



CENTRO DE MASSA DE MATRIZES SIMÉTRICAS DEFINIDAS
POSITIVAS: DESCRIÇÃO DE UM ALGORITMO E SUA IMPLEMENTAÇÃO

Gleydson José Bianquini Couto

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Paulo Roberto Oliveira

Rio de Janeiro
Agosto de 2012

CENTRO DE MASSA DE MATRIZES SIMÉTRICAS DEFINIDAS
POSITIVAS:DESCRIÇÃO DE UM ALGORITMO E SUA IMPLEMENTAÇÃO

Gleydson José Bianchini Couto

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

Examinada por:

Prof. Paulo Roberto Oliveira, Dr.Ing.

Prof. Ronaldo Gregório, D.Sc.

Prof^ª. Nair Maria Maia de Abreu, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
AGOSTO DE 2012

Couto, Gleydson José Bianquini

Centro de Massa de Matrizes Simétricas Definidas Positivas: Descrição de um Algoritmo e sua Implementação/Gleydson José Bianquini Couto. – Rio de Janeiro: UFRJ/COPPE, 2012.

IX, 46 p.: il.; 29, 7cm.

Orientador: Paulo Roberto Oliveira

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2012.

Referências Bibliográficas: p. 45 – 46.

1. matrizes simétricas definidas positivas. 2. centro de massa riemanniano. 3. algoritmo de ponto proximal. 4. decomposição de schur. I. Oliveira, Paulo Roberto. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Agradecimentos

A Deus, pela saúde.

Ao professor Paulo Roberto Oliveira, por sua orientação.

Ao professor Ronaldo Gregório do Instituto Multidisciplinar de Nova Igaraçu (IM/UFRuralRJ), pelo imenso apoio prestado.

Aos professores do PESC, pela formação.

Ao professor Marcello Goulart Teixeira do DCC/IM/NCE/UFRJ, pelo curso introdutório de linguagem C.

Ao Programa de Engenharia Química (PEQ), em especial aos professores Arge-miro e Príamo, por fornecer espaço nas instalações do programa e a licença oficial do Matlab.

A Capes, pela bolsa concedida.

A minha esposa Daniele Machado Couto, por seu amor.

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

CENTRO DE MASSA DE MATRIZES SIMÉTRICAS DEFINIDAS
POSITIVAS: DESCRIÇÃO DE UM ALGORITMO E SUA IMPLEMENTAÇÃO

Gleydson José Bianquini Couto

Agosto/2012

Orientador: Paulo Roberto Oliveira

Programa: Engenharia de Sistemas e Computação

Apresenta-se neste trabalho, uma evolução do algoritmo de ponto proximal com decomposições de Schur, inicialmente proposto por GREGÓRIO e OLIVEIRA [1] em 2009, aplicado a resolução do problema de centro de massa riemannianno (PCMR) na variedade das matrizes simétricas definidas positivas. O método baseia-se na fatoração de uma matriz simétrica definida positiva X na forma $Q\Lambda Q^T$, onde Λ e Q são matrizes diagonal definida positiva e ortogonal, respectivamente. Em nossa versão, é proposta uma nova atualização para a matriz ortogonal Q , tendo em vista a estrutura riemanniana do grupo ortogonal \mathbb{O}^n . A solução da iteração principal, passa então a ser obtida em duas etapas iterativas e recursivas. Primeiro, fixamos uma matriz ortogonal e obtemos a solução de um problema não-linear riemanniano no conjunto das matrizes diagonais definidas positivas, que é isomorfo ao octante positivo de \mathbb{R}^n . Depois fixamos a solução obtida anteriormente e computamos a solução de um problema não-linear riemanniano no grupo ortogonal \mathbb{O}^n . Em ambas as etapas é empregado o método de Armijo generalizado para variedades riemannianas e elas são aplicadas recursivamente, nessa ordem, até que se obtenha a solução da iteração principal. Ao final, são apresentadas simulações computacionais para o algoritmo.

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

CENTER OF MASS OF SYMMETRIC POSITIVE DEFINITE MATRICES:
DESCRIPTION OF AN ALGORITHM AND ITS IMPLEMENTATION

Gleydson José Bianquini Couto

August/2012

Advisor: Paulo Roberto Oliveira

Department: Systems Engineering and Computer Science

It is presented in this work, an evolution of the proximal point algorithm with Schur decomposition, first proposed by GREGÓRIO and OLIVEIRA [1] in 2009, applied to solve the riemannian center of mass problem in the manifold of symmetric positive definite matrices. The method is based on factorization of a symmetric positive definite matrix X in the form $Q\Lambda Q^T$, where Λ and Q are diagonal positive definite and orthogonal matrices, respectively. In our version, we propose a new update to the orthogonal matrix Q that takes accounting the riemannian structure of the orthogonal group \mathbb{O}^n . The solution of the main iteration is obtained in two interactive and recursive steps. First, we fix an orthogonal matrix and we obtain the solution of a nonlinear riemannian problem in the set of diagonal positive definite matrices, which is isomorphic to the positive octant of \mathbb{R}^n . After we fix the solution obtained above and compute the solution of a nonlinear riemannian problem in the real orthogonal group \mathbb{O}^n . In both steps is employed the generalized armijo method on riemannian manifolds and they are applied recursively until the solution of the main iteration be computed. Finally, some computational simulations are presented for the algorithm.

Sumário

Lista de Figuras	viii
Lista de Tabelas	ix
1 Introdução	1
2 Conceitos preliminares	4
2.1 Alguns conceitos de geometria riemanniana	4
2.2 Alguns conceitos sobre matrizes	7
2.2.1 Exponencial e logaritmo de uma matriz	12
2.2.2 Espaço tangente à S_{++}^n : produto interno e norma	14
2.2.3 Distância e segmento geodésico em S_{++}^n	15
2.3 Algoritmo de Ponto Proximal no Cone das Matrizes Simétricas Defi- nidas Positivas	15
3 Problema de Centro de Massa e algoritmo proposto	17
3.1 Descrição do Problema de Centro de massa	17
3.2 Algoritmo para o PCMR	20
3.2.1 Cálculo de Λ_{j+1}^k	21
3.3 Nova atualização para Q_{j+1}^k e modificação do algoritmo proposto . . .	25
3.4 Estrutura riemanniana de \mathbb{O}^n	26
3.5 Método iterativo para computar Q_{j+1}^k	28
4 Implementação do algoritmo proposto <i>MSDPProx</i>	32
4.1 Descrição das variáveis e dos parâmetros	32
4.2 Experimentos numéricos	35
5 Conclusões e trabalhos futuros	44
Referências Bibliográficas	45

Lista de Figuras

4.1	Comportamento típico da norma de g , teste3.	42
4.2	Comportamento típico da norma de g , teste4.	42
4.3	Comportamento típico da norma do gradiente G , teste3.	43
4.4	Comportamento típico da norma do gradiente G , teste4.	43

Lista de Tabelas

4.1	Rotina1, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	35
4.2	Rotina1, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.5, \delta_0^k = 10^{-3}, \theta_3 = 0.1.$	36
4.3	Rotina2, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	36
4.4	Rotina2, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.5, \delta_0^k = 10^{-3}, \theta_3 = 0.1.$	36
4.5	Rotina3, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.5, \delta_0^k = 10^{-3}, \theta_3 = 0.1.$	37
4.6	Rotina4, $m = 10, n = 2, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	37
4.7	$m = 1000, n = 2, \beta = 10^{-6}, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	37
4.8	Rotina3, $m = 10, n = 3, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	38
4.9	Rotina3, $m = 10, n = 3, \epsilon^0 = 10^{-3}, \theta_2 = 0.5, \delta_0^k = 10^{-3}, \theta_3 = 0.8.$	38
4.10	Rotina4, $m = 10, n = 3, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	38
4.11	Rotina4, $m = 10, n = 3, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.9.$	39
4.12	$m = 1000, n = 3, \beta = 10^{-6}, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	39
4.13	Rotina3, $m = 10, n = 5, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	39
4.14	Rotina4, $m = 10, n = 5, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	40
4.15	Rotina4, $m = 10, n = 7, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	40
4.16	Rotina3, $m = 10, n = 10, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	40
4.17	Rotina3, $n = 10, \beta = 10^{-6}, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	41
4.18	Rotina4, $n = 10, \beta = 10^{-6}, \epsilon^0 = 10^{-3}, \theta_2 = 0.8, \delta_0^k = 10^{-3}, \theta_3 = 0.5.$	41

Capítulo 1

Introdução

Historicamente, como destacado em DO CARMO [2], a geometria riemanniana é um desenvolvimento natural da geometria diferencial das superfícies em \mathbb{R}^3 . Dessa forma, é possível transportar técnicas de otimização em \mathbb{R}^n , tais como método do gradiente, Newton, ponto proximal, dentre outros, para variedades riemannianas.

A otimização em *variedades de Hadamard* (variedades riemannianas completas, conexas e de curvatura seccional não-positiva) tem ganhado destaque devido ao difeomorfismo entre tais variedades e o \mathbb{R}^n (ver teorema de Holpf-Hinow e teorema de Hadamard em DO CARMO [2], SAKAI [3]). Um resultado importante diz respeito a unicidade dos segmentos geodésicos que unem dois quaisquer de seus pontos, fato esse similar ao caso euclideano em que os segmentos geodésicos são segmentos de reta (únicos).

A variedade das matrizes simétricas definidas positivas munida da métrica dada pela Hessiana da barreira logarítmica para o problema de programação semidefinida padrão

$$F(X) = -\ln \det(X) \tag{1.1}$$

representa um exemplo de variedade de Hadamard (ROTHAUS [4]).

Em FERREIRA e OLIVEIRA [5] é introduzido um método de ponto proximal com convergência global assegurada para determinar minimizadores de funções geodesicamente convexas em variedades de Hadamard.

A partir de MOAKHER [6], o problema de centro de massa riemanniano (PCMR) no conjunto das matrizes simétricas definidas positivas é definido por

$$\min_{X \in S_{++}^n} \left\{ \frac{1}{2} \sum_{i=1}^m d^2(X^i, X) \right\},$$

onde X^i ($i = 1, \dots, m$) são matrizes simétricas definidas positivas previamente estabelecidas e $d(.,.)$ é a distância riemanniana em S_{++}^n , (ROTHAUS [4]), dada pela

métrica definida pela hessiana de (1.1), definida por

$$d(X^i, X) = \left[\sum_{l=1}^n \ln^2 \theta_l(X^{i-\frac{1}{2}} X X^{i-\frac{1}{2}}) \right]^{\frac{1}{2}},$$

onde $\theta_l(A)$ é o l -ésimo autovalor de A .

Definimos a função objetivo do PCMR por

$$f(X) = \frac{1}{2} \sum_{i=1}^m d^2(X^i, X).$$

Tal função é geodesicamente convexa estrita e coerciva (ver lema 4.1 em FERREIRA e OLIVEIRA [5]), assim o PCMR possui única solução, que corresponde a uma matriz \bar{X} tal que o gradiente da função objetivo em \bar{X} se anula, ou seja, para que a solução do PCMR seja calculada exatamente, deve ser satisfeita a condição

$$\text{grad } f(\bar{X}) = 0.$$

No entanto, computando a solução por métodos iterativos, tal condição pode ser reescrita como

$$\|\text{grad } f(\bar{X})\| < \epsilon,$$

onde ϵ é uma precisão fixada.

O (PCMR) de matrizes simétricas definidas positivas é também conhecido como *média riemanniana de tensores*, cujas possíveis aplicações podem ser encontradas na teoria da elasticidade (COWIN e YANG [7]) e na difusão de imagens de ressonância magnética (BASSER *et al.* [8] e ARSIGNY *et al.* [9]).

Para problemas de otimização riemannianos, geodesicamente convexos, definidos na variedade das matrizes simétricas definidas positivas, GREGÓRIO e OLIVEIRA [1] propôs uma metodologia baseada em fatorações de Schur para resolução da iteração principal do método apresentado em FERREIRA e OLIVEIRA [5], resultando em um método denominado algoritmo de ponto proximal com decomposições de Schur.

Neste algoritmo, a idéia consiste em decompor uma matriz simétrica definida positiva X na forma $Q\Lambda Q^T$, onde Λ e Q são matrizes diagonal definida positiva e ortogonal, respectivamente. A resolução da iteração principal do algoritmo de ponto proximal é substituída então pela resolução de uma sucessão de subproblemas riemannianos cujas variáveis são matrizes diagonais definidas positivas.

Em nosso trabalho, adaptamos a versão inexata do algoritmo introduzido por GREGÓRIO e OLIVEIRA [1] à resolução do PCMR de matrizes simétricas definidas positivas. Além disso, uma nova atualização da matriz ortogonal Q é discutida,

visto que depois de computar a matriz diagonal definida positiva Λ (solução do sub-problema), GREGÓRIO e OLIVEIRA [1] estabelece que a variável Q é atualizada de maneira a preservar o valor da função objetivo regularizada.

A solução da iteração principal passa a ser obtida em duas etapas iterativas e recursivas. Na primeira etapa, é fixada uma matriz ortogonal Q e obtida a solução de um problema não-linear riemanniano na variedade das matrizes diagonais definidas positivas Ω_{++}^n , que é isomorfo ao octante positivo de \mathbb{R}^n , o que simplifica o cálculo das expressões da geodésica e do gradiente riemanniano.

Logo após, fixa-se a solução obtida anteriormente, ou seja a matriz diagonal definida positiva, e computa-se a solução de um novo problema não-linear riemanniano no grupo ortogonal \mathbb{O}^n , de forma que o valor da função objetivo regularizada tenha seu valor minimizado. Para tal fim, é discutida a estrutura riemanniana de \mathbb{O}^n (métrica, geodésica e gradiente riemanniano).

Em ambas as etapas é empregado o método de Armijo generalizado para variedades riemannianas, apresentado por YANG [10] e elas são aplicadas recursivamente até que a solução da iteração principal seja obtida.

No capítulo seguinte, são relatados alguns conceitos preliminares sobre geometria riemanniana e matrizes simétricas definidas positivas cujo objetivo é facilitar o entendimento do problema e do algoritmo apresentados no capítulo 3, onde estão presentes as principais idéias discutidas nesse trabalho. Em seguida, no capítulo 4, são abordadas rotinas para gerar aleatoriamente matrizes simétricas definidas positivas para os testes computacionais, cujos resultados são expostos em tabelas. E por fim, no último capítulo, alguns comentários sobre os resultados e sugestões de trabalhos futuros.

Capítulo 2

Conceitos preliminares

Neste capítulo serão abordados apenas alguns conceitos pertinentes à variedades riemannianas, matrizes e algoritmo de Ponto Proximal no Cone das Matrizes Simétricas Definidas Positivas, afim de facilitar a compreensão de nosso trabalho.

2.1 Alguns conceitos de geometria riemanniana

Nesta seção serão ressaltados apenas alguns elementos de geometria riemanniana, conforme feito em GREGÓRIO [11]. Para demonstrações de teoremas e maiores detalhes ver em SAKAI [3] ou DO CARMO [2].

Definição 2.1.1. Uma *variedade diferenciável* de dimensão n é um conjunto \mathbb{M} , não vazio, e uma família de aplicações biunívocas $\mathbf{x}_\alpha : \mathbb{U}_\alpha \subset \mathbb{R}^n \rightarrow \mathbb{M}$ de abertos \mathbb{U}_α de \mathbb{R}^n em \mathbb{M} tais que:

- (1) $\bigcup_{\alpha} \mathbf{x}_\alpha(\mathbb{U}_\alpha) = \mathbb{M}$.
- (2) Para todo par α, β com $\mathbf{x}_\alpha(\mathbb{U}_\alpha) \cap \mathbf{x}_\beta(\mathbb{U}_\beta) = \mathbb{W} \neq \emptyset$, os conjuntos $\mathbf{x}_\alpha^{-1}(\mathbb{W})$ e $\mathbf{x}_\beta^{-1}(\mathbb{W})$ são abertos em \mathbb{R}^n e as aplicações $\mathbf{x}_\beta^{-1} \circ \mathbf{x}_\alpha$ são diferenciáveis.
- (3) A família $\{(\mathbb{U}_\alpha, \mathbf{x}_\alpha)\}$ é máxima relativamente às condições (1) e (2).

Definição 2.1.2. Seja \mathbb{M} uma variedade diferenciável. Uma *aplicação diferenciável* $\gamma : (-\epsilon, \epsilon) \rightarrow \mathbb{M}$ é dita uma *curva* diferenciável em \mathbb{M} . Sejam $\gamma(0) = p \in \mathbb{M}$ e D o conjunto das funções de \mathbb{R}^n diferenciáveis em p . O *vetor tangente* à curva γ em $t = 0$ é a função $\gamma'(0) : D \rightarrow \mathbb{R}^n$ dada por

$$\gamma'(0) f = \left. \frac{d(f \circ \gamma)}{dt} \right|_{t=0}, \quad f \in D.$$

Um *vetor tangente* em p é o vetor tangente em $t = 0$ de alguma *curva* γ , com $\gamma(0) = p$. O conjunto de todos os vetores tangentes a \mathbb{M} em p é chamado de espaço tangente a \mathbb{M} em p e será denotado por $T_p\mathbb{M}$.

Definição 2.1.3. Uma métrica riemaninana em uma variedade diferenciável \mathbb{M} é uma correspondência que associa a cada ponto $p \in \mathbb{M}$ um produto interno $\langle \cdot, \cdot \rangle_p$ diferenciável em $T_p\mathbb{M}$. E $\|\bullet\|_p$ é a norma correspondente à $\langle \cdot, \cdot \rangle_p$ dada por $\|v\|_p = \langle v, v \rangle_p^{\frac{1}{2}}$, $v \in T_p\mathbb{M}$.

Uma variedade diferenciável \mathbb{M} munida de uma métrica Riemanniana é chamada *variedade Riemanniana*.

Conhecida a métrica riemanniana, podemos calcular o comprimento de uma curva diferenciável γ ligando dois pontos quaisquer $p, q \in \mathbb{M}$. Considere $\gamma : [t_1, t_2] \rightarrow \mathbb{M}$ satisfazendo $\gamma(t_1) = p$ e $\gamma(t_2) = q$. Assim, o comprimento riemanniano de γ é definido por

$$L(\gamma) = \int_{t_1}^{t_2} \|\gamma'(t)\|_{\gamma(t)} dt, \quad (2.1)$$

e a distância riemanniana d entre p e q defini-se como

$$d(p, q) = \inf_{\gamma \in C_{p,q}} \{L(\gamma)\}, \quad (2.2)$$

onde $C_{p,q}$ é o conjunto de todas as curvas diferenciáveis ligando p à q . De forma intuitiva, uma curva ligando p à q , tal que o comprimento desta curva seja menor ou igual à qualquer outra curva ligando p à q , é conhecida como *geodésica*.

Seja a aplicação exponencial $exp_p : T_p\mathbb{M} \rightarrow \mathbb{M}$, que associa a cada $v \in T_p\mathbb{M}$ a um ponto de \mathbb{M} . Geometricamente, $exp_p(v)$ é o ponto de \mathbb{M} obtido percorrendo um comprimento igual a $\|v\|$, a partir de p , sobre a geodésica que passa por p com velocidade igual a $\frac{v}{\|v\|}$.

Definição 2.1.4. Uma variedade \mathbb{M} é geodesicamente completa se para todo $p \in \mathbb{M}$, a aplicação exponencial exp_p está definida para todo $v \in T_p\mathbb{M}$, ou seja, se as geodésicas $\gamma(t)$ que partem de p estão definidas para todos os valores do parâmetro $t \in \mathbb{R}$.

Veremos nos teoremas 2.1.1 e 2.1.2 uma propriedade geométrica similar à do \mathbb{R}^n , que diz respeito a unicidade dos segmentos geodésicos ligando dois pontos quaisquer.

Teorema 2.1.1. (Holpf-Hinow) Seja \mathbb{M} uma variedade Riemanniana e seja $p \in \mathbb{M}$. As seguintes afirmações são equivalentes:

(1) exp_p está definida em todo o $T_p\mathbb{M}$.

- (2) Os limitados e fechados de \mathbb{M} são compactos.
- (3) \mathbb{M} é completa como espaço métrico.
- (4) \mathbb{M} é geodesicamente completa.
- (5) Existe uma sucessão de compactos $K_n \subset \mathbb{M}$, $K_n \subset \text{int } K_{n+1}$ e $\bigcup_n K_n = \mathbb{M}$, tais que se $q_n \notin K_n$ então $d(p, q_n) \rightarrow \infty$. Em que $\text{int } X$ indica o interior do conjunto X .

Além disso, cada uma das afirmações acima implica que

- (6) Para todo $q \in \mathbb{M}$ existe uma geodésica γ ligando p a q com $L(\gamma) = d(p, q)$.

Variedades Riemannianas completas, conexas e com curvatura seccional não-positiva são ditas *variedades de Hadamard*. O teorema a seguir segue como aplicação do teorema 2.1.1.

Teorema 2.1.2. Seja \mathbb{M} uma *variedades de Hadamard*. Então \mathbb{M} é difeomorfa ao espaço euclidiano \mathbb{R}^n , $n = \dim \mathbb{R}^n$. Assim, para todo ponto $p \in \mathbb{M}$, aplicação exponencial \exp_p é um difeomorfismo.

Cabe ressaltar a relevância deste teorema, pois se \mathbb{M} é uma variedade de Hadamard, então \mathbb{M} tem a mesma topologia e estrutura diferenciável do espaço euclidiano \mathbb{R}^n . De forma similar ao \mathbb{R}^n , a propriedade geométrica que garante a unicidades dos segmentos geodésicos ligando dois pontos quaisquer de \mathbb{M} é preservada. Esse resultado é válido para a variedade das matrizes simétricas definidas positivas, munido da métrica dada pela hessiana de (1.1).

Definição 2.1.5. Um subconjunto S de uma *variedade de Hadamard* \mathbb{M} é dito geodesicamente convexo se para quaisquer par de pontos p e q em S , o único segmento geodésico que os une está contido em S .

Definição 2.1.6. $f : S \rightarrow \mathbb{R}$ é dita geodesicamente convexa (estrita) se a restrição de f a qualquer segmento geodésico de S é uma função convexa (estrita).

Em outras palavras, a definição anterior assume que se γ é um segmento geodésico conectando $p, q \in S$, isto é, $\gamma(0) = p$, $\gamma(1) = q$ e $\gamma(t) \in S$, para todo $t \in (0, 1)$, então f é convexa se e somente se $f(\gamma(1 - \lambda)) \leq \lambda f(\gamma(0)) + (1 - \lambda) f(\gamma(1))$. Além disso, para que f seja geodesicamente convexa estrita é necessário e suficiente que a desigualdade seja estrita.

Teorema 2.1.3. Seja \mathbb{M} uma *variedade de Hadamard* e $p \in \mathbb{M}$. A função $\rho_p : \mathbb{M} \mapsto \mathbb{R}_+$, definida por

$$\rho_p(q) = \frac{1}{2}d^2(p, q), \quad (2.3)$$

é geodesicamente convexa estrita e seu gradiente é dado por $\text{grad } \rho_p(q) = -\exp_q^{-1}p$, onde $\exp_q^{-1}p$ é o vetor tangente à geodésica que conecta q a p , em q .

2.2 Alguns conceitos sobre matrizes

A seguir são colocadas algumas definições e teoremas sobre os tipos de matrizes que serão trabalhadas ao definir e resolver o problema de centro de massa riemanniano (*PCMR*). A partir de LEON [12], são destacados conceitos dentro do contexto de matrizes reais $\mathbb{R}^{m \times n}$, de modo que fiquem adaptados ao algoritmo apresentado.

Definição 2.2.1. No espaço vetorial $\mathbb{R}^{m \times n}$, podemos introduzir o seguinte produto interno

$$\langle A, B \rangle_F = \sum_{i=1}^m \sum_{j=1}^n a_{ij}b_{ij},$$

onde $A, B \in \mathbb{R}^{m \times n}$, denominado *produto interno de Frobenius*, que pode ser expresso ainda como

$$\langle A, B \rangle_F = \text{Tr}(B^T A). \quad (2.4)$$

Definição 2.2.2. Para o espaço vetorial $\mathbb{R}^{m \times n}$, a norma definida pelo produto interno (2.4) é chamada de *norma de Frobenius* e é denotada por $\|\bullet\|_F$. Logo, se $A \in \mathbb{R}^{m \times n}$,

$$\|A\|_F = (\langle A, A \rangle_F)^{\frac{1}{2}} = \sqrt{\text{Tr}(A^T A)}. \quad (2.5)$$

A seguir relembremos o conceito de autovalores, assim como o autovetor associado.

Definição 2.2.3. Dada a equação matricial

$$A\mathbf{x} = \lambda\mathbf{x}, \quad (2.6)$$

se a equação (2.6) tiver solução não-trivial \mathbf{x} , dizemos que λ é um autovalor de A e \mathbf{x} , o autovetor associado a λ . Observe que a equação (2.6) pode ser reescrita como

$$(A - \lambda I)\mathbf{x} = \mathbf{0}, \quad (2.7)$$

onde podemos concluir que teremos solução não-trivial \mathbf{x} , se somente se, $\det(A - \lambda I) = 0$. Chamamos $p(\lambda) = \det(A - \lambda I)$ de polinômio característico, cujas

raízes são os autovalores de A . O núcleo da transformação linear $(A - \lambda I)$, denotado por $N(A - \lambda I) = \{\mathbf{x} \in \mathbb{R}^n | (A - \lambda I)\mathbf{x} = \mathbf{0}\}$, se denomina autoespaço associado a λ . Qualquer vetor não-nulo do conjunto $N(A - \lambda I)$ é um autovetor de A .

Definição 2.2.4. Uma matriz A , $n \times n$, é dita **diagonalizável** se existirem uma matriz invertível X e uma matriz diagonal D satisfazendo

$$X^{-1}AX = D.$$

Neste caso dizemos que X **diagonaliza** A .

Teorema 2.2.1. Uma matriz A , $n \times n$, é diagonalizável se e somente se A tem n autovetores linearmente independentes.

Demonstração. Suponha que A tem n autovetores linearmente independentes $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$. Seja λ_i o autovalor associado a \mathbf{x}_i ($i = 1, \dots, n$). Alguns λ_i 's podem ser iguais. Seja X a matriz cujo j -ésimo vetor coluna é o vetor \mathbf{x}_j ($j = 1, \dots, n$). Então $A\mathbf{x}_j = \lambda_j\mathbf{x}_j$ é o j -ésimo vetor coluna de AX . Logo,

$$\begin{aligned} AX &= (A\mathbf{x}_1, A\mathbf{x}_2, \dots, A\mathbf{x}_n) \\ &= (\lambda_1\mathbf{x}_1, \lambda_2\mathbf{x}_2, \dots, \lambda_n\mathbf{x}_n) \\ &= (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix} \\ &= XD. \end{aligned}$$

Como X tem n colunas linearmente independentes, X é invertível e, portanto,

$$XD = AX \Leftrightarrow X^{-1}XD = X^{-1}AX \Leftrightarrow D = X^{-1}AX.$$

Reciprocamente, suponha que A é diagonalizável. Então, existe uma matriz invertível X tal que $AX = XD$. Se $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ são os vetores colunas de X , temos $A\mathbf{x}_j = \lambda_j\mathbf{x}_j$, onde $\lambda_j = d_{jj}$ ($j = 1, \dots, n$). Logo, para cada j , λ_j é um autovalor de A com autovetor associado \mathbf{x}_j . Como as colunas de X são linearmente independentes, A tem n autovetores linearmente independentes. ■

Definição 2.2.5. Uma matriz A é dita **simétrica** se $A = A^t$.

Teorema 2.2.2. Todos os autovalores de uma matriz simétrica são reais. Além disso, autovetores associados a autovalores distintos são ortogonais.

Demonstração. Seja A uma matriz simétrica. Seja λ um autovalor de A e seja \mathbf{x} um autovetor associado. Se $\alpha = \mathbf{x}^t A \mathbf{x}$, então

$$\bar{\alpha} = \alpha^t = (\mathbf{x}^t A \mathbf{x})^t = \mathbf{x}^t A \mathbf{x} = \alpha,$$

logo α é real. Temos que

$$\alpha = \mathbf{x}^t A \mathbf{x} = \mathbf{x}^t \lambda \mathbf{x} = \lambda \|\mathbf{x}\|^2$$

e, portanto,

$$\lambda = \frac{\alpha}{\|\mathbf{x}\|^2}$$

é real. Se \mathbf{x}_1 e \mathbf{x}_2 são autovetores associados a autovalores distintos λ_1 e λ_2 , respectivamente, então

$$(A\mathbf{x}_1)^t \mathbf{x}_2 = \mathbf{x}_1^t A^t \mathbf{x}_2 = \mathbf{x}_1^t A \mathbf{x}_2 = \mathbf{x}_1^t \lambda_2 \mathbf{x}_2 = \lambda_2 \mathbf{x}_1^t \mathbf{x}_2$$

e

$$(A\mathbf{x}_1)^t \mathbf{x}_2 = (\mathbf{x}_2^t A \mathbf{x}_1)^t = (\mathbf{x}_2^t \lambda_1 \mathbf{x}_1)^t = (\lambda_1 \mathbf{x}_2^t \mathbf{x}_1)^t = \lambda_1 \mathbf{x}_1^t \mathbf{x}_2,$$

logo,

$$\lambda_1 \mathbf{x}_1^t \mathbf{x}_2 = \lambda_2 \mathbf{x}_1^t \mathbf{x}_2,$$

que implica em

$$\lambda_1 \mathbf{x}_1^t \mathbf{x}_2 - \lambda_2 \mathbf{x}_1^t \mathbf{x}_2 = (\lambda_1 - \lambda_2) \mathbf{x}_1^t \mathbf{x}_2 = 0$$

e, como $\lambda_1 \neq \lambda_2$, temos

$$\mathbf{x}_1^t \mathbf{x}_2 = \langle \mathbf{x}_1, \mathbf{x}_2 \rangle = 0.$$

■

Definição 2.2.6. Uma matriz Q , $n \times n$, é dita **ortogonal** se seus vetores colunas formam um conjunto ortonormal em \mathbb{R}^n .

Logo, Q é **ortogonal** se e somente se $Q^T Q = I$. Segue que

$$Q^t Q Q^{-1} = I Q^{-1} \Leftrightarrow Q^t = Q^{-1}.$$

Corolário 2.2.3. Se os autovalores de uma matriz simétrica A forem distintos, então existe uma matriz ortogonal Q que diagonaliza A .

Demonstração. Seja \mathbf{x}_i um autovetor associado a λ_i para cada autovalor λ_i de A . Seja $\mathbf{q}_i = \frac{1}{\|\mathbf{x}_i\|} \mathbf{x}_i$. Então, \mathbf{q}_i é um autovetor unitário associado a λ_i para cada i . Pelo teorema 2.2.2, $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n\}$ é um conjunto ortonormal em \mathbb{R}^n . Seja Q a matriz

cuja i -ésima coluna é o vetor \mathbf{q}_i para cada i , então Q é ortogonal e diagonaliza A . ■

Teorema 2.2.4 (Teorema de Schur). Se A é uma matriz $n \times n$, então existe uma matriz ortogonal Q tal que $Q^t A U$ é triangular superior.

Demonstração. A demonstração é por indução em n . O resultado é óbvio para $n = 1$. Suponha que a hipótese é válida para matrizes $k \times k$ e seja A uma matriz $(k+1) \times (k+1)$. Sejam λ_1 um autovalor de A e \mathbf{w}_1 um autovetor associado. Usando o processo de Gram-Schmidt¹, construa $\mathbf{w}_2, \dots, \mathbf{w}_{k+1}$ de modo que $\{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{k+1}\}$ seja uma base ortonormal para \mathbb{R}^{k+1} . Seja W a matriz cuja i -ésima coluna é o vetor \mathbf{w}_i ($i = 1, \dots, k+1$). Então, por construção, W é ortogonal. A primeira coluna de $W^t A W$ vai ser $W^t A \mathbf{w}_1$. De forma que

$$W^t A \mathbf{w}_1 = W^t \lambda_1 \mathbf{w}_1 = \lambda_1 W^t \mathbf{w}_1 = \lambda_1 \mathbf{e}_1.$$

Onde $\mathbf{e}_1 = (1, 0, 0, \dots, 0)^t$. Então $W^t A W$ é uma matriz da forma

$$\left(\begin{array}{c|ccc} \lambda_1 & \times & \dots & \times \\ \hline 0 & & & \\ \vdots & & M & \\ 0 & & & \end{array} \right)$$

onde M é uma matriz $k \times k$. Pela hipótese de indução, existe uma matriz $k \times k$ ortogonal V_1 tal que $V_1^t M V_1 = T_1$, onde T_1 é triangular superior. Seja

$$V = \left(\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline 0 & & & \\ \vdots & & V_1 & \\ 0 & & & \end{array} \right).$$

Então V é ortogonal e

¹Dada uma base de um espaço vetorial munido de produto interno, é um processo utilizado para encontrar uma base ortonormal. Ver detalhes em LEON [12].

$$\begin{aligned}
V^t(W^tAW)V &= \left(\begin{array}{c|ccc} \lambda_1 & \times & \dots & \times \\ \hline 0 & & & \\ \vdots & & & \\ 0 & & & \end{array} \right) \\
&= \left(\begin{array}{c|ccc} \lambda_1 & \times & \dots & \times \\ \hline 0 & & & \\ \vdots & & & \\ 0 & & & \end{array} \right) \\
&= T
\end{aligned}$$

Seja $Q = WV$. A matriz Q é ortogonal, já que

$$Q^tQ = (WV)^tWV = V^tW^tWV = I$$

e $Q^tAQ = T$. ■

A fatoração da matriz $A = QTQ^t$ é conhecida como a **decomposição de Schur** de A . No caso em que A é simétrica, a matriz T é diagonal.

Teorema 2.2.5 (Teorema Espectral). Se A é simétrica, então existe uma matriz ortogonal Q que diagonaliza A .

Demonstração. Pelo teorema 2.2.4, existe uma matriz ortogonal Q tal que $Q^tAQ = T$, onde T é triangular superior. Veja que,

$$T^t = (Q^tAQ)^t = Q^tA^tQ = Q^tAQ = T.$$

Logo, T é simétrica e, portanto, tem que ser diagonal. ■

Definição 2.2.7. Uma matriz A real simétrica $n \times n$ é dita

- i) **definida positiva** se $\mathbf{x}^tA\mathbf{x} > 0$ para todo $\mathbf{x} \in \mathbb{R}^n$ não nulo.
- ii) **definida negativa** se $\mathbf{x}^tA\mathbf{x} < 0$ para todo $\mathbf{x} \in \mathbb{R}^n$ não nulo.
- iii) **semidefinida positiva** se $\mathbf{x}^tA\mathbf{x} \geq 0$ para todo $\mathbf{x} \in \mathbb{R}^n$ não nulo.
- iv) **semidefinida negativa** se $\mathbf{x}^tA\mathbf{x} \leq 0$ para todo $\mathbf{x} \in \mathbb{R}^n$ não nulo.

Teorema 2.2.6. Seja uma matriz A real simétrica $n \times n$. Então, A é definida positiva se e somente se todos os seus autovalores são positivos.

Demonstração. Se A é definida positiva e se λ é um autovalor de A , então, para qualquer autovetor \mathbf{x} associado a λ ,

$$\mathbf{x}^t A \mathbf{x} = \mathbf{x}^t \lambda \mathbf{x} = \lambda \mathbf{x}^t \mathbf{x} = \lambda \|\mathbf{x}\|^2.$$

Logo,

$$\lambda = \frac{\mathbf{x}^t A \mathbf{x}}{\|\mathbf{x}\|^2} > 0.$$

Reciprocamente, suponha que os autovalores da matriz A são positivos. Como o produto de todos os autovalores é igual ao determinante de uma matriz, então o determinante de A é não-nulo, ou seja, matriz A é invertível. Seja $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ um conjunto ortonormal de autovetores de A . Se \mathbf{x} pode ser escrito como

$$\mathbf{x} = \alpha_1 \mathbf{x}_1 + \alpha_2 \mathbf{x}_2 + \dots + \alpha_n \mathbf{x}_n$$

onde

$$\alpha_i = \mathbf{x}^t \mathbf{x}_i \quad \text{para } i = 1, \dots, n \quad \text{e} \quad \sum_{i=1}^n (\alpha_i)^2 = \|\mathbf{x}\|^2 > 0.$$

Temos então, que

$$\begin{aligned} \mathbf{x}^t A \mathbf{x} &= (\alpha_1 \mathbf{x}_1 + \dots + \alpha_n \mathbf{x}_n)^t (\alpha_1 \lambda_1 \mathbf{x}_1 + \dots + \alpha_n \lambda_n \mathbf{x}_n) \\ &= \sum_{i=1}^n (\alpha_i)^2 \lambda_i \\ &\geq (\min \lambda_i) \|\mathbf{x}\|^2 > 0, \end{aligned}$$

e, portanto, A é definida positiva. ■

Denotemos por S_{++}^n o conjunto de matrizes simétricas definidas positivas, por S_+^n o conjunto de matrizes simétricas semidefinidas positivas e por S^n o conjunto de matrizes simétricas. De forma análoga, denotemos por Ω_{++}^n o conjunto de matrizes diagonais definidas positivas, por Ω_+^n o conjunto de matrizes diagonais semidefinidas positivas e por Ω^n o conjunto de matrizes diagonais.

2.2.1 Exponencial e logaritmo de uma matriz

Dado um escalar a , a exponencial e^a pode ser expressa como uma série de potências

$$e^a = 1 + a + \frac{1}{2!} a^2 + \frac{1}{3!} a^3 + \dots$$

Analogamente, para uma matriz A $n \times n$, podemos definir sua exponencial e^A

através de uma série de potências convergente

$$e^A = I + A + \frac{1}{2!}A^2 + \frac{1}{3!}A^3 + \dots \quad (2.8)$$

Para uma matriz diagonal

$$D = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}$$

é fácil calcular sua exponencial:

$$\begin{aligned} e^D &= \lim_{m \rightarrow \infty} \left(I + D + \frac{1}{2!}D^2 + \dots + \frac{1}{m!}D^m \right) \\ &= \lim_{m \rightarrow \infty} \begin{pmatrix} \sum_{k=1}^m \frac{1}{k!} \lambda_1^k & & & \\ & \ddots & & \\ & & \sum_{k=1}^m \frac{1}{k!} \lambda_n^k & \\ & & & \end{pmatrix} \\ &= \begin{pmatrix} e^{\lambda_1} & & & \\ & e^{\lambda_2} & & \\ & & \ddots & \\ & & & e^{\lambda_n} \end{pmatrix}. \end{aligned}$$

É mais difícil calcular a exponencial de uma matriz geral $A \ n \times n$. Se, no entanto, A for diagonalizável, então

$$A^k = X D^k X^{-1} \quad \text{para } k = 1, 2, \dots$$

e, portanto,

$$\begin{aligned} e^A &= X \left(I + D + \frac{1}{2!}D^2 + \frac{1}{3!}D^3 + \dots \right) X^{-1} \\ &= X e^D X^{-1}. \end{aligned}$$

De uma forma geral, adaptado de GOLUB e LOAN [13], se $A = QTQ^t$ é a decomposição de schur de A então

$$f(A) = Qf(T)Q^t. \quad (2.9)$$

No entanto, no caso de A ser diagonalizável, a equação (2.9) pode ser escrita como

$$f(A) = Xf(D)X^{-1}.$$

Em que $f(D) = \text{diag}(f(d_{11}), f(d_{22}), \dots, f(d_{nn}))$.

Sendo assim, o logaritmo natural de uma matriz A diagonalizável, denotado por $\ln(A)$, pode ser calculado por

$$\ln(A) = X\ln(D)X^{-1}.$$

2.2.2 Espaço tangente à S_{++}^n : produto interno e norma

De acordo com MOAKHER [6], o espaço tangente à S_{++}^n em X , denotado por $T_X S_{++}^n$, é o espaço de matrizes reais simétricas S^n , ou seja,

$$T_X S_{++}^n = S^n.$$

para todo $X \in S_{++}^n$.

Seja $X \in S_{++}^n$ um ponto fixo. A métrica riemanniana em $T_X S_{++}^n$, com respeito a hessiana de (1.1), é dada por

$$\langle S_1, S_2 \rangle_X = \langle F''(X) S_1, S_2 \rangle_F, \quad (2.10)$$

$S_1, S_2 \in T_X S_{++}^n$.

Em GREGÓRIO [11], na demonstração da proposição 2.3.1, analisando as diferenciais de

$$F(X) = -\ln \det(X),$$

conclui-se que F'' é a transformação linear que satisfaz

$$F''(X)H = X^{-1}HX^{-1}. \quad (2.11)$$

com $H \in S^n$. Segue ainda, que a relação inversa a relação dada em (2.11) é dada por

$$[F''(X)]^{-1}H = XHX. \quad (2.12)$$

A partir de (2.11) conclui-se que (2.10) pode ser reescrita como

$$\langle S_1, S_2 \rangle_X = \langle X^{-1}S_1X^{-1}, S_2 \rangle_F = \text{Tr} \{ X^{-1}S_1X^{-1}S_2 \},$$

onde $S_1, S_2 \in T_X S_{++}^n$. A norma associada de um vetor em $T_X S_{++}^n$, localmente

em X , é definida por

$$\|S\|_X = \sqrt{\langle S, S \rangle_X} = \sqrt{\langle F''(X)S, S \rangle_F} = \sqrt{\langle X^{-1}SX^{-1}, S \rangle_F} = \sqrt{\text{Tr}\{X^{-1}SX^{-1}S\}},$$

onde $S \in T_X S_{++}^n$.

2.2.3 Distância e segmento geodésico em S_{++}^n

Sejam $A, B \in S_{++}^n$, denote por $\theta_l(X)$ o l -ésimo autovalor de X , $l = 1, \dots, n$. De acordo com ROTHBAUS [4], a distância riemanniana $d(A, B)$ entre A e B , com respeito a métrica riemanniana definida pela hessiana da função barreira (1.1) é dada por

$$d(A, B) = \left[\sum_{l=1}^n \ln^2 \theta_l(A^{-\frac{1}{2}}BA^{-\frac{1}{2}}) \right]^{\frac{1}{2}} \quad (2.13)$$

e seja $\gamma : [0, 1] \rightarrow S_{++}^n$, de acordo com teorema 6.1 NESTEROV e TODD [14] o único segmento geodésico conectando A a B é dado por

$$\gamma(t) = A^{\frac{1}{2}} \left(A^{-\frac{1}{2}}BA^{-\frac{1}{2}} \right)^t A^{\frac{1}{2}}. \quad (2.14)$$

Note que, sendo γ o único segmento geodésico ligando A e B tal que $\gamma(0) = A$ e $\gamma(1) = B$,

$$\exp_A^{-1}B = \dot{\gamma}(0) = A^{\frac{1}{2}} \text{Ln} \left(A^{-\frac{1}{2}}BA^{-\frac{1}{2}} \right) A^{\frac{1}{2}},$$

onde $\dot{\gamma}(t)$ denota a derivada de γ com respeito a t .

2.3 Algoritmo de Ponto Proximal no Cone das Matrizes Simétricas Definidas Positivas

Como o objetivo deste trabalho é apresentar uma evolução do algoritmo de ponto proximal com decomposições de Schur, aplicado a resolução do problema de centro de massa riemanniano (PCMR) na variedade das matrizes simétricas definidas positivas, seguem a descrição do método de ponto proximal nesta variedade e o esquema de algoritmo, proposto por GREGÓRIO e OLIVEIRA [1], onde maiores detalhes e demonstrações podem ser encontrados.

Considere o problema

$$\min_{X \in S_{++}^n} \{f(X)\}, \quad (2.15)$$

onde f é uma função geodesicamente convexa em S_{++}^n .

Dados β^0 e $X^0 \in S_{++}^n$, a metodologia de ponto proximal consiste em gerar uma

seqüência de minimizadores $(X^k)_{k \in N}$ definida por

$$X^{k+1} = \underset{Y \in S_{++}^n}{\operatorname{argmin}} \left\{ f(Y) + \frac{\beta^k}{2} \sum_{l=1}^n \ln^2 \theta_l \left(X^{k-\frac{1}{2}} Y X^{k-\frac{1}{2}} \right) \right\}, \quad (2.16)$$

com β^k satisfazendo

$$\sum_{k=0}^{\infty} \frac{1}{\beta^k} = +\infty. \quad (2.17)$$

Em FERREIRA e OLIVEIRA [5], a convergência global do algoritmo de ponto proximal é assegurada para a classe de funções geodesicamente convexas, com respeito à métrica (2.13).

Baseado na metodologia de ponto proximal apresentada em FERREIRA e OLIVEIRA [5], GREGÓRIO e OLIVEIRA [1] proporam o seguinte esquema de algoritmo

Algoritmo de Ponto Proximal (versão inexata). Dados $\beta^0, \epsilon^0 > 0$ e $X^0 \succ 0$;

$k \rightarrow 0$;

Passo 1 **enquanto** $0 \notin \partial f(X^k)$ **faça**

Escolha $\Lambda_0^k \in \Omega_{++}^n$, $Q_0^k \in \mathbb{O}^n$;

$Y_0^k = (X^k)^{\frac{1}{2}} Q_0^k \Lambda_0^k Q_0^{kT} (X^k)^{\frac{1}{2}}$;

$j \rightarrow 0$;

Passo 1.1 **enquanto** $\beta^k \exp_{Y_j^k}^{-1} X^k \notin \partial_{\epsilon^k} f(Y_j^k)$ **faça**

$\bar{\Lambda}_{j+1}^k = \underset{\Lambda \in \Omega_{++}^n}{\operatorname{argmin}} \{ \phi_j^k(\Lambda) + \beta^k \rho_I(\Lambda) \}$;

$[\Lambda_{j+1}^k, Q_{j+1}^k] = \mathbf{Schur} \left(Q_j^k \bar{\Lambda}_{j+1}^k Q_j^{kT} \right)$;

$Y_{j+1}^k = (X^k)^{\frac{1}{2}} Q_{j+1}^k \Lambda_{j+1}^k Q_{j+1}^{kT} (X^k)^{\frac{1}{2}}$;

atualizar j e **retornar** para o passo 1.1;

fim;

$X^{k+1} = Y_j^k$;

atualizar β^k, ϵ^k, k e **retornar** para o passo 1;

fim.

Onde $\partial f(X)$ representa o *subdiferencial* de f em X e se f é diferenciável, pode ser escrito como $\operatorname{grad} f(X)$.

Capítulo 3

Problema de Centro de Massa e algoritmo proposto

3.1 Descrição do Problema de Centro de massa

Como descrito em MOAKHER [6], o problema de centro de massa riemanianno (PCMR) no conjunto das matrizes simétricas definidas positivas é definido por

$$\min_{X \in S_{++}^n} \left\{ \sum_{i=1}^m d^2(X^i, X) \right\}, \quad (3.1)$$

onde X^i ($i = 1, \dots, m$) são matrizes simétricas definidas positivas previamente estabelecidas e $d(\cdot, \cdot)$ é a distância riemanniana em S_{++}^n , definida em (2.13).

Seja a função objetivo do PCMR é definida por

$$f(X) = \frac{1}{2} \sum_{i=1}^m d^2(X^i, X),$$

que é a adição de m funções $f^i : S_{++}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, dadas por

$$f^i(X) = \frac{1}{2} d^2(X^i, X). \quad (3.2)$$

Pelo teorema 2.1.3, a função objetivo do PCMR é geodesicamente convexa estrita. Como tal função é coerciva (ver lema 4.1 em FERREIRA e OLIVEIRA [5]) o PCMR possui única solução.

Seja $\gamma^i : [0, 1] \rightarrow S_{++}^n$ o segmento geodésico ligando X a X^i em S_{++}^n . De acordo com teorema 6.1 NESTEROV e TODD [14] o único segmento geodésico conectando X a X^i é

$$\gamma^i(t) = X^{\frac{1}{2}} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right)^t X^{\frac{1}{2}}, \quad (3.3)$$

sejam Q^i e Λ^i matrizes ortogonal e diagonal, respectivamente, tal que

$$X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}} = Q^i\Lambda^iQ^{iT} \quad (3.4)$$

então da equação anterior,

$$\left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right)^t = Q^i\Lambda^{it}Q^{iT}.$$

Temos que γ^i é diferenciável e denotando por $\dot{\gamma}^i(t)$ sua derivada com respeito a t , vem que

$$\dot{\gamma}^i(t) = X^{\frac{1}{2}} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right)^t \text{Ln} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right) X^{\frac{1}{2}}.$$

De forma similar, a partir da equação (3.4), temos que

$$\text{Ln} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right) = Q^i \text{Ln} \Lambda^i Q^{iT},$$

onde $\text{Ln} \Lambda^i$ é a matriz diagonal cujos elementos da diagonal são os logaritmos naturais dos elementos da diagonal da matriz Λ^i .

Em geral, para uma matriz real simétrica S , a geodésica em S_{++}^n , que tem um ponto inicial $X \in S_{++}^n$ e velocidade S , em X , é definida por $\gamma(t) = \exp_X(tS)$, cuja forma explícita é apresentada por MOAKHER [6] como

$$\gamma(t) = X^{\frac{1}{2}} e^{tX^{-\frac{1}{2}}SX^{-\frac{1}{2}}} X^{\frac{1}{2}}. \quad (3.5)$$

SAKAI [3] estabelece que as funções f^i , $i = 1, \dots, m$ descritas em (3.2) são diferenciáveis e seus gradientes naturais, em $T_X S_{++}^n$, são dados por

$$\text{grad} f^i(X) = -\exp_X^{-1} X^i, \quad (3.6)$$

onde $\exp_X^{-1} X^i$ é o vetor tangente ao segmento geodésico γ^i , cuja expressão é dada em (3.3), em $t = 0$, dado por

$$\exp_X^{-1} X^i = \dot{\gamma}^i(0) = X^{\frac{1}{2}} \text{Ln} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right) X^{\frac{1}{2}}.$$

As observações acima implicam que

$$\text{grad} f^i(X) = -X^{\frac{1}{2}} \text{Ln} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right) X^{\frac{1}{2}},$$

que

$$\text{grad} f(X) = -\sum_{i=1}^m X^{\frac{1}{2}} \text{Ln} \left(X^{-\frac{1}{2}}X^iX^{-\frac{1}{2}}\right) X^{\frac{1}{2}} \quad (3.7)$$

e ainda que a solução \bar{X} do (PCMR) é caracterizada por

$$\text{grad } f(\bar{X}) = - \sum_{i=1}^m \bar{X}^{\frac{1}{2}} \text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) \bar{X}^{\frac{1}{2}} = 0. \quad (3.8)$$

Desde que

$$\sum_{i=1}^m \bar{X}^{\frac{1}{2}} \text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) \bar{X}^{\frac{1}{2}} = \bar{X}^{\frac{1}{2}} \sum_{i=1}^m \left[\text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) \right] \bar{X}^{\frac{1}{2}}$$

e $\bar{X}^{\frac{1}{2}}$ é não singular, a equação (3.8) pode ser reescrita como

$$\sum_{i=1}^m \text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) = 0.$$

Para que a solução do (PCMR) seja calculada exatamente, deve ser satisfeita a condição (3.8). No entanto, computando a solução por métodos iterativos, a condição (3.8) pode ser reescrita como

$$\|\text{grad } f(\bar{X})\| = \|\text{grad } f\|_{\bar{X}} < \epsilon, \quad (3.9)$$

onde ϵ é uma precisão fixada.

Uma vez que $\text{grad } f$ é calculado localmente em $X \in S_{++}^n$, então sua norma é calculada localmente por

$$\begin{aligned} \|\text{grad } f\|_X &\stackrel{(1)}{=} \sqrt{\text{Tr} \{X^{-1} [\text{grad } f(X)] X^{-1} [\text{grad } f(X)]\}} \\ &\stackrel{(2)}{=} \sqrt{\text{Tr} \left\{ X^{-1} \left[- \sum_{i=1}^m X^{\frac{1}{2}} \text{Ln} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right) X^{\frac{1}{2}} \right] X^{-1} \left[- \sum_{i=1}^m X^{\frac{1}{2}} \text{Ln} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right) X^{\frac{1}{2}} \right] \right\}} \\ &\stackrel{(3)}{=} \sqrt{\text{Tr} \left\{ \left[\sum_{i=1}^m \text{Ln} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right) \right] \left[\sum_{i=1}^m \text{Ln} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right) \right] \right\}} \\ &\stackrel{(4)}{=} \left\| \sum_{i=1}^m \text{Ln} \left(X^{-\frac{1}{2}} X^i X^{-\frac{1}{2}} \right) \right\|_F \end{aligned}$$

A igualdade (1) segue como descrito na equação (2.2.2), a igualdade (2) segue da substituição de $\text{grad } f$ dado em (3.7), a igualdade (3) segue da propriedade do traço, $\text{Tr}(AB) = \text{Tr}(BA)$, e a igualdade (4) segue da norma de Frobenius, dada em (2.5).

Desde que

$$\|\text{grad } f\|_{\bar{X}} = \left\| \sum_{i=1}^m \text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) \right\|_F,$$

a condição (3.9) pode ser reescrita como

$$\left\| \sum_{i=1}^m \text{Ln} \left(\bar{X}^{-\frac{1}{2}} X^i \bar{X}^{-\frac{1}{2}} \right) \right\|_F < \epsilon.$$

3.2 Algoritmo para o PCMR

De acordo com o que foi discutido, a versão inexata do algoritmo inicialmente proposto por GREGÓRIO e OLIVEIRA [1] pode ser reescrito como:

<p>Algoritmo <i>SDPProx</i> versão inexata. Dados $\epsilon, \beta^0, \epsilon^0 > 0$ e $X^0 \succ 0$;</p> <p>$k \rightarrow 0$;</p> <p>Passo 1 enquanto $\left\ \sum_{i=1}^m \text{Ln} \left(X^{k-\frac{1}{2}} X^i X^{k-\frac{1}{2}} \right) \right\ _F > \epsilon$ faça</p> <p style="padding-left: 20px;">Dados $\Lambda_0^k \in \Omega_{++}^n, Q_0^k \in \mathbb{O}^n$;</p> <p style="padding-left: 20px;">$Y_0^k = (X^k)^{\frac{1}{2}} Q_0^k \Lambda_0^k Q_0^{kT} (X^k)^{\frac{1}{2}}$;</p> <p style="padding-left: 20px;">$j \rightarrow 0$;</p> <p style="padding-left: 20px;">Passo 1.1 enquanto $\left\ \sum_{i=1}^m \text{Ln} \left(Y_j^{k-\frac{1}{2}} X^i Y_j^{k-\frac{1}{2}} \right) + \beta^k \text{Ln} \left(Y_j^{k-\frac{1}{2}} X^k Y_j^{k-\frac{1}{2}} \right) \right\ _F > \epsilon^k$ faça</p> <p style="padding-left: 40px;">Passo 1.1.1 calcular $\bar{\Lambda}_{j+1}^k = \underset{\Lambda \in \Omega_{++}^n}{\text{argmin}} \{ \phi_j^k(\Lambda) + \beta^k \rho_I(\Lambda) \}$;</p> <p style="padding-left: 40px;">$[\Lambda_{j+1}^k, Q_{j+1}^k] = \text{Schur} \left(Q_j^k \bar{\Lambda}_{j+1}^k Q_j^{kT} \right)$;</p> <p style="padding-left: 40px;">$Y_{j+1}^k = (X^k)^{\frac{1}{2}} Q_{j+1}^k \Lambda_{j+1}^k Q_{j+1}^{kT} (X^k)^{\frac{1}{2}}$;</p> <p style="padding-left: 40px;">atualizar j e retornar para o passo 1.1;</p> <p style="padding-left: 20px;">fim;</p> <p style="padding-left: 20px;">$X^{k+1} = Y_j^k$;</p> <p style="padding-left: 20px;">atualizar β^k, ϵ^k, k e retornar para o passo 1;</p> <p>fim.</p>
--

O teorema 2 em GREGÓRIO e OLIVEIRA [1] estabelece que a sequência gerada pelo algoritmo converge para o mínimo de f se a hipótese de convexidade é garantida, com respeito à métrica Riemanniana dada pela hessiana de (1.1), e as sequências $\{\beta^k\}_{k \in \mathbb{N}}$ e $\{\epsilon^k\}_{k \in \mathbb{N}}$ satisfazem

$$\sum_{k=0}^{\infty} \frac{1}{\beta^k} = +\infty, \sum_{k=0}^{\infty} \epsilon^k < +\infty \quad \text{e} \quad \sum_{k=0}^{\infty} \frac{\epsilon^k}{\beta^k} < +\infty. \quad (3.10)$$

Para atender estas condições, basta tomar por exemplo a sequência $\{\beta^k\}_{k \in \mathbb{N}}$ constante e a série $\sum_{k=0}^{\infty} \epsilon^k$ como uma série geométrica convergente.

3.2.1 Cálculo de Λ_{j+1}^k

Note que a função f^i , $i = 1, \dots, m$, definida em (3.2), é dada explicitamente por

$$f^i(X) = \frac{1}{2} \sum_{l=1}^n \ln^2 \theta_l \left(X^{i-\frac{1}{2}} X X^{i-\frac{1}{2}} \right).$$

Defina $\phi_j^{k_i} : \Omega_{++}^n \mapsto \mathbb{R}$ por

$$\phi_j^{k_i}(\Lambda) = f^i \left(X^{k\frac{1}{2}} Q_j^k \Lambda Q_j^{kT} X^{k\frac{1}{2}} \right). \quad (3.11)$$

Assim, de acordo com (3.2), a equação (3.11) pode ser reescrita como

$$\phi_j^{k_i}(\Lambda) = \frac{1}{2} d^2 \left(X^i, X^{k\frac{1}{2}} Q_j^k \Lambda Q_j^{kT} X^{k\frac{1}{2}} \right),$$

ou ainda, de forma explícita, como

$$\phi_j^{k_i}(\Lambda) = \frac{1}{2} \sum_{l=1}^n \ln^2 \theta_l \left(X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q_j^k \Lambda Q_j^{kT} X^{k\frac{1}{2}} X^{i-\frac{1}{2}} \right).$$

Denote por $W_j^{k_i}$ a matriz $X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q_j^k$. Assim, note que $W_j^{k_i T} = Q_j^{kT} X^{k\frac{1}{2}} X^{i-\frac{1}{2}}$, com $W_j^{k_i}$ não-singular (invertível) e $W_j^{k_i} \Lambda W_j^{k_i T}$, simétrica. Pode-se então escrever a função $\phi_j^{k_i}$ como

$$\phi_j^{k_i}(\Lambda) = \frac{1}{2} \sum_{l=1}^n \ln^2 \theta_l \left(W_j^{k_i} \Lambda W_j^{k_i T} \right).$$

De acordo com a definição 4.5.4 em HORN e JOHNSON [15], as matrizes Λ e $W_j^{k_i} \Lambda W_j^{k_i T}$ são \mathbb{T} congruentes. O teorema espectral para matrizes simétricas indica que existe uma matriz $Q_j^{k_i}$ ortogonal tal que $Q_j^{k_i T} W_j^{k_i} \Lambda W_j^{k_i T} Q_j^{k_i} = \Sigma_j^{k_i}$, onde $\Sigma_j^{k_i}$ é uma matriz diagonal com autovalores de $W_j^{k_i} \Lambda W_j^{k_i T}$ em sua diagonal. Desde que a matriz $W_j^{k_i}$ é fixada, $\Sigma_j^{k_i}$ é uma aplicação contínua que depende de Λ , indicado por

$$\Sigma_j^{k_i} = \Sigma_j^{k_i}(\Lambda).$$

De acordo com RINEHART [16], $\Sigma_j^{k_i}$ e $Q_j^{k_i}$ são aplicações diferenciáveis, com respeito a $W_j^{k_i} \Lambda W_j^{k_i T}$ desde que todos os autovalores de $W_j^{k_i} \Lambda W_j^{k_i T}$ sejam distintos, para toda $\Lambda \in \Omega_{++}^n$. Como $W_j^{k_i} \Lambda W_j^{k_i T}$ é uma aplicação que depende de Λ , temos que

$$\Sigma_{ll}^{k_{ij}} = \Sigma_{ll}^{k_{ij}}(\Lambda_{11}, \dots, \Lambda_{nn}).$$

Se $W_j^{k_i} \Lambda W_j^{k_i T}$ tem algum autovalor $\Sigma_{qq}^{k_{ij}}$, $1 \leq q \leq n$, com multiplicidade ≥ 2 , para algum $\Lambda \in \Omega_{++}^n$, então a diferenciabilidade desse autovalor, com respeito a Λ , não é

garantida. Entretanto, SUN e SUN [17] mostra que autovalores de matrizes simétricas positivas são funções semidiferenciáveis dos elementos da matriz.

Por outro lado, existe uma correspondência um a um entre os elementos de Ω_{++}^n e os elementos de \mathbb{R}_{++}^n ,

$$\begin{aligned}\Omega_{++}^n &\longmapsto \mathbb{R}_{++}^n \\ \Lambda &\longmapsto \lambda = \text{diag}(\Lambda),\end{aligned}$$

onde $\text{diag}(D)$ é o vetor cuja i -ésima componente é o elemento D_{ii} da matriz diagonal D .

Agora, podemos trabalhar com $\phi_j^{k_i} : \mathbb{R}_{++}^n \longmapsto \mathbb{R}$. Isto simplifica o cálculo do gradiente euclidiano de $\phi_j^{k_i}$, denotado por $\nabla \phi_j^{k_i}(\lambda)$. Note que, $\phi_j^{k_i}$ pode ser escrito de outra forma, como

$$\phi_j^{k_i}(\lambda) = \frac{1}{2} \sum_{l=1}^n \ln^2(\sigma_l^{k_{ij}}),$$

com $\sigma^{k_{ij}} \in \mathbb{R}^n$ dado por $\sigma^{k_{ij}} = \text{diag}(\Sigma_j^{k_i})$. Supondo que todos $\Sigma_{ll}^{k_{ij}}$, $1 \leq l \leq n$, são distintos, segue que a q -ésima componente de $\nabla \phi_j^{k_i}(\lambda)$ é dado por

$$(\nabla \phi_j^{k_i}(\lambda))_q = \sum_{l=1}^n \frac{\ln(\sigma_l^{k_{ij}})}{\sigma_l^{k_{ij}}} \frac{\partial \sigma_l^{k_{ij}}}{\partial \lambda_q}. \quad (3.12)$$

Em GREGÓRIO [11], na definição 1.2.8 é observado que o gradiente riemanniano de uma função diferenciável f , denotado por $\text{grad } f$, depende da métrica riemanniana. Em particular, se a métrica é dada por uma matriz não singular $G(\bullet)$ então

$$\text{grad } f(p) = G^{-1}(p) f'(p), \quad (3.13)$$

onde $f'(p)$ é a derivada usual de f no ponto p . Assim, decorre da relação riemanniana (3.13) que o gradiente riemanniano de $\phi_j^{k_i}$ em $T_\Lambda \Omega_{++}^n$, denotado por $\text{grad } \phi_j^{k_i}$, satisfaz

$$\text{grad } \phi_j^{k_i}(\Lambda) = [F''(\Lambda)]^{-1} \nabla \phi_j^{k_i}(\Lambda), \quad (3.14)$$

onde $F''(\Lambda)$ é a hessiana de (1.1), em Λ . Segue da relação (2.12), que

$$[F''(\Lambda)]^{-1} \nabla \phi_j^{k_i}(\Lambda) = \Lambda \nabla \phi_j^{k_i}(\Lambda) \Lambda. \quad (3.15)$$

Uma vez que Λ é substituído por $\lambda = \text{diag}(\Lambda)$, a relação (3.14) pode ser reescrita como

$$\text{grad } \phi_j^{k_i}(\lambda) = (\lambda \odot \lambda) \odot \nabla \phi_j^{k_i}(\Lambda), \quad (3.16)$$

onde $(x_1, x_2, \dots, x_n) \odot (y_1, y_2, \dots, y_n) = (x_1 y_1, x_2 y_2, \dots, x_n y_n)$ representa o produto de hadamard entre vetores do \mathbb{R}^n . Portanto, a q -ésima componente do gradiente

riemanniano de $\phi_j^{k_i}$ em λ , é dada por

$$\left(\text{grad } \phi_j^{k_i}(\lambda)\right)_q = \lambda_q^2 \sum_{l=1}^n \frac{\ln(\sigma_l^{k_{ij}})}{\sigma_l^{k_{ij}}} \frac{\partial \sigma_l^{k_{ij}}}{\partial \lambda_q}. \quad (3.17)$$

Assumindo que todos $\sigma_l^{k_{ij}}$, $1 \leq l \leq n$, são distintos, a equação (3.17) está bem definida. No entanto, na prática $\text{grad } \phi_j^{k_i}(\lambda)$ não pode ser calculada explicitamente pois não temos a regra analítica para $\Sigma_j^{k_i}$ com respeito a Λ ou não é possível garantir que todos $\sigma_l^{k_{ij}}$, $1 \leq l \leq n$, são distintos.

Por outro lado, para cada $\Lambda \in \Omega_{++}^n$, $\Sigma_j^{k_i}(\Lambda)$ pode ser computada exatamente. Isto implica que uma aproximação para a derivada direcional $\sigma_l^{k_{ij}}(\lambda, e_q) = \lim_{h \rightarrow 0} \frac{\sigma_l^{k_{ij}}(\lambda + he_q) - \sigma_l^{k_{ij}}(\lambda)}{h}$, onde $e_q = (0, \dots, 1, \dots, 0)^T \in \mathbb{R}^n$, isto é, a derivada parcial $\frac{\partial \sigma_l^{k_{ij}}}{\partial \lambda_q}$, pode ser computada a cada iteração usando um método de extrapolação. Por exemplo, podemos calcular o quociente de um lado

$$\Delta \sigma_{l^q}^{k_{ij}} = \frac{\sigma_l^{k_{ij}}(\lambda^+) - \sigma_l^{k_{ij}}(\lambda)}{h}, \quad (3.18)$$

ou diferença quociente central

$$\Delta \sigma_{l^q}^{k_{ij}} = \frac{\sigma_l^{k_{ij}}(\lambda^+) - \sigma_l^{k_{ij}}(\lambda^-)}{h}, \quad (3.19)$$

onde $\lambda^+ = \lambda + he_q$ e $\lambda^- = \lambda - he_q$, com $h > 0$ suficiente pequeno. De acordo com STOER e BULIRSCH [18], o segundo caso é uma melhor aproximação no caso de um método de extrapolação.

Na prática, substituímos a q -ésima componente de $\text{grad } \phi_j^{k_i}(\lambda)$, dada em 3.17, por uma aproximação dada por

$$\left(\widetilde{\text{grad}} \phi_j^{k_i}(\lambda)\right)_q = \lambda_q^2 \sum_{l=1}^n \frac{\ln(\sigma_l^{k_{ij}})}{\sigma_l^{k_{ij}}} \Delta \sigma_{l^q}^{k_{ij}}.$$

Desde que $\phi_j^k(\lambda) = \sum_{i=1}^m \phi_j^{k_i}(\lambda)$, temos que

$$\left(\widetilde{\text{grad}} \phi_j^k(\lambda)\right)_q = \lambda_q^2 \sum_{i=1}^m \sum_{l=1}^n \frac{\ln(\sigma_l^{k_{ij}})}{\sigma_l^{k_{ij}}} \Delta \sigma_{l^q}^{k_{ij}}.$$

Note que $\widetilde{\text{grad}} \phi_j^k(\lambda)$ é uma aproximação para algum subgradiente de ϕ_j^k , em λ . A regularização $\rho_I(\Lambda) = d^2(I, \Lambda)$ nesta etapa do algoritmo de ponto proximal

também pode ser escrita como uma função de \mathbb{R}_{++}^n em \mathbb{R} , como

$$\rho_I(\lambda) = \sum_{l=1}^n \ln^2 \lambda_l.$$

É fácil ver que a q -ésima componente gradiente euclidiano de ρ_I é dada por

$$(\nabla \rho_I(\lambda))_q = \frac{\ln(\lambda_q)}{\lambda_q},$$

e mais uma vez, pelas relações (3.13), (2.11) e (2.12) para a função ρ_I , segue que

$$(\text{grad } \rho_I(\lambda))_q = \lambda_q \ln(\lambda_q).$$

A geodésica (3.5), restrita a Ω_{++}^n , pode ser escrita na forma de um vetor como

$$\gamma(t) = \lambda \odot e^{t\lambda^{-1} \odot s},$$

para algum vetor $s \in \mathbb{R}^n$, onde λ^{-1} é o vetor cuja q -ésima componente é o inverso da q -ésima componente de λ e $e^{\mathbf{x}} = (e^{x_1}, e^{x_2}, \dots, e^{x_n})$, para algum $\mathbf{x} \in \mathbb{R}^n$.

Agora, podemos enunciar um método iterativo para resolver o problema do passo 1.1.1 do **Algoritmo SDPProx versão inexata**

Algoritmo 1. Dados $\delta_j^k > 0$ e $\Lambda \in \Omega_{++}^n$;
 $\lambda = \text{diag}(\Lambda)$;
computar $g = [\widetilde{\text{grad}} \phi_j^k + \beta^k \text{grad } \rho_I](\lambda)$;
 Passo 1 **enquanto** $\|g\| > \delta_j^k$ faça
 $d = -g$;
 $\alpha = \underset{t \geq 0}{\text{argmin}} \left\{ [\phi_j^k + \beta^k \rho_I](\lambda \odot e^{t\lambda^{-1} \odot d}) \right\}$;
 $\lambda = \lambda \odot e^{\alpha \lambda^{-1} \odot d}$;
 Atualizar g **retornar** para o passo 1.
fim.

Note que, no algoritmo 1, α é calculado exatamente. Na prática, empregamos o algoritmo de Armijo generalizado, presente em YANG [10], para computar aproximadamente o passo α .

3.3 Nova atualização para Q_{j+1}^k e modificação do algoritmo proposto

O algoritmo *SDPProx*, em GREGÓRIO e OLIVEIRA [1], atualiza a matriz Q_j^k , depois de computar $\bar{\Lambda}_{j+1}^k$, através de uma decomposição de schur de $Q_j^k \bar{\Lambda}_{j+1}^k Q_j^{kT}$. Eles estabelecem que o valor da função $(f + \beta^k \rho_{X^k})$ no novo ponto $Y_{j+1}^k = (X^k)^{\frac{1}{2}} Q_{j+1}^k \Lambda_{j+1}^k Q_{j+1}^{kT} (X^k)^{\frac{1}{2}}$ é preservado, com respeito ao valor da mesma função em $\bar{Y}_{j+1}^k = (X^k)^{\frac{1}{2}} Q_{j+1}^k \bar{\Lambda}_{j+1}^k Q_{j+1}^{kT} (X^k)^{\frac{1}{2}}$, isto é,

$$f(Y_{j+1}^k) + \beta^k \rho_{X^k}(Y_{j+1}^k) = f(\bar{Y}_{j+1}^k) + \beta^k \rho_{X^k}(\bar{Y}_{j+1}^k).$$

Nesse ponto, iniciamos uma nova atualização para Q_{j+1}^k que satisfaz

$$f(Y_{j+1}^k) + \beta^k \rho_{X^k}(Y_{j+1}^k) \leq f(\bar{Y}_{j+1}^k) + \beta^k \rho_{X^k}(\bar{Y}_{j+1}^k).$$

Seja $\mathbb{O}^n \subset \mathbb{R}^{n \times n}$, o conjunto das matrizes ortogonais reais, denominado grupo ortogonal. Fixado o iterado $\bar{\Lambda}_{j+1}^k$, defina $\varphi_j^{k_i}, \psi : \mathbb{O}^n \mapsto \mathbb{R}$ como

$$\varphi_j^{k_i}(Q) = \frac{1}{2} d^2 \left(X^i, X^{k^{\frac{1}{2}}} Q \bar{\Lambda}_{j+1}^k Q^T X^{k^{\frac{1}{2}}} \right) = \frac{1}{2} \text{Tr} \left\{ \text{Ln}^2 \left(X^{i-\frac{1}{2}} X^{k^{\frac{1}{2}}} Q \bar{\Lambda}_{j+1}^k Q^T X^{k^{\frac{1}{2}}} X^{i-\frac{1}{2}} \right) \right\},$$

e

$$\psi(Q) = \frac{1}{2} d^2 \left(X^k, X^{k^{\frac{1}{2}}} Q \bar{\Lambda}_{j+1}^k Q^T X^{k^{\frac{1}{2}}} \right) = \frac{1}{2} \text{Tr} \left\{ \text{Ln}^2 \left(X^{k-\frac{1}{2}} X^{k^{\frac{1}{2}}} Q \bar{\Lambda}_{j+1}^k Q^T X^{k^{\frac{1}{2}}} X^{k-\frac{1}{2}} \right) \right\}.$$

Por similaridade de $Q \bar{\Lambda}_{j+1}^k Q^T$ e $\bar{\Lambda}_{j+1}^k$, a última equação mostra que ψ é escrita como

$$\psi(Q) = \frac{1}{2} \text{Tr} \left\{ \text{Ln}^2 \left(\bar{\Lambda}_{j+1}^k \right) \right\},$$

que não depende de Q . Isto implica que $\psi(Q) = \psi(Q_j^k)$ para todo $Q \in \mathbb{O}^n$, ou seja, ψ é constante.

Denote $\varphi_j^k : \mathbb{O}^n \mapsto \mathbb{R}$ por

$$\varphi_j^k(Q) = \sum_{i=1}^m \varphi_j^{k_i}(Q).$$

Definimos o iterado Q_{j+1}^k como uma solução local do problema

$$\min_{Q \in \mathbb{O}^n} \left\{ (\varphi_j^k + \beta \psi)(Q) \right\}.$$

Uma vez que ψ é constante, podemos omitir a última parcela da soma na função

objetivo do problema acima. Portanto, Q_{j+1}^k é uma solução local do problema

$$\min_{Q \in \mathbb{O}^n} \{\varphi_j^k(Q)\}.$$

Proposição 3.3.1. φ_j^k possui mínimo global em \mathbb{O}^n .

Demonstração. Desde que \mathbb{O}^n é compacto e φ_j^k é contínua, a proposição segue imediatamente como uma consequência do teorema de Weierstrass. ■

Com isso, propõe-se o **algoritmo modificado MSDPProx** onde a matriz Q é atualizado conforme acima

Algoritmo MSDPProx versão inexata. Dados $\epsilon, \beta^0, \epsilon^0 > 0$ e $X^0 \succ 0$;

$k \rightarrow 0$;

Passo 1 **enquanto** $\left\| \sum_{i=1}^m \text{Ln} \left(X^{k-\frac{1}{2}} X^i X^{k-\frac{1}{2}} \right) \right\|_F > \epsilon$ **faça**

Dados $\Lambda_0^k \in \Omega_{++}^n, Q_0^k \in \mathbb{O}^n$;

$Y_0^k = (X^k)^{\frac{1}{2}} Q_0^k \Lambda_0^k Q_0^{kT} (X^k)^{\frac{1}{2}}$;

$j \rightarrow 0$;

Passo 1.1 **enquanto** $\left\| \sum_{i=1}^m \text{Ln} \left(Y_j^{k-\frac{1}{2}} X^i Y_j^{k-\frac{1}{2}} \right) + \beta^k \text{Ln} \left(Y_j^{k-\frac{1}{2}} X^k Y_j^{k-\frac{1}{2}} \right) \right\|_F > \epsilon^k$ **faça**

Passo 1.1.1 **calcular** $\Lambda_{j+1}^k = \text{argmin}_{\Lambda \in \Omega_{++}^n} \{\phi_j^k(\Lambda) + \beta^k \rho_I(\Lambda)\}$;

Passo 1.1.2 **calcular** $Q_{j+1}^k \in \text{argmin} \{\varphi_j^k(Q) : Q \in \mathbb{O}_n\}$;

$Y_{j+1}^k = (X^k)^{\frac{1}{2}} Q_{j+1}^k \Lambda_{j+1}^k Q_{j+1}^{kT} (X^k)^{\frac{1}{2}}$;

atualizar j e **retornar** para o passo 2;

fim;

$X^{k+1} = Y_j^k$;

atualizar β^k, ϵ^k, k e **retornar** para o passo 1;

fim.

3.4 Estrutura riemanniana de \mathbb{O}^n

\mathbb{O}^n é um espaço métrico compacto com respeito a métrica $\|Q_1 - Q_2\|_2$, onde para uma matriz quadrada $A_{n \times n}$, $\|A\|_2 = \sup_{\|x\|_2=1, x \in \mathbb{R}^n} \|Ax\|_2$. Este resultado é apresentado com detalhes em PUTNAM e WINTNER [19]. HORN e JOHNSON [15] também cita este resultado para \mathbb{O}^n , como um caso particular do conjunto de matrizes unitárias. Como para toda matriz $Q \in \mathbb{O}^n$, $\|Q\|_F = (\text{Tr } Q^T Q)^{\frac{1}{2}} = (\text{Tr } I)^{\frac{1}{2}} = \sqrt{n}$, \mathbb{O}^n é limitado. Pela argumentação presente no Lema 2.1.8, em HORN e JOHN-

SON [15], desde que \mathbb{O}^n pode ser visto como um subconjunto particular do espaço euclidiano \mathbb{R}^{n^2} , \mathbb{O}^n é fechado. Estas considerações implicam que \mathbb{O}^n é compacto.

Por outro lado, alguns fatos sobre grupos ortogonais, como espaço tangente, métricas, gradientes naturais e geodésicas na forma riemanniana, com relação ao produto interno $\langle X_1, X_2 \rangle_Q = \text{Tr} \left\{ (QX_2)^T QX_1 \right\} = \text{Tr} \left\{ X_2^T Q^T QX_1 \right\} = \text{Tr} \left\{ X_2^T X_1 \right\}$ no espaço tangente de \mathbb{O}^n em $Q \in \mathbb{O}^n$, denotado por $T_Q\mathbb{O}^n$, são apresentados em uma breve revisão em NISHIMORI e AKAHO [20]. Neste trabalho, eles mostram que o espaço tangente à \mathbb{O}^n em I , denotado por $T_I\mathbb{O}^n$, é o conjunto de matrizes anti-simétricas, ou seja,

$$T_I\mathbb{O}^n = \{X \in \mathbb{R}^{n \times n} / X^T = -X\}.$$

Eles concluem que o espaço tangente à \mathbb{O}^n em Q , denotado por $T_Q\mathbb{O}^n$, é obtido por uma translação à esquerda (ou à direita) de $T_I\mathbb{O}^n$, isto é,

$$T_Q\mathbb{O}^n = \{X \in \mathbb{R}^{n \times n} / X^T Q + Q^T X = 0\}.$$

A geodésica em \mathbb{O}^n , com Q como ponto inicial e velocidade $V \in T_Q\mathbb{O}^n$, é outro importante fato presente em NISHIMORI e AKAHO [20]. Esta pode ser computada através da aplicação exponencial para matrizes

$$e^{tX} = \sum_{k=0}^{\infty} \frac{(tX)^k}{k!},$$

de duas maneiras equivalentes

$$\Gamma(t) = e^{tVQ^T} Q,$$

ou

$$\Gamma(t) = Qe^{tQ^T V}.$$

Para finalizar esta revisão sobre a estrutura riemanniana de \mathbb{O}^n , o gradiente natural de uma função $f : \mathbb{O}^n \mapsto \mathbb{R}$, em Q , com respeito a métrica em $T_Q\mathbb{O}^n$ definida por

$$\langle X_1, X_2 \rangle_Q = \text{Tr} \left\{ X_2^T X_1 \right\},$$

é dado por

$$\text{grad } f(Q) = \frac{1}{2} \left(\nabla f(Q) - Q [\nabla f(Q)]^T Q \right), \quad (3.20)$$

onde $\nabla f(Q)$ é o gradiente euclidiano de f em Q .

3.5 Método iterativo para computar Q_{j+1}^k

Na seção anterior alguns aspectos de compacidade de \mathbb{O}^n e a existência de Q_{j+1}^k foram discutidas. Para computar Q_{j+1}^k , o método do gradiente geodésico em \mathbb{O}^n pode ser explorado. Para conseguir isso computamos o gradiente euclidiano de φ_j^k .

Lema 3.5.1. Dada $A(t), B(t) \in \mathbb{R}^{n \times n}$, $t \in \mathbb{R}$ ser caminho suave. Então

$$[A(t) B(t)]' = A'(t) B(t) + A(t) B'(t).$$

Demonstração. Denote $C(t), D(t), E(t) \in \mathbb{R}^{n \times n}$, $t \in \mathbb{R}$, onde $C(t) = A(t) B(t)$, $D(t) = A'(t) B(t)$, $E(t) = A(t) B'(t)$. Desde que $A(t), B(t)$ são diferenciáveis, com respeito a t , todos os elementos $c_{ij}(t)$ de $C(t)$ são funções diferenciáveis, com respeito a t , e

$$c'_{ij}(t) = \left(\sum_{k=1}^n a_{ik}(t) b_{kj}(t) \right)' = \left(\sum_{k=1}^n a'_{ik}(t) b_{kj}(t) \right) + \left(\sum_{k=1}^n a_{ik}(t) b'_{kj}(t) \right) = d_{ij}(t) + e_{ij}(t).$$

Portanto, concluímos que

$$C'(t) = D(t) + E(t)$$

e segue o lema. ■

Como $X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q$ é não singular e $\Lambda_{j+1}^k \in \Omega_{++}^n$, $X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q \Lambda_{j+1}^k Q^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}}$ é simétrica definida positiva, para toda matriz ortogonal $Q \in \mathbb{O}^n$.

Proposição 3.5.2. Seja $Q(t) \in \mathbb{O}^n$ um caminho suave. Defina

$$R^{k_i}(t) = X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}}$$

e

$$g(t) = \varphi_j^{k_i}(Q(t)) = \frac{1}{2} \text{Tr} \{ L n^2 R^{k_i}(t) \}.$$

a derivada de g , com respeito a t , é dada por

$$g'(t) = 2 \left\langle Q'(t), X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k \right\rangle_F. \quad (3.21)$$

Demonstração. Como $R^{k_i}(t)$ é simétrica definida positiva, para todo $t \in \mathbb{R}$, a proposição 2.1 em MOAKHER [6] estabelece que

$$g'(t) = \text{Tr} \left\{ L n R^{k_i}(t) R^{k_i^{-1}}(t) R^{k_i'}(t) \right\}.$$

De acordo com o lema 3.5.1,

$$R^{k'_i}(t) = X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} + X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k Q'(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}}.$$

Isto implica que

$$\begin{aligned} g'(t) &\stackrel{(1)}{=} \text{Tr} \left\{ L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} \right\} \\ &\quad + \text{Tr} \left\{ L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k Q'(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} \right\} \\ &\stackrel{(2)}{=} \text{Tr} \left\{ \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \right\} \\ &\quad + \text{Tr} \left\{ X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k Q'(t)^T \right\} \\ &\stackrel{(3)}{=} \text{Tr} \left\{ \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \right\} \\ &\quad + \text{Tr} \left\{ Q'(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k \right\} \\ &\stackrel{(4)}{=} \text{Tr} \left\{ \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \right\} \\ &\quad + \text{Tr} \left\{ \Lambda_{j+1}^k Q(t)^T X^{k\frac{1}{2}} X^{i-\frac{1}{2}} R^{k_i^{-1}}(t) L_n R^{k_i}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q'(t) \right\} \\ &\stackrel{(5)}{=} \left\langle Q'(t), X^{k\frac{1}{2}} X^{i-\frac{1}{2}} R^{k_i^{-1}}(t) L_n R^{k_i}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k \right\rangle_F \\ &\quad + \left\langle Q'(t), X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k \right\rangle_F \end{aligned}$$

A igualdade (1) segue da substituição de $R^{k'_i}(t)$ em $g'(t)$. As igualdades (2), (3) e (4) seguem da propriedade invariante do traço sob transformações similares,

$$\text{Tr}(AB) = \text{Tr}(BA) \quad \text{e} \quad \text{Tr}(A) = \text{Tr}(A^T),$$

para matrizes quadradas A e B , respectivamente. A igualdade (5) segue do *produto interno de Frobenius* (2.4). Agora, sejam $Q^{k_i}(t)$ e $\Lambda^{k_i}(t)$ caminhos definidos no grupo ortogonal \mathbb{O}^n e no conjunto das matrizes diagonais definidas positivas Ω_{++}^n , respectivamente, tais que

$$R^{k_i}(t) = Q^{k_i}(t) \Lambda^{k_i}(t) Q^{k_i^T}(t).$$

Pela equação anterior

$$R^{k_i^{-1}}(t) = Q^{k_i}(t) \Lambda^{k_i^{-1}}(t) Q^{k_i^T}(t) \quad \text{e} \quad L_n R^{k_i}(t) = Q^{k_i}(t) L_n \Lambda^{k_i}(t) Q^{k_i^T}(t).$$

Isto implica que

$$\begin{aligned}
Ln R^{k_i}(t) R^{k_i^{-1}}(t) &= Q^{k_i}(t) Ln \Lambda^{k_i}(t) Q^{k_i^T}(t) Q^{k_i}(t) \Lambda^{k_i^{-1}}(t) Q^{k_i^T}(t) \\
&= Q^{k_i}(t) Ln \Lambda^{k_i}(t) \Lambda^{k_i^{-1}}(t) Q^{k_i^T}(t) \\
&= Q^{k_i}(t) \Lambda^{k_i^{-1}}(t) Ln \Lambda^{k_i}(t) Q^{k_i^T}(t) \\
&= Q^{k_i}(t) \Lambda^{k_i^{-1}}(t) Q^{k_i^T}(t) Q^{k_i}(t) Ln \Lambda^{k_i}(t) Q^{k_i^T}(t) \\
&= R^{k_i^{-1}}(t) Ln R^{k_i}(t).
\end{aligned}$$

Em outras palavras $Ln R^{k_i}(t)$ e $R^{k_i^{-1}}(t)$ comutam. Portanto, podemos concluir que

$$g'(t) = 2 \left\langle Q'(t), X^{k\frac{1}{2}} X^{i-\frac{1}{2}} R^{k_i^{-1}}(t) Ln R^{k_i}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k \right\rangle_F.$$

■

Pela teoria diferencial,

$$g'(t) = \left\langle \nabla \varphi_j^{k_i}(Q(t)), Q'(t) \right\rangle_F. \quad (3.22)$$

Comparando (3.21) e (3.22), concluímos que

$$\nabla \varphi_j^{k_i}(Q(t)) = 2X^{k\frac{1}{2}} X^{i-\frac{1}{2}} Ln R^{k_i}(t) R^{k_i^{-1}}(t) X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q(t) \Lambda_{j+1}^k,$$

portanto

$$\nabla \varphi_j^k(Q) = 2 \sum_{i=1}^m X^{k\frac{1}{2}} X^{i-\frac{1}{2}} Ln R^{k_i} R^{k_i^{-1}} X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q \Lambda_{j+1}^k.$$

Pela equação (3.20),

$$grad \varphi_j^k(Q) = \frac{1}{2} \left(\nabla \varphi_j^k(Q) - Q [\nabla \varphi_j^k(Q)]^T Q \right).$$

Substituindo $R^{k_i^{-1}} = X^{i\frac{1}{2}} X^{k-\frac{1}{2}} Q \Lambda_{j+1}^{k-1} Q^T X^{k-\frac{1}{2}} X^{i\frac{1}{2}}$ em $\nabla \varphi_j^k(Q)$ temos

$$\nabla \varphi_j^k(Q) = 2 \sum_{i=1}^m X^{k\frac{1}{2}} X^{i-\frac{1}{2}} Ln R^{k_i} X^{i\frac{1}{2}} X^{k-\frac{1}{2}} Q \Lambda_{j+1}^{k-1} Q^T X^{k-\frac{1}{2}} X^{i\frac{1}{2}} X^{i-\frac{1}{2}} X^{k\frac{1}{2}} Q \Lambda_{j+1}^k,$$

que simplificando resulta em

$$\nabla \varphi_j^k(Q) = 2 \sum_{i=1}^m X^{k\frac{1}{2}} X^{i-\frac{1}{2}} Ln R^{k_i} X^{i\frac{1}{2}} X^{k-\frac{1}{2}} Q.$$

Denotando $P^{k_i} = X^{k\frac{1}{2}} X^{i-\frac{1}{2}} L_n R^{k_i} X^{i\frac{1}{2}} X^{k-\frac{1}{2}}$, conclui-se que

$$\text{grad } \varphi_j^k(Q) = \frac{1}{2} \left(2 \sum_{i=1}^m P^{k_i} Q - Q \left(2 \sum_{i=1}^m Q^T P^{k_i^T} \right) Q \right),$$

ou ainda, simplificando,

$$\text{grad } \varphi_j^k(Q) = \left(\sum_{i=1}^m P^{k_i} - P^{k_i^T} \right) Q.$$

Agora, com os resultados anteriores segue um método iterativo para resolver o problema do passo 1.1.2 do **Algoritmo MSDPProx versão inexata**

Algoritmo 2. Dados $\delta_j^k > 0$ e $Q \in \mathbb{O}^n$;
computar $G = \text{grad } \varphi_j^k(Q)$;
 Passo 1 **enquanto** $\|G\| > \delta_j^k$ faça
 $D = -G$;
 $\alpha = \underset{t \geq 0}{\text{argmin}} \left\{ \varphi_j^k(Q e^{tQ^T D}) \right\}$;
 $Q = Q e^{\alpha Q^T D}$;
 Atualizar G **retornar** para o passo 1.
fim.

Novamente, utiliza-se o algoritmo de Armijo generalizado, presente em YANG [10], para computar aproximadamente o passo α .

Capítulo 4

Implementação do algoritmo proposto *MSDPProx*

Neste capítulo são detalhados os parâmetros utilizados na implementação, assim como rotinas diferentes para gerar matrizes simétricas definidas positivas aleatoriamente. Em seguida, apresenta-se os resultados obtidos. Tais rotinas e o algoritmo *MSDPProx* foram escritos em linguagem Matlab. Para realização dos testes computacionais, foi utilizado o Matlab versão 7.6.0.324 (R2008a), licença número 564368, no laboratório de informática do Programa de Engenharia Química (PEQ).

4.1 Descrição das variáveis e dos parâmetros

Em todos os testes, as matrizes X^0 , Λ_0^k e Q_0^k para $k = 1, 2, \dots$ receberam a matriz identidade. Também em todos os testes foi utilizado $\epsilon = 10^{-3}$, de modo que para o teste convergir é necessário que no algoritmo modificado *MSDPProx*

$$\left\| \sum_{i=1}^m \text{Ln} \left(X^{k-\frac{1}{2}} X^i X^{k-\frac{1}{2}} \right) \right\|_F \leq 10^{-3}.$$

No algoritmo 1, foi utilizado $h = \lambda_q \sqrt{\text{eps}}$ no cálculo de $\Delta \sigma_{l^a}^{kij}$ em $\widetilde{\text{grad}} \phi_j^k$, onde eps é o menor número representável da aritmética de ponto flutuante no Matlab.

O Algoritmo 1 e algoritmo 2 são métodos iterativos, onde depois de cada iteração j a precisão δ_j^k é atualizada através da expressão $\delta_{j+1}^k = \delta_j^k \theta_3$, com $0 < \theta_3 < 1$.

Os experimentos com o *MSDPProx* foram feitos com a sequência $\{\beta^k\}_{k \in \mathbb{N}}$ constante, de modo que a série $\sum_{k=0}^{\infty} \frac{1}{\beta^k}$ diverge e a sequência $\{\epsilon^k\}_{k \in \mathbb{N}}$ satisfazendo

$\epsilon^{k+1} = \epsilon^k \theta_2$, com $0 < \theta_2 < 1$, de modo que a série $\sum_{k=0}^{\infty} \epsilon^k$ converge, assim como a

série $\sum_{k=0}^{\infty} \frac{\epsilon^k}{\beta^k}$.

Para os experimentos, cada matriz X^i ($i = 1, \dots, m$) é gerada aleatoriamente no Matlab por 4 rotinas apresentadas a seguir.

```

rotina1(m,n)
  Para  $i = 1 : m$ 
     $X(:, :, i) = rand(n);$ 
     $X(:, :, i) = X(:, :, i) * X(:, :, i)';$ 
  fm.
fm

```

Na rotina1, é gerada uma matriz quadrada aleatoriamente e atribuída à matriz X^i , em seguida X^i recebe ela mesma multiplicada por sua transposta. Note que por definição, uma matriz $A_{n \times n}$ é semidefinida positiva se $x^t A x \geq 0$, daí tomando $A = X^i X^{i^t}$ e $y = X^{i^t} x$, verificamos que $x^t X^i X^{i^t} x = y^t y \geq 0$, o que permite concluir que a rotina1 retorna matrizes simétricas semidefinidas positivas. Mas isso implica que a matriz X^i pode ter autovalor nulo. Seleccionamos então somente as matrizes que tem somente autovalores estritamente positivos. As matrizes com autovalores nulos, podem ser substituídas por matrizes simétricas definidas positivas geradas da mesma forma.

```

rotina2(m,n)
  Para  $i = 1 : m$ 
     $X(:, :, i) = rand(n) - rand(n);$ 
     $X(:, :, i) = X(:, :, i) * X(:, :, i)';$ 
     $X(:, :, i) = \frac{1}{norm(X(:, :, i))} * X(:, :, i);$ 
  fm.
fm

```

Na rotina2, $norm(X^i)$ é o maior autovalor de X^i , para uma matriz simétrica X . Denote $\lambda_{max}(X^i)$ como o maior autovalor de X^i , daí temos que a rotina2 retorna matrizes da forma

$$\frac{1}{\lambda_{max}(X^i)} X^i. \quad (4.1)$$

Suponha que v_{max} seja autovetor associado ao autovalor $\lambda_{max}(X^i)$. Então,

$$v_{max}^t \left(\frac{1}{\lambda_{max}(X^i)} X^i \right) v_{max} = \frac{1}{\lambda_{max}(X^i)} (v_{max}^t X^i v_{max}) = \frac{1}{\lambda_{max}(X^i)} \lambda_{max}(X^i) = 1.$$

Sendo assim, a rotina2 retorna aleatoriamente matrizes simétricas semidefinidas positivas cujos autovalores pertencem ao intervalo $[0, 1]$. Novamente, seleccionamos

então somente as matrizes que tem somente autovalores estritamente positivos. As matrizes com autovalores nulos, podem ser substituídas por matrizes simétricas definidas positivas geradas da mesma forma.

```

rotina3(m,n)
  Para  $i = 1 : m$ 
     $X(:, :, i) = rand(n) - rand(n);$ 
     $X(:, :, i) = X(:, :, i) * X(:, :, i)';$ 
     $X(:, :, i) = X(:, :, i) - eye(n) * min(eig(X(:, :, i)));$ 
     $X(:, :, i) = \frac{1}{norm(X(:, :, i))} * X(:, :, i);$ 
     $X(:, :, i) = X(:, :, i) + eye(n);$ 
     $X(:, :, i) = \frac{1}{norm(X(:, :, i))} * X(:, :, i);$ 
  fim.
fim

```

Na rotina3, note que, depois de receber o produto $X^i X^{it}$, a matriz X^i é simétrica semidefinida positiva. Denote $\lambda_{min}(X^i)$ e $\|A\|_2 = \sqrt{\lambda_{max}(AA^t)} = \lambda_{max}(A)$, sendo A matriz simétrica, como o menor autovalor de X^i e maior autovalor de A , respectivamente. Daí, temos que a rotina3 retorna matrizes da forma

$$a[b(X^i - \lambda_{min}(X^i)I) + I], \quad (4.2)$$

onde,

$$b = \frac{1}{\|X^i - \lambda_{min}(X^i)I\|_2} \quad e \quad a = \frac{1}{\|b(X^i - \lambda_{min}(X^i)I) + I\|_2}.$$

Suponha que v_{min} seja autovetor associado ao autovalor $\lambda_{min}(X^i)$.

$$\begin{aligned}
 v_{min}^t \{a[b(X^i - \lambda_{min}(X^i)I) + I]\} v_{min} &= ab[v_{min}^t X^i v_{min} - \lambda_{min}(X^i)v_{min}^t v_{min}] + av_{min}^t v_{min} \\
 &= ab[\lambda_{min}(X^i) - \lambda_{min}(X^i)] + a \\
 &= a.
 \end{aligned}$$

Isto implica que a é o menor autovalor de (4.2). Além disso, suponha $B = b(X^i - \lambda_{min}(X^i)I)$, onde $\lambda_{max}(B) = 1$ e que \hat{v}_{max} seja autovetor associado ao autovalor $\lambda_{max}(B)$. Seque que,

$$\hat{v}_{max}^t (B + I) \hat{v}_{max} = \hat{v}_{max}^t B \hat{v}_{max} + \hat{v}_{max}^t \hat{v}_{max} = 1 + 1 = 2.$$

Isto implica que 2 é o maior autovalor de $(B + I)$ e que $a = \frac{1}{2}$. Sendo assim, a

rotina3 retorna matrizes simétricas definidas positivas cujos autovalores pertencem ao intervalo $[\frac{1}{2}, 1]$.

As rotinas anteriormente apresentadas foram motivadas pela rotina *random.m* em BINI e IANNAZZO [21].

```

rotina4(m,n)
  Para  $i = 1 : m$ 
     $X(:, :, i) = rand(n) - rand(n);$ 
     $X(:, :, i) = \frac{1}{2} * (X(:, :, i) + X(:, :, i)');$ 
     $X(:, :, i) = expm(X(:, :, i));$ 
  fim.
fim

```

Na rotina4, note que, depois de receber $\frac{1}{2}(X^i + X^{it})$, X^i é simétrica, de modo que todos os autovalores são reais. Assim, a rotina4 retorna matrizes definidas positivas da forma e^{X^i} .

4.2 Experimentos numéricos

Nesta seção, são exibidas tabelas com os resultados dos experimentos numéricos. Foram realizados testes com matrizes simétricas definidas positivas nos conjuntos S_{++}^2 (tabelas 4.1, 4.2, 4.3, 4.4, 4.5, 4.6 e 4.7), S_{++}^3 (tabelas 4.8, 4.9, 4.10, 4.11 e 4.12), S_{++}^5 (tabelas 4.13 e 4.14), S_{++}^7 (tabela 4.15) e S_{++}^{10} (tabela 4.16), explorando principalmente a variação do parâmetro β .

Denota-se $\|grad f(X^k)\|$ como a norma do gradiente da função objetivo na k -ésima iteração.

Os testes da tabela 4.1 variam apenas o parâmetro β da regularização.

$\ grad f(X^0)\ = 35.9507$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.3213	3	2.6667
2	10^{-1}	#	0.0455	2	3.5
3	10^{-2}	#	0.0042	2	4.5
4	10^{-3}	#	0,0520	1	3
5	10^{-4}	#	0,0028	2	2
6	10^{-5}	convergiu	8,8617E-04	2	3
7	10^{-6}	convergiu	8,1887E-04	1	3

Tabela 4.1: Rotina1, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Os testes da tabela 4.2 utilizam o mesmo conjunto de matrizes da tabela 4.1, modificando os parâmetros θ_2 e θ_3 , variando apenas o parâmetro β .

$\ grad f(X^0)\ = 35.9507$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	45.6975	1	3
2	10^{-1}	#	6.0556	1	3
3	10^{-2}	#	0,6243	1	3
4	10^{-3}	#	0,0628	1	3
5	10^{-4}	#	0,0031	2	2
6	10^{-5}	convergiu	8,1389E-04	1	3
7	10^{-6}	convergiu	2,2896E-04	1	3

Tabela 4.2: Rotina1, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.5$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.1$.

Os testes da tabela 4.3 variam apenas o parâmetro β , onde são geradas novas matrizes pela rotina2.

$\ grad f(X^0)\ = 16.7405$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0428	3	3.6667
2	10^{-1}	#	0.0015	3	3
3	10^{-2}	convergiu	6.2773E-04	2	7
4	10^{-3}	convergiu	8.9197E-04	2	2
5	10^{-4}	convergiu	3.1902E-04	2	2
6	10^{-5}	convergiu	1.5765E-04	2	2
7	10^{-6}	convergiu	4.2231E-04	2	2

Tabela 4.3: Rotina2, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Para os testes da tabela 4.4 novas matrizes são geradas.

$\ grad f(X^0)\ = 21.1688$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	8.5326	1	3
2	10^{-1}	#	1.0594	1	3
3	10^{-2}	#	0,1085	1	3
4	10^{-3}	#	0,0014	2	2
5	10^{-4}	convergiu	2.3107E-04	2	2
6	10^{-5}	convergiu	2.1938E-04	1	3
7	10^{-6}	convergiu	2,1296E-04	1	3

Tabela 4.4: Rotina2, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.5$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.1$.

Os testes da tabela 4.5 e 4.6 variam apenas o parâmetro β .

$\ grad f(X^0)\ = 4.9150$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0575	2	2.5
2	10^{-1}	convergiu	6.8423E-04	2	3
3	10^{-2}	convergiu	1.99E-04	2	2
4	10^{-3}	convergiu	7.04E-04	1	3
5	10^{-4}	convergiu	1.0795E-04	1	3
6	10^{-5}	convergiu	8.1472E-05	1	3
7	10^{-6}	convergiu	8.1047E-05	1	3

Tabela 4.5: Rotina3, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.5$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.1$.

Gerando novas matrizes pela rotina4.

$\ grad f(X^0)\ = 1.4609$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	convergiu	4.8744E-04	4	4
2	10^{-1}	convergiu	4.8835E-04	2	6
3	10^{-2}	#	0.0014	1	2
4	10^{-3}	convergiu	3.2117E-04	1	2
5	10^{-4}	convergiu	2.6683E-04	1	2
6	10^{-5}	convergiu	2.6359E-04	1	2
7	10^{-6}	convergiu	2.6328E-04	1	2

Tabela 4.6: Rotina4, $m = 10$, $n = 2$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

A seguir, na tabela 4.7 testes com 1000 matrizes 2×2 , geradas apenas com as rotinas 3 e 4 para garantir que as matrizes geradas aleatoriamente sejam simétricas definidas positivas como descrito na seção anterior.

rotina	estado	$\ grad f(X^0)\ $	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
3	convergiu	490.6214	9.5777E-05	1	6
4	convergiu	18.5317	2.5523E-04	1	7

Tabela 4.7: $m = 1000$, $n = 2$, $\beta = 10^{-6}$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Inicia-se na tabela 4.8 testes com matrizes 3×3 , variando apenas o parâmetro β .

$\ grad f(X^0)\ = 6.6536$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0825	2	4
2	10^{-1}	convergiu	4.3583E-04	3	4.3333
3	10^{-2}	#	0.0011	2	7.5
4	10^{-3}	convergiu	4.8032E-04	2	2.5
5	10^{-4}	convergiu	8.6429E-04	1	4
6	10^{-5}	convergiu	8.6193E-04	1	4
7	10^{-6}	convergiu	8.6222E-04	1	4

Tabela 4.8: Rotina3, $m = 10$, $n = 3$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Os testes da tabela 4.9 utilizam o mesmo conjunto de matrizes da tabela 4.8, modificando os parâmetros θ_2 e θ_3 , variando novamente apenas o parâmetro β .

$\ grad f(X^0)\ = 6.6536$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0075	3	12.3333
2	10^{-1}	convergiu	9.8874E-04	2	17.5
3	10^{-2}	#	0.0100	1	8
4	10^{-3}	convergiu	4.8409E-04	2	4.5
5	10^{-4}	convergiu	8.6910E-04	1	8
6	10^{-5}	convergiu	8.6682E-04	1	8
7	10^{-6}	convergiu	8.6711E-04	1	8

Tabela 4.9: Rotina3, $m = 10$, $n = 3$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.5$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.8$.

Agora, para a tabela 4.10, matrizes geradas pela rotina4.

$\ grad f(X^0)\ = 3.0958$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0024	3	4.3333
2	10^{-1}	#	0.0325	1	5
3	10^{-2}	#	0.0034	1	5
4	10^{-3}	convergiu	8.2244E-04	1	5
5	10^{-4}	convergiu	7.4915E-04	1	5
6	10^{-5}	convergiu	7.4785E-04	1	5
7	10^{-6}	convergiu	7.4778E-04	1	5

Tabela 4.10: Rotina4, $m = 10$, $n = 3$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Os testes da tabela 4.11 utilizam o mesmo conjunto de matrizes da tabela 4.10, modificando o parâmetro θ_3 , variando novamente apenas o parâmetro β .

$\ grad f(X^0)\ = 3.0958$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0,0024	3	25.6666
2	10^{-1}	#	0.0325	1	26
3	10^{-2}	convergiu	5.70E-04	2	40.5
4	10^{-3}	convergiu	8.2163E-04	1	27
5	10^{-4}	convergiu	7.4919E-04	1	27
6	10^{-5}	convergiu	7.4799E-04	1	27
7	10^{-6}	convergiu	7.4793E-04	1	27

Tabela 4.11: Rotina4, $m = 10$, $n = 3$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.9$.

A seguir, na tabela 4.12 testes com 1000 matrizes 3×3 , geradas apenas com as rotinas 3 e 4.

rotina	estado	$\ grad f(X^0)\ $	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
3	convergiu	651.4991	9.1589E-04	2	4
4	convergiu	33.5277	8.6848E-04	1	9

Tabela 4.12: $m = 1000$, $n = 3$, $\beta = 10^{-6}$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Nas tabelas 4.13 e 4.14, testes com matrizes 5×5 .

$\ grad f(X^0)\ = 9.0374$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.1138	2	5.5
2	10^{-1}	#	0.1363	1	7
3	10^{-2}	#	0.0138	1	7
4	10^{-3}	convergiu	6.4907E-04	2	4
5	10^{-4}	convergiu	5.4691E-04	2	4
6	10^{-5}	convergiu	6.6834E-04	2	4
7	10^{-6}	convergiu	6.6784E-04	2	4

Tabela 4.13: Rotina3, $m = 10$, $n = 5$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Gerando novas matrizes com a rotina4.

$\ grad f(X^0)\ = 5.5459$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0440	2	5
2	10^{-1}	convergiu	7.9065E-04	2	5.5
3	10^{-2}	#	0.0056	1	6
4	10^{-3}	convergiu	7.7820E-04	1	6
5	10^{-4}	convergiu	5.5010E-04	1	6
6	10^{-5}	convergiu	5.4708E-04	1	6
7	10^{-6}	convergiu	5.4704E-04	1	6

Tabela 4.14: Rotina4, $m = 10$, $n = 5$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Em seguida, nas tabelas 4.15 e 4.16 testes com matrizes 7×7 e 10×10 , respectivamente.

$\ grad f(X^0)\ = 6.5779$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	0.0419	2	5
2	10^{-1}	#	0.0551	1	7
3	10^{-2}	#	0.0056	1	7
4	10^{-3}	convergiu	7.6932E-04	1	7
5	10^{-4}	convergiu	5.1238E-04	1	7
6	10^{-5}	convergiu	5.2383E-04	1	7
7	10^{-6}	convergiu	5.2393E-04	1	7

Tabela 4.15: Rotina4, $m = 10$, $n = 7$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

$\ grad f(X^0)\ = 14.1329$					
teste	β	estado	$\ grad f(X^k)\ $	nº de iterações em k	nº médio de iterações em j/k
1	1	#	1.9579	1	8
2	10^{-1}	#	0.2215	1	8
3	10^{-2}	#	0.0225	1	8
4	10^{-3}	convergiu	9.9672E-04	2	4.5
5	10^{-4}	#	0.0010	2	4.5
6	10^{-5}	convergiu	9.7972E-04	2	4.5
7	10^{-6}	convergiu	9.7974E-04	2	4.5

Tabela 4.16: Rotina3, $m = 10$, $n = 10$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

Na tabela 4.17, a dimensão n das matrizes está fixada em 10 e apenas a quantidade m de matrizes está variando em cada teste.

teste	m	estado	$\ grad f(X^0)\ $	$\ grad f(X^k)\ $	n° de iterações em k	n° médio de iterações em j/k
1	20	convergiu	28.0181	9.3984E-04	1	10
2	40	convergiu	55.5717	9.2941E-04	1	10
3	60	convergiu	83.1532	8.2226E-04	1	10
4	80	convergiu	110.7869	5.2974E-04	1	6

Tabela 4.17: Rotina3, $n = 10$, $\beta = 10^{-6}$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

teste	m	estado	$\ grad f(X^0)\ $	$\ grad f(X^k)\ $	n° de iterações em k	n° médio de iterações em j/k
1	20	convergiu	15.2223	9.6230E-04	1	6
2	40	convergiu	19.5246	7.4947E-004	1	7
3	60	convergiu	24.4086	8.6849e-004	1	6
4	80	convergiu	27.8719	8.8713E-04	1	7

Tabela 4.18: Rotina4, $n = 10$, $\beta = 10^{-6}$, $\epsilon^0 = 10^{-3}$, $\theta_2 = 0.8$, $\delta_0^k = 10^{-3}$, $\theta_3 = 0.5$.

A seguir, com base nos dados da tabela 4.18, uma vez verificado que a condição do passo 1 não é satisfeita para X^0 , e que a condição do passo 1.1 também não é satisfeita para Y_0^0 , o algoritmo modificado *MSDPProx* recorre ao algoritmo 1 para determinar Λ_1^0 solução do passo 1.1.1, podemos ver nas figuras abaixo o comportamento monótono decrescente da evolução de $\|g\|$ no decorrer das iterações para solução deste último subproblema. É importante observar, que a figura apresentada é um exemplo de comportamento de $\|g\|$, apenas para solução de Λ_1^0 , no entanto é um comportamento típico da norma de g .

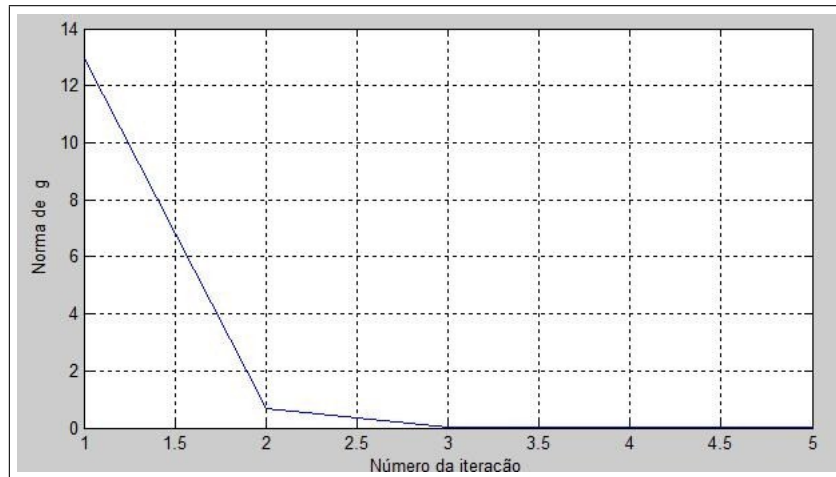


Figura 4.1: Comportamento típico da norma de g , teste3.

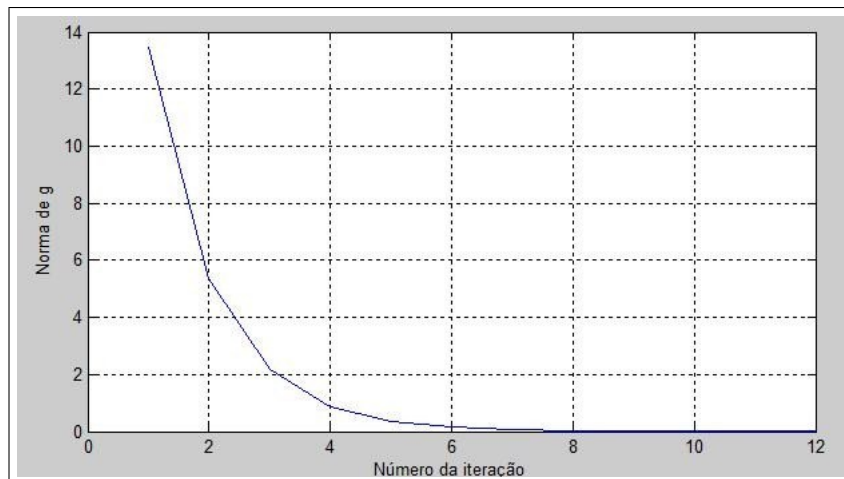


Figura 4.2: Comportamento típico da norma de g , teste4.

Ainda com base nos dados da tabela 4.18, ao recorrer ao algoritmo 2 para determinar Q_1^0 solução do passo 1.1.2, podemos ver nas figuras abaixo o comportamento oscilatório da evolução de $\|G\|$ no decorrer das iterações para solução deste último subproblema. É importante observar, que a figura apresentada é um exemplo de comportamento de $\|G\|$, apenas para solução de Q_1^0 , no entanto é um comportamento típico da norma do gradiente G .

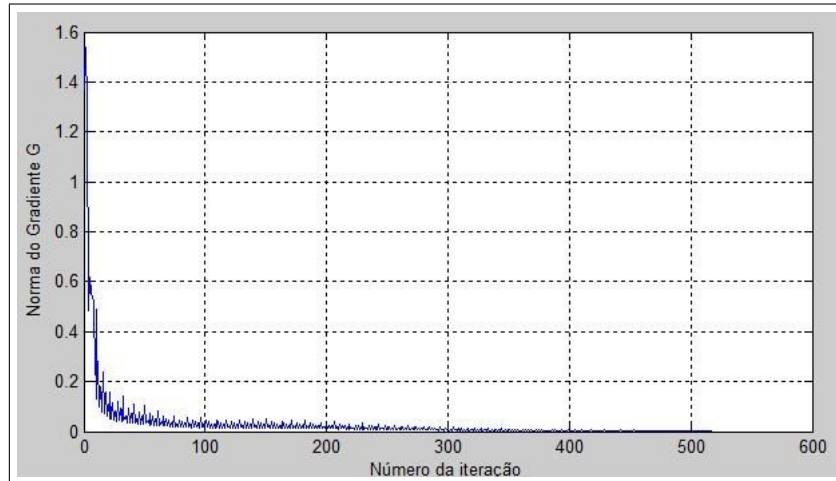


Figura 4.3: Comportamento típico da norma do gradiente G , teste3.

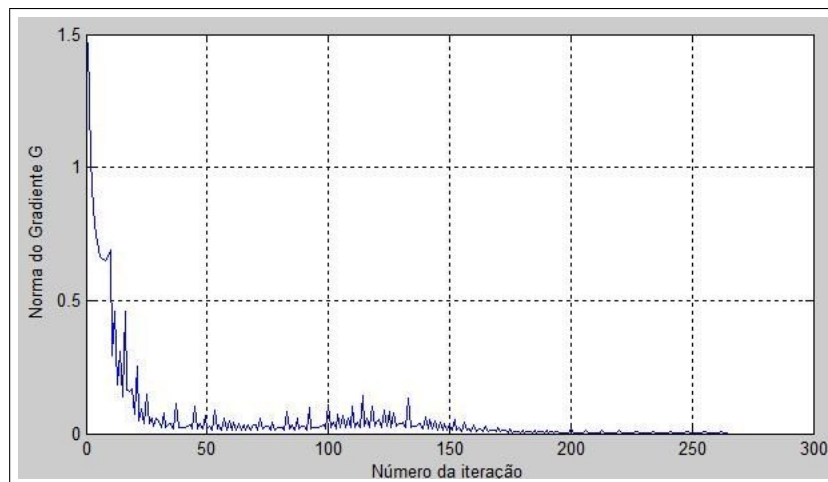


Figura 4.4: Comportamento típico da norma do gradiente G , teste4.

Capítulo 5

Conclusões e trabalhos futuros

Em geral, para os parâmetros θ_2 , θ_3 e β os valores 0.8, 0.5 e 10^{-6} respectivamente, funcionaram bem para os testes computacionais realizados.

Uma alternativa para o critério de parada do algoritmo 1 seria $d(\Lambda_j^k, \Lambda_{j+1}^k) > 0$ e para o algoritmo 2, $|\varphi_j^k(Q_j^k) - \varphi_j^k(Q_{j+1}^k)| > 0$.

Percebemos na solução da iteração principal em duas etapas como foi proposto, que na primeira etapa, o valor do gradiente da função regularizada em Λ teve comportamento monótono decrescente, no entanto na segunda etapa, o valor do gradiente da função regularizada em Q teve comportamento oscilatório, como é ilustrado nas figuras no final do capítulo 4.

Como sugestão de trabalho futuro, propomos a adaptação do método de gradiente conjugado para variedades riemannianas, em SMITH [22], para substituir o algoritmo 2 e uma variante do método de direção conjugada no caso riemanniano para substituir o algoritmo 1.

Referências Bibliográficas

- [1] GREGÓRIO, R., OLIVEIRA, P. R. “Proximal point algorithm with Schur decomposition on the cone of symmetric semidefinite positive matrices”, *J. Math. Anal. Appl.*, v. 355, n. 2, pp. 469–479, 2009.
- [2] DO CARMO, M. P. *Geometria Riemanniana*. 4 ed. Rio de Janeiro, IMPA, 2008.
- [3] SAKAI, T. *Riemannian geometry, Translations of mathematical monographs*, v. 149. Providence, R.I., American Mathematical Society, 1996.
- [4] ROTHUS, O. “Domains of positivity”, *Abh. Math. Sem. Univ. Hamburg*, v. 24, n. 3, pp. 189–235, 1960.
- [5] FERREIRA, O. P., OLIVEIRA, P. R. “Proximal point algorithm on Riemannian manifolds”, *Optimization*, v. 51, n. 2, pp. 257–270, 2002.
- [6] MOAKHER, M. “A differential geometry approach to the geometric mean of symmetric positive-definite matrices”, *SIAM Journal of Matrix Analysis and Applications*, v. 26, n. 3, pp. 735–747, 2005.
- [7] COWIN, S. C., YANG, G. “Averaging anisotropic elastic constant data”, *J. Elasticity*, v. 46.
- [8] BASSER, P. J., MATIELLO, J., BIHAN, D. L. “MR diffusion tensor spectroscopy and imaging”, *Biophysical Journal*, v. 66.
- [9] ARSIGNY, V., FILLARD, P., ET, X. P. “Geometric means in a novel vector space structure on symmetric positive-definite matrices”, *SIAM Journal of Matrix Analysis and Applications*, v. 29, n. 1, pp. 328–347, 2008.
- [10] YANG, Y. “Optimization on riemannian manifolds”, *Proceedings of the 38th conference on decision & control, Phoenix, Arizona, USA*, pp. 888–893, 1999.
- [11] GREGÓRIO, R. *Algoritmo de Ponto Proximal no Cone das Matrizes Simétricas Semidefinidas Positivas e um Método de Escalarização Proxi-*

mal Log-Quadrático para Programação Multiobjetivo. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2008.

- [12] LEON, S. J. *Álgebra Linear com Aplicações*. 4 ed. Rio de Janeiro, LTC, 1999.
- [13] GOLUB, G. H., LOAN, C. F. V. *Matrix Computations*. 3 ed. Baltimore and London, Johns Hopkins University Press, 1996.
- [14] NESTEROV, Y., TODD, M. “On the Riemannian geometry defined by self-concordant barriers and interior-point methods”, *Foundations of Computational Mathematics*, v. 2, n. 4, pp. 333–361, 2002.
- [15] HORN, R. A., JOHNSON, C. *Matrix Analysis*. 1 ed. Cambridge, Cambridge University Press, 1985.
- [16] RINEHART, R. F. “P and D in $P^{-1}XP = dg(\lambda_1, \dots, \lambda_n) = D$ as a matrix function of X”, *Canadian Journal of Mathematics*, v. 18, n. 4, pp. 832–837, 1996.
- [17] SUN, D., SUN, J. “Strong semismoothness of eigenvalues of symmetric matrices and its application to inverse eigenvalue problems”, *SIAM Journal of Numerical Analysis*, v. 40, n. 6, pp. 2352–2367, 2003.
- [18] STOER, J., BULIRSCH, R. *Matrix Analysis*. 3 ed. New York, Springer-Verlag, 2010.
- [19] PUTNAM, C. R., WINTNER, A. “Orthogonal Group in Hilbert Space”, *American Journal of Mathematics, Johns Hopkins University Press*, v. 74, n. 1, pp. 52–78, 1952.
- [20] NISHIMORI, Y., AKAHO, S. “Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold”, *Journal of Neurocomputing, Elsevier*, v. 67, n. 1, pp. 106–135, 2005.
- [21] BINI, D. A., IANNAZZO, B. “The matrix means toolbox”, *Disponível em: <http://bezout.dm.unipi.it/software/mmtree/toolbox/>. Acesso em: 27/12/11*.
- [22] SMITH, S. T. “Optimization techniques on riemannian manifolds, Hamiltonian and gradient flows, algorithms and control, Fields Int. Commun”, *Amer. Math Soc.*, v. 3.