazem uso de heurísticas), possivelmente conseguimos melhores resultados de convergência. Quando o valor ótimo f^* é conhecido, a estratégia mais simples de atualização é escolher um valor $\alpha \in (0, 1)$, fixo, e tomar $\ell_k := \alpha f^* + (1 - \alpha) f(x^k)$. No entanto, mesmo que não se conheça o valor ótimo f^* , basta trabalharmos com um limitante inferior f_k^{low} e um limitante superior f_k^{up} para estabelecermos o valor de $\ell_k = \alpha f_k^{low} + (1 - \alpha) f_k^{up}$, e atualizarmos tais limitantes com cautela ao longo das iterações.

Perceba que esse valor $\alpha \in (0, 1)$ vai determinar o quão exigentes queremos ser. Quanto mais próximo de 0, mais próximo ℓ_k vai ficar do limitante superior. Faz sentido supor que, nesse caso, maiores serão as chances do conjunto viável do subproblema ser diferente de vazio e/ou do conjunto viável ser muito grande. Em outras palavras, as chances do centro de estabilidade pertencer a esse conjunto são maiores, logo não sairíamos do lugar. Assim, essa estratégia pode aumentar a quantidade de passos nulos do método, pois só vamos aceitar um decréscimo da função que seja uma fração α do decréscimo nominal.

Em contrapartida, quanto mais próximo de 1, mais próximo ℓ_k vai ficar do limitante inferior e por consequência, maiores serão as chances do conjunto viável do subproblema ser muito pequeno, favorecendo a situação em que o centro de estabilidade não pertence ao conjunto viável e dessa forma, obtemos um passo sério. Obviamente, se o conjunto viável for vazio vamos dar um passo nulo e atualizar o limitante inferior.

Na Figura 14 ilustramos uma iteração do método de feixes em níveis:

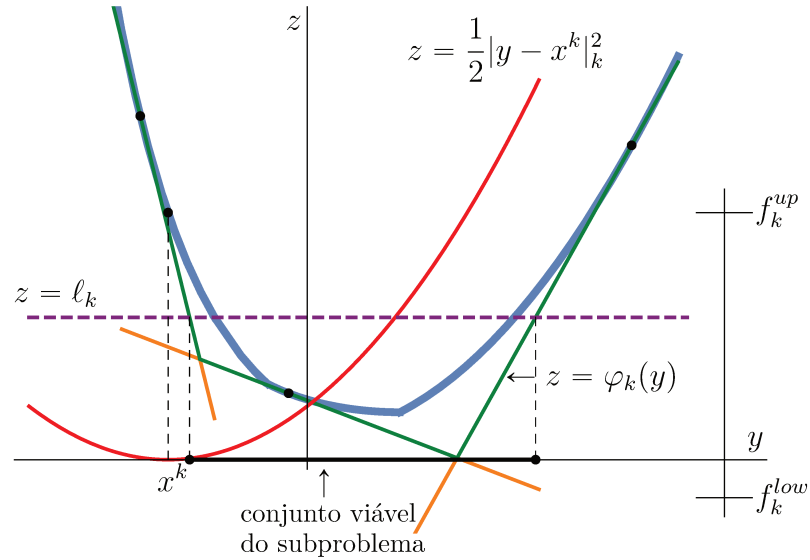


Figura 14 – Visualização dos principais elementos envolvidos no Método de Feixes em Níveis. A função objetivo original f está em em destaque (azul), assim como o segmento que define o conjunto viável do subproblema na instância ilustrada.

E assim como foi feito para a variante (2.11), por meio do Lema 2, vamos caracterizar a solução do subproblema (2.10). Para isso, vamos reescrevê-lo expandindo as suas restrições da seguinte maneira:

$$\begin{aligned} \min \quad & \frac{1}{2} |y - x^k|^2 \\ \text{s.a} \quad & f(x^k) + \langle s^i, (y - x^k) \rangle - \xi_i \leq \ell_k, \quad i = 1, \dots, k. \end{aligned} \tag{2.37}$$

Vamos trabalhar sob duas hipóteses:

$$(H1) \quad \{y \in \mathbb{R}^n \mid \varphi_k(y) \leq \ell_k\} \neq \emptyset;$$

$$(H2) \quad \varphi_k(x^k) > \ell_k.$$

A hipótese (H1) assegura que (2.10) e (2.37) estão bem definidos, e sendo problemas quadráticos estritamente convexos, possuem solução única, que chamaremos de y^{k+1} . Já (H2) elimina o caso simples em que $\varphi_k(x^k) \leq \ell_k$, para o qual $y^{k+1} = x^k$, concentrando a análise na situação não trivial.

Sob (H1) e (H2), e tomando $\lambda \in \mathbb{R}_+^{m_k}$ definimos a função Lagrangiana para o problema expandido (2.37):

$$\mathcal{L}(y, \lambda) = \frac{1}{2} \|y - x^k\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) + \langle s^j, (y - x^k) \rangle) - \xi_j - \ell_k.$$

Vamos construir o problema dual para (2.37). Para que $\sup_{\lambda \geq 0} \mathcal{L}(y, \lambda) < \infty$ é preciso que, nos termos da somatória na Lagrangiana, como $\lambda \geq 0$, tenhamos a **viabilidade** primal, isto é, para todo $j = 1, \dots, m_k$

$$f(x^k) + \langle s^j, (y - x^k) \rangle - \xi_j - \ell_k \leq 0.$$

Por outro lado, para assegurar que $\inf_{y \in \mathbb{R}^n} \mathcal{L}(y, \lambda) > -\infty$ será preciso impor que $\lambda_j = 0$ sempre que tivermos viabilidade estrita, isto é, $f(x^k) + \langle s^j, (y - x^k) \rangle - \xi_j < \ell_k$ (o que assegurará a complementaridade), pois como $\lambda \geq 0$, se para tal j tivermos $\lambda_j > 0$, a Lagrangiana pode ficar ilimitada inferiormente.

Vamos portanto definir o conjunto de índices

$$I_k(y) := \{i \in \{1, \dots, m_k\} \mid f(x^k) + \langle s^i, (y - x^k) \rangle - \xi_i \geq \ell_k\} \quad (2.38)$$

para trabalhar com as componentes não necessariamente nulas do vetor de multiplicadores.

O problema dual é então

$$\begin{aligned} \max \quad & \phi(\lambda) \\ \text{s.a} \quad & \lambda \geq 0 \end{aligned} \quad (2.39)$$

em que $\phi(\lambda) = \inf_{y \in \mathbb{R}^n} \mathcal{L}(y, \lambda)$, com y tal que $\inf_y \mathcal{L}(y, \lambda) > -\infty$. Como $\mathcal{L}(y, \lambda)$ é uma função convexa e suave, o $\inf_y \mathcal{L}(y, \lambda)$ se realiza em y tal que $\nabla_y \mathcal{L}(y, \lambda) = 0$, ou seja,

$M_k(y - x^k) + \sum_{j=1}^{m_k} \lambda_j s^j = 0$. Chamando tal ponto de $y(\lambda)$ temos

$$y(\lambda) = x^k - M_k^{-1} \sum_{j=1}^{m_k} \lambda_j s^j. \quad (2.40)$$

Dessa forma, a função dual é expressa por

$$\begin{aligned}
 \phi(\lambda) &= \mathcal{L}(y(\lambda), \lambda) \\
 &= \frac{1}{2} \|y(\lambda) - x^k\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) + \langle s^j, (y(\lambda) - x^k) \rangle - \xi_j - \ell_k) \\
 (2.40) \quad &= \frac{1}{2} \left\| -M_k^{-1} \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) + \langle s^j, -M_k^{-1} \sum_{i=1}^{m_k} \lambda_i s^i \rangle - \xi_j - \ell_k) \\
 &= \frac{1}{2} \left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) - \xi_j - \ell_k) - \underbrace{\left\langle \sum_{j=1}^{m_k} \lambda_j s^j, M_k^{-1} \sum_{i=1}^{m_k} \lambda_i s^i \right\rangle}_{\left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2} \\
 &= -\frac{1}{2} \left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) - \xi_j - \ell_k),
 \end{aligned}$$

de maneira que o problema dual fica:

$$\begin{aligned}
 \max_{\lambda \in \mathbb{R}^{m_k}} \quad & -\frac{1}{2} \left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 + \sum_{j=1}^{m_k} \lambda_j (f(x^k) - \xi_j - \ell_k) \\
 \text{s.a.} \quad & \lambda \geq 0
 \end{aligned} \tag{2.41}$$

ou, equivalentemente

$$\begin{aligned}
 \min_{\lambda \in \mathbb{R}^{m_k}} \quad & \frac{1}{2} \left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 - \sum_{j=1}^{m_k} \lambda_j (f(x^k) - \xi_j - \ell_k) \\
 \text{s.a.} \quad & \lambda \geq 0.
 \end{aligned} \tag{2.42}$$

Cabe destacar que, a menos que o conjunto de direções $\{s^j\}_{j=1}^{m_k}$ seja linearmente independente, o problema dual pode possuir infinitas soluções. No entanto, como (2.42) é um problema quadrático convexo, o valor ótimo de (2.41) e de (2.10) coincidem, ou seja, não há [folga de dualidade](#).

Vamos denotar por $\bar{\lambda} \in \mathbb{R}_+^n$ a solução de (2.41) (ou de (2.10)) de norma mínima, e escrever

$$y(\bar{\lambda}) \equiv y^{k+1} \quad \text{e} \quad I_k(y^{k+1}) \equiv I_k.$$

Levando em conta que $\bar{\lambda}_j = 0$ para $j \in \{1, \dots, m_k\} \setminus I_k$, o valor objetivo dual ótimo do problema (2.41) é dado por:

$$\phi(\bar{\lambda}) = -\frac{1}{2} \left\| \sum_{j=1}^{m_k} \bar{\lambda}_j s^j \right\|_k^2 + \sum_{j=1}^{m_k} \bar{\lambda}_j (f(x^k) - \xi_j - \ell_k), \tag{2.43}$$

e a solução do problema primal será

$$y^{k+1} = x^k - M_k^{-1} \sum_{j \in I_k} \bar{\lambda}_j s^j. \quad (2.44)$$

Note que o valor ótimo primal é expresso por:

$$\begin{aligned} \frac{1}{2} \|y^{k+1} - x^k\|_k^2 &= \frac{1}{2} \left\| -M_k^{-1} \sum_{j \in I_k} \bar{\lambda}_j s^j \right\|_k^2 \\ &= \frac{1}{2} \left\langle M_k M_k^{-1} \sum_{j \in I_k} \bar{\lambda}_j s^j, M_k^{-1} \sum_{j \in I_k} \bar{\lambda}_j s^j \right\rangle \\ &= \frac{1}{2} \left\langle \sum_{j \in I_k} \bar{\lambda}_j s^j, M_k^{-1} \sum_{j \in I_k} \bar{\lambda}_j s^j \right\rangle \\ &= \frac{1}{2} \left\| \sum_{j \in I_k} \bar{\lambda}_j s^j \right\|_k^2. \end{aligned} \quad (2.45)$$

Além disso, das restrições do problema primal temos

$$f(x^k) + \langle s^j, (y - x^k) \rangle - \xi_j = \ell_k, \quad \forall j \in I_k.$$

Logo

$$f(x^k) - \xi_j - \ell_k \stackrel{(2.44)}{=} \left\langle s^j, M_k^{-1} \sum_{i \in I_k} \bar{\lambda}_i s^i \right\rangle, \quad \forall j \in I_k \quad (2.46)$$

e portanto

$$\begin{aligned} \sum_{j \in I_k} \bar{\lambda}_j (f(x^k) - \xi_j - \ell_k) &= \left\langle \sum_{j \in I_k} \bar{\lambda}_j s^j, M_k^{-1} \sum_{i \in I_k} \bar{\lambda}_i s^i \right\rangle \\ &= \left\| \sum_{j \in I_k} \bar{\lambda}_j s^j \right\|_k^2. \end{aligned}$$

Substituindo esta última relação em (2.43), obtemos

$$\phi(\bar{\lambda}) = \frac{1}{2} \left\| \sum_{j \in I_k} \bar{\lambda}_j s^j \right\|_k^2,$$

comprovando que os valores ótimos primal e dual coincidem.

Queremos exibir uma caracterização para o par primal-dual do método de feixe em níveis na mesma linha do Lema 2 ([5, Lema 10.8]). Utilizaremos a definição de decréscimo nominal dada por

$$\delta_{k+1} := f(x^k) - \varphi_k(y^{k+1}).$$

No contexto de feixes em níveis, e sob a hipótese (H2), vale $\varphi_k(y^{k+1}) = \ell_k$, logo

$$\delta_{k+1} = f(x^k) - \ell_k. \quad (2.47)$$

Substituindo (2.47) na relação (2.46) temos

$$\delta_{k+1} - \xi_j = \left\langle s^j, M_k^{-1} \sum_{i \in I_k} \bar{\lambda}_i s^i \right\rangle, \forall j \in I_k.$$

Multiplicando a igualdade acima por $\bar{\lambda}_j$, para $j \in I_k$, e somando vem

$$\sum_{j \in I_k} \bar{\lambda}_j (\delta_{k+1} - \xi_j) = \left\langle \sum_{j \in I_k} \bar{\lambda}_j s^j, M_k^{-1} \sum_{i \in I_k} \bar{\lambda}_i s^i \right\rangle$$

e então

$$\delta_{k+1} = \frac{\sum_{j \in I_k} \bar{\lambda}_j \xi_j}{\sum_{j \in I_k} \bar{\lambda}_j} + \frac{1}{\sum_{j \in I_k} \bar{\lambda}_j} \left\| \sum_{j \in I_k} \bar{\lambda}_j s^j \right\|_k^2. \quad (2.48)$$

Temos assim todos os elementos para enunciar e provar o resultado desejado:

Proposição 2 (Caracterização da solução do subproblema em níveis). *Seja y^{k+1} a solução única do problema (2.10), sob as hipóteses (H1) e (H2). Então*

$$y^{k+1} = x^k - M_k^{-1} \bar{s}^k, \quad (2.49)$$

em que $\bar{s}^k = \sum_{j \in I_k} \bar{\lambda}_j s^j$, com $I_k = I_k(y^{k+1})$ definido em (2.38) e $\bar{\lambda}_j \in \mathbb{R}^{m_k}$ é a solução de norma mínima do problema

$$\min_{\lambda \in \mathbb{R}^{m_k}} \frac{1}{2} \left\| \sum_{j=1}^{m_k} \lambda_j s^j \right\|_k^2 - \sum_{i=j}^{m_k} \lambda_j (f(x^k) - \xi_j - \ell_k) \quad \text{s.a. } \lambda \geq 0. \quad (2.50)$$

Adicionalmente, definindo a direção $s_N^k := \frac{1}{\sum_{j \in I_k} \bar{\lambda}_j} \bar{s}^k$, valem as relações

(i) $s_N^k \in \partial \varphi_k(y^{k+1})$;

(ii) $\delta_{k+1} = \varepsilon_k + \sum_{j \in I_k} \bar{\lambda}_j \|s_N^k\|_k^2$, em que definimos $\varepsilon_k := \frac{\sum_{j \in I_k} \bar{\lambda}_j \xi_j}{\sum_{j \in I_k} \bar{\lambda}_j}$ e

(iii) $s_N^k \in \partial_{\varepsilon_k} f(x^k)$.

Demonstração. A primeira parte da proposição foi provada no desenvolvimento preparatório que a antecede. Vejamos que se verificam as relações adicionais:

(i) Temos que

$$\varphi_k(y) = f(x^k) + \max_{1 \leq i \leq m_k} \{ \langle s^i, y - x^k \rangle - \xi_i \},$$

de modo que o subdiferencial desta função em um ponto genérico é o conjunto

$$\partial\varphi_k(y) = \left\{ \sum_{i=1}^{m_k} \theta_i s^i \mid \sum_{i=1}^{m_k} \theta_i = 1, \theta_i \geq 0, \forall i = 1, \dots, m_k \right\}.$$

Em y^{k+1} temos

$$\partial\varphi_k(y^{k+1}) = \left\{ \sum_{i \in I_k} \theta_i s^i : \sum_{i \in I_k} \theta_i = 1, \text{ em que } \theta_i \geq 0, \text{ se } i \in I_k \text{ e } \theta_i = 0, \text{ se } i \notin I_k \right\}.$$

Como

$$s_N^k := \frac{1}{\sum_{i \in I_k} \bar{\lambda}_i} \sum_{j \in I_k} \bar{\lambda}_j s^j = \sum_{j \in I_k} \frac{\bar{\lambda}_j}{\sum_{i \in I_k} \bar{\lambda}_i},$$

temos s_N^k expresso como combinação linear com coeficientes não negativos e que somam 1, o que mostra a primeira relação.

(ii) Da relação (2.48), com as substituições:

$$\varepsilon_k := \frac{\sum_{j \in I_k} \bar{\lambda}_j \xi_j}{\sum_{j \in I_k} \bar{\lambda}_j} \quad \text{e} \quad \bar{s}^k = \sum_{j \in I_k} \bar{\lambda}_j s^j$$

temos

$$\delta_{k+1} = \varepsilon_k + \frac{1}{\sum_{j \in I_k} \bar{\lambda}_j} \|\bar{s}^k\|_k^2.$$

(iii) Usando o fato que os modelos φ_k são subaproximações para f , ou seja, $f(y) \geq \varphi_k(y)$ para todo y , e pelo item (i) escrevemos

$$f(y) \geq \varphi_k(y) \geq \varphi_k(y^{k+1}) + \langle s_N^k, y - y^{k+1} \rangle.$$

Somando e subtraindo x^k no segundo termo do produto interno, separando e usando a relação (2.49) vem

$$\begin{aligned} f(y) &\geq \varphi_k(y^{k+1}) + \langle s_N^k, y - x^k \rangle - \langle s_N^k, y^{k+1} - x^k \rangle \\ &= \varphi_k(y^{k+1}) + \langle s_N^k, y - x^k \rangle + \langle s_N^k, M_k^{-1} \bar{s}^k \rangle. \end{aligned}$$

Como $\varphi_k(y^{k+1}) = \ell_k$, somando e subtraindo $f(x^k)$ obtemos

$$\begin{aligned} f(y) &\geq f(x^k) + \langle s_N^k, y - x^k \rangle + \ell_k - f(x^k) + \langle s_N^k, M_k^{-1} \bar{s}^k \rangle \\ &\stackrel{(2.8)}{=} f(x^k) + \langle s_N^k, y - x^k \rangle - \delta_{k+1} + \sum_{j \in I_k} \bar{\lambda}_j \langle s_N^k, M_k^{-1} s^j \rangle \\ &= f(x^k) + \langle s_N^k, y - x^k \rangle - \delta_{k+1} + \sum_{j \in I_k} \bar{\lambda}_j \|s_N^k\|_k^2 \\ &\stackrel{(ii)}{=} f(x^k) + \langle s_N^k, y - x^k \rangle - \varepsilon_k \end{aligned}$$

Portanto, pela Definição 5 segue que $s_N^k \in \partial_{\varepsilon_k} f(x^k)$. \square

Na prática, podemos resolver tanto o problema (2.10), quanto o seu dual (2.50). Inclusive, pensando no critério de parada, é preferível testar se

$$\varepsilon_k \leq \text{tol}_\varepsilon \quad \text{e} \quad \|\hat{s}^k\|_k \leq \text{tol}_s \quad (2.51)$$

do que testar se $\delta_{k+1} \leq \varepsilon_{\text{tol}}$, para tolerâncias pré estabelecidas tol_ε e tol_s .

Adicionalmente, esse resultado vai nos permitir elaborar uma rotina de compressão para a variante do método de feixes em níveis. Na Figura 15 ilustramos uma iteração do método de feixes em níveis na qual aplicamos as técnicas de compressão. Podemos, inclusive, comparar o modelo desagregado na Figura 14 com o seu respectivo modelo agregado.

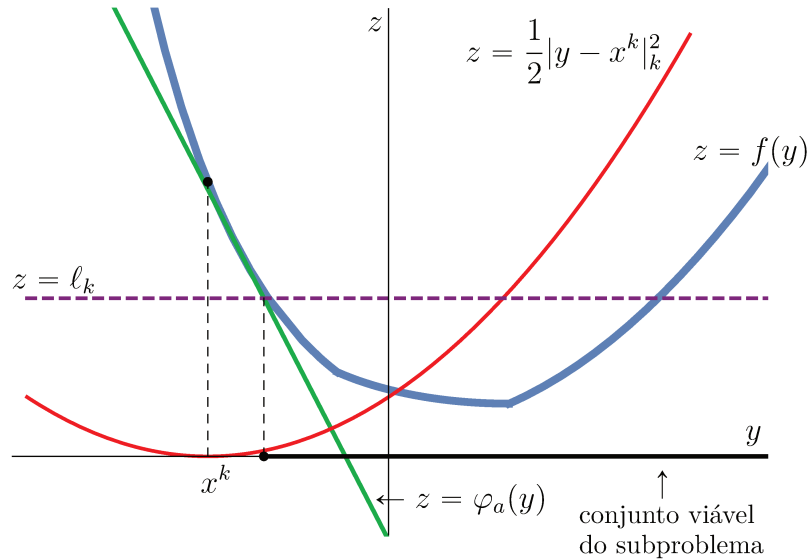


Figura 15 – Visualização dos principais elementos envolvidos no Método de Feixes em Níveis, com ênfase para o modelo agregado $\varphi_a(y) = f(x^k) - \varepsilon_k + \langle s_N^k, y - x^k \rangle$, e elementos do subproblema corrente (função objetivo e conjunto viável) para a instância ilustrada.

Vamos primeiramente apresentar o algoritmo sem nenhuma técnica de compressão e em seguida apresentamos a rotina de compressão:

Algoritmo 4 – MÉTODO DE FEIXES EM NÍVEIS

Entrada: $y^1 \in \mathbb{X}$, $\epsilon_{\text{tol}} > 0$, $f_1^{\text{low}} \leq f^*$, $\alpha \in (0, 1)$ e $itmax \in \mathbb{N}$.
 Inicializamos: $f_1^{\text{up}} = \infty$, $\hat{\delta}_1 = \infty$, $m_{\text{max}} \in \mathbb{N}$, $\mathcal{B} = \emptyset$, $x^1 = y^1$ e $k = 1$
 Envia y^1 para o oráculo.

repita

Passo 1: Receber informação do oráculo

$\mathcal{B} \leftarrow \mathcal{B} \cup \{(s^k, \xi_k)\}$

se $f(y^k) < f_k^{\text{up}}$ **então**

$f_k^{\text{up}} = f(y^k)$
 $x^* = y^k$

fim

Passo 2: Testar otimalidade e decréscimo suficiente

Atualiza o *gap*: $\delta_k = f_k^{\text{up}} - f_k^{\text{low}}$

se $\delta_k \leq \epsilon_{\text{tol}}$ **então**

retorna x^*

fim

se $\delta_k \leq \alpha \hat{\delta}$ **então**

$x^k = x^*$
 $\hat{\delta} = \delta_k$
 $m_k \leftarrow |\mathcal{B}|$ (cardinalidade de \mathcal{B})
 se $m_k \geq m_{\text{max}}$ **então**
 | Aciona rotina de compressão
 fim

fim

Atualiza o *nível*: $\ell_k = f_k^{\text{up}} - \alpha \delta_k$

Passo 3: Computar o próximo iterado

Resolve o problema quadrático (2.10) para obter \bar{y}

se (2.10) *é viável* **então**

$y^{k+1} = \bar{y}$
 $f_{k+1}^{\text{low}} = f_k^{\text{low}}$
 $f_{k+1}^{\text{up}} = f_k^{\text{up}}$

senão

$f_k^{\text{low}} = \ell_k$
 Volte para o **Passo 2**

fim

Passo 4: Devolver a informação para o oráculo

Envia y^{k+1} para o oráculo

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

até $k \geq itmax$;

Saída: x^*

Embora para a versão do método de feixes em níveis seja possível efetuar a compressão de forma mais específica, utilizando, por exemplo elementos de Oliveira [9], neste trabalho, optamos por seguir a rotina apresentada na referência [5], desenvolvida para a variante Proximal, mas adequando o problema dual à variante em Níveis, com base na Proposição 2. Para incluir as técnicas de compressão, basta inicializarmos $m_{max} < \infty$ e inserirmos a rotina abaixo no Passo 2 do Algoritmo 4.

Algoritmo 5 – ROTINA DE COMPRESSÃO

Entrada: m_{max} , $m_k \in \mathbb{N}$ e \mathcal{B} .

Passo 1: Resolver o problema dual (2.50)

Defina o conjunto de índices dos \mathcal{B} -pares indispensáveis:

$$J := \{i \leq m_k : \bar{\lambda}_i > 0\}$$

Determine a cardinalidade de J :

$$n_{act} \leftarrow |J|$$

Passo 2: Realizar seleção

se $n_{act} \leq m_{max} - 1$ **então**

 Descarte todos os \mathcal{B} -pares dispensáveis de \mathcal{B}

 Guarde a nova cardinalidade do feixe:

$$n_{left} \leftarrow n_{act}$$

 Defina: $m_{k+1} = n_{left} + 1$

senão

 Descarte dois ou mais \mathcal{B} -pares de \mathcal{B}

 Defina: $m_{k+1} = n_{left} + 2$

fim

Passo 3: Testar cardinalidade do feixe e comprimir

se $n_{left} \neq n_{act}$ **então**

$$\mathcal{B} \leftarrow \mathcal{B} \cup \{(s^{m_{k+1}-1}, \xi_{m_{k+1}-1}) := (\bar{s}_N^k, \varepsilon_k)\}$$

fim

Passo 4: Atualizar e aperfeiçoar o feixe

Defina: $s^{m_{k+1}} := s(y^{k+1})$

se y^{k+1} *é um passo sério* **então**

 Defina: $\xi_{m_{k+1}} := 0$

 Atualiza erros de linearização:

para $i = 1 : m_{k+1} - 1$ **faça**

$$\quad \xi_i := \xi_i + f(y^{k+1}) - f(x^k) - \langle s^i, y^{k+1} - x^k \rangle$$

fim

senão

 Defina: $\xi_{m_{k+1}} := f(x^k) - (f(y^{k+1}) + \langle s^{k+1}, x^k - y^{k+1} \rangle)$

fim

$$\mathcal{B} \leftarrow \mathcal{B} \cup \{(s^{m_{k+1}}, \xi_{m_{k+1}})\}$$

Saída: m_{k+1} e \mathcal{B}

2.4 Métodos de Segunda Ordem

Os métodos de segunda ordem baseiam-se em utilizar informações de segunda ordem da função para construir a direção de descida (em geral são aproximações secantes, e não a matriz hessiana propriamente dita) e com isso conseguem obter uma taxa de convergência *superlinear*.

Ainda pensando no nosso contexto de feixes, poderíamos construir um modelo que usa a informação de segunda ordem da seguinte maneira

$$\varphi_k(y) = \max_{i \leq k} \left\{ f_i + \langle s^i, (y - x^k) \rangle + \frac{1}{2} \langle x^k - y^i, H_k(x^k - y^i) \rangle \right\},$$

em que H_k é uma aproximação para a matriz hessiana. A ideia aqui é tentar capturar a curvatura da função objetivo para que, de alguma forma, o modelo se torne uma aproximação ainda melhor do que a aproximação por planos cortantes. Perceba que assim como a função objetivo pode possuir derivadas primeiras na maior do domínio, não é extraordinário pensar que ela também possua derivadas segundas na maior maior parte do domínio.

Porém, tudo tem seu preço. Ao ganhar uma melhor aproximação para a função objetivo, alteramos a estrutura do nosso problema, pois agora nossas restrições não são mais lineares, mas sim restrições não lineares. Se temos um *solver* que é capaz de resolver um problema não linear com restrições quadráticas, de forma tão eficiente e rápida quanto resolver um problema quadrático com restrições, é possível que o custo benefício valha a pena.

Os métodos de feixe de segunda ordem mais conhecidos são os de decomposição $\mathcal{V}\mathcal{U}$, que consistem em decompor o domínio da função em dois subespaços ortogonais: \mathcal{V} e \mathcal{U} , em que \mathcal{V} contém as características não suaves e \mathcal{U} contém as características suaves da função. Note que no subespaço \mathcal{U} é possível calcular passos de Newton, ou quase-Newton, o que acelera a convergência. Os autores [Mifflin e Sagastizábal](#) explicam com mais detalhes este método no trabalho [\[23\]](#).

3 Problema: Árvore Euclidiana de Steiner

Neste capítulo, faremos uma breve apresentação dos *problemas bidimensionais* que foram utilizados para testarmos o método, ou seja, um problema convexo, não suave e que pudéssemos ter de antemão a solução para analisarmos a acurácia do método. O problema que escolhemos foi o de obter o ponto de Fermat¹ de um triângulo, ou seja:

Problema 1. *Dados A , B e C , pontos não colineares em \mathbb{R}^2 , queremos obter o ponto S tal que a soma das distâncias entre S e cada um dos pontos dados seja a menor possível. Matematicamente podemos escrever o problema da seguinte forma:*

$$\min_{S \in \mathbb{R}^2} \|S - A\|_2 + \|S - B\|_2 + \|S - C\|_2 \quad (3.1)$$

Inicialmente, trabalhamos com o Problema 1, pois dessa forma, além da análise do custo computacional e tempo de execução do método, pudemos também visualizar geometricamente a convergência do método. Em seguida, trabalhamos com os problemas de Steiner e Steiner Generalizado, a fim de analisar a *robustez* do método para dimensões maiores. Recomendamos a referência [16] para um estudo mais aprofundado, tanto sobre a origem quanto sobre os algoritmos para resolução do problema da árvore de Steiner.

3.1 O problema de Steiner

Primeiramente, vamos contar um pouco do contexto histórico do problema, o que vai nos ajudar a entender como que o problema proposto por Fermat, atualmente é popularmente conhecido como problema de Steiner.

O problema de Steiner nada mais é do que problema de Fermat generalizado e podemos descrevê-lo da seguinte maneira:

Problema 2. *Dados N pontos em um plano (ou no espaço), queremos conectá-los por segmentos de reta de tal forma que a soma dos segmentos seja mínima e que quaisquer dois pontos podem ser interligados por segmentos de reta diretamente ou através de outros pontos e segmentos de reta.*

Para entendermos porque a generalização do Problema 1 é conhecida como Problema de Steiner, vale a pena acompanhar a cronologia dos fatos. Antes de 1640,

¹ Também é comum ser chamado de ponto de Torricelli, pois apesar do problema ter sido proposto por Pierre de Fermat no início do século XVII, quem o resolveu foi Evangelista Torricelli.

Torricelli propôs uma solução geométrica para o problema de Fermat, que consistia em construir um triângulo com os três pontos iniciais e a partir dos lados desse triângulo construir três triângulos equiláteros, cada um deles inscritos em uma circunferência, conforme ilustrado na Figura 16. O ponto de intersecção das três circunferências ficou conhecido como ponto de Fermat. Pouco tempo depois, em 1647, Cavalieri mostrou em seu livro intitulado “*Excercitationes Geometricae*” que os segmentos com uma extremidade no ponto de Torricelli e a outra em cada um dos três pontos formavam um ângulo de 120° entre si.

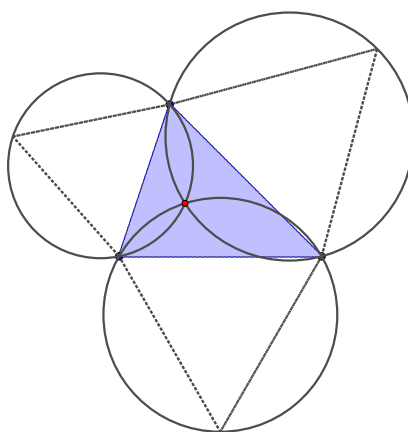


Figura 16 – Resolução proposta por Torricelli e, em vermelho, o ponto de Torricelli.

No entanto, em 1834, Heinen observou que se um dos ângulos internos do triângulo é igual ou superior a 120° , então o ponto de Torricelli fica fora do triângulo e não é mais o minimizador de (3.1). Quando isso ocorre, o minimizador do problema é justamente o vértice com ângulo interno obtuso.

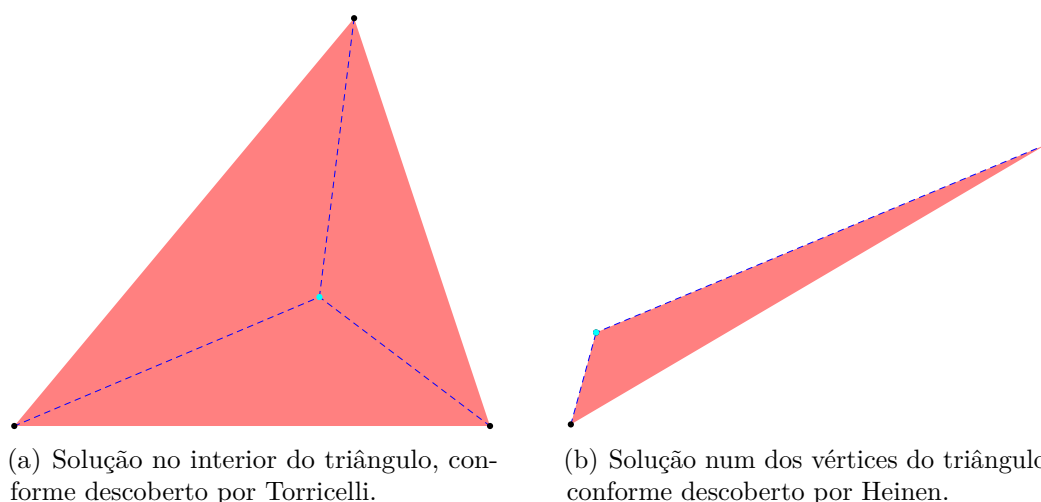


Figura 17 – Exemplos de triângulos e respectivos pontos de Fermat: solução no interior do triângulo em 17(a), e solução no vértice do triângulo obtusângulo cujo ângulo interno é igual ou superior a 120° , em 17(b).

Depois disso, o problema generalizado de Fermat, Problema 2, começou a chamar a atenção de muitos matemáticos da época, inclusive Steiner. Mas apesar do trabalho de Steiner, conforme [16, Sec. 1.1] o que realmente fez com que o problema recebesse o seu nome foi o fato de que os autores Courant e Robbins do famoso livro intitulado “*What Is Mathematics?*” (ver referência [8]), não fizeram referência a Fermat, que formulou o problema para o caso $N = 3$, e sim a Steiner.

Com o advento da Teoria dos Grafos, viu-se que o problema de Steiner poderia ser estendido. Por exemplo, um problema de Steiner em \mathbb{R}^4 pode ser interpretado como um problema bidimensional, no qual queremos encontrar dois pontos (ou vértices, usando a nomenclatura da Teoria dos Grafos), S_1 e S_2 , conforme ilustrado na Figura 18:

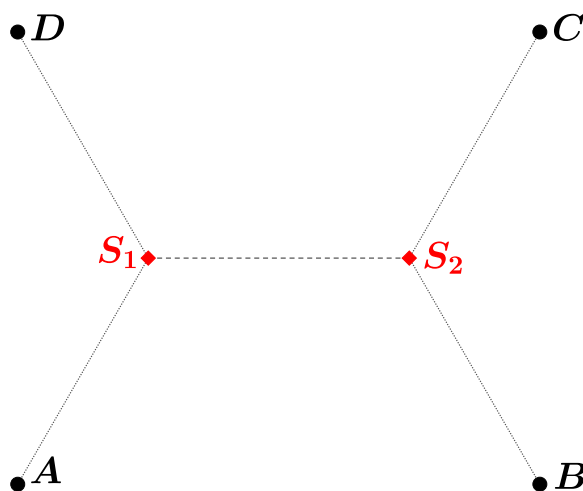


Figura 18 – Um exemplo de problema de Steiner com $N = 4$.

O problema de Steiner com $N = 4$ pode ser formulado como segue:

Problema 3. *Dados quatro pontos A, B, C e D no plano e supondo uma topologia pré-estabelecida, conforme Figura 18, podemos formular o problema de Steiner de encontrar $S = [S_1, S_2] \in \mathbb{R}^4$ da seguinte maneira:*

$$\min_{S \in \mathbb{R}^4} f(S) \quad (3.2)$$

em que

$$f(S) := \|A - S_1\|_2 + \|B - S_2\|_2 + \|C - S_2\|_2 + \|D - S_1\|_2 + \|S_1 - S_2\|_2$$

Cabe observar que, para um quadrilátero genérico, a solução exata desse problema é desconhecida.

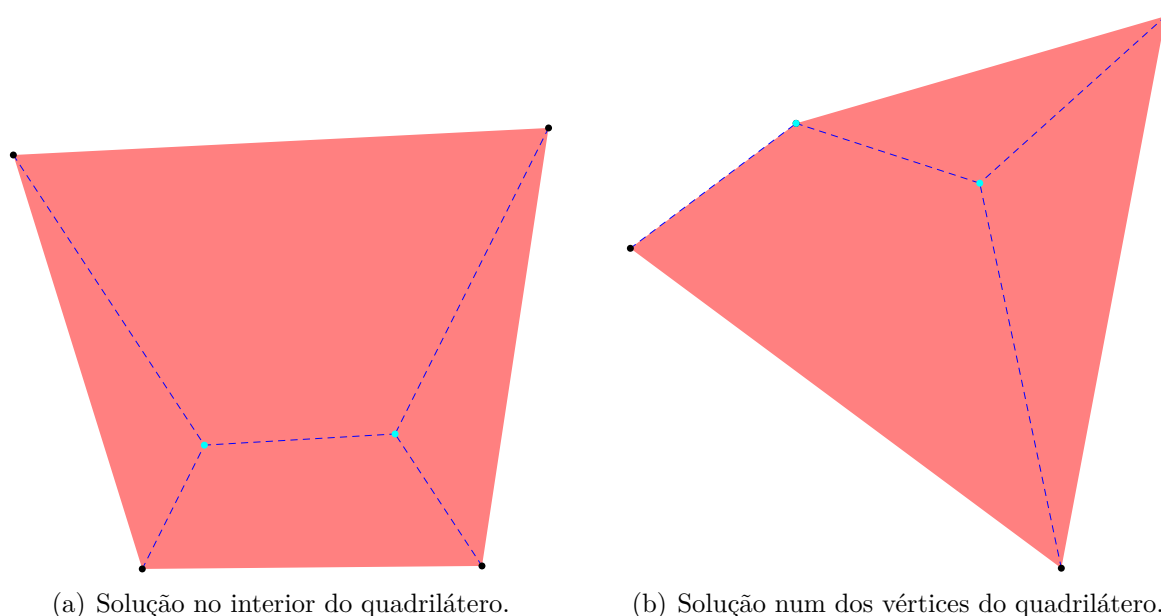


Figura 19 – Exemplos de quadriláteros com os respectivos pontos de Steiner: em 19(a) os dois pontos estão no interior, e em 19(b), um dos pontos coincide com um vértice do quadrilátero..

3.2 O problema de Steiner Generalizado

O problema de Steiner também foi muito estudado em teoria dos grafos, pois pode-se interpretar os pontos como sendo os vértices do grafo e os segmentos de reta como sendo as suas arestas (conforme Winter em [26]). Formulações matemáticas para o problema de Steiner envolvendo variáveis inteiras, e que demandam técnicas de programação inteira-mista foram propostas, ver por exemplo [7, 13, 21]. Algumas generalizações do Problema 2 foram sendo estudadas, e na que iremos apresentar aqui, as arestas possuem pesos distintos:

Problema 4. *Dados N vértices, queremos conectá-los por arestas de tal forma que a soma ponderada dos comprimentos das arestas seja mínima e que quaisquer dois vértices podem ser interligados por arestas diretamente ou por meio de outros vértices e arestas.*

No capítulo 4 vamos resolver dois problemas desse tipo, que também foram analisados por Donno e Pesamosca em [11] e por Facchinei e Lucidi em [12]. O primeiro deles pode ser formulado como segue:

Problema 5. *Dados os vértices*

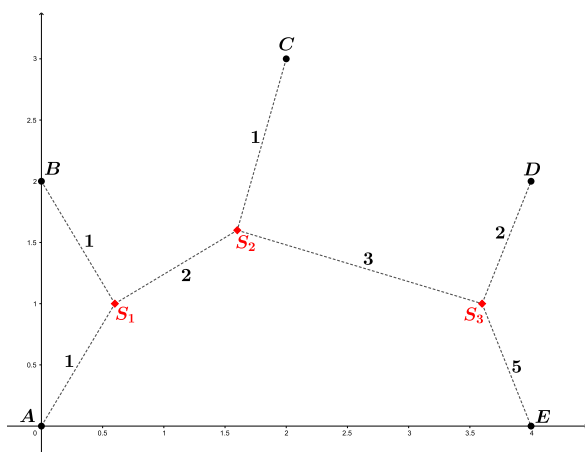
$$A = (0, 0), \quad B = (0, 2), \quad C = (2, 3), \quad D = (4, 2), \quad E = (4, 0) \in \mathbb{R}^2,$$

seja $S \in \mathbb{R}^6$ tal que $S = [S_1, S_2, S_3]$, com $S_i \in \mathbb{R}^2$ e $i = 1, 2, 3$. Então, queremos resolver o problema:

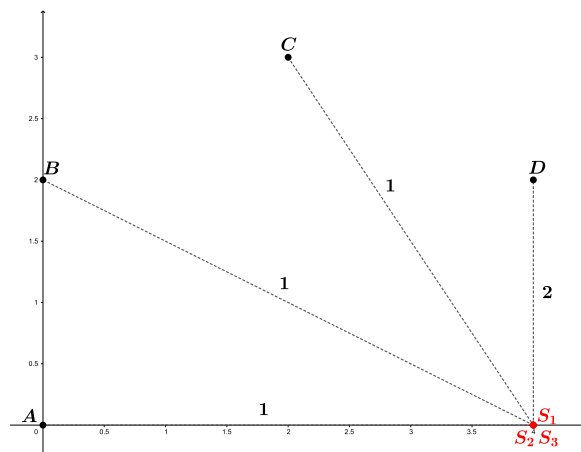
$$\min_{S \in \mathbb{R}^6} f(S), \tag{3.3}$$

em que $f(S)$ é definida da seguinte maneira:

$$\begin{aligned} f(S) := & \|S_1 - A\|_2 + \|S_1 - B\|_2 + \|S_2 - C\|_2 + \\ & + 2\|S_3 - D\|_2 + 2\|S_1 - S_2\|_2 + \\ & + 3\|S_2 - S_3\|_2 + 5\|S_3 - E\|_2. \end{aligned}$$



(a) Ilustração do grafo do problema.



(b) Ilustração do grafo solução.

Figura 20 – Ilustração com o grafo representando o problema e o grafo representando a solução do problema resolvido por [Donno e Pesamosca](#).

Apresentamos a seguir a formulação do segundo problema generalizado com o qual trabalharemos:

Problema 6. *Dados os vértices*

$$A = (0, 0), \quad B = (0, 2), \quad C = (2, 3), \quad D = (3, -1),$$

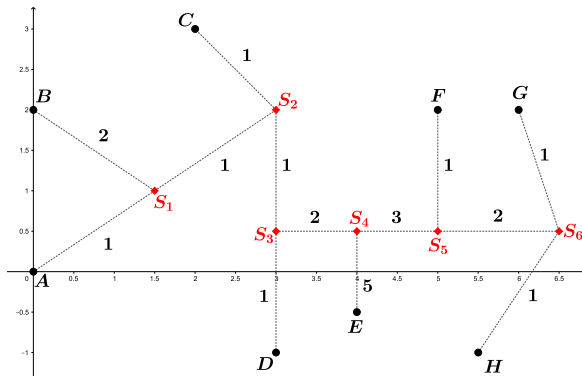
$$E = (4, -0.5), \quad F = (5, 2), \quad G = (6, 2), \quad H = (5.5, -1) \in \mathbb{R}^2,$$

seja $S \in \mathbb{R}^{12}$ tal que $S = [S_1, S_2, S_3, S_4, S_5, S_6]$, com $S_i \in \mathbb{R}^2$ e $i = 1, 2, 3, 4, 5, 6$. Então, queremos resolver o problema:

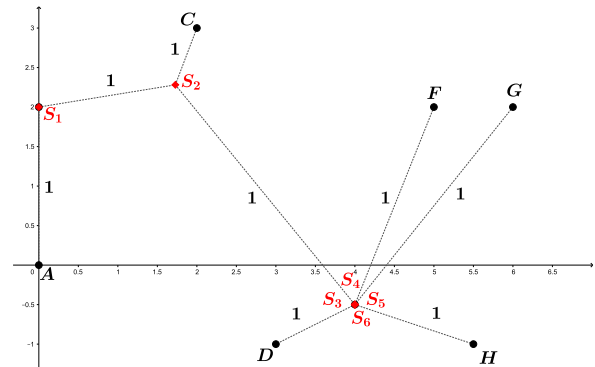
$$\min_{S \in \mathbb{R}^{12}} f(S), \tag{3.4}$$

em que $f(S)$ é definida da seguinte maneira:

$$\begin{aligned} f(S) := & \|S_1 - A\|_2 + \|S_1 - S_2\|_2 + \|S_2 - S_3\|_2 + \|S_2 - C\|_2 + \\ & + \|S_3 - D\|_2 + \|S_5 - F\|_2 + \|S_6 - G\|_2 + \|S_6 - H\|_2 + \\ & + 2\|S_1 - B\|_2 + 2\|S_3 - S_4\|_2 + 2\|S_5 - S_6\|_2 + \\ & + 3\|S_4 - S_5\|_2 + 5\|S_4 - E\|_2. \end{aligned}$$



(a) Ilustração do grafo do problema.



(b) Ilustração do grafo solução.

Figura 21 – Ilustração com o grafo representando o problema e o grafo representando a solução do problema resolvido por [Facchinei e Lucidi](#).

Em [12] foram apresentadas as duas formulações, e em [11] é exibida a solução gráfica de um dos problemas. Visando computar uma solução de referência para ser comparada com a solução encontrada pelo método de feixes em níveis, implementamos ambos os problemas no Mathematica e resolvemos com NMinimize, precisão estendida (AccuracyGoal -> 20, PrecisionGoal -> 18 e WorkingPrecision -> 20), e as quatro opções de métodos possíveis, nomeadamente NelderMead, RandomSearch, SimulatedAnnealing e DifferentialEvolution.

As soluções obtidas foram $S = (4, 0, 4, 0, 4, 0)^T$ para o problema 5 e $S = (0, 2, 1.7277, 2.2813, 4, -0.5, 4, -0.5, 4, -0.5, 4, -0.5)^T$ para o problema 6.

Dessa maneira, temos os elementos necessários para verificar a eficiência do método quando estamos trabalhando com dimensões maiores.

4 Experimentos Numéricos

Os experimentos foram todos realizados em um *desktop* com processador **Intel Core i7 8700**, com **32Gb** de RAM com 6 núcleos disponíveis. Todos os algoritmos foram implementados em **Julia** e o *solver* utilizado para resolver os problemas quadráticos foi o **Gurobi 9.1.0**.

4.1 Metodologia

4.1.1 Geração dos problemas teste

Primeiramente, resolvemos o Problema 1. Para tanto geramos um conjunto de testes com 3200 triângulos, em que os vértices A , B e C foram gerados aleatoriamente nas regiões indicadas abaixo:

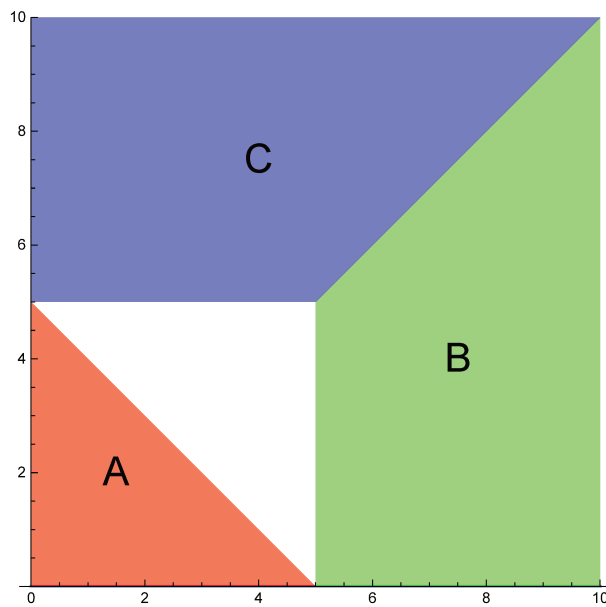


Figura 22 – Domínio dos vértices A , B e C , separados por cor.

Calculamos numericamente o ponto de Fermat de cada um dos triângulos gerados. Inicialmente, verificamos se algum dos ângulos internos era maior ou igual a 120 graus e, caso contrário, resolvemos o sistema não linear de duas equações e duas incógnitas para encontrar o ponto de intersecção entre os dois arcos capazes de 120 graus. Esse conjunto de testes foi utilizado para validar a eficiência do método.

Em seguida, resolvemos o Problema 2, com $N = 4$. Vale ressaltar que não calculamos a solução para esse conjunto de testes, sorteamos uniformemente 1% da amostra para conferirmos a solução. Nesse conjunto de testes quisemos analisar apenas o tempo

computacional gasto para resolver os problemas com $N = 4$, formulados em (3.2). Para tanto, geramos um conjunto de testes com 3200 quadriláteros, e fixamos a topologia do problema, ou seja, os vértices A , D estão conectados ao ponto de Steiner S_1 e os vértices B e C estão conectados ao ponto de Steiner S_2 , conforme ilustrado na Figura 18.

Para essa família de problemas-teste, os vértices foram gerados aleatoriamente nas regiões indicadas abaixo:

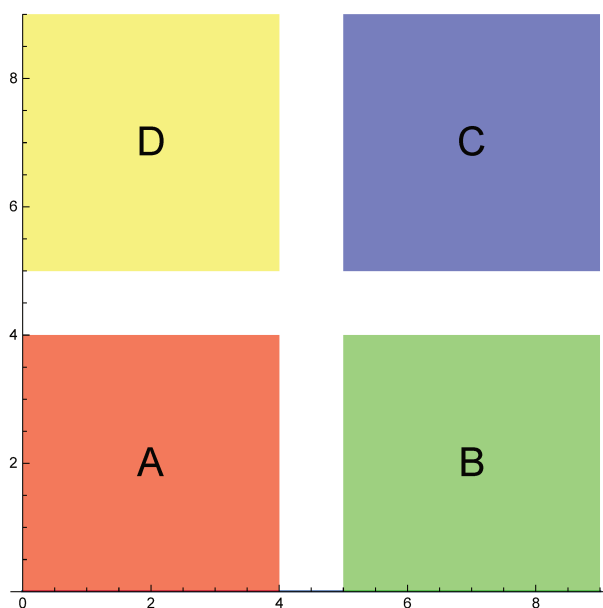


Figura 23 – Domínio dos vértices A , B , C e D , separados por cor.

Por fim, trabalhamos com instâncias do Problema 4, nomeadamente os exemplos resolvidos por Donno e Pesamosca e Facchinei e Lucidi que foram apresentados no capítulo anterior.

4.1.2 Implementação

Implementamos quatro possíveis versões para o método de feixes em níveis, sendo elas:

- SP - sem compressão e resolvendo subproblema primal;
- CP - com compressão e resolvendo subproblema primal;
- SD - sem compressão e resolvendo subproblema dual;
- CD - com compressão e resolvendo subproblema dual.

Nossa intenção foi verificar se alguma dessas quatro abordagens poderia nos oferecer um desempenho melhor para o método de feixes em níveis aplicado ao problema de Steiner. E em caso positivo, identificar as razões pelas quais uma abordagem seria mais vantajosa que as outras.

Os parâmetros algorítmicos utilizados foram: $\epsilon_{tol} = 10^{-5}$, $f^{low} = 0.1$ e $itmax = 500$.

4.1.3 Indicadores analisados

A fim de analisarmos as quatro versões apresentadas anteriormente, precisamos estabelecer alguns indicadores, capazes de medir o desempenho dessas versões e dessa forma sermos capazes de compará-las e identificar as vantagens de uma versão sobre as outras.

Como no nosso primeiro conjunto de testes temos as coordenadas do ponto de Fermat de cada um dos 3200 triângulos, isso significa que podemos comparar se a solução obtida pelas versões está próxima da solução esperada, ou seja, conseguimos mensurar a eficácia dessas versões.

Além disso, em todos os experimentos, conseguimos medir o tempo gasto por cada uma das versões para convergir para a solução e também a quantidade de problemas quadráticos que foram resolvidos, ou seja, temos elementos para medir o esforço computacional de cada versão.

Com isso em mente, podemos descrever três possíveis indicadores de desempenho:

- indicador de **eficácia**: a norma euclidiana da diferença entre a solução exata e a solução obtida pelo algoritmo;
- indicador de **eficiência**: tempo, em segundos, necessário para o método satisfazer o critério de parada estabelecido;
- indicador de **esforço**: a quantidade de problemas quadráticos resolvidos para obter a solução.

Vamos definir alguns atributos baseados nesses indicadores, dessa forma a simplificar a análise das versões.

Definição 7. A **acurácia** do método é a mediana¹ dos valores absolutos dos logaritmos (base 10) dos indicadores de eficácia (menores ou iguais a 1) de um conjunto de testes. Se a acurácia for relativamente grande/pequena o método será dito **eficaz/ineficaz**.

Definição 8. A **morosidade** do método é a média aritmética dos indicadores de eficiência de um conjunto de testes. Se a morosidade for relativamente pequena/grande, o método será dito **eficiente/ineficiente**.

¹ complementada pela análise da dispersão dos valores, que pode ser feita visualmente com base em diagramas do tipo *boxplot*.

Definição 9. O *custo computacional* do método é a mediana dos indicadores de esforço de um conjunto de testes. Se o custo computacional for relativamente pequeno/grande, o método será dito *econômico/custoso*.

4.2 Experimento 1: Influência do Ponto Inicial

Objetivo: Analisar a influência do ponto inicial para as quatro versões. Escolhemos testar o baricentro e um ponto gerado aleatoriamente na caixa $[-10, 10] \times [-10, 10]$.

Primeiro, vamos analisar a acurácia de cada versão, com o ponto inicial aleatoriamente gerado (A) e com o baricentro (B):

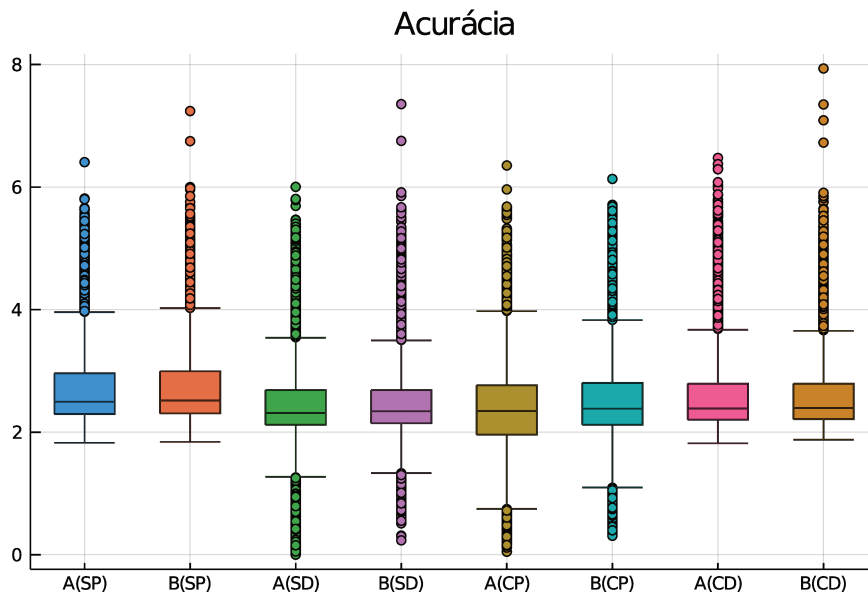


Figura 24 – Comparando a acurácia das versões utilizando o ponto gerado aleatoriamente e o baricentro como ponto inicial.

Em seguida, apresentamos duas tabelas, uma para cada tipo de ponto inicial, com a morosidade de cada versão:

Tabela 1 – Tempo e Alocação de memória das versões com ponto inicial sendo o baricentro.

Versões	Tempo	%	Média	Alocação	%	Média
CD	248s	30.3%	0.08s	17.8GiB	28.3%	0.0056GiB
SP	219s	26.7%	0.07s	19.9GiB	31.7%	0.0062GiB
CP	206s	25.2%	0.06s	14.5GiB	23.1%	0.0045GiB
SP	147s	17.9%	0.05s	10.6GiB	16.9%	0.0033GiB
<i>Total</i>	819s	100%	0.26s	62.7GiB	100%	0.0196GiB

Tabela 2 – Tempo e Alocação de memória das versões com ponto inicial aleatório.

Versões	Tempo	%	Média	Alocação	%	Média
CD	288s	29.4%	0.09s	18.9GiB	26.9%	0.0059GiB
SP	234s	26.6%	0.07s	22.0GiB	31.3%	0.0069GiB
CP	232s	26.4%	0.07s	18.1GiB	25.8%	0.0057GiB
SP	154s	17.5%	0.05s	11.2GiB	16.0%	0.0035GiB
<i>Total</i>	878s	100%	0.28s	70.2GiB	100%	0.0220GiB

Por último, exibimos um gráfico para analisarmos o custo computacional de cada versão:

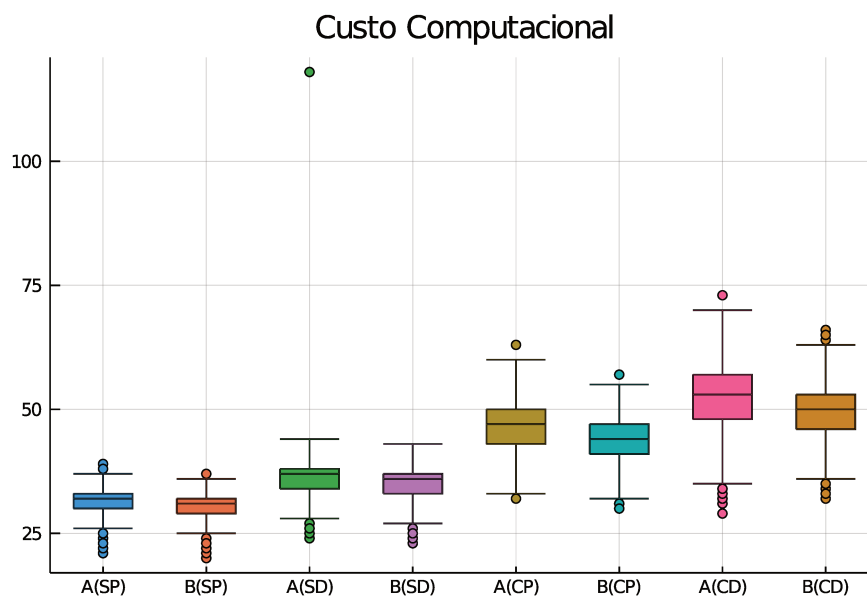


Figura 25 – Comparando o custo computacional dos métodos alterando o ponto inicial.

4.2.1 Conclusão do Experimento 1

Conclusão: Com base nos indicadores apresentados, podemos concluir que o baricentro como ponto inicial proporcionou os menores valores de morosidade e custo computacional. Observamos ainda que a acurácia não apresentou variabilidade significativa quanto à inicialização.

Do ponto de vista geométrico, faz sentido que isso ocorra, uma vez que o baricentro sempre é um ponto no interior do triângulo, conseqüentemente mais próximo do ponto de Fermat do que os pontos aleatórios, principalmente daqueles que possuem alguma coordenada negativa. Apoiados neste resultado, iremos utilizar o baricentro como ponto inicial para os demais testes.

4.3 Experimento 2: Influência do parâmetro α

Objetivo: Analisar a influência da escolha do parâmetro α para as quatro versões. Escolhemos três valores para α : 0.2, 0.5 e 0.8.

Separamos o experimento em duas partes: na primeira analisamos as versões sem rotina de compressão e, na segunda, as versões com rotina de compressão.

4.3.1 Versões sem rotina de compressão: SD e SP

Primeiro, vamos analisar a acurácia de cada versão, com os diferentes valores de α :

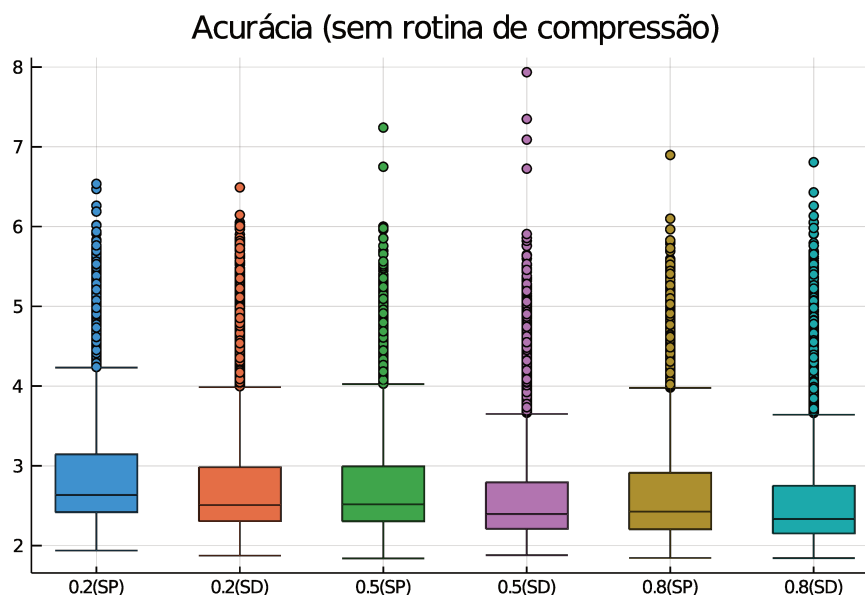


Figura 26 – Comparando a acurácia das versões sem compressão.

Em seguida, apresentamos três tabelas, uma para cada valor de α , com a morosidade de cada versão:

Tabela 3 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.2$.

Versões	Tempo	%	Média	Alocação	%	Média
SD	205s	59.5%	0.06s	14.9GiB	60.5%	0.005GiB
SP	140s	40.5%	0.04s	9.73GiB	39.5%	0.003GiB
<i>Total</i>	345s	100%	0.11s	24.6GiB	100%	0.008GiB

Tabela 4 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.5$.

Versões	Tempo	%	Média	Alocação	%	Média
SD	254s	54.7%	0.08s	17.8GiB	55%	0.006GiB
SP	210s	45.3%	0.07s	14.5GiB	45%	0.005GiB
<i>Total</i>	464s	100%	0.15s	32.3GiB	100%	0.01GiB

Tabela 5 – Tempo e Alocação de memória das versões SD e SP com $\alpha = 0.8$.

Versões	Tempo	%	Média	Alocação	%	Média
SD	495s	52.5%	0.15s	34.1GiB	52.7%	0.01GiB
SP	448s	47.5%	0.14s	30.5GiB	47.3%	0.01GiB
<i>Total</i>	943s	100%	0.29s	64.6GiB	100%	0.02GiB

Também exibimos um gráfico para analisarmos o custo computacional de cada versão:

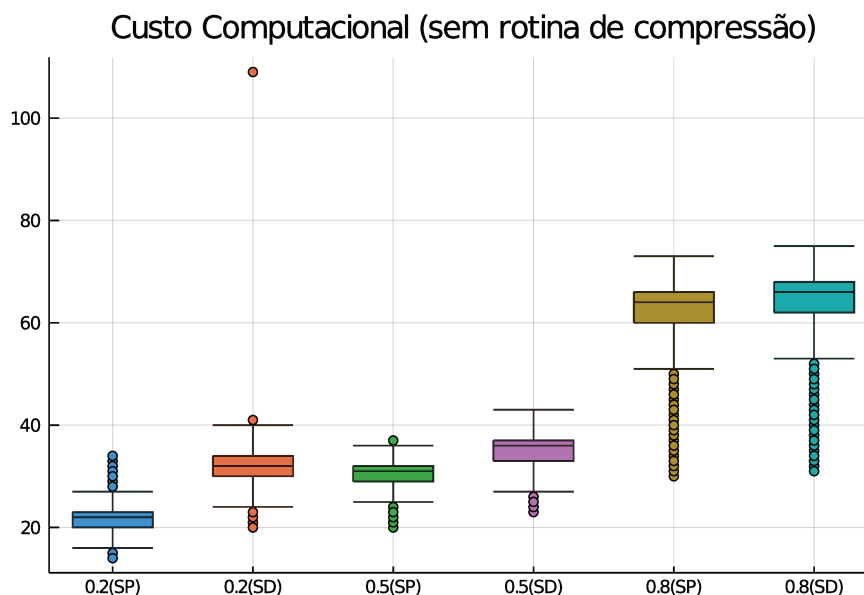


Figura 27 – Comparando o custo computacional das versões sem compressão para cada valor do parâmetro α .

E por fim, verificamos a quantidade de passos sérios e nulos:

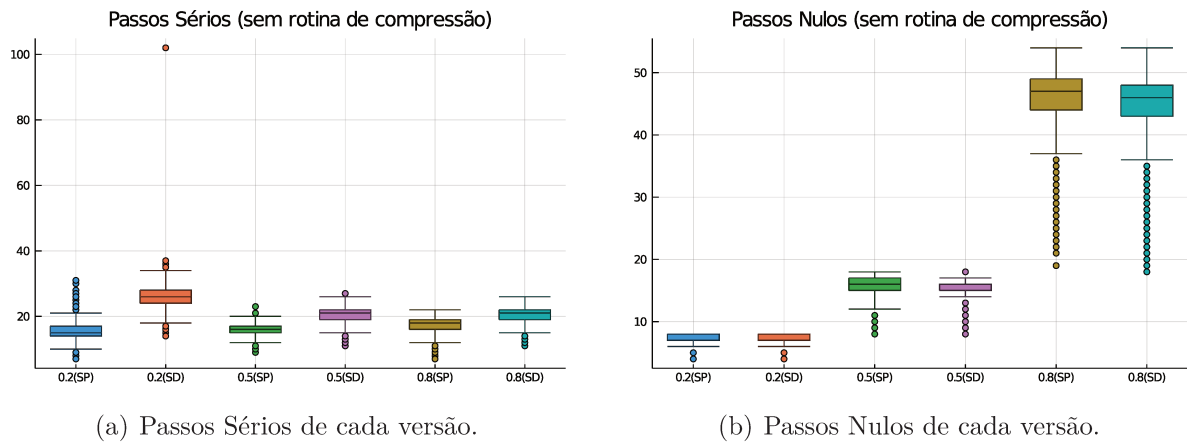


Figura 28 – Distribuição dos passos de cada versão sem rotina de compressão, para cada valor do parâmetro α .

4.3.2 Versões com rotina de compressão: CD e CP

Primeiro, vamos analisar a acurácia de cada versão, com os diferentes valores de α :

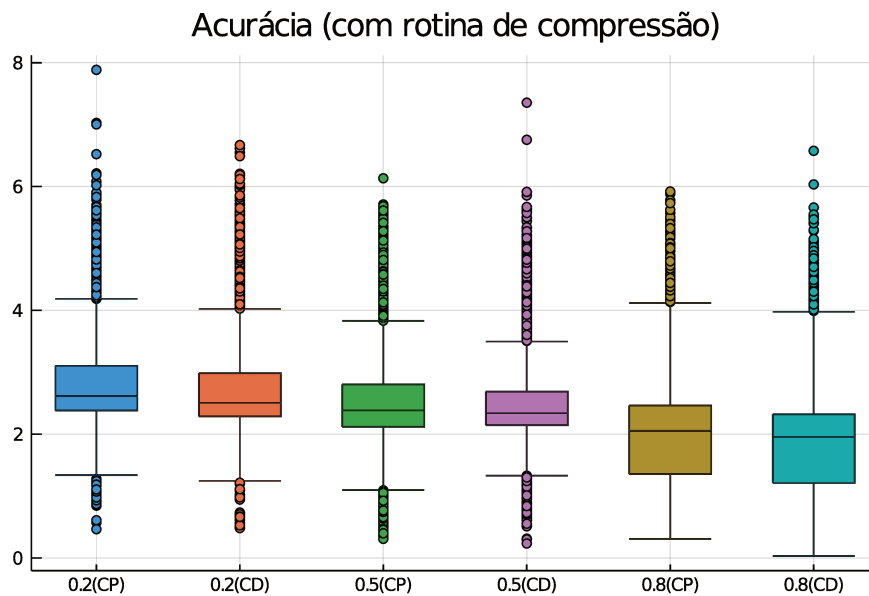


Figura 29 – Comparando a acurácia das versões com compressão para cada valor do parâmetro α .

Em seguida, apresentamos três tabelas, uma para cada valor de α , com a morosidade de cada versão:

Tabela 6 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.2$.

Versões	Tempo	%	Média	Alocação	%	Média
CD	205s	59.5%	0.06s	14.9GiB	60.5%	0.005GiB
CP	140s	40.5%	0.04s	9.73GiB	39.5%	0.003GiB
<i>Total</i>	345s	100%	0.11s	24.6GiB	100%	0.008GiB

Tabela 7 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.5$.

Versões	Tempo	%	Média	Alocação	%	Média
CD	254s	54.7%	0.08s	17.8GiB	55%	0.006GiB
CP	210s	45.3%	0.07s	14.5GiB	45%	0.005GiB
<i>Total</i>	464s	100%	0.15s	32.3GiB	100%	0.01GiB

Tabela 8 – Tempo e Alocação de memória das versões CD e CP com $\alpha = 0.8$.

Versões	Tempo	%	Média	Alocação	%	Média
CD	495s	52.5%	0.15s	34.1GiB	52.7%	0.01GiB
CP	448s	47.5%	0.14s	30.5GiB	47.3%	0.01GiB
<i>Total</i>	943s	100%	0.29s	64.6GiB	100%	0.02GiB

Também exibimos um gráfico para analisarmos o custo computacional de cada versão:

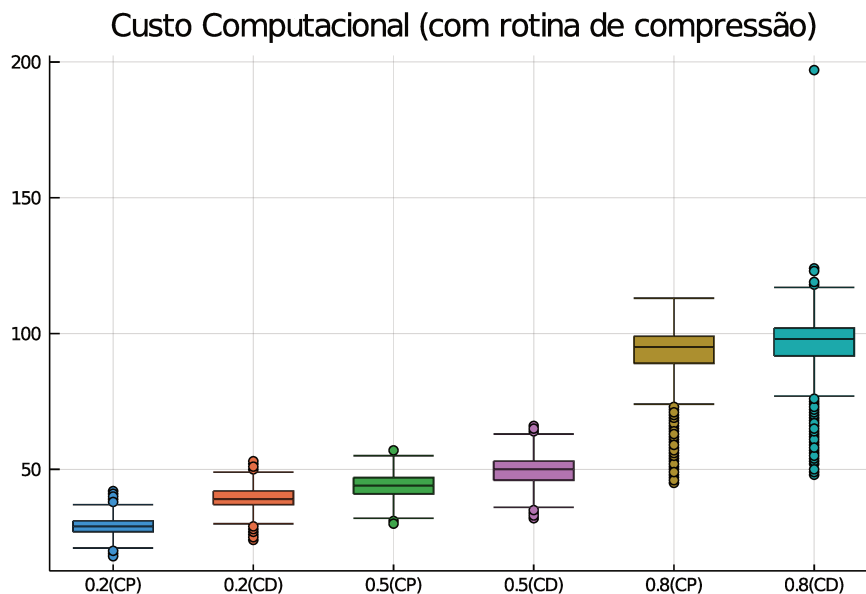


Figura 30 – Comparando o custo computacional das versões com compressão para cada valor do parâmetro α .

E por fim, verificamos a quantidade de passos sérios e nulos:

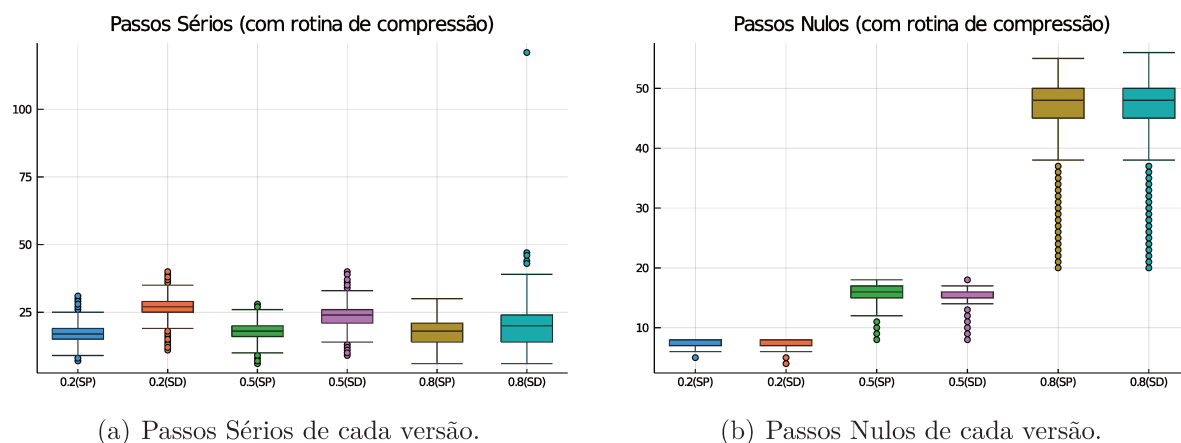


Figura 31 – Distribuição dos passos de cada versão com rotina de compressão, para cada valor do parâmetro α .

4.3.3 Conclusão do Experimento 2

Conclusão: Com base nos indicadores apresentados, podemos concluir que independentemente do uso da rotina de compressão, o uso do parâmetro $\alpha = 0.2$ foi o que proporcionou os melhores valores de acurácia, morosidade e custo computacional. Também podemos concluir que as versões que resolvem o subproblema primal (SP e CP) tiveram uma leve vantagem com relação às versões que resolvem o subproblema dual (SD e CD, respectivamente), para os três valores de α . Apoiados neste resultado, iremos utilizar o parâmetro $\alpha = 0.2$ nos demais testes.

Note que conforme observado na seção 2.3.5, foi justamente o valor de α mais próximo de 1 ($\alpha = 0.8$), que gerou mais passos nulos, como possível consequência de que muitos problemas inviáveis foram produzidos com tal escolha.

4.4 Experimento 3: Quadriláteros

Objetivo: Analisar o desempenho de cada uma das quatro versões quando aumentamos a dimensão do problema. A partir dessa análise iremos escolher a melhor versão com rotina de compressão e a melhor versão sem rotina de compressão para resolvermos problemas de dimensões maiores.

Como vimos no Capítulo 3, como desconhecemos a solução exata para quadriláteros genéricos, não temos como calcular a acurácia desse conjunto de testes. Então nos basearemos apenas nos indicadores de eficiência e de esforço.

Primeiro, verificamos o custo computacional de cada versão para resolver o conjunto de quadriláteros.

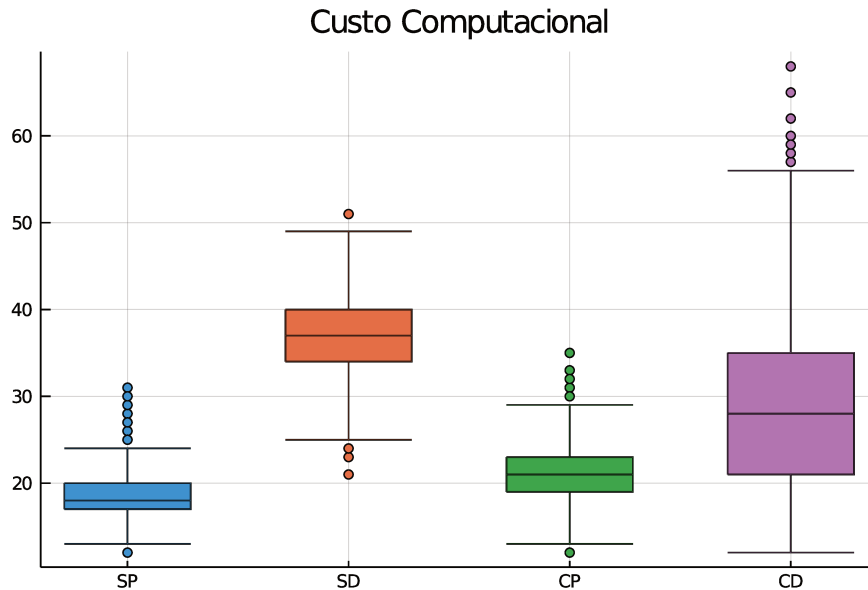


Figura 32 – Comparando o custo computacional das versões aplicadas no conjunto de quadriláteros.

E em seguida, apresentamos uma tabela para verificarmos a morosidade de cada versão para resolver o conjunto de quadriláteros.

Tabela 9 – Tempo e Alocação de memória das versões aplicadas no conjunto de quadriláteros.

Versões	Tempo	%	Média	Alocação	%	Média
CD	305s	46.0%	0.10s	34.4GiB	53.6%	0.0108GiB
SD	164s	24.7%	0.05s	14.6GiB	22.7%	0.0046GiB
CP	103s	15.5%	0.03s	8.05GiB	12.5%	0.0025GiB
SP	91.3s	13.8%	0.03s	7.19GiB	11.2%	0.0022GiB
<i>Total</i>	<i>663s</i>	<i>100%</i>	<i>0.21s</i>	<i>64.20GiB</i>	<i>100%</i>	<i>0.0201GiB</i>

4.4.1 Conclusão do Experimento 3

Conclusão: Com base nesses indicadores, podemos concluir que as versões SP e CP apresentam uma performance melhor com relação às outras. Apoiados neste resultado, bem como nos dos demais experimentos, iremos utilizar as versões SP e CP para resolvermos os problemas do próximo experimento.

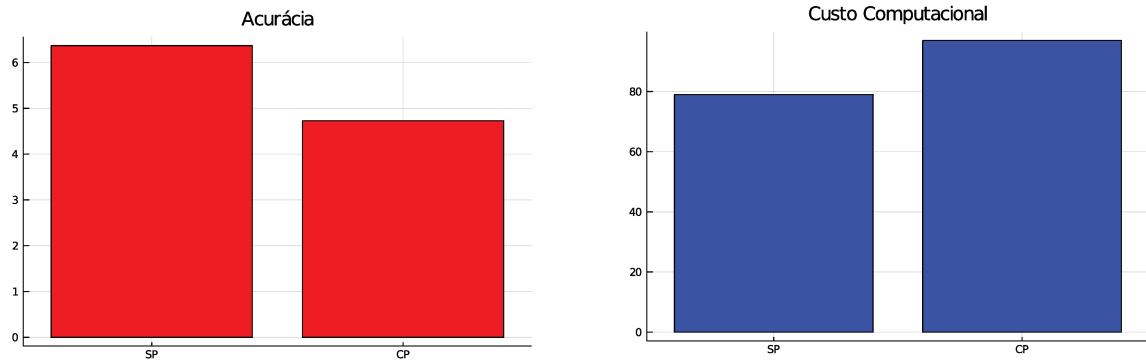
4.5 Experimento 4: Problemas de dimensão maiores

Objetivo: Analisar o desempenho das versões SP e CP quando a dimensão do problema aumenta. Para isso iremos resolver o Problema 5 e o Problema 6 apresentados no capítulo anterior.

Separamos o experimentos em duas partes: no primeiro analisamos o Problema 5 e, na segunda, o Problema 6. Em ambos os problemas utilizamos o mesmo ponto inicial utilizado por Facchinei e Lucidi em [12], que foi $y = \left(\frac{2}{3}, \frac{5}{3}, \frac{20}{9}, \frac{20}{9}, \frac{92}{27}, \frac{38}{27}\right)^T$ para o Problema 5 e $y = \left(\frac{2}{3}, \frac{5}{3}, \frac{17}{9}, \frac{11}{9}, \frac{80}{27}, \frac{-5}{54}, \frac{323}{81}, \frac{38}{81}, \frac{1214}{243}, \frac{362}{243}, \frac{8017}{1458}, \frac{605}{729}\right)^T$ para o Problema 6.

4.5.1 Problema 5

Primeiro, exibimos um gráfico para analisarmos a acurácia e o custo computacional das duas versões para o problema em \mathbb{R}^6 :



(a) Comparação da acurácia obtida pelas duas versões.

(b) Comparação do custo computacional obtido pelas duas versões.

Figura 33 – Acurácia e Custo Computacional das versões na resolução do Problema 5

Em seguida, apresentamos uma tabela, comparando a morosidade entre as duas versões para o problema em \mathbb{R}^6 :

Tabela 10 – Tempo e Alocação de memória das versões aplicadas no Problema 5.

Versões	Tempo	%	Alocação	%
CP	687ms	76.2%	110MiB	84.6%
SP	211ms	23.4%	20.1MiB	15.4%
<i>Total</i>	902ms	100%	131MiB	100%

E por último, acompanhamos as iterações de cada uma das versões para verificar o comportamento de cada uma das duas versões:

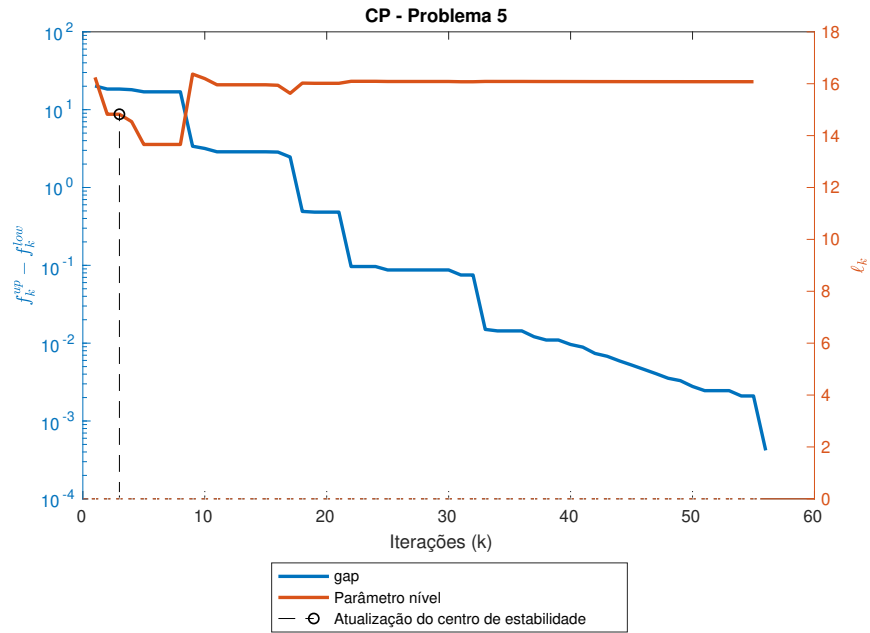


Figura 34 – Acompanhando a evolução do *gap*, do nível e atualização de centro de estabilidade na versão CP para o Problema 5.

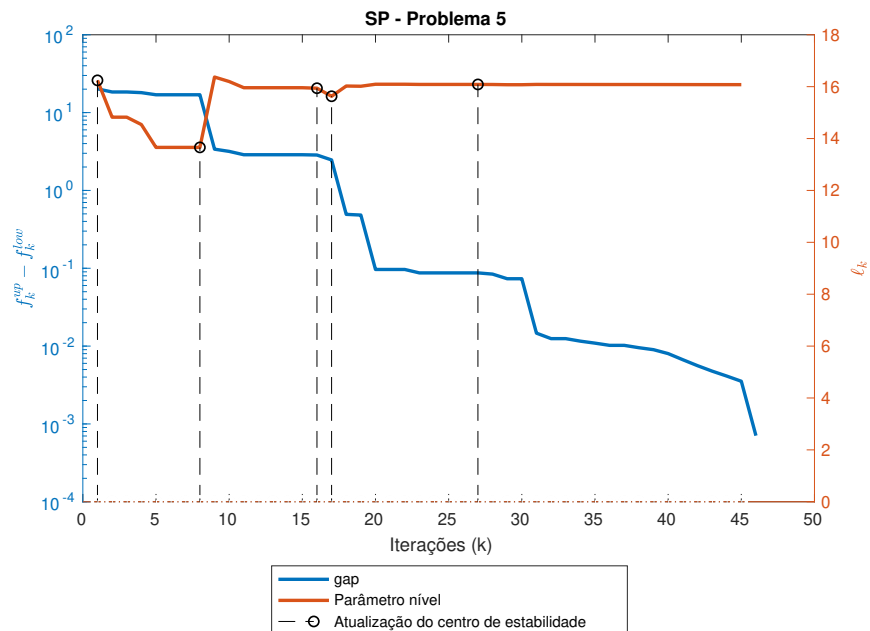
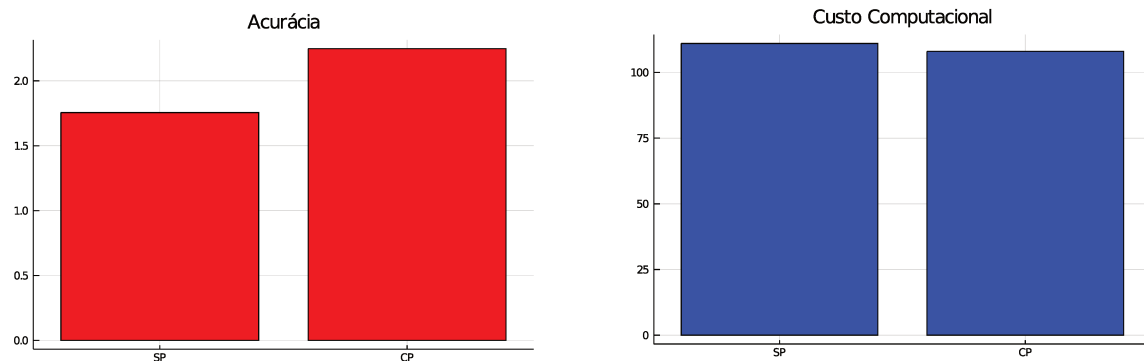


Figura 35 – Acompanhando a evolução do *gap*, do nível e da atualização de centro de estabilidade na versão SP para o Problema 5.

4.5.2 Problema 6

Primeiro, exibimos um gráfico para analisarmos a acurácia e o custo computacional das duas versões para o problema em \mathbb{R}^{12} :



(a) Comparação da acurácia obtida pelas duas versões.

(b) Comparação do custo computacional obtido pelas duas versões.

Figura 36 – Acurácia e Custo Computacional das versões na resolução do Problema 6

Em seguida, apresentamos uma tabela, comparando a morosidade entre as duas versões para o problema em \mathbb{R}^{12} :

Tabela 11 – Tempo e Alocação de memória das versões aplicado no Problema 6.

Versões	Tempo	%	Alocação	%
CP	620ms	57.7%	76.6MiB	61.0%
SP	454ms	42.3%	48.9MiB	39.0%
<i>Total</i>	1.07s	100%	125MiB	100%

E por último, acompanhamos as iterações de cada uma das versões para verificar o comportamento de cada uma das duas versões:

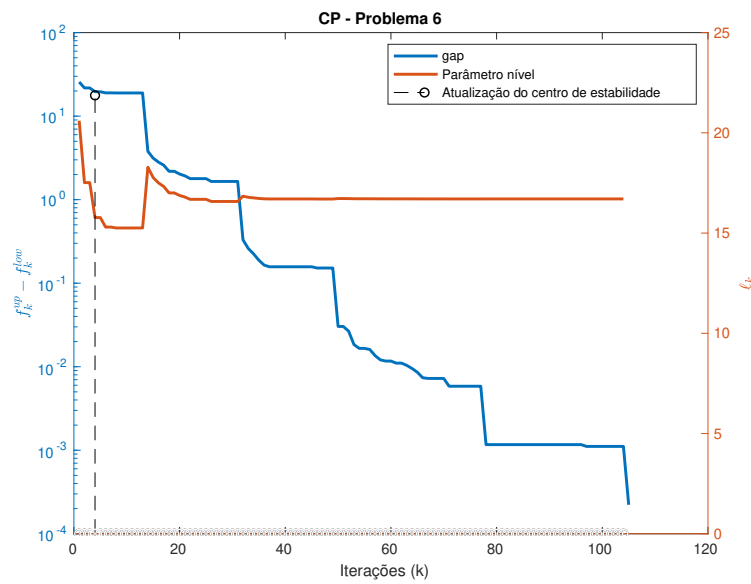


Figura 37 – Acompanhando a evolução do *gap*, do nível e da atualização de centro de estabilidade na versão CP para o Problema 6.

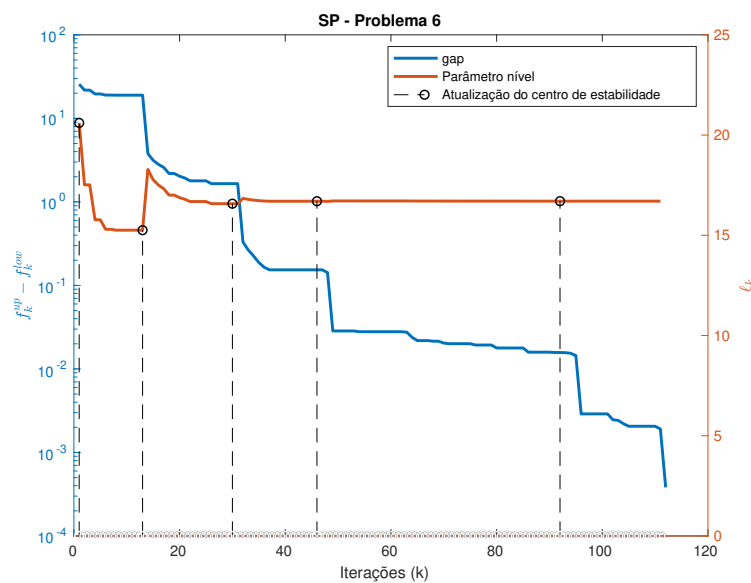


Figura 38 – Acompanhando a evolução do *gap*, do nível e atualização de centro de estabilidade na versão SP para o Problema 6.

4.5.3 Conclusão do Experimento 4

Conclusão: Com base nos indicadores apresentados, podemos concluir que conforme a dimensão do problema aumenta, o custo computacional acaba sendo similar em ambas as versões. Em contrapartida, a versão CP apresenta uma acurácia melhor para o problema de maior dimensão, enquanto que a versão SP apresenta uma acurácia melhor para o problema de menor dimensão.

5 Considerações Finais

Neste trabalho consideramos o problema irrestrito de minimização convexa, em que a função objetivo não é necessariamente suave. Nosso objetivo foi compreender o método de feixes para otimização convexa, implementá-lo em `Julia` e analisar os resultados computacionais em uma família de problemas com apelo geométrico, os problemas de Steiner. Fizemos quatro implementações para o Método de Feixes em Níveis nos beneficiando das vantagens da linguagem `Julia`, que para o problema estudado, sempre nos gerou um vetor numérico ao calcular um subgradiente.

Em preparação para nosso estudo, revisamos os conceitos fundamentais envolvidos, com destaque para convexidade, dualidade e as noções de subgradiente e ϵ -subgradiente. Para motivar a técnica de interesse nesse trabalho, nosso ponto de partida foi o método do subgradiente. Em seguida abordamos o método de planos cortantes, para então chegar aos métodos de feixes. Estudamos os aspectos teóricos subjacente à convergência para a variante Proximal. Finalmente, os aspectos computacionais relacionados à variante em Feixes, bem como os algoritmos foram estabelecidos, para então desenvolvermos a implementação computacional.

Nossa principal contribuição com esse trabalho foi o estudo comparativo entre quatro possíveis formas de implementação em `Julia` para o método de feixes baseado em conjuntos de nível. A análise dos experimentos nos apresenta indícios da eficiência, eficácia e robustez das versões implementadas.

Cabe destacar que nossa abordagem para o problema de Steiner teve uma perspectiva da otimização não suave, diferentemente dos estudos feitos com base em otimização combinatória que têm sido feitos extensivamente ao longo dos anos.

Por fim, uma possível extensão deste trabalho seria a análise de funções não convexas, neste caso o estudo iria explorar técnicas de otimização local. Assim como o algoritmo teórico proposto por [Bonnans et al.](#), nossos experimentos também utilizaram uma técnica de compressão que não explorava diretamente as peculiaridades da variante em níveis, conforme apresentada no trabalho [9]. Uma implementação específica também pode servir de motivação para trabalhos futuros.

Referências

- [1] IVO BALBAERT. *Getting started with Julia*. Community experience distilled. Packt Publishing, 2015. Citado na página 15.
- [2] MOKHTAR S. BAZARAA, HANIF D. SHERALI E CHITHARANJAN M. SHETTY. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, third edition, 2013. Citado na página 24.
- [3] NICOLAS BEHR, GIUSEPPE DATTOLI, AMBRA LATTANZI E SILVIA LICCIARDI. Dual Numbers and Operational Umbral Methods. *Axioms*, 8(3):77, 2019. Disponível em: <https://doi.org/10.3390/axioms8030077>. Citado na página 28.
- [4] DIMITRI PANTELI BERTSEKAS. *Convex optimization theory*. Athena Scientific optimization and computation series. Athena Scientific, Belmont, Massachusetts, 2009. Citado 2 vezes nas páginas 18 e 44.
- [5] JOSEPH-FRÉDÉRIC BONNANS, JEAN CHARLES GILBERT, CLAUDE LEMARÉCHAL E CLAUDIA ALEJANDRA SAGASTIZÁBAL. *Numerical Optimization Theoretical and Practical Aspects*. Universitext. Springer-Verlag Berlin Heidelberg, 2006. Disponível em: [10.1007/978-3-540-35447-5](https://doi.org/10.1007/978-3-540-35447-5). Citado 9 vezes nas páginas 15, 21, 31, 32, 44, 59, 63, 68 e 92.
- [6] MARCELO MARCEL CORDOVA. *Técnicas de otimização não-diferenciável para a resolução do problemas do comissionamento de unidades geradoras termelétricas*. Dissertação (Mestrado), Universidade Federal de Santa Catarina, 2014. Disponível em: <https://repositorio.ufsc.br/xmlui/handle/123456789/130963>. Citado na página 43.
- [7] VIRGÍNIA COSTA, MÁRCIA HELENA COSTA FAMPA E NELSON MACULAN. Um modelo matemático para o problema euclidiano de Steiner em \mathbb{R}^n . In *A Investigação Operacional em Portugal: Novos Desafios, Novas Ideias*, pages 145–158. IST Press, 2016. Citado na página 73.
- [8] RICHARD COURANT E HERBERT ROBBINS. *What Is Mathematics?* Oxford University Press, New York, 1 edition, 1941. Citado na página 72.
- [9] WELINGTON DE OLIVEIRA. Target radius methods for nonsmooth convex optimization. *Operations Research Letters*, 46(6):659–664, 2017. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167637717300238>. Citado 2 vezes nas páginas 68 e 92.

- [10] WELINGTON DE OLIVEIRA E MIKHAIL SOLODOV. A doubly stabilized bundle method for nonsmooth convex optimization. *Mathematical Programming*, 1-2(156):125–159, 2016. Disponível em: <https://doi.org/10.1007/s10107-015-0873-6>. Citado na página 43.
- [11] FLAVIA DONNO E GIANCARLO PESAMOSCA. On the Solution of the Generalized Steiner Problem by the Subgradient Method. *ZOR - Zeitschrift für Operations Research*, 34(5):335–352, 1990. Citado 5 vezes nas páginas 10, 73, 74, 75 e 78.
- [12] FRANCISCO FACCHINEI E STEFANO LUCIDI. Nonmonotone bundle-type scheme for convex nonsmooth minimization. *Journal of Optimization Theory and Applications*, 76(2):241–257, 1993. Disponível em: <https://doi.org/10.1007/BF00939607>. Citado 6 vezes nas páginas 10, 16, 73, 75, 78 e 88.
- [13] MÁRCIA HELENA COSTA FAMPA E NELSON MACULAN. Using a conic formulation for finding Steiner minimal trees. *Numerical Algorithms*, 35(2):315–330, 2004. Disponível em: <https://link.springer.com/article/10.1023/B:NUMA.0000021765.17831.bc>. Citado na página 73.
- [14] ANTONIO FRANGIONI. Generalized bundle methods. *SIAM Journal on Optimization*, 13(1):117–156, 2002. Disponível em: <https://doi.org/10.1137/S1052623498342186>. Citado na página 37.
- [15] ANTONIO FRANGIONI. Standard bundle methods: untrusted models and duality. In *Numerical Nonsmooth Optimization*, pages 61–226. Springer, 2020. Disponível em: https://doi.org/10.1007/978-3-030-34910-3_3. Citado 2 vezes nas páginas 37 e 59.
- [16] FRANK KWANG-MING HWANG, DANA RICHARDS E PAWEL WINTER. *The Steiner Tree Problem*, volume 53 of *Annals of Discrete Mathematics*. Elsevier, 1992. Disponível em: <https://www.sciencedirect.com/bookseries/annals-of-discrete-mathematics/vol/53/suppl/C>. Citado 2 vezes nas páginas 70 e 72.
- [17] FRANCK IUTZELER, JÉRÔME MALICK E WELINGTON DE OLIVEIRA. Asynchronous level bundle methods. *Mathematical Programming*, 2019. Disponível em: <https://doi.org/10.1007/s10107-019-01414-y>. Citado na página 15.
- [18] ALEXEY IZMAILOV E MIKHAIL SOLODOV. *Otimização*, volume 1. IMPA, 3ª edition, 2014. Citado 2 vezes nas páginas 22 e 24.
- [19] ALEXEY IZMAILOV E MIKHAIL SOLODOV. *Otimização*, volume 2. IMPA, 3ª edition, 2018. Citado na página 31.

- [20] CLAUDE LEMARÉCHAL, ARKADI NEMIROVSKI E YURII NESTEROV. New variants of bundle methods. *Mathematical Programming*, 69(1–3):111–147, 1995. Disponível em: <https://doi.org/10.1007/BF01585555>. Citado 2 vezes nas páginas 15 e 38.
- [21] NELSON MACULAN, PHILIPPE MICHELON E ADILSON ELIAS XAVIER. The Euclidean Steiner tree problem in R^n : A mathematical programming formulation. *Annals of Operations Research*, 96(1):209–220, 2000. Disponível em: <https://link.springer.com/article/10.1023/A:1018903619285>. Citado na página 73.
- [22] JEAN CARLOS A MEDEIROS E SANDRA AUGUSTA SANTOS. Estudo de um método baseado em autovalores generalizados para o subproblema de região de confiança. Relatório de pesquisa de iniciação científica, 2018. Disponível em: <https://www.ime.unicamp.br/pesquisa/relatorios/2019/1>. Citado 2 vezes nas páginas 39 e 42.
- [23] ROBERT MIFFLIN E CLAUDIA ALEJANDRA SAGASTIZÁBAL. A VU-algorithm for convex minimization. *Mathematical programming*, 104(2-3):583–608, 2005. Disponível em: <https://doi.org/10.1007/s10107-005-0630-3>. Citado na página 69.
- [24] JARRETT REVELS, MILES LUBIN E THEODORE PAPAMARKOU. Forward-mode Automatic Differentiation in Julia. *arXiv:1607.07892 [cs.MS]*, 2016. Disponível em: <https://arxiv.org/abs/1607.07892>. Citado na página 28.
- [25] ELIANE QUELHO FROTA REZENDE E MARIA LÚCIA BONTORIM DE QUEIROZ. *Geometria euclidiana plana e construções geométricas*. Editora da Unicamp, Campinas, SP, 2008. Citado na página 18.
- [26] PAWEL WINTER. Steiner problem in networks: A survey. *Networks*, 17(2):129–167, 1987. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/net.3230170203>. Citado na página 73.

APÊNDICE A – Glossário

Glossário

abscissa refere-se ao eixo horizontal, em um sistema cartesiano de coordenadas.

acurácia grande precisão entre um valor obtido experimentalmente e seu o valor verdadeiro.

AD do inglês *Automatic differentiation*, também chamada de diferenciação algorítmica, é um conjunto de técnicas para avaliar a derivada de função numericamente de forma exata.

cardinalidade é uma medida da quantidade de elementos de um conjunto. A cardinalidade de um conjunto X é usualmente denotada por $|X|$.

compacto conjunto fechado e limitado.

1) **compacidade** qualidade do conjunto que é dito compacto.

condição de otimalidade condição necessária e suficiente para caracterizar os minimizadores (maximizadores) de um problema de minimização (maximização).

condição de Slater é uma condição suficiente para Dualidade Forte, ou seja, os valores ótimos primal e dual são iguais. Em particular, se o problema primal satisfaz a condição de Slater, então a folga de dualidade é 0 e se o valor dual é finito, então ele é atingido.

Condições de Wolfe conjunto das condições de decréscimo suficiente e de curvatura.

1) **Condição de Armijo** desigualdade utilizada na busca linear inexata para encontrar um tamanho de passo α que forneça um decréscimo suficiente da função.

epígrafo conjunto de pontos que estão sobre o gráfico, ou acima dele.

folga de dualidade é a diferença entre o valor objetivo de uma solução primal e o valor objetivo de uma solução dual.

gradiente vetor das derivadas parciais da função.

hessiana matriz das derivadas parciais do gradiente função (ou das derivadas parciais de segunda ordem da função).

heurística diz-se de um algoritmo que fornece soluções sem uma demonstração formal de qualidade, usualmente avaliado de forma empírica em termos de complexidade e qualidade das soluções.

ínfimo o maior elemento entre as cotas inferiores.

inviável diz-se da solução que não satisfaz as restrições do problema. Quando um problema não possui uma solução viável, ele também é dito inviável.

1) **inviabilidade** quando se é inviável.

iteração é o processo de repetição de uma ou mais ações de um algoritmo.

1) **iterado** nome dado à variável que é atualizada a cada iteração de um método iterativo.

ponto corrente variável atualizada na iteração atual (corrente).

Lipschitz contínua uma função f é dita Lipschitz contínua se existir uma constante real $L > 0$ tal que para todo x e y no domínio da f têm-se:

$$|f(x) - f(y)| \leq L|x - y|,$$

1) **constante de Lipschitz** diz-se do ínfimo das constantes L que satisfazem a desigualdade de Lipschitz.

2) **localmente Lipschitz contínua** uma função f é dita localmente Lipschitz contínua se para cada ponto x do domínio, existe uma vizinhança de x em que f é Lipschitz contínua.

minimal diz-se de um elemento que não é maior que nenhum outro elemento do conjunto.

monótona diz-se de uma sequência crescente ou decrescente.

ordenada refere-se ao eixo vertical, em um sistema cartesiano de coordenadas.

poliedro no contexto de otimização, diz-se de um conjunto de pontos no espaço que satisfazem as restrições lineares de desigualdade de um problema.

1) **poliedral** que tem forma de poliedro.

robustez diz-se de um algoritmo que mantém um bom desempenho ainda que a dimensão do problema aumente.

secante reta que intercepta o gráfico da função em dois pontos quaisquer.

simplex generalização do conceito de triângulo a outras dimensões. É o invólucro convexo de $(n + 1)$ pontos independentes em \mathbb{R}^n .

soma telescópica são somas finitas nas quais pares de termos consecutivos se cancelam, deixando apenas os termos inicial e final. Uma clara referência ao telescópio, cuja a função é encurtar a distância entre nossos olhos e os corpos celestes.

superlinear diz-se de um algoritmo que converge com uma ordem p , em que $1 < p < 2$.

supremo o menor elemento entre as cotas superiores.

Teoria dos Grafos ramo da matemática que estuda os grafos, que são estruturas matemáticas, que podem ser denotadas por $G(V, E)$, em que V é um conjunto não vazio de pontos, também chamados de *vértices/nós* e E é um subconjunto de pares não ordenados de V , também chamados de *arestas*.

topologia neste contexto, refere-se à configuração dos vértices e arestas do problema.

viável diz-se da solução que satisfaz as restrições do problema. Quando um problema possui uma solução viável, ele também é dito viável.

1) **viabilidade** quando se é viável.