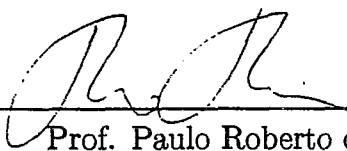


DUAS BARREIRAS PARA PROGRAMAÇÃO SEMIDEFINIDA

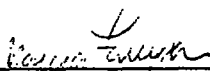
Sérgio Assunção Monteiro

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

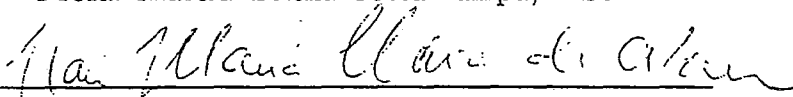
Aprovada por:



Prof. Paulo Roberto de Oliveira, Dr. Ing.



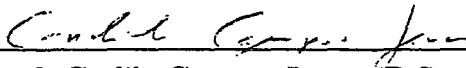
Prof. Márcia Helena Costa Fampa, D.Sc.



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



Prof. Manoel Bezerra Campelo Neto, D.Sc.



Prof. Carlile Campos Lavor, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 2007

MONTEIRO, SÉRGIO ASSUNÇÃO

Duas barreiras para programação semidefinida [Rio de Janeiro] 2007

XI, 121 p. 29,7 cm (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 2007)

Tese – Universidade Federal do Rio de Janeiro, COPPE

1 - Programação Semidefinida

2 - Métodos de Pontos Interiores

3 - Otimização Combinatória

I. COPPE/UFRJ II. Título (série)

Agradecimentos

A Deus: único, real e todo-poderoso.

À minha amada esposa Mirian, ao meu filho Sérgio Absalão e à minha filha Ana Clara pelo amor incondicional e por nunca duvidarem de mim, mesmo nos momentos mais difíceis

Aos meus queridos pais Antônio e Izabel e aos meus irmãos Tony e Andréia e aos meus sobrinhos Hysrael, Leandrinho e Belinha, pelo amor e incentivo dispensados a mim.

Ao meu orientador Paulo Roberto, por sua atenção, competência, amizade e importantes lições de vida.

Aos professores Márcia Fampa, Nair, Carlile e Manoel Campelo por participarem da minha banca.

Aos professores que me ensinaram importantes lições de vida e me deram apoio: Nelson Maculan, Abílio Lucena, Marcos Negreiros e Plácido Rogério.

A Alexandre Donato, em especial, vai a minha homenagem pelo profissional, amigo e importante apoio na reta final da minha tese.

A todos os amigos, Carlos Henrique, Talita, Michele, Luidi, Ana Lúcia, Prata, Tibérius, Mara, Ronaldo, Sissy, Yuri, Kilbert, meus irmãos da igreja metodista de Botafogo, meus amigos do Karatê, muito obrigado.

A todos os colegas de estudos, funcionários e professores do Programa.

Ao CNPq e à CAPES pelas bolsas de estudos concedidas, tanto na época de graduação quanto durante o período de mestrado e doutorado.

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

DUAS BARREIRAS PARA PROGRAMAÇÃO SEMIDEFINIDA

Sérgio Assunção Monteiro

Dezembro/2007

Orientador: Paulo Roberto Oliveira

Programa: Engenharia de Sistemas e Computação

A programação semidefinida tem recebido a atenção de vários pesquisadores nas últimas décadas. Em especial, por volta de 1995, com o trabalho de Goemans e Williansom [13], quando apresentaram uma heurística para o problema de corte máximo, baseada em um problema de programação semidefinida, que produz um corte com a garantia de estar a 14% da solução ótima. Neste trabalho, apresentamos duas novas metodologias para tratar problemas de programação semidefinida: o método de centros analíticos e a barreira hiper-cúbica. Associamos estes métodos a problemas de otimização combinatória, além de avaliá-los em relação aos melhores métodos existentes.

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

TWO BARRIERS TO SEMIDEFINITE PROGRAMMING

Sérgio Assunção Monteiro

December/2007

Advisor: Paulo Roberto Oliveira

Department: Computing and Systems Engineering

The semidefinite programming has been receiving the attention of several researchers' in the last decades. Especially, around the year of 1995 with the work of Goemans and Williamson [13], where they presented a heuristic for the max-cut problem, based on a semidefinite programming produces a cut with the guarantee of being to 14% of the optimal solution. In this work, we presented two new methodologies to treat problems of semidefinite programming : the method of analytic centers and the hiper-cubic barrier. We associate these methods to problems of combinatorial optimization, besides evaluating them in relation to the best existent methods.

Conteúdo

1	Introdução	1
2	Programação Semidefinida	3
2.1	Alguns conceitos e propriedades de Álgebra Linear	3
2.2	Problema primal de programação semidefinida	8
2.3	O problema dual	9
2.4	Condições de otimalidade	12
2.5	Métodos de Pontos Interiores para Programação Semidefinida	
	Positiva	12
2.5.1	Métodos de Pontos Interiores Seguidores da Trajetória	14
2.6	Resultados de Convergência em Programação Semidefinida . . .	17
2.7	Convergência Global do Algoritmo Primal-Dual em PSD . . .	21
3	Centros Analíticos	28

3.1	O Método de Centros Analíticos Aplicado à Programação Semidefinida	29
3.1.1	O Algoritmo Conceitual	37
3.1.2	O Método de Newton	37
3.2	O Algoritmo	40
3.3	Trajectoria Central	40
3.3.1	Propriedades próximo à Trajetória Central	43
3.3.2	Complexidade	53
3.4	Aspectos Computacionais	58
3.4.1	Obtenção da viabilidade	58
3.4.2	Proximidade inicial	59
4	Uma nova barreira para PSDP	61
4.1	Definição do Problema e da Barreira Hipercúbica para Programação Semidefinida Positiva	62
4.2	Cálculo do Gradiente e Hessiana de $B(X)$	64
4.3	A Trajetória Central	65
4.4	Aplicação para $f(X) = TrCX$	73

4.4.1	O Algoritmo	77
4.4.2	Aplicação	78
4.5	Aspectos Computacionais	79
4.5.1	Proximidade inicial	79
5	Problemas de Otimização Combinatória e SDP	81
5.1	Relaxação de Programação Semidefinida para Problemas de Otimização Combinatória	82
5.1.1	Relaxação de Programação Semidefinida para o Pro- blema do Corte Máximo	82
5.1.2	Relaxação de Programação Semidefinida para o Pro- blema de Equipartição de Grafos	84
5.1.3	Problema de k -Equipartições	85
5.1.4	Relaxação de Programação Semidefinida para o Pro- blema do Clique Máximo	87
5.1.5	Relaxação de Programação Semidefinida para o Pro- blema de Atribuição Quadrática	88
6	Experiência Numérica	94
6.1	Busca linear	94

6.2	Precondicionamento do sistema de Newton	95
6.3	Resultados Computacionais	95
7	Propostas e Conclusões	102
A	Notações	103
A.1	Operações	103
A.2	Conjuntos	103
A.3	Vetores	104
A.4	Elementos de Álgebra	104
A.5	Derivadas de $Tr(XLnX)$	105
A.6	Cálculo da Direção	107
B	Estruturas de Dados para Programação Semidefinida	110
B.1	Matrizes Esparsas	110
B.2	Exemplo do Problema de Corte Máximo	112

Lista de Tabelas

2.1	algoritmo seguidor da trajetória primal-dual com a direção HKM	18
3.1	algoritmo de centros analíticos	41
3.2	algoritmo de centros analíticos BFGS	42
4.1	algoritmo para barreira Hipercúbica $B(X)$	78
6.1	Interpretação das nomenclaturas	96
6.2	Dados para problemas de otimização combinatória da biblioteca SDPLib	97
6.3	Resultados para problemas de otimização combinatória com Algoritmo de Centros Método de Newton	98
6.4	Resultados para problemas de otimização combinatória com Algoritmo de Centros Método de QuasiNewton BFGS	99
6.5	Resultados para problemas de otimização combinatória com o pacote SDPT	100

6.6	Dados para problemas de k-partições	100
6.7	Resultados para problemas de k-partições de grafos com a barreira-hipercúbica	101
6.8	Resultados para problemas de k-partições de grafos com o modelo propostos por Rendl e Karisch	101

Capítulo 1

Introdução

A Programação Semidefinida é uma generalização da programação linear onde as variáveis vetores são substituídas por matrizes simétricas e, em particular, as restrições de não-negatividade são substituídas por restrições onde as matrizes devem ser semidefinidas positivas.

Ao longo deste trabalho apresentamos os conceitos de programação semidefinida, o seu relacionamento com a otimização combinatória e a aplicação do algoritmo de centros para o formato dual, para programação semidefinida positiva e a posterior análise de seus resultados computacionais. Além disto, apresentamos uma nova barreira para o problema primal de programação semidefinida com variáveis canalizadas. Estas são as principais contribuições deste trabalho.

Este trabalho é dividido por mais seis capítulos e dois apêndices. O capítulo 2 trata dos aspectos teóricos e análise dos principais métodos utilizados na programação semidefinida, importados dos métodos de pontos interiores para a programação linear. O capítulo 3 apresenta o método de centros que aplicamos para a programação semidefinida. No capítulo 4 apresentamos uma barreira para o problema primal de programação semidefi-

nida com variáveis canalizadas, apresentando aspectos teóricos e aplicações. No capítulo 5 apresentamos a relação entre a programação semidefinida e a otimização combinatória, analisando problemas como o do corte máximo, entre outros, detalhadamente. O capítulo 6 trata dos aspectos computacionais dos métodos apresentados no capítulo 3, resolvendo problemas da biblioteca SDPLib [7]. No capítulo 7 analisamos os resultados obtidos e apresentamos um conjunto de propostas para dar continuidade a este trabalho. Este trabalho foi escrito usando o miktex 2.1 e as fontes de referências usadas foram [3] e [33].

Os principais resultados deste trabalho estão na aplicação do algoritmo de centros, usando as direções de Newton e quase Newton, para programação semidefinida e da aplicação da barreira hiper-cúbica para o problema primal de programação semidefinida. Além disso, reunimos a análise dos principais problemas de otimização combinatória que são estudados em programação semidefinida. Desenvolvemos, ainda, uma ferramenta computacional, implementada em Matlab 7.0, que interpreta um problema no formato de armazenamento da SDPLib, ver [7], e o carrega em uma estrutura de dados para tratar com matrizes esparsas (consultar o apêndice B para mais detalhes). Deste modo, acrescentamos nesta área da programação matemática, que tem sido tão estudada nos últimos anos, duas novas metodologias para resolver problemas de programação semidefinida, além de apresentar um material com caráter didático.

Capítulo 2

Programação Semidefinida

Nos últimos anos, a programação semidefinida tem sido objeto de intenso estudo, devido a muitas aplicações em otimização combinatória, em engenharia de sistemas e de controle, estatística, entre outras, ver [1] e [15]. Este interesse aumentou muito devido ao fato de que problemas de programação semidefinida podem ser resolvidos com muita eficiência por métodos de pontos interiores. Neste capítulo, iniciando com elementos de álgebra linear necessários, apresentamos os principais conceitos de programação semidefinida, além de analisar, detalhadamente, o algoritmo primal-dual de pontos interiores aplicado à programação semidefinida.

2.1 Alguns conceitos e propriedades de Álgebra Linear

Necessitamos da definição de produto escalar no espaço de matrizes. Sejam A e B duas matrizes quadradas n por n . O produto escalar é definido por:

$$A \bullet B = \text{Tr}(A^\top B) = \sum_{i=1}^n \sum_{j=1}^n A_{ij} B_{ij} \quad (2.1)$$

onde $Tr(C)$ representa o traço da matriz C , dado por:

$$Tr(C) = \sum_{i=1}^n C_{ii} \quad (2.2)$$

Facilita a apresentação a introdução da notação do operador aplicado ao conjunto das matrizes simétricas representadas por S^n dado por: $\mathcal{A} : S^n \longrightarrow R^m$, definido por $(\mathcal{A}X)_i = A_i \bullet X$, $i = 1, \dots, m$ e $X \in S^n$. Onde A_i representa a i -ésima matriz. Neste caso, o operador adjunto $\mathcal{A}^\top : R^m \longrightarrow S^n$ se escreve:

$$\mathcal{A}^\top y = \sum_{i=1}^m y_i A_i, \quad y \in R^m.$$

De fato, \mathcal{A}^\top é adjunto de \mathcal{A} devido a que

$$\begin{aligned} \langle \mathcal{A}X, y \rangle &= \sum_{i=1}^m (\mathcal{A}X)_i y_i = \sum_{i=1}^m (A_i \bullet X) y_i = \sum_{i=1}^m y_i A_i \bullet X \\ &= (\mathcal{A}^\top y) \bullet X. \end{aligned} \quad (2.3)$$

Definição 1 *Produto de Hadamard: Sejam $A \in R^{m \times n}$ e $B \in R^{m \times n}$. Então $A \circ B$ é definido como a matriz $m \times n$ dada por:*

$$A \circ B = |A_{ij} B_{ij}|, i = 1, \dots, m, j = 1, \dots, n \quad (2.4)$$

Definição 2 *Produto de Kronecker: Sejam $G \in R^{m \times n}$ e $K \in R^{r \times s}$. Então $G \otimes K$ é definido como a matriz $mr \times ns$ dada por:*

$$G \otimes K = |g_{ij} K|, i = 1, \dots, m, j = 1, \dots, n \quad (2.5)$$

Definição 3 *Operador \odot : Sejam as matrizes $P \in S^n$, $Q \in S^n$ e $U \in S^n$, então o operador \odot de S^n em S^n é definido como:*

$$(P \odot Q)U = \frac{1}{2}(PUQ^\top + QUP^\top) \quad (2.6)$$

Definição 4 $vec(\cdot)$: Seja $A \in R^{m \times n}$. Então $vec(A)$ é definido como o vetor dado por:

$$vec(A) = [A_{11}, A_{21}, \dots, A_{m1}, A_{12}, A_{22}, \dots, A_{m2}, \dots, A_{1n}, A_{2n}, \dots, A_{mn}]^T \quad (2.7)$$

A seguir apresentamos alguns resultados que serão necessários nos capítulos seguintes:

Proposição 1 Definimos a norma de Frobenius por $\|A\| = (\sum_{i=1}^n \lambda_i(A)^2)^{\frac{1}{2}}$, onde $\lambda_i(A)$, $i = 1, \dots, n$ são os auto-valores de A . Para qualquer matriz semidefinida positiva simétrica A , temos que:

$$\|A\| \leq Tr(A) \quad (2.8)$$

Demonstração:

$$\begin{aligned} \|A\|^2 &= Tr\{A^T A\} \\ &= \sum_{i=1}^n \lambda_i(A)^2 \\ &\leq \left(\sum_{i=1}^n \lambda_i(A)\right)^2 \text{ (dado que } A \text{ é semidefinida positiva)} \\ &= (Tr\{A\})^2 \end{aligned}$$

■

Proposição 2 Sejam A e B matrizes simétricas e definidas positivas, onde $\lambda_i(A)$ representa o i -ésimo auto-valor de A , $\beta_j(B)$ representa o j -ésimo auto-valor de B e $R := P^T Q$, onde P e Q são as matrizes de autovetores de A e B , respectivamente, então:

$$TrAB = \sum_{i=1}^n \sum_{j=1}^n \lambda_i(A) \beta_j(B) R_{ij}^2 \quad (2.9)$$

onde R_{ij} é o (ij) -ésimo elemento de R .

Demonstração: Como A e B são simétricas, existem matrizes ortogonais P e Q e matrizes diagonais D e E , tais que:

$$\begin{aligned} A &= PDP^\top, PP^\top = I, D = \text{Diag}(\lambda) \\ B &= QEQ^\top, QQ^\top = I, E = \text{Diag}(\beta), \end{aligned}$$

onde usamos $\lambda = (\lambda_1(A), \dots, \lambda_n(A))$ e $\beta = (\beta_1(B), \dots, \beta_n(B))$. Definimos por $R := P^\top Q$. Portanto obtemos:

$$\begin{aligned} \text{Tr}(AB) &= \text{Tr}(PDP^\top QEQ^\top) \\ &= \text{Tr}(DP^\top QEQ^\top P) \\ &= \text{Tr}(DRER^\top) \\ &= \sum_{i=1}^n \lambda_i(A) (RER^\top)_{ii} \end{aligned}$$

Consideramos a expressão (RER^\top) e obtemos:

$$\begin{aligned} (RER^\top)_{ii} &= R_{i.}(ER^\top)_{.i} \\ &= R_{i.}(RE)_{.i}^\top \\ &= \sum_{j=1}^n \beta_j(B) R_{ij}^2 \end{aligned}$$

onde $R_{i.}$ indica a i -ésima linha da matriz R e $(ER)_{.i}^\top$ representa a i -ésima coluna de ER^\top . Portanto, temos:

$$\text{Tr}\{AB\} = \sum_{i=1}^n \sum_{j=1}^n \lambda_i(A) \beta_j(B) R_{ij}^2$$

■

Teorema 1 *Sejam A e B matrizes simétricas e definidas positivas, onde $\lambda_{\min}(A)$ representa o menor auto-valor de A , então:*

$$\text{Tr}\{AB\} \geq \lambda_{\min}(A) \text{Tr}(B) \tag{2.10}$$

Demonstração: Seja $R := P^\top Q$, onde P e Q são as matrizes de autovetores de A e B , respectivamente. Então, pela proposição 2, temos que:

$$\begin{aligned}
Tr\{AB\} &= \sum_{i=1}^n \sum_{j=1}^n \lambda_i(A) \beta_j(B) R_{ij}^2 \\
&\geq \lambda_{\min}(A) \sum_{j=1}^n \beta_j(B) \sum_{i=1}^n R_{ij}^2 \\
&= \lambda_{\min}(A) \sum_{j=1}^n \beta_j(B) \quad (\text{dado que } \sum_{i=1}^n R_{ij}^2 = 1) \\
&= \lambda_{\min}(A) Tr(B)
\end{aligned}$$

■

Lema 1 Denote por $M_{m,n}(C)$ o espaço vetorial das matrizes complexas $m \times n$ e A^* a matriz adjunta Hermitiana de $A \in M_{m,n}(C)$. Sejam $A, B \in (M_{m,n}(C))^2$ matrizes hermitianas. Existe uma matriz não-singular $S \in M_{m,n}(C)$, tal que $A = SBS^*$, se e somente se A e B tem a mesma inércia, ou seja, o mesmo número de autovalores positivos, negativos e nulos.

Demonstração: Ver [20], teorema 4.5.8. ■

Lema 2 Suponha que $(Z, X) \in (S_{++}^n \times S_{++}^n)$. Seja $\lambda_j(ZX)$ ($j = 1, \dots, n$) os auto-valores de ZX . Então $\lambda_j(ZX)$ é real e positivo para todo $j = 1, \dots, n$.

Demonstração: Dado que Z é uma matriz definida positiva, existe uma matriz simétrica não-singular e única \sqrt{Z} , tal que $Z = \sqrt{Z}\sqrt{Z}$. Portanto a matriz $\sqrt{Z}X\sqrt{Z}$ é definida positiva simétrica pelo lema 1. Como ZX é similar a $\sqrt{Z}X\sqrt{Z}$, elas possuem os mesmos auto-valores. Isto completa a demonstração. ■

2.2 Problema primal de programação semi-definida

A forma padrão do problema de programação semidefinida é como se segue:

(PSD) :

$$\min \quad C \bullet X \quad (2.11)$$

s.a :

$$A_i \bullet X = b_i \quad i = 1, \dots, m$$

$$X \succeq 0 \quad X \in S^n$$

onde X , C e A_i são matrizes quadradas reais e simétricas, S^n representa o conjunto das matrizes simétricas $n \times n$ e a matriz incógnita X é, além disto, semidefinida positiva ($X \succeq 0$). Como em qualquer problema de otimização contínua, estamos naturalmente interessados em obter condições de otimalidade, o problema dual, entre outros, de modo a termos condições de estabelecer algoritmos adequados. Observamos que o (PSD) pode ser comparado com um problema de programação linear na forma padrão, em que o cone das matrizes simétricas semidefinidas positivas corresponde ao ortante positivo. Pode-se, aliás, ver a programação linear como um caso particular da PSD, em que X , C e A_i são expressos na forma diagonal. Entretanto, é essencial que se tenha em mente o caráter não-linear, mais precisamente convexo, da PSD. Esta propriedade se verifica facilmente. A função custo é linear, portanto convexa. Da mesma forma, as restrições de igualdade são também lineares. Já as restrições de desigualdade verificam a propriedade de convexidade (temos, na verdade, um cone convexo).

2.3 O problema dual

Usaremos a função de Lagrange para a obtenção do dual do problema de PSD. Para isto, seja a função de Lagrange,

$$L(X, y) = C \bullet X + \sum_{i=1}^m y_i (b_i - A_i \bullet X), X \succeq 0, y \in R^m. \quad (2.12)$$

A função dual resulta em

$$\varphi(y) = y^\top b + \min_{X \succeq 0} (C - \sum_{i=1}^n y_i A_i) \bullet X, y \in R^m. \quad (2.13)$$

onde b é o vetor de componentes b_i . Como $X \succeq 0$, verifica-se que este problema de minimização só fornecerá resultado finito (nulo) se a matriz coeficiente dada por $C - \sum_{i=1}^n y_i A_i$ também for semidefinida positiva.

Temos, por conseguinte, o problema dual dado por:

(PSD_d) :

$$\max \quad b^\top y \quad (2.14)$$

s.a :

$$\sum_{i=1}^n y_i A_i + Z = C$$

$$Z \succeq 0, \quad Z \in S^n$$

Obteremos a seguir o teorema da dualidade fraca. Considere a diferença entre os custos primal e dual, supondo que as variáveis primal X e duais y e Z são, respectivamente, primal e dual viáveis. Temos, então:

$$C \bullet X - b^\top y = C \bullet X - \sum_{i=1}^m b_i y_i = C \bullet X - \sum_{i=1}^m y_i A_i \bullet X = Z \bullet X \geq 0 \quad (2.15)$$

Teorema 2 (dualidade fraca) *Suponha que ambos os problemas primal e dual possuam soluções viáveis. Então os valores das funções objetivo verificam $v(PSD) \geq v(PSD_d)$, sendo $Z \bullet X \geq 0$ a diferença.*

Com o objetivo de obter o teorema da dualidade forte, necessitamos de um lema do tipo Farkas.

Lema 3 (*Farkas*) *Sejam E_i , $i = 1, \dots, m$, e B matrizes simétricas n por n dadas. O sistema $\sum_{i=1}^n y_i E_i - B \succ 0$ não tem solução $y = (y_1, \dots, y_m)$ se e somente se existe uma matriz simétrica Z não-nula tal que $E_i \bullet Z = 0$, $i = 1, \dots, m$ e $B \bullet Z \geq 0$, $Z \succeq 0$.*

Demonstração: Veja [1]. ■

Teorema 3 (*dualidade forte*) *Se o problema dual tiver uma solução estritamente viável, tal que $Z \succ 0$ e $v(PSD_d)$ for finito, então o ótimo primal é atingido, e $v(PSD) = v(PSD_d)$.*

Demonstração: Considere o sistema em $y \in R^m$, dado por $\sum_{i=1}^m b_i y_i > v(PSD_d)$, $\sum_{i=1}^m y_i A_i - C \preceq 0$. Devido à definição do problema dual, este sistema não possui solução. A fim de utilizarmos o lema de Farkas, reescrevemos estas desigualdades através da seguinte notação, para $i = 1, \dots, m$:

$$E_i = \begin{bmatrix} b_i & 0 \\ 0 & -A_i \end{bmatrix}$$

$$B = \begin{bmatrix} v(PSD_d) & 0 \\ 0 & -C \end{bmatrix}$$

Então o sistema $\sum_{i=1}^m y_i E_i - B \succ 0$ não tem solução y . De acordo com o lema de Farkas, isto implica na existência de uma matriz Z simétrica não-nula tal que:

$$E_i \bullet Z = 0, \quad i = 1, \dots, m.$$

$$B \bullet Z \geq 0, \quad Z \succeq 0$$

Para retornarmos à notação inicial, escrevemos

$$Z = \begin{bmatrix} \alpha & u^\top \\ u & X \end{bmatrix}$$

obtendo $\alpha b_i - A_i \bullet X = 0$, $i = 1, \dots, n$ e $\alpha v(PSD_d) - C \bullet X \geq 0$. Mostraremos agora que $\alpha \neq 0$. Por absurdo, suponha que $\alpha = 0$. Como Z é semidefinida e não-nula, deve-se ter $u = 0$, e, portanto $X \neq 0$. Teríamos, em consequência, $A_i \bullet X = 0$, $i = 1, \dots, n$, $-C \bullet X \geq 0$, e, pelo lema de Farkas, inexistiria solução para o sistema $-\sum_{i=1}^m y_i A_i + C \succ 0$, o que é falso, devido à hipótese de existência de uma solução dual estritamente viável. Logo, $\alpha \neq 0$, sendo α de fato positivo. Como o sistema é homogêneo, podemos definir $\alpha = 1$. Temos, finalmente, $X \succeq 0$, $A_i \bullet X = b_i$, $i = 1, \dots, m$ e $C \bullet X \leq v(PSD_d)$.

Assim, X é primal viável, e o custo correspondente é inferior ao do dual. O uso do teorema da dualidade fraca completa a demonstração. ■

Definição 5 *Condição de Slater:*

1. O (PSD) satisfaz a condição de Slater, se existe um $X \in S_{++}^n$, tal que $AX = b$.
2. O (PSD_d) satisfaz a condição de Slater, se existe um par (y, Z) com $Z \in S_{++}^n$, tal que $Z + \mathcal{A}^\top y = C$.

Observação: A condição do tipo Slater, de existência de uma solução estritamente viável, é comum em problemas de programação não-linear. Deveríamos, portanto, admiti-la como natural, por ser, como vimos acima, a programação semidefinida um problema convexo (não-linear).

2.4 Condições de otimalidade

Como consequência dos teoremas 2 e 3, e do fato que $X \bullet Z = 0$ se, e somente se, $XZ = 0$, sendo $X \in S_+^n$ e $Z \in S_+^n$, mostra-se que:

Corolário 1 *Sejam X e (y, Z) soluções viáveis dos problemas primal e dual, respectivamente. Então $v(PSD) = v(PSD_d)$, e ambos são soluções ótimas se e somente se $X \bullet Z = 0$ (ou $XZ = 0$).*

Com isto, as condições de otimalidade, determinadas no corolário 1, são:

[KKT] :

$$AX = b$$

$$A^\top y + Z = C$$

$$XZ = 0$$

$$X, Z \succeq 0$$

$$y \in R^m$$

2.5 Métodos de Pontos Interiores para Programação Semidefinida Positiva

Nesta seção, abordamos o algoritmo primal-dual seguidor trajetória de pontos interiores, ver [19]. Para obter mais detalhes sobre este algoritmo, consultar [25][37].

Nesterov e Nemirovskii, ver [44], mostraram que problemas de otimização sobre cones podem ser resolvidos em tempo polinomial usando técnicas de pontos interiores. O principal elemento dos seus métodos é a existência de

uma função que possua a propriedade de auto-concordância. Nesterov e Nemirovskii, ver [44], mostraram que uma função barreira auto-concordante computável em S_{++}^n é dada por:

$$f(X) = -\log \det X$$

Portanto, existe uma técnica padrão para se construir algoritmos de pontos interiores seguidores da trajetória para (*PSD*). É importante observar que $f(X)$ é finita para $X \in S_{++}^n$ e tende a infinito, quando X se aproxima da fronteira de S_+^n . Usando o termo barreira $-\log \det X$, muitos algoritmos do tipo seguidor da trajetória primal foram estudados para resolver problemas de programação semidefinida, ver, por exemplo, [4]. O método de Newton é usado para resolver a família parametrizada de problemas 2.16, abaixo, sendo o parâmetro barreira dado por $\mu > 0$, reduzido iterativamente.

$$\min \quad C \bullet X - \mu \log \det X \tag{2.16}$$

s.a :

$$\mathcal{A}X = b$$

Deste modo, se tem $X \succ 0$ como uma restrição implícita na função objetivo. Podemos observar que a função objetivo em 2.16 é convexa em X e as restrições são lineares. Portanto, as condições de Karush-Kuhn-Tucker (KKT), associadas a este problema, garantem a otimalidade. Denotando por y o multiplicador de Lagrange associado à restrição $\mathcal{A}X = b$, então as condições de otimalidade são dadas por:

$$\mathcal{A}^\top y + Z = C, \tag{2.17}$$

$$\mathcal{A}X = b$$

$$ZX = \mu I$$

$$X \succ 0, \quad Z \succ 0$$

2.5.1 Métodos de Pontos Interiores Seguidores da Trajetória

Suponhamos que ambos os problemas primal e dual satisfaçam à condição de Slater. Portanto, o mínimo em (PSD) e o máximo em (PSD_d) são atingidos. Ambos os problemas possuem soluções finitas e seus valores ótimos são iguais, ou seja, a dualidade forte é assegurada. Qualquer solução ótima (X^*, y^*, Z^*) é caracterizada pelas condições de otimalidade.

Agora, vamos analisar a condição de complementaridade associada ao problema dual 2.18 que é dado por:

$$\max \quad b^\top y + \mu \log \det Z \quad (2.18)$$

s.a :

$$Z + \mathcal{A}^\top y = C$$

Observamos que este problema é bem definido para qualquer $\mu > 0$, uma vez que a função objetivo é estritamente côncava. Suas condições KKT são as mesmas do problema anterior, dadas em 2.17. O conjunto de soluções $\{(X_\mu, y_\mu, Z_\mu), \mu > 0\}$ define uma curva suave, chamada de trajetória central primal-dual. Em cada iteração, com μ fixo, o algoritmo gera um ponto próximo da trajetória central de acordo com um conceito de proximidade bem estabelecido, como veremos posteriormente. Este é o motivo do nome algoritmo seguidor da trajetória. Fica claro também que se $\mu \rightarrow 0$, recupera-se as condições de otimalidade do problema inicial, em que $ZX = 0$ (condição de complementaridade), as demais condições sendo sempre verificadas.

Direções de Busca

Seja $p = (X, y, Z) \in S^n \times R^m \times S^n$ e seja $F_\mu : S^n \times R^m \times S^n \mapsto R^m \times S^n \times M_n$ definido como a seguinte aplicação não-linear :

$$F_\mu(p) := \begin{bmatrix} \mathcal{A}X - b \\ Z + \mathcal{A}^\top y - C \\ ZX - \mu I \end{bmatrix}$$

M_n denota o espaço vetorial das matrizes $n \times n$.

Temos que um ponto p^* satisfaz as condições de otimalidade de Karush-Kuhn-Tucker de 2.17, se e somente se $F_\mu(p^*) = 0$. O método de Newton é usado para resolver este problema aproximadamente: se a direção de Newton é $\Delta p = (\Delta X, \Delta y, \Delta Z)$, então:

$$F_\mu(p) + \nabla F_\mu(p)\Delta p = 0.$$

No nosso caso, este sistema é dado por:

$$\begin{aligned} \mathcal{A}\Delta X &= -r_P \\ \Delta Z + \mathcal{A}^\top \Delta y &= -r_D \\ \Delta ZX + Z\Delta X &= -r_{pC} \end{aligned} \tag{2.19}$$

onde $r_P = \mathcal{A}X - b$ e $r_D = Z + \mathcal{A}^\top y - C$ são os resíduos primal e dual, respectivamente, e $r_{pC} = ZX - \mu I$ é a condição de complementaridade perturbada. Esta direção é conhecida como direção de Helmberg-Kojima-Monteiro, normalmente, chamada de direção HKM, ver [17][28][37].

A condição de complementaridade perturbada é usada para calcular o valor de μ .

$$\mu = \frac{Z \bullet X}{n}$$

Depois de calcular este parâmetro, todo o processo é repetido com redução de μ até que se obtenha a precisão desejada.

Normalmente, esta direção de busca não produz um componente ΔX simétrico, porque XZ não é simétrico. Isto ocorre porque os espaços domínio e imagem de $F_\mu(\cdot)$ têm dimensões diferentes. Para os métodos de pontos interiores é muito importante manter a simetria das variáveis, uma vez que as variáveis X e Z são simétricas. Monteiro introduziu a simetrização da terceira equação de 2.19 por um processo que ele chamou de escalonamento e simetrização, ver [37]. Ele sugeriu que fossem multiplicados ambos os lados da equação, respectivamente, por P e P^{-1} e, em seguida, a simetrizasse. A aproximação de Monteiro abordou apenas os casos onde $P = Z^{-1/2}$ e $P = X^{1/2}$. Zhang generalizou este esquema para matrizes P de escalonamento geral, ver [60]. Ele introduziu uma família de operadores de simetrização, $H_p(M) : R^{n \times n} \mapsto S^n$ definidos por:

$$H_p(M) = \frac{1}{2} \left(PMP^{-1} + (PMP^{-1})^\top \right),$$

para todo P invertível. Como consequência, conforme mostrou Zhang em [60]:

$$H_p(ZX) = \mu I, \text{ se e somente se } ZX = \mu I, \quad (Z, X) \in \left(S_{++}^n, S_{++}^n \right)$$

Portanto o sistema 2.19 é equivalente a:

$$\mathcal{A}\Delta X = -r_p$$

$$\Delta Z + \mathcal{A}^\top \Delta y = -r_D$$

$$H_p(\Delta ZX + Z\Delta X) = \mu I - H_p(ZX)$$

Esta direção de busca é chamada de direção unificada de Monteiro-Zhang, ver [41][60]. Ela produz uma classe de direções de busca parametrizadas por P . As direções mais usadas na prática, dependendo da escolha de P , são:

1. Direção NT, $P = W^{-1/2}$, satisfazendo $WZW = X$, ver [45][57].

2. Direção HKM, $P = Z^{-1/2}$, ver [17][28][37].
3. Direção AHO, $P = I$, ver [2].

Na sequência, apresentamos o algoritmo primal-dual dado em [17][16], antes, no entanto, precisamos do lema a seguir:

Lema 4 *Suponha que $X \in S_{++}^n$ e $\Delta X \in S^n$. Seja:*

$$\begin{aligned}\bar{\alpha} &= \sup \left\{ \alpha : 1 + \alpha \frac{\lambda_{\min}(\Delta X)}{\lambda_{\min}(X)} \geq 0, \alpha \geq 0, \lambda_{\min}(X) \neq 0 \right\} \\ &= \begin{cases} -\frac{\lambda_{\min}(X)}{\lambda_{\min}(\Delta X)}, & \text{se } \lambda_{\min}(\Delta X) < 0 \\ \geq 0, & \text{caso contrário} \end{cases}\end{aligned}$$

Então $X + \alpha \Delta X \in S_{++}^n$, se e somente se $\alpha < \bar{\alpha}$.

Demonstração: Para $\alpha \geq 0$, temos que:

$$\begin{aligned}\lambda_{\min}(X + \alpha \Delta X) &\geq \lambda_{\min}(X) + \lambda_{\min}(\alpha \Delta X) \\ &= \lambda_{\min}(X) + \alpha \lambda_{\min}(\Delta X).\end{aligned}\tag{2.20}$$

Então, para que 2.20 seja positivo, concluímos que:

$$\alpha = \begin{cases} -\lambda_{\min}(X)/\lambda_{\min}(\Delta X), & \text{se } \lambda_{\min}(\Delta X) < 0 \\ \geq 0, & \text{caso contrário} \end{cases}$$

■

2.6 Resultados de Convergência em Programação Semidefinida

A convergência dos métodos primais-duais seguidores da trajetória foi analisada, inicialmente, por Kojima, Shindoh e Hara [28] e por Nesterov e Todd [46][45]. O trabalho [28] introduz a família Kojima, Shindoh e Hara de

1. **Algoritmo:** ALGORITMO SEGUIDOR DA TRAJETÓRIA
2. PARA PROGRAMAÇÃO SEMIDEFINIDA
3. **Inicialização:**
4. $p^0 = (X^0, y^0, Z^0)$ com $X^0 \in S_{++}^n$ e $Z^0 \in S_{++}^n$
5. $\sigma \in (0, 1)$
6. $k = 0$
7. **repita**
8. $p = p^k$
9. $r_p = AX^k - b$
10. $r_d = Z^k + A^\top y^k - C$
11. $\mu = \langle Z^k, X^k \rangle$
12. $r_{pC} = Z^k X^k - \sigma \mu I$
13. **calcular a direção de busca** $\Delta p = (\Delta X, \Delta y, \Delta Z)$ **resolvendo** 2.19
14. **simetrizar** ΔX **tal que:** $\Delta X := \frac{\Delta X^\top + \Delta X}{2}$
15. **executar busca linear (com teste da razão), ou seja,**
16. **calcular** α_p e $\alpha_d \in (0, 1)$ (aplicar lema 4)
17. **tal que** $X^k + \alpha_p \Delta X \in S_{++}^n$ e $Z^k + \alpha_d \Delta Z \in S_{++}^n$
18. **calcular** $\alpha := \min\{\alpha_p, \alpha_d\}$
19. $X^{k+1} := X^k + \alpha \Delta X$
20. $y^{k+1} := y^k + \alpha \Delta y$
21. $Z^{k+1} := Z^k + \alpha \Delta Z$
22. $k = k + 1$
23. $p^k = (X^k, y^k, Z^k)$
24. **até que** p^k **esteja suficientemente próximo da solução ótima.**

Tabela 2.1: algoritmo seguidor da trajetória primal-dual com a direção HKM

direções de busca. Além de um método de pontos interiores seguidor da trajetória viável, Kojima et al. apresentaram também um algoritmo de pontos interiores de redução potencial viável e um inviável, ver [28]. Independente do trabalho [28], Helmberg, Rendl, Vanderbei e Wolkowicz [17] analisaram um algoritmo de ponto interior muito semelhante. Em particular, os autores em [17] apresentaram convergência global do algoritmo, enquanto Kojima et al. [28] estabeleceram a convergência polinomial para os métodos de pontos interiores propostos. Eles estabeleceram a polinomialidade baseada na direção de busca HKM, que está contida na família de direções de busca KSH. Em [42], Monteiro e Tsuchiya mostraram que este resultado é verdadeiro para qualquer direção da família KSH.

A análise de convergência local dos algoritmos de pontos interiores em programação semidefinida foi dada inicialmente por Kojima, Shida e Shindoh [30] e independentemente por Potra e Sheng [50]. Em ambos os artigos os autores apresentam convergência superlinear de um algoritmo que é uma extensão do algoritmo preditor-corretor de Mizuno-Todd-Ye dado em [36] para programação linear baseado na direção HKM. Outro artigo que trata de convergência local de métodos de pontos interiores para programação semidefinida foi publicado por Kojima, Shida e Shindoh [29].

A família Monteiro-Zhang de direções de busca foi analisada em [41]. Neste trabalho Monteiro e Zhang apresentam uma análise unificada para uma classe de algoritmos primais-duais seguidores da trajetória de passos longos para programação semidefinida cujas direções de busca são baseadas no esquema de escalonamento e simetrização introduzido por Zhang, ver [60] (seção 1.4.1). A matriz de escalonamento P é escolhida de modo que $PZXP^{-1}$ seja simétrica. Considerando esta restrição, a classe de direções de

busca obtida deste maneira inclui as direções HKM e NT, mas não a direção AHO.

A direção AHO foi analisada, recentemente, em [38] e [31]. Em [31], Kojima et al. propuseram um algoritmo primal-dual preditor-corretor em que mostraram convergência global, mas não polinomial. Eles também apresentam convergência local quadrática sob a condição de complementaridade estrita. Monteiro [38] estabelece a convergência polinomial da classe de algoritmos de pontos interiores primais-duais para programação semidefinida baseada na família Monteiro Zhang de direções de busca. Diferente de [41], em [38] nenhuma condição é dada para a matriz de escalonamento P . Em particular, dado que a direção AHO está contida na família Monteiro Zhang, Monteiro estabeleceu também a convergência polinomial dos algoritmos baseados nesta direção de busca. Em [38], Monteiro mostrou a complexidade computacional de tempo polinomial do algoritmo de passos curtos seguidor da trajetória de Kojima, Mizuno e Yoshise [27] e Monteiro e Adler [39][40] e o algoritmo preditor corretor de Mizuno, Todd e Ye [36] estendem-se da programação linear para a programação semidefinida. Tseng [55] também estabeleceu a complexidade de iteração polinomial para um algoritmo preditor-corretor para programação semidefinida usando a direção AHO.

Em [43], Monteiro e Tsuchiya estabeleceram a convergência polinomial de uma nova classe de algoritmos primais-duais seguidores da trajetória para programação semidefinida. Eles introduziram uma família de direções de busca (chamadas de família NT) as quais são baseadas na equação dada por:

$$(PZP^\top)^{\frac{1}{2}}(P^{-\top}ZP^{-1})(PZP^\top)^{\frac{1}{2}} - \mu I = 0,$$

onde P é uma matriz não-singular.

2.7 Convergência Global do Algoritmo Primal-Dual em PSD

Nesta seção, apresentamos a convergência global do algoritmo primal-dual para programação semidefinida. A primeira análise unificada para demonstrar a convergência dos algoritmos de programação não-linear foi dada em Zangwill [61]. Fazemos uso destas técnicas para demonstrar a convergência global.

Sob as hipóteses de que o conjunto $\{A_i\}, (i = 1, \dots, m)$ é linearmente independente e que ambos os problemas primal e dual satisfazem a condição de Slater, mostra-se que existe uma trajetória central para o par primal-dual (PSD) e (PSD_d) e a sequência (X_μ, y_μ, Z_μ) converge para uma solução ótima quando μ vai para zero. Para qualquer $\mu > 0$ fixo, (X_μ, y_μ, Z_μ) é caracterizado como uma solução única pelas condições de Karush-Kunh-Tucker.

Usamos p para denotar o triplo (X, y, Z) e Δp para denotar a direção de busca $(\Delta X, \Delta y, \Delta Z)$. Além disso, usamos $F(PSD)$ e $F(PSD_d)$ para indicar o conjunto das soluções viáveis de PSD e PSD_d , respectivamente e $\text{int}(\cdot)$ para caracterizar o interior de um determinado conjunto.

Em [59], sugere-se que uma função de mérito possa ser usada para medir o progresso até se atingir a solução ótima. Este função foi usada em [16, 17] para demonstrar a convergência de um algoritmo de pontos interiores para programação semidefinida. Esta demonstração segue a prova dada em Anstreicher e Vial [5] para problemas de programação convexa. Para nossa

de-monstração, vamos usar a função de mérito definida por, ver [14]:

$$f_\mu(p) := Z \bullet X - \mu \log \det(ZX) + \frac{1}{2} \|r_P\|^2 + \frac{1}{2} \|r_D\|^2, \quad (2.21)$$

onde $p = (X, y, Z)$, r_P e r_D são, respectivamente, os resíduos primal e dual, veja 2.19. A função de mérito 2.21 possui três propriedades que vamos explorar. A primeira é de que f_μ é estritamente convexa no interior do conjunto viável. A segunda é de que f_μ é continuamente diferenciável se $(Z, X) \in (S_{++}^n \times S_{++}^n)$. A última é que f_μ é limitada inferiormente e tende a infinito na fronteira do conjunto de pontos viáveis.

O lema a seguir usa o fato que $\log \det X$ é infinitamente diferenciável em $\{Y : Y \in M_n, \det Y > 0\}$, onde M_n denota o espaço das matrizes $n \times n$, e do fato que a r -ésima diferencial é, ver [35], dada por:

$$d^r \log \det X = (-1)^{r-1} (r-1)! \text{Tr}(X^{-1} \Delta X)^r, \quad r = 1, 2, \dots$$

Lema 5 *Suponha que $\mu > 0$ e $(Z, X) \in \text{int}(F(\text{PSD}) \times F(\text{PSD}_d))$. Então*

$$\begin{aligned} f_\mu(Z + \Delta Z, X + \Delta X) - f_\mu(Z, X) &= \Phi_1(\Delta Z, \Delta X) + \Phi_2(\Delta Z, \Delta X) \\ &\quad o(\|\Delta Z\|^2) + o(\|\Delta X\|^2) \end{aligned}$$

onde

$$\Phi_1(\Delta Z, \Delta X) = \langle X - \mu Z^{-1}, \Delta Z \rangle + \langle Z - \mu X^{-1}, \Delta X \rangle$$

e

$$\Phi_2(\Delta Z, \Delta X) = \frac{\mu}{2} \|Z^{-1} \Delta Z\|^2 + \frac{\mu}{2} \|X^{-1} \Delta X\|^2$$

para todo $(\Delta Z, \Delta X) \in S^n \times S^n$, onde $\Delta Z = Z - Z'$ e $\Delta X = X - X'$ para $(Z', X') \in F(\text{PSD}) \times F(\text{PSD}_d)$, respectivamente. O símbolo $o(\cdot)$ representa o termo residual.

Demonstração: Esta demonstração pode ser encontrada em [28]. ■

Corolário 2 *Suponha que $\mu > 0$. Então f_μ é estritamente convexa em $\text{int}(F(PSD) \times F(PSD_d))$.*

Demonstração: A forma quadrática $\Phi_2(\Delta Z, \Delta X)$ é positiva para qualquer $(\Delta Z, \Delta X) \in S^n \times S^n$ não-nulos. Além disso, $\Phi_2(\Delta Z, \Delta X)$ domina $o(\|\Delta Z\|^2) + o(\|\Delta X\|^2)$. Logo

$$f_\mu(Z + \Delta Z, X + \Delta X) - f_\mu(Z, X) > \Phi_1(\Delta Z, \Delta X).$$

A $\Phi_1(\Delta Z, \Delta X)$ é a derivada direcional de f_μ na direção $(\Delta Z, \Delta X)$. ■

Kojima et al. [28] usou o seguinte resultado para demonstrar a monotonicidade da trajetória central.

Lema 6 *Seja $p^0 = (X^0, y^0, Z^0) \in \text{int}(F(PSD) \times F(PSD_d))$ e seja $\zeta \in \Re$, $\zeta > 0$, tal que $f_\mu(p^0) \leq \zeta$. Então o conjunto de nível dado por:*

$$\Gamma = \{p = (X, y, Z) \in \text{int}(F(PSD) \times F(PSD_d)) : f_\mu \leq \zeta\}$$

é um subconjunto limitado e fechado de $S^n \times \mathbb{R} \times S^n$.

Demonstração: Esta demonstração pode ser encontrada em [28]. ■

Seja $p = (X, y, Z) \in \text{int}(F(PSD) \times F(PSD_d))$ e sejam $\lambda_j(ZX)$, ($j = 1, \dots, n$) os autovalores da matriz ZX . Consideramos, agora, os termos dependentes de Z e X . Tem-se, então:

$$\begin{aligned} \langle Z, X \rangle - \mu \log \det(ZX) &= \text{tr}(ZX) - \mu \left(\prod_{i=1}^n \lambda_i(ZX) \right) \\ &= \sum_{j=1}^n (\lambda_j(ZX) - \mu \log \lambda_j(ZX)) \quad (2.22) \end{aligned}$$

Segue do lema 2, seção 2.1, que $\lambda_j(ZX) > 0$ para todo $j = 1, \dots, n$. Portanto cada termo $\lambda_j(ZX) - \mu \log \lambda_j(ZX)$ atinge o mínimo se e somente se $\lambda_j(ZX) = \mu$, ($j = 1, \dots, n$). Logo, como os termos residuais são não-negativos, f_μ é limitado inferiormente. Em particular, $f_\mu(p) \geq n\mu(1 - \log \mu)$ e $F_\mu(p) = 0$, se e somente se $f_\mu(p) = n\mu(1 - \log \mu)$. A parte “se”, segue do fato que $p \in \text{int}(F(PSD) \times F(PSD_d))$, das considerações dos parágrafos anteriores, da equivalência de $ZX = \mu I$ e $\lambda_j(ZX) = \mu$, ($j = 1, \dots, n$). A parte “somente se”, segue do fato de que, dado que o $\text{int}(F(PSD) \times F(PSD_d))$ é não-vazio, existe um único minimizador global de f_μ sobre $\text{int}(F(PSD) \times F(PSD_d))$. Isto segue do corolário 2, do lema 6 e da continuidade de f_μ .

O lema a seguir demonstra que a direção de busca Δp é de descida para f_μ .

Lema 7 *A derivada direcional de f_μ na direção Δp satisfaz*

$$\left\langle \frac{\partial f_\mu(p)}{\partial p}, \Delta p \right\rangle \leq 0.$$

Esta igualdade é válida se e somente se $F_\mu(p) = 0$.

Demonstração: Esta demonstração pode ser encontrada em [16, 17]. ■

Definição 6 *O critério de Goldstein e Armijo para o tamanho do passo α , tal que $p^{k+1} = p^k + \alpha^k \Delta p^k$ e $0 < \delta < \frac{1}{2}$ é dado por:*

$$\alpha^k \text{ é aceito se e somente se: } \delta \leq \frac{f_\mu(p^{k+1}) - f_\mu(p^k)}{\alpha \left\langle \frac{\partial f_\mu(p)}{\partial p}, \Delta p \right\rangle} \leq 1 - \delta.$$

Dado que f_μ é inferiormente limitado e continuamente diferenciável em $\text{int}(F(PSD) \times F(PSD_d))$, α sempre existe. Esta condição garante que o

tamanho do passo α nem é muito grande e nem muito pequeno e que leva a um algoritmo de busca limitado, ver Luenberger [34]. Assim, executando uma busca linear em relação à função de mérito, o critério de Goldstein Armijo e um ponto inicial $p^0 \in \text{int}(F(PSD) \times F(PSD_d))$ garantem que a iteração gerada pelo algoritmo esteja em um conjunto compacto e se mova em direção à solução ótima. Em particular, dado que Δp é uma direção de descida para f_μ e f_μ tende a infinito na fronteira do conjunto de pontos viáveis, concluimos que todos os iterados restantes sejam pontos estritamente interiores.

Um algoritmo A pode ser definido como o mapeamento no espaço X que associa cada ponto $x \in X$ a um subconjunto de X . Definimos este subconjunto por $A(x)$. Logo o algoritmo A é um mapeamento ponto conjunto.

Definição 7 *Sejam X e Y conjuntos fechados não-vazios. Seja $A : X \mapsto Y$ um mapeamento ponto-conjunto. Então A é dito ser fechado em $x \in X$, se as suposições*

$$\begin{aligned} x_k \in X, \quad x_k \mapsto x \\ y_k \in A(x_k), \quad y_k \mapsto y \end{aligned}$$

implicam que $y \in A(x)$. O mapeamento ponto-conjunto A é fechado em $Z \subseteq X$ se for fechado em cada ponto de Z .

Teorema 4 *Suponha que $p \in (F(PSD) \times F(PSD_d))$ e $\Delta p \in S^n \times \mathbb{R}^m \times S^n$. Para um δ fixo, tal que $0 < \delta < \frac{1}{2}$, o mapeamento $S : \text{int}(F(PSD) \times F(PSD_d)) \times (S^n \times \mathbb{R}^m \times S^n) \mapsto \text{int}(F(PSD) \times F(PSD_d))$ definido por:*

$$S(p, \Delta p) = \left\{ \bar{p} : \bar{p} = p + \alpha \Delta p, \quad \text{para } \alpha \geq 0, \quad \delta \leq \frac{f_\mu(\bar{p}) - f_\mu(p)}{\alpha \left\langle \frac{\partial f_\mu(p)}{\partial p}, \Delta p \right\rangle} \leq 1 - \delta \right\}$$

é fechado em $(p, \Delta p)$, se $\Delta p \neq 0$.

Demonstração: Suponha que $(p^k)_{k \in \mathbb{N}}$ e $(\Delta p^k)_{k \in \mathbb{N}}$ são sequências tais que $(p^k) \rightarrow p$ e $(\Delta p^k) \rightarrow \Delta p$ e que $(\bar{p}^k \rightarrow \bar{p})$, onde $\bar{p}^k \in S(p^k, \Delta p^k)$. Temos que mostrar que $\bar{p} \in S(p, \delta p)$. Note que para cada k , $\bar{p}^k = p^k + \alpha^k \Delta p^k$ para algum α^k . Consequentemente, vale que:

$$\alpha^k = \frac{\|\bar{p}^k - p^k\|}{\|\Delta p^k\|} \rightarrow \frac{\|\bar{p} - p\|}{\|\Delta p\|} \equiv \alpha$$

Portanto, α^k converge para algum α e $\bar{p} = p + \alpha \Delta p$. Seja:

$$\phi(p, \Delta p, \alpha) = \frac{f_\mu(p + \alpha \Delta p) - f_\mu(p)}{\alpha \left\langle \frac{\partial f_\mu(p)}{\partial p}, \Delta p \right\rangle}$$

Notamos que $\delta \leq \phi(p^k, \Delta p^k, \alpha^k) \leq 1 - \delta$ para todo k . Como f_μ é contínua e diferenciável, segue que $\phi(p, \Delta p, \alpha)$ é contínua. Logo, temos que $\phi(p, \Delta p, \alpha)$ e $\delta \leq \phi(p, \Delta p, \alpha) \leq 1 - \delta$. Assim, $\bar{p} \in S(p, \Delta p)$. Deste modo, concluímos a demonstração. ■

Se usarmos o critério de Goldstein e Armijo para escolher o comprimento do passo α , então usamos o seguinte teorema para apresentar a convergência do algoritmo primal-dual seguidor da trajetória.

Teorema 5 *Suponha que $\text{int}(F(\text{PSD}) \times F(\text{PSD}_d)) \neq \emptyset$ e seja o conjunto solução não-vazio $\hat{\Gamma} \subseteq \text{int}(F(\text{PSD}) \times F(\text{PSD}_d))$ definido como*

$$\hat{\Gamma} = \{(X_\mu, y_\mu, Z_\mu), \mu > 0 : (X_\mu, y_\mu, Z_\mu) \text{ resolve}(KKT)_\mu\}. \quad (2.23)$$

Seja $S(p, \Delta p) : \text{int}(F(\text{PSD}) \times F(\text{PSD}_d)) \times (S^n \times \mathbb{R} \times S^n) \mapsto \text{int}(F(\text{PSD}) \times F(\text{PSD}_d))$ o mapeamento ponto-conjunto definido pelo teorema 4. Dado $p^0 \in \text{int}(F(\text{PSD}) \times F(\text{PSD}_d))$ tal que $f_\mu(p^0) \leq \zeta$, para algum $\zeta > 0$ e com um parâmetro $\epsilon > 0$, a sequência $(p^k)_{k \in \mathbb{N}}$ é gerada iterativamente como se segue:

Se $p^k \in \hat{\Gamma}$ e $\mu^k \leq \epsilon$, então pare; caso contrário, para o valor atual de μ que gera p^{k+1} , trocar k por $k + 1$ e repita para $\|Z^k X^k - \mu I\| \leq \epsilon$. Se $p^k \in \hat{\Gamma}$ reduzir μ por um fator constante $\rho \in (0, 1)$.

Demonstração: Os iterados gerados pelo algoritmo primal-dual são contidos no conjunto compacto Γ , definido como acima. f_μ é uma função decrescente contínua em relação à direção de busca Δp e o mapeamento $S(p, \Delta p)$ é fechado nos pontos fora do conjunto $\hat{\Gamma}$. Essas são as condições que garantem a convergência do algoritmo de mapeamento. Uma demonstração detalhada deste resultado pode ser encontrada em, por exemplo, [34]. ■

Capítulo 3

Centros Analíticos

Para tratar a dificuldade associada com o cálculo de centros em poliedros, o método de centros analíticos tem-se mostrado muito eficiente para se obter soluções com precisão de boa qualidade. A cada iteração do método, é gerada uma aproximação do centro analítico. Esta noção de centro foi introduzida por Huard [21].

Neste capítulo, aplicamos o método de centros analíticos para a programação semidefinida usando duas técnicas para obtenção de direções de descida: o método de Newton e o método quase Newton BFGS. Por outro lado, o método que desenvolvemos faz parte da classe de algoritmos no formato dual, que, juntamente com os algoritmos primais-duais, apresentam as melhores performances computacionais. Neste capítulo, abordaremos os conceitos fundamentais do algoritmo para programação semidefinida e explicaremos o funcionamento do método. Os resultados numéricos e exemplos são mostrados no capítulo 6.

3.1 O Método de Centros Analíticos Aplicado à Programação Semidefinida

O centro analítico de um conjunto \mathcal{F} [18] é o único máximo de uma determinada função distância $d(x, \mathcal{F})$. Assim, ele não está associado a um conceito geométrico e, por isto, o seu cálculo é computacionalmente mais simples.

Relembremos o dual de (PSD), ou seja, o dual de 2.11 mostrado no capítulo 2, 2.14,

(PSD_d) :

$$\max \quad b^\top y \quad (3.1)$$

s.a :

$$\sum_{i=1}^m y_i A_i + Z = C$$

$$Z \succeq 0, \quad Z \in S^n$$

Substituindo $y = x$, $c = -b$, $A_0 = C$ e $A_i = -A_i$ para $i = 1, \dots, n$, reescrevemos o problema PSD_d como o seguinte problema de minimização que contém, exclusivamente, restrições de desigualdades:

(P) :

$$\min \quad c^\top x \quad (3.2)$$

s.a :

$$A(x) \succeq 0$$

$$x \in R^m$$

onde

- $x \in R^m$ é o vetor de decisão;

- $c \in R^m$ é o vetor custo;
- $A(x)$ é uma matriz simétrica formada pela combinação linear das matrizes A_0, A_1, \dots, A_m , onde cada matriz $A_i \in S^n$, ou seja, denotando por \mathcal{A}^\top o operador associado ao conjunto de matrizes $A_i, i = 1, \dots, m$:

$$A(x) = A_0 + \mathcal{A}^\top x$$

onde $\mathcal{A}^\top x = \sum_{i=1}^m x_i A_i$.

Faremos as seguintes suposições sobre (P) :

1. O problema (P) é estritamente viável, ou seja, existe um vetor que satisfaz $A(x) \succ 0$ (condição de Slater);
2. O conjunto de soluções de (P) é não-vazio e compacto;
3. $A(x)$ é diferente de zero sempre que x seja diferente de zero (suposição de não-degenerescência), ou seja, as matrizes são linearmente independentes.

Seja \mathcal{F} o conjunto viável definido como:

$\mathcal{F} = \{x \in R^m : A(x) \succeq 0\}$. Para $\rho > c^\top x^*$, onde x^* é o valor ótimo de 3.2 (suposto existente), $x \in \text{int}(\mathcal{F})$ e $q \geq m + 1$, define-se a função barreira logarítmica [18] associada a \mathcal{F} como:

$$f_\rho(x) = -q \ln(\rho - c^\top x) - \text{Tr} \ln(A(x)) \quad (3.3)$$

O método de centros analíticos associa a 3.2 a seguinte família de problemas não-lineares e convexos:

$$\min f_\rho(x) : x \in \text{int}(\mathcal{F}) \quad (3.4)$$

Seja

$$x(\rho) = \arg \min \{f_\rho(x) : x \in \text{int}(\mathcal{F}_\rho)\} \quad (3.5)$$

com \mathcal{F}_ρ dado por:

$$\mathcal{F}_\rho = \{x \in \mathcal{F} : \underbrace{c^\top x \leq \rho, \dots, c^\top x \leq \rho}_{q \text{ vezes}}\}, \quad (3.6)$$

onde $c^\top x \leq \rho$ é repetido q vezes e \mathcal{F}_ρ é diferente de vazio e limitado. A curva contínua que une os pontos $x(\rho)$ é conhecida como trajetória central e cada $x(\rho)$ é o centro analítico de \mathcal{F}_ρ . O centro analítico pode também ser interpretado, geometricamente, como o ponto que maximiza o produto das distâncias a todas as faces de \mathcal{F}_{ρ^k} , ver [18].

O problema dual associado a 3.2, que, a menos da notação, é o primal 2.11, é dado por:

(D) :

$$\max \quad -A_0 \bullet U \quad (3.7)$$

s.a :

$$AU = c$$

$$U \succeq 0, \quad U \in S^n$$

onde $\mathcal{A} : S^n \longrightarrow R^m$ é definido por $(AU)_i = A_i \bullet U$, $i = 1, \dots, m$ e $U \in S^n$.

Com o objetivo de obter as condições de otimalidade associadas a 3.5, observamos que f_ρ é estritamente convexa em \mathcal{F}_ρ , sendo o único ponto de mínimo (interior) caracterizado por:

$$q \frac{c_i}{\rho - c^\top x} - A(x)^{-1} \bullet A_i = 0, \quad i = 1, \dots, m.$$

Considere, agora, a substituição da variável

$$U := \frac{\rho - c^\top x}{q} A(x)^{-1},$$

que leva a $A_i \bullet U = c_i$, $i = 1, \dots, m$. Temos, portanto:

$$A(x) \succeq 0, x \in \mathbb{R}^m \quad (3.8)$$

$$\mathcal{A}U = c, U \succeq 0 \quad (3.9)$$

$$A(x)U = \frac{\rho - c^\top x}{q} I \quad (3.10)$$

Agora, com o objetivo de simplificar a notação do conjunto viável \mathcal{F}_ρ , definimos:

$$\tilde{A}(x) = \tilde{A}_0 + \tilde{\mathcal{A}}^\top x = \tilde{A}_0 + \sum_{i=1}^m x_i \tilde{A}_i \quad (3.11)$$

$$(\tilde{\mathcal{A}}\tilde{U})_i := \text{Tr} \tilde{A}_i \tilde{U}, i = 1, \dots, m \text{ e } \tilde{U} \in S^{n+q} \quad (3.12)$$

e as matrizes de entrada dadas por:

$$\tilde{A}_0 = \begin{bmatrix} A_0 & 0 & \cdots & 0 \\ 0 & \rho & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \rho \end{bmatrix}_{(n+q) \times (n+q)}$$

$$\tilde{A}_i = \begin{bmatrix} A_i & 0 & \cdots & 0 \\ 0 & -c_i & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -c_i \end{bmatrix}_{(n+q) \times (n+q)}$$

$$\tilde{W} = \begin{bmatrix} W & 0 & \cdots & 0 \\ 0 & w_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_q \end{bmatrix}_{(n+q) \times (n+q)}$$

onde $W \in S^n$ e $w \in R^q$. Consequentemente, 3.5 se reescreve:

(Dm) :

$$\min f_\rho(x) = -\text{Tr}\ln(\tilde{A}(x)) \quad (3.13)$$

s.a :

$$\begin{aligned} \tilde{A}(x) - \tilde{W} &= 0 \\ \tilde{W} &\succ 0, \quad \tilde{W} \in S^{n+q} \end{aligned}$$

Analogamente ao sistema 3.8 a 3.10, as condições de otimalidade associadas ao problema 3.13, usando esta notação, são dadas por:

$$\tilde{A}(x) - \tilde{W} = 0, \quad \tilde{W} \succ 0 \quad (3.14)$$

$$\tilde{A}\tilde{U} = 0, \quad \tilde{U} \succ 0 \quad (3.15)$$

$$\tilde{U}\tilde{W} = I \quad (3.16)$$

Por outro lado, é fácil ver da notação acima que tendo $\tilde{A}(x)$ a forma diagonal nas últimas q linhas e colunas, os correspondentes autovalores de $\tilde{A}(x) = \tilde{W}$, são iguais e dados por:

$$\lambda_{n+1}(\tilde{W}) = \rho - c^\top x \quad (3.17)$$

Além disso, de 3.16, temos que:

$$\lambda_{n+1}(\tilde{U}) = \frac{1}{\lambda_{n+1}(\tilde{W})} = \frac{1}{\rho - c^\top x} \quad (3.18)$$

A seguir, mostramos que a função objetivo primal $c^\top x(\rho)$ e $\rho - c^\top x(\rho)$ são monotonicamente não-crescentes e a função objetivo dual $-\text{Tr}A_0U(\rho)$ é monotonicamente não-decrescente, se ρ diminui.

Lema 8 *Seja $x(\rho_i) := \arg \min\{f_{\rho_i}(x) : \tilde{A}(x) \succ 0, x \in \mathbb{R}^m\}$, $\forall i = 1, 2$, e seja $\rho_2 < \rho_1$. Então, os resultados a seguir são verdadeiros:*

1. $\rho_2 - c^\top x(\rho_2) \leq \rho_1 - c^\top x(\rho_1)$;
2. $c^\top x(\rho_2) \leq c^\top x(\rho_1)$;
3. $-Tr A_0 U(\rho_1) \geq -Tr A_0 U(\rho_2)$;

Demonstração:

1. Consideremos a função:

$$\phi_D(\tilde{U}, \rho) = Tr \tilde{A}_0 \tilde{U} + q\rho\lambda_{n+1}(\tilde{U}) - Tr Ln\tilde{U} - (n + q) \quad (3.19)$$

Denotaremos por $\tilde{U}(\rho)$ o ponto mínimo de $\phi_D(\tilde{U}, \rho)$ sobre a região $\{\tilde{U} : \mathcal{A}\tilde{U} = 0\}$. Seja $\hat{U}(\rho)$ a submatriz correspondente aos primeiros n auto-valores da matriz $\tilde{U}(\rho)$. Agora, uma vez que $\tilde{U}(\rho_1)$ minimiza $\phi_D(\tilde{U}(\rho_1), \rho_1)$ e $\tilde{U}(\rho_2)$ minimiza $\phi_D(\tilde{U}(\rho_2), \rho_2)$, temos que:

$$\begin{aligned} Tr A_0 \hat{U}(\rho_2) + q\rho_2\lambda_{n+1}(\tilde{U}(\rho_2)) - Tr Ln\tilde{U}(\rho_2) - (n + q) &\leq \\ Tr A_0 \hat{U}(\rho_1) + q\rho_2\lambda_{n+1}(\tilde{U}(\rho_1)) - Tr Ln\tilde{U}(\rho_1) - (n + q) &\quad (3.20) \end{aligned}$$

e

$$\begin{aligned} Tr A_0 \hat{U}(\rho_1) + q\rho_1\lambda_{n+1}(\tilde{U}(\rho_1)) - Tr Ln\tilde{U}(\rho_1) - (n + q) &\leq \\ Tr A_0 \hat{U}(\rho_2) + q\rho_1\lambda_{n+1}(\tilde{U}(\rho_2)) - Tr Ln\tilde{U}(\rho_2) - (n + q). &\quad (3.21) \end{aligned}$$

Somando as respectivas parcelas de 3.20 e 3.21, temos:

$$q(\rho_2\lambda_{n+1}(\tilde{U}(\rho_2)) + \rho_1\lambda_{n+1}(\tilde{U}(\rho_1))) \leq q(\rho_2\lambda_{n+1}(\tilde{U}(\rho_1)) + \rho_1\lambda_{n+1}(\tilde{U}(\rho_2))), \quad (3.22)$$

que implica em:

$$(\rho_1 - \rho_2)(\lambda_{n+1}(\tilde{U}(\rho_2)) - \lambda_{n+1}(\tilde{U}(\rho_1))) \geq 0.$$

Como $\rho_1 > \rho_2$ e usando 3.17 e 3.18, obtemos:

$$\rho_1 - c^\top x(\rho_1) \geq \rho_2 - c^\top x(\rho_2), \quad (3.23)$$

como queríamos demonstrar.

2. Por definição, temos:

$$f_{\rho_2}(x(\rho_2)) = -q \ln(\rho_2 - c^\top x(\rho_2)) - \text{TrLn}A(x(\rho_2))$$

Sabemos, ainda, que:

$$-q \ln(\rho_2 - c^\top x(\rho_2)) - \text{TrLn}A(x(\rho_2)) \leq -q \ln(\rho_2 - c^\top x) - \text{TrLn}A(x), \quad (3.24)$$

para x viável. Em particular, fazemos $x = x(\rho_1)$:

$$-q \ln(\rho_2 - c^\top x(\rho_2)) - \text{TrLn}A(x(\rho_2)) \leq -q \ln(\rho_2 - c^\top x(\rho_1)) - \text{TrLn}A(x(\rho_1)). \quad (3.25)$$

Aplicando um raciocínio similar para $f_{\rho_1}(x(\rho_1))$, obtemos:

$$-q \ln(\rho_1 - c^\top x(\rho_1)) - \text{TrLn}A(x(\rho_1)) \leq -q \ln(\rho_1 - c^\top x(\rho_2)) - \text{TrLn}A(x(\rho_2)) \quad (3.26)$$

Somando os termos de 3.25 e 3.26, obtemos, após algumas manipulações:

$$\frac{\rho_2 - c^\top x(\rho_2)}{\rho_1 - c^\top x(\rho_2)} \geq \frac{\rho_2 - c^\top x(\rho_1)}{\rho_1 - c^\top x(\rho_1)} \quad (3.27)$$

Da hipótese inicial, temos que $\rho_2 < \rho_1$, logo:

$$c^\top x(\rho_2) \leq c^\top x(\rho_1) \quad (3.28)$$

como queríamos demonstrar.

3. Apresentamos, agora, a demonstração da última afirmação.

Seja a função barreira associada ao problema 3.7 definida por:

$$\phi(U(\mu(\rho)), \mu(\rho)) = Tr \frac{A_0 U(\mu(\rho))}{\mu(\rho)} - Tr Ln U(\mu(\rho)) \quad (3.29)$$

onde $\mu(\rho)$ é definido por:

$$\mu(\rho) = \frac{\rho - c^\top x(\rho)}{q}$$

De 3.23 e da hipótese inicial em que $\rho_1 > \rho_2$, obtemos:

$$\mu(\rho_1) \geq \mu(\rho_2). \quad (3.30)$$

Sabemos ainda que:

$$U(\mu(\rho)) := \arg \min \{ \phi(U(\mu(\rho)), \mu(\rho)) \text{ s.a. } AU(\mu(\rho)) = c \}$$

e as condições de otimalidade associadas a 3.29 são iguais a 3.8, 3.9 e

3.10. Temos ainda que:

$$Tr \frac{A_0 U(\mu(\rho_1))}{\mu(\rho_1)} - Tr Ln U(\mu(\rho_1)) \leq Tr \frac{A_0 U(\mu(\rho_2))}{\mu(\rho_1)} - Tr Ln U(\mu(\rho_2)) \quad (3.31)$$

e que:

$$Tr \frac{A_0 U(\mu(\rho_2))}{\mu(\rho_2)} - Tr Ln U(\mu(\rho_2)) \leq Tr \frac{A_0 U(\mu(\rho_1))}{\mu(\rho_2)} - Tr Ln U(\mu(\rho_1)). \quad (3.32)$$

Somando as respectivas parcelas de 3.31 e 3.32, temos:

$$\left(\frac{1}{\mu(\rho_2)} - \frac{1}{\mu(\rho_1)} \right) (Tr A_0 U(\mu(\rho_2)) - Tr A_0 U(\mu(\rho_1))) \leq 0. \quad (3.33)$$

Agora, de 3.30, temos que:

$$Tr A_0 U(\mu(\rho_2)) \leq Tr A_0 U(\mu(\rho_1)). \quad (3.34)$$

Como queríamos demonstrar. ■

3.1.1 O Algoritmo Conceitual

O algoritmo inicia com $x^0 \in \text{int}(\mathcal{F})$ e $c^\top x^0 \leq \rho^0$. A cada iteração, é encontrado o novo centro analítico aproximado da região atual e o limitante superior é decrescido. A restrição associada à função objetivo, replicada q vezes, é removida e outra é adicionada, definindo, assim, uma nova região. Esse procedimento é repetido até que seja alcançado um ponto próximo à trajetória central, e suficientemente próximo da solução ótima.

3.1.2 O Método de Newton

Para encontrar um ponto na vizinhança de $x(\rho^k)$, o qual é o centro analítico da região \mathcal{F}_{ρ^k} , utilizar-se-á o método de Newton e o método Quase Newton, BFGS.

O método de Newton consiste na minimização da aproximação quadrática da função a ser minimizada .

Considerando a função f_ρ definida em 3.3, temos:

$$f_\rho(x) \cong f_\rho(x^k) + (x - x^k)^\top \nabla f_\rho(x^k) + \frac{1}{2}(x - x^k)^\top H_\rho(x^k)(x - x^k), \quad (3.35)$$

O vetor gradiente e a matriz hessiana são dados por:

$$\nabla f_\rho(x) = \left[\frac{q}{\rho - c^\top x} c - (A(x)^{-1} \bullet A_i)_{i=1 \dots m} \right] \quad (3.36)$$

e

$$H_\rho(x) = \left[\frac{q}{(\rho - c^\top x)^2} cc^\top + ((A(x)^{-1} A_i A(x)^{-1}) \bullet A_j)_{i,j=1 \dots m} \right]. \quad (3.37)$$

Observamos que 3.37 pode ser reescrita, com vantagens computacionais, como:

$$H_\rho(x) = \left[\frac{q}{(\rho - c^\top x)^2} cc^\top + ((A_i A(x)^{-1}) \bullet (A_j A(x)^{-1}))_{i,j=1 \dots m} \right] \quad (3.38)$$

É de simples verificação que a primeira parcela de H_ρ é semidefinida positiva (sdp) no interior de \mathcal{F}_ρ . Veremos também que a segunda parcela é definida positiva (dp) no interior de \mathcal{F} , logo f_ρ é estritamente convexa em \mathcal{F}_ρ , e possui um único ponto de mínimo.

Lema 9 *A segunda parcela de 3.38 é definida positiva.*

Demonstração:

Denotamos por $G(x)$ a segunda parcela de H_ρ .

Cada componente de $G(x)$ pode ser escrita como: $G(x)_{ij} = A(x)^{-1}A_i \bullet A(x)^{-1}A_j$. Portanto, tomando um vetor $s \in R^n$, $s \neq 0$, temos que:

$$\begin{aligned}
s^\top G(x)s &= \\
&= \sum_{i=1}^m \sum_{j=1}^m s_i s_j A(x)^{-1}A_i \bullet A(x)^{-1}A_j \\
&= A(x)^{-1} \sum_{i=1}^m s_i A_i \bullet A(x)^{-1} \sum_{j=1}^m s_j A_j \\
&= A(x)^{-1} \mathcal{A}^\top s \bullet A(x)^{-1} \mathcal{A}^\top s \\
&= \|A(x)^{-1} \mathcal{A}^\top s\|_F^2 \geq 0
\end{aligned}$$

Suponha que:

$$\|A(x)^{-1} \mathcal{A}^\top s\|_F^2 = 0 \rightarrow A(x)^{-1} \mathcal{A}^\top s = 0$$

Temos

$$\mathcal{A}^\top s = 0 \tag{3.39}$$

E como as matrizes A_i , $\forall i = 1, \dots, m$ são linearmente independentes, 3.39 só

será verdadeiro se $s = 0$, logo

$$s^\top G(x)s > 0, \quad \forall s \neq 0$$

■

Se x^{k+1} é o ponto mínimo em $f_\rho(x^{k+1})$, então $\frac{\partial f_\rho(x^{k+1})}{\partial x} = 0$. Deste modo, derivando 3.35 e igualando a zero, tem-se que:

$$\nabla f_\rho(x^k) + H_\rho(x^k)(x^{k+1} - x^k) = 0 \quad (3.40)$$

Substituindo $(x^{k+1} - x^k) = d^k$, obtém-se:

$$H_\rho(x^k)d^k = -\nabla f_\rho(x^k) \quad (3.41)$$

Como $H_\rho(x)$ é uma matriz definida positiva em \mathcal{F}_ρ , d^k sempre será uma direção de descida.

O método é executado em um processo iterativo, tal que:

$$x^{k+1} = x^k + \lambda d^k,$$

onde $\lambda = \arg \min\{f_\rho(x^k + \lambda d^k) : (x^k + \lambda d^k) \in \text{int}(\mathcal{F}_\rho)\}$, caso se escolha a busca linear exata.

Uma característica dos algoritmos polinomiais é a necessidade de se estabelecer um critério de aproximação do ponto x^{k+1} ao centro analítico. Seja a norma induzida por $H_\rho(x)$, dada por:

$$\|x\|_{H_\rho(x)} = \sqrt{x^\top H_\rho(x)x}$$

Então, a proximidade de x a $x(\rho)$ é dada através da direção de Newton por:

$$\delta_\rho(x) = \|H_\rho(x)^{-1}\nabla f_\rho(x)\|_{H_\rho} = \sqrt{\nabla f_\rho(x)^\top H_\rho(x)^{-1}\nabla f_\rho(x)} \quad (3.42)$$

Os critérios de convergência (em tempo polinomial) exigirão apenas que $\delta_\rho(x^{k+1}) \leq \epsilon$, para algum $\epsilon \in (0, 1)$.

Logo, conhecidos ρ^0 , um limitante superior para o valor da função objetivo, e $x^0 \in \text{int}(\mathcal{F}_\rho)$ tal que $\delta_\rho(x^0) \leq \epsilon$, o método é iniciado com a subsequente aproximação à trajetória central.

3.2 O Algoritmo

A seguir, apresentamos o algoritmo de centros analíticos usando o método de Newton e o método BFGS aplicado ao problema 3.2 com as notações anteriores. No algoritmo 2, temos o método de Newton aplicado para programação semidefinida. Nas linhas de 4 – 7, temos os parâmetros iniciais necessários para a execução do algoritmo. Das linhas 9 – 25, temos o laço principal do algoritmo, que garante uma ϵ -solução ótima. O laço das linhas 15 – 22 garante a proximidade da solução à trajetória central. O algoritmo 3 é semelhante ao algoritmo 2, com exceção do cálculo aproximado da hessiana, proporcionando ganhos nos cálculos computacionais.

3.3 Trajetória Central

A seguir, vamos analisar o comportamento do algoritmo de centros aplicado à programação semidefinida, quando o centro analítico está próximo à trajetória central.

Nosso primeiro passo consiste em reescrever o gradiente e a hessiana da função 3.3 de forma a facilitar às demonstrações adiante utilizando as notações apresentadas na seção 3.1.

1.	Algoritmo: ALGORITMO DE CENTROS ANALÍTICOS
2.	PARA PROGRAMAÇÃO SEMIDEFINIDA
3.	Inicialização:
4.	$\theta \in (0, 1)$ (atualização da cota);
5.	$\epsilon \in (0, 1)$ (proximidade);
6.	$x^0 \in \text{int}(\mathcal{F}_{\rho^0}) : \delta_{\rho^0} \leq \epsilon;$
7.	$\rho^0 > c^\top x^0;$
8.	$k = 0;$
9.	repita
10.	atualizar a cota superior da função custo
11.	$\rho^{k+1} = \rho^k + \theta(c^\top x^k - \rho^k)$
12.	cálculo do centro
13.	$x = x^k$
14.	repita
15.	calcular $H_{\rho^{k+1}}$ e $H_{\rho^{k+1}}^{-1}$
16.	$i = 1$
17.	$h = -H_{\rho^{k+1}}^{-1} \nabla f_{\rho^{k+1}}(x^k)$
18.	$\lambda = \arg \min_{\lambda \geq 0} \{f_{\rho^{k+1}}(x + \lambda h) : x + \lambda h \in \mathcal{F}_{\rho^{k+1}}\}$
19.	$x = x + \lambda h$
20.	$v = \ h\ _{H_{\rho^{k+1}}}$
21.	$i = i + 1$
22.	até $v \leq \epsilon$
23.	$x^{k+1} = x$
24.	$k = k + 1$
25.	até $(\rho^k - c^\top x^k) / \max\{1, c^\top x^k \} < \epsilon$

Tabela 3.1: algoritmo de centros analíticos

1. **Algoritmo:** ALGORITMO DE CENTROS ANALÍTICOS BFGS
2. PARA PROGRAMAÇÃO SEMIDEFINIDA
3. **Inicialização:**
4. $\theta \in (0, 1)$ (atualização da cota);
5. $\epsilon \in (0, 1)$ (proximidade);
6. $x^0 \in \text{int}(\mathcal{F}_{\rho^0}) : \delta_{\rho^0} \leq \epsilon$;
7. $\rho^0 > c^\top x^0$;
8. $k = 0$
9. **Obter** $x^1 \in \mathcal{F}_{\rho^0}$ e $D^k \in S_{++}^m$
10. **repita**
11. **atualizar a cota superior da função custo**
12. $\rho^{k+1} = \rho^k + \theta(c^\top x^k - \rho^k)$
13. **cálculo do centro**
14. $x = x^k$
15. **repita**
16. **calcular a inversa de $H_{\rho^{k+1}}^{-1}$ aproximada**
17. $q^k = \nabla f(x^{k+1}) - \nabla f(x^k)$
18. $p^k = x^{k+1} - x^k$
19. $\tau^k = q^k D^k q^{k\top}$
20. $w^k = \frac{p^k}{p^{k\top} q^k} - \frac{D^k q^k}{\tau^k}$
21. $D^{k+1} = D^k + \frac{p^k p^{k\top}}{p^{k\top} q^k} - \frac{D^k q^k q^{k\top} D^k}{q^{k\top} D^k q^k} + \tau^k w^k w^{k\top}$
22. $H_{\rho^{k+1}}^{-1} = D^{k+1}$
23. $h = -H_{\rho^{k+1}}^{-1} g_{\rho^{k+1}}$
24. $v = \|h\|_{H_{\rho^{k+1}}}$
25. $\lambda = \arg \min_{\lambda \geq 0} \{f_{\rho^{k+1}}(x + \lambda h) : x + \lambda h \in \mathcal{F}_{\rho^{k+1}}\}$
26. **até** $v \leq \epsilon$
27. $x^{k+1} = x$
28. $k = k + 1$
29. **até** $(\rho^k - c^\top x^k) / \max\{1, |c^\top x^k|\} < \epsilon$

Tabela 3.2: algoritmo de centros analíticos BFGS

O gradiente de $f_\rho(x)$ 3.3, é dado por:

$$g(x, \rho) = q \frac{1}{\rho - c^\top x} c - \mathcal{A}A(x)^{-1} = \tilde{\mathcal{A}}\tilde{\mathcal{A}}(x)^{-1}, \quad (3.43)$$

e a hessiana de 3.3 é dada por:

$$H(x, \rho) = q \frac{1}{(\rho - c^\top x)^2} cc^\top + \mathcal{A}A(x)^{-1}(\mathcal{A}A(x)^{-1})^\top = (\tilde{\mathcal{A}}\tilde{\mathcal{A}}(x)^{-1})(\tilde{\mathcal{A}}\tilde{\mathcal{A}}(x)^{-1})^\top \quad (3.44)$$

onde $\tilde{\mathcal{A}}(x)$ e $\tilde{\mathcal{A}}(\cdot)$ são dados, respectivamente, por 3.11 e 3.12.

3.3.1 Propriedades próximo à Trajetória Central

Consideramos três medidas de distância entre x , um ponto interior de 3.2, e um ponto da trajetória central $x(\rho)$. A primeira medida é uma adaptação para programação semidefinida introduzida em [53]. Esta medida é dada por:

$$\delta(x, \rho) = \min \|\tilde{\mathcal{A}}(x)^{\frac{1}{2}} \tilde{U} \tilde{\mathcal{A}}(x)^{\frac{1}{2}} - I\| \quad (3.45)$$

s.a :

$$\tilde{\mathcal{A}}\tilde{U} = 0$$

$$\tilde{U} \succeq 0$$

que pode ser reescrita como:

$$\delta(x, \rho) = \min \|\tilde{U} \tilde{\mathcal{A}}(x) - I\| \quad (3.46)$$

s.a :

$$\tilde{\mathcal{A}}\tilde{U} = 0$$

$$\tilde{U} \succeq 0$$

As outras duas medidas são associadas com o passo de Newton para a função $f_\rho(\cdot)$. Usamos $p = p(x, \rho)$ para definir o passo de Newton e $p_A = p_A(x, \rho)$

para definir o passo resultante na matriz $\tilde{A}(\cdot)$. Sendo g e H o gradiente e a hessiana, respectivamente, de $f_\rho(\cdot)$ em um dado par (x, ρ) , então temos:

$$p = -H^{-1}g, \quad (3.47)$$

$$p_A = \sum_{i=1}^m p_i \tilde{A}_i$$

O segundo e terceiro critérios são dados por:

$$\|p\|_H \equiv (p^\top H p)^{\frac{1}{2}}, \quad \|\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}\| \quad (3.48)$$

A medida de distância $\|p\|_H$ é utilizada como forma padrão na análise de algoritmos de pontos interiores, ver [44], por exemplo. A métrica $\|\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}\|$ pode ser considerada como a norma do passo de Newton na matriz $\tilde{A}(\cdot)$.

Lema 10 *Temos que:*

$$\rho - \rho^* \leq (\rho - c^\top x(\rho)) \left[1 + \frac{n}{q}\right] \quad (3.49)$$

Demonstração: O centro exato $x(\rho)$ minimiza a função distância 3.42 para ρ . As condições necessárias e suficientes para $x(\rho)$ são dadas em 3.8, 3.9 e 3.10. Dessas condições, obtemos que $x(\rho)$ é primal viável. Além disso, usando que $\rho^* \geq -\text{Tr} A_0 U(\rho)$, segue que:

$$\rho^* - c^\top x(\rho) \geq -c^\top x(\rho) - \text{Tr} A_0 U(\rho) = -\frac{n}{q}(\rho - c^\top x(\rho)) \quad (3.50)$$

onde a última igualdade vem da condição de otimalidade dada por 3.10, ou seja:

$$\begin{aligned} A(x) \bullet U &= (A^\top x + A_0) \bullet U \\ &= (x^\top A U + A_0 \bullet U) \\ &= c^\top x + A_0 \bullet U \\ &= \frac{\rho - c^\top x}{q} n \end{aligned}$$

Então, temos, de 3.50:

$$-\rho^* \leq \frac{n}{q}(\rho - c^\top x(\rho)) - c^\top x(\rho) \quad (3.51)$$

Somando ρ em ambos os lados de 3.51, chegamos ao resultado desejado. ■

Lema 11 *Para um dado x e ρ , temos que $\|p\|_H = \|\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}\|$.*

Demonstração:

Temos que:

$$\begin{aligned} \|\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}\|^2 &= \left\| \sum_{i=1}^m p_i \tilde{A}(x)^{-\frac{1}{2}} \tilde{A}_i \tilde{A}(x)^{-\frac{1}{2}} \right\|^2 \\ &= \text{Tr} \left(\sum_{i=1}^m p_i \tilde{A}(x)^{-\frac{1}{2}} \tilde{A}_i \tilde{A}(x)^{-\frac{1}{2}} \right) \left(\sum_{j=1}^m p_j \tilde{A}(x)^{-\frac{1}{2}} \tilde{A}_j \tilde{A}(x)^{-\frac{1}{2}} \right) \\ &= \sum_{i=1}^m \sum_{j=1}^m p_i p_j \text{Tr} \left(\tilde{A}(x)^{-\frac{1}{2}} \tilde{A}_i \tilde{A}(x)^{-\frac{1}{2}} \right) \left(\tilde{A}(x)^{-\frac{1}{2}} \tilde{A}_j \tilde{A}(x)^{-\frac{1}{2}} \right) \\ &= \sum_{i=1}^m \sum_{j=1}^m p_i H_{ij} p_j \\ &= p^\top H p \\ &= \|p\|_H^2 \end{aligned}$$

■

Passemos a analisar, agora, o problema dado por:

$$\min \quad \sigma(\tilde{U}) \quad (3.52)$$

s.a :

$$\tilde{A}\tilde{U} = 0 \quad (3.53)$$

$$\tilde{U} \succeq 0$$

onde $\sigma(\tilde{U}) = \|\tilde{A}(x)^{\frac{1}{2}} \tilde{U} \tilde{A}(x)^{\frac{1}{2}} - I\|^2$ e $\tilde{A}(x)$ é dado por 3.11, seção 3.1.

Lema 12 *As condições de otimalidade de primeira ordem para 3.52 e 3.53 são dadas por:*

$$\tilde{A}(x)\tilde{U}\tilde{A}(x) - \tilde{A}(x) = \sum_{i=1}^m \alpha_i \tilde{A}_i \quad (3.54)$$

$$\tilde{A}\tilde{U} = 0, U \succeq 0 \quad (3.55)$$

onde $\alpha_i \in \Re, \forall i = 1 \dots, m$.

Demonstração:

Temos que:

$$\begin{aligned} \sigma(\tilde{U}) &= \|\tilde{A}(x)^{\frac{1}{2}}\tilde{U}\tilde{A}(x)^{\frac{1}{2}} - I\|^2 \\ &= \sum_{i,j=1}^{n+q} (\tilde{A}(x)_{i\bullet}^{\frac{1}{2}}\tilde{U}\tilde{A}(x)_{\bullet j}^{\frac{1}{2}})^2 + \sum_{i=1}^{n+q} (1 - 2\tilde{A}(x)_{i\bullet}^{\frac{1}{2}}\tilde{U}\tilde{A}(x)_{\bullet i}^{\frac{1}{2}}), \end{aligned} \quad (3.56)$$

onde $\tilde{A}(x)_{i\bullet}$ e $\tilde{A}(x)_{\bullet j}$ são a i -ésima linha e j -ésima coluna de $\tilde{A}(x)$, respectivamente.

Dados $w \in \Re^{n+q}$ e $v \in \Re^{n+q}$, obtemos:

$$w^\top \tilde{U} v = \sum_{i=1}^{n+q} \sum_{j=1}^{n+q} w_i \tilde{U}_{ij} v_j,$$

então,

$$\frac{\partial w^\top \tilde{U} v}{\partial \tilde{U}_{kl}} = w_k v_l, \quad k, l = 1, \dots, n+q. \quad (3.57)$$

Portanto, aplicando a regra da cadeia e 3.57 a 3.56, obtemos:

$$\frac{\partial \sigma(\tilde{U})}{\partial \tilde{U}_{kl}} = 2(\tilde{A}(x)\tilde{U}_{kl}\tilde{A}(x) - \tilde{A}(x))_{kl}, \quad k, l = 1, \dots, n+q. \quad (3.58)$$

E, dado que:

$$Tr(\tilde{A}_i \tilde{U}) = \sum_{j=1}^m Tr(\tilde{A}_i)_{j\bullet} \tilde{U}, \quad (3.59)$$

onde $(\tilde{A}_i)_{j\bullet}$ é a j -ésima linha de \tilde{A}_i . Então:

$$\frac{\partial Tr(\tilde{A}_i \tilde{U})}{\partial \tilde{U}_{kl}} = (\tilde{A}_i)_{kl}. \quad (3.60)$$

Denominando por $\alpha \in \mathbb{R}^m$ o multiplicador associado à igualdade em 3.53, obtemos a função lagrangeana dada por:

$$L(\tilde{U}, \alpha) = \|\tilde{A}(x)^{-\frac{1}{2}}\tilde{U}\tilde{A}(x)^{-\frac{1}{2}} - I\|^2 - \alpha^\top \tilde{A}\tilde{U}, \quad \tilde{A}(x) \succeq 0. \quad (3.61)$$

O lema segue imediatamente de 3.58, 3.60 e 3.61. \blacksquare

Lema 13 Para x e ρ dados, $\delta(x, \rho) = \|\tilde{A}(x)^{-\frac{1}{2}}\tilde{p}_A\tilde{A}(x)^{-\frac{1}{2}}\|$.

Demonstração: É suficiente mostrar que $U(x, \rho)$ é uma solução ótima para 3.52. Além disso, uma vez que $\sigma(\cdot)$ é uma função convexa, é suficiente mostrar que $U(x, \rho)$ satisfaz as condições de otimalidade de primeira ordem 3.54 e 3.55. Temos, ainda, por 3.44 e 3.47, temos:

$$\begin{aligned} g_i = -\sum_{j=1}^m H_{ij}p_j &= -\sum_{j=1}^m \text{Tr}(\tilde{A}(x)^{-1}\tilde{A}_j\tilde{A}(x)^{-1}\tilde{A}_i)p_j \\ &= -\text{Tr}\tilde{A}(x)^{-1}\left(\sum_{j=1}^m p_j\tilde{A}_j\right)\tilde{A}(x)^{-1}\tilde{A}_i \\ &= -\text{Tr}\tilde{A}(x)^{-1}p_A\tilde{A}(x)^{-1}\tilde{A}_i \end{aligned} \quad (3.62)$$

e que de 3.43, temos:

$$g = \tilde{A}\tilde{A}(x)^{-1} \quad (3.63)$$

A equação 3.63 implica em :

$$\begin{aligned} g_i &= -\langle \tilde{A}_i, \tilde{A}(x)^{-1} \rangle \\ &= -\text{Tr}\tilde{A}_i\tilde{A}(x)^{-1} \end{aligned} \quad (3.64)$$

Portanto, de 3.62 e de 3.64, obtemos:

$$-\text{Tr}\tilde{A}_i\tilde{A}(x)^{-1} = -\text{Tr}\tilde{A}(x)^{-1}p_A\tilde{A}(x)^{-1}\tilde{A}_i, \quad (3.65)$$

de que se deduz:

$$\text{Tr}(\tilde{A}(x)^{-1} - \tilde{A}(x)^{-1}p_A\tilde{A}(x)^{-1})\tilde{A}_i = 0. \quad (3.66)$$

Consideremos, agora, a substituição da variável:

$$\tilde{U}(x, \rho) := \tilde{A}(x)^{-1} - \tilde{A}(x)^{-1} p_A \tilde{A}(x)^{-1} = \tilde{A}(x)^{-1} (\tilde{A}(x) - p_A) \tilde{A}(x)^{-1}, \quad (3.67)$$

que leva a:

$$\text{Tr} \tilde{U}(x, \rho) \tilde{A}_i = 0, \quad i = 1, \dots, m.$$

Substituindo 3.67 em 3.54, obtemos:

$$\begin{aligned} \tilde{A}(x) \tilde{A}(x)^{-1} (\tilde{A}(x) - p_A) \tilde{A}(x)^{-1} - \tilde{A}(x) &= \sum_{i=1}^m \alpha_i \tilde{A}_i \\ \tilde{A}(x) - p_A - \tilde{A}(x) &= \sum_{i=1}^m \alpha_i \tilde{A}_i \\ -p_A &= \sum_{i=1}^m \alpha_i \tilde{A}_i \end{aligned} \quad (3.68)$$

Portanto, de 3.54 e 3.68, temos:

$$\tilde{A}(x) \tilde{U}(x, \rho) \tilde{A}(x) - \tilde{A}(x) = -p_A. \quad (3.69)$$

Multiplicando 3.69 por $\tilde{A}(x)^{-\frac{1}{2}}$ pela esquerda e pela direita, chegamos a:

$$\tilde{A}(x)^{-\frac{1}{2}} (\tilde{A}(x) \tilde{U}(x, \rho) \tilde{A}(x) - \tilde{A}(x)) \tilde{A}(x)^{-\frac{1}{2}} = -\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}},$$

obtendo:

$$\tilde{A}(x)^{\frac{1}{2}} \tilde{U}(x, \rho) \tilde{A}(x)^{\frac{1}{2}} - I = -\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}} \quad (3.70)$$

■

Lema 14 *Suponha que $\delta(x, \rho) < 1$ e que $x^* = x + p$. Então $\tilde{A}(x^*) \succ 0$ e $\delta(x^*, \rho) \leq \delta(x, \rho)^2$.*

Demonstração: Lembremos que δ é definido em 3.45 e que o raio espectral de uma matriz quadrada G é dado por $\varrho(G) = \max\{|\lambda| : \lambda \in \lambda(G)\}$, onde $\lambda(G)$ indica o conjunto dos auto-valores de G . Temos, então, que:

$$\begin{aligned}\tilde{A}(x^*) &= \tilde{A}(x + p) \\ &= \tilde{A}(x) + p_A \\ &= \tilde{A}(x)^{\frac{1}{2}}(I + \tilde{A}(x)^{-\frac{1}{2}}p_A\tilde{A}(x)^{-\frac{1}{2}})\tilde{A}(x)^{\frac{1}{2}} \\ &:= \tilde{A}(x)^{\frac{1}{2}}\varrho(I + V)\tilde{A}(x)^{\frac{1}{2}}.\end{aligned}$$

Porém, sabemos que:

$$\varrho(V) = \varrho(\tilde{A}(x)^{-\frac{1}{2}}p_A\tilde{A}(x)^{-\frac{1}{2}}) = \|\tilde{A}(x)^{-\frac{1}{2}}p_A\tilde{A}(x)^{-\frac{1}{2}}\| < 1. \quad (3.71)$$

em que a igualdade vem do fato de p_A ser simétrica e a desigualdade vem da hipótese inicial de que $\delta(x, \rho) < 1$. Portanto, retornando à expressão obtida para $\tilde{A}(x)$ acima, concluímos que $\tilde{A}(x) \succ 0$. Dando continuidade à demonstração, pela definição de $\delta(x, \rho)$ em 3.45, temos:

$$\delta(x^*, \rho)^2 \leq \|(\tilde{A}(x) + p_A)^{\frac{1}{2}}\tilde{U}(x, \rho)(\tilde{A}(x) + p_A)^{\frac{1}{2}} - I\|^2 \quad (3.72)$$

Mas,

$$\|(\tilde{A}(x) + p_A)^{\frac{1}{2}}\tilde{U}(x, \rho)(\tilde{A}(x) + p_A)^{\frac{1}{2}} - I\| = \|(\tilde{A}(x) + p_A)\tilde{U}(x, \rho) - I\| \quad (3.73)$$

Substituindo 3.67 em 3.73, obtemos:

$$\|(\tilde{A}(x) + p_A)^{\frac{1}{2}}\tilde{U}(x, \rho)(\tilde{A}(x) + p_A)^{\frac{1}{2}} - I\|.$$

Que é equivalente a:

$$\|(\tilde{A}(x) + p_A)(\tilde{A}(x)^{-1} - \tilde{A}(x)^{-1}p_A\tilde{A}(x)^{-1}) - I\|.$$

Finalmente, obtemos:

$$\|\tilde{A}(x)^{-\frac{1}{2}}(\tilde{A}(x)^{-\frac{1}{2}}p_A\tilde{A}(x)^{-1}p_A\tilde{A}(x)^{-\frac{1}{2}})\tilde{A}(x)^{-\frac{1}{2}}\|. \quad (3.74)$$

Portanto, de 3.72 e 3.74, obtemos:

$$\begin{aligned}
\delta(x^*, \rho) &\leq \left(\sum_{i=1}^m \lambda_i^2 (\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-1} p_A \tilde{A}(x)^{-\frac{1}{2}}) \right)^{\frac{1}{2}} \\
&= \left(\sum_{i=1}^m \lambda_i^4 (\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}) \right)^{\frac{1}{2}} \\
&\leq \sum_{i=1}^m \lambda_i^2 (\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}) \\
&= \|\tilde{A}(x)^{-\frac{1}{2}} p_A \tilde{A}(x)^{-\frac{1}{2}}\|^2 \\
&= \delta(x, \rho)^2
\end{aligned}$$

■

Lema 15 Se $\delta := \delta(x, \rho) < 1$, então $f_\rho(x) - f_\rho(x(\rho)) \leq \frac{\delta^2}{1-\delta^2}$.

Demonstração: A função $f_\rho(\cdot)$ é convexa em x , portanto:

$$f_\rho(x+p) \geq f_\rho(x) + p^\top g,$$

Como $p = -H^{-1}g$, temos:

$$p^\top g = -p^\top H p = -\delta^2$$

Onde a última igualdade vem de 3.70. Fazendo as substituições apropriadas, obtemos:

$$f_\rho(x) - f_\rho(x+p) \leq \delta^2 \tag{3.75}$$

Agora, seja $x^0 := x$ e definimos por x^1, x^2, \dots , a sequência de pontos obtidos por passos de Newton iniciando em x^0 . Pelo lema 14, temos:

$$\delta(x^i, \rho) \leq \delta(x^0, \rho)^{2^i} = \delta^{2^i} \tag{3.76}$$

Então, usando 3.75, temos:

$$\begin{aligned}
f_\rho(x) - f_\rho(x(\rho)) &= \sum_{i=0}^{\infty} f_\rho(x^i) - f_\rho(x^{i+1}) \\
&\leq \sum_{i=0}^{\infty} \delta(x^i, \rho)^2 \\
&\leq \sum_{i=0}^{\infty} \delta^{2^{i+1}} \\
&\leq \sum_{i=0}^{\infty} \delta^{2(i+1)} \\
&= \frac{\delta^2}{1 - \delta^2}
\end{aligned}$$

■

Lema 16 *Se $\delta := \delta(x, \rho) < 1$ e $q \geq 2\sqrt{n}$, então:*

$$\rho - c^\top x(\rho) \leq \left(1 + \frac{2\sqrt{n}}{q} \frac{\delta}{1 - \delta}\right) (\rho - c^\top x)$$

Demonstração: Sabemos que:

$$\begin{aligned}
p^\top g &= p^\top \left(q \frac{1}{\rho - c^\top x} c - \mathcal{A}A(x)^{-1} \right) \\
&= q \frac{1}{\rho - c^\top x} c^\top p - p^\top \mathcal{A}A(x)^{-1}
\end{aligned} \tag{3.77}$$

Uma vez que temos $p^\top g = \delta^2$, substituímos em 3.77 e obtemos:

$$q \frac{1}{\rho - c^\top x} c^\top p - p^\top \mathcal{A}A(x)^{-1} = \delta^2 \tag{3.78}$$

Fazendo as devidas manipulações em 3.78, obtemos:

$$\begin{aligned}
q \frac{1}{\rho - c^\top x} c^\top p &= \delta^2 + p^\top \mathcal{A}A(x)^{-1} \\
&= \delta^2 + \langle p, \mathcal{A}A(x)^{-1} \rangle \\
&= \delta^2 + \langle A(p), A(x)^{-1} \rangle \\
&= \delta^2 + \langle p_A, A(x)^{-1} \rangle \\
&= \delta^2 + \langle A(x)^{-\frac{1}{2}} p_A A(x)^{-\frac{1}{2}}, I \rangle
\end{aligned} \tag{3.79}$$

Aplicando a desigualdade de Cauchy-Schwartz a 3.79, obtemos:

$$\begin{aligned} |\langle A(x)^{-\frac{1}{2}} p_A A(x)^{-\frac{1}{2}}, I \rangle| &\leq \|A(x)^{-\frac{1}{2}} p_A A(x)^{-\frac{1}{2}}\| \|I\| \\ &= \delta \sqrt{n} \end{aligned}$$

A partir deste resultado, concluimos que:

$$\begin{aligned} q \frac{1}{\rho - c^\top x} c^\top p &\leq \delta^2 + \delta \sqrt{n} \\ c^\top p &\leq \frac{(\rho - c^\top x)}{q} (\delta^2 + \delta \sqrt{n}) \\ &\leq \delta(1 + \delta) \frac{(\rho - c^\top x)}{q} \sqrt{n} \\ &\leq \frac{2\delta(\rho - c^\top x) \sqrt{n}}{q} \end{aligned}$$

Novamente, seja $x^0 := x$ e x^1, x^2, \dots , a sequência de pontos obtidos pela repetição de passos de Newton iniciando em x^0 . De acordo com o lema 14, esta sequência converge para $x(\rho)$ e, além disso:

$$\delta(x^k, \rho) \leq \delta(x^{k-1}, \rho)^2 \leq \dots \leq \delta(x^0, \rho)^{2^k} = \delta(x, \rho)^{2^k}$$

Portanto, podemos escrever:

$$\begin{aligned} \rho - c^\top x^{k+1} &= \rho - c^\top (x^k + p(x^k, \rho)) \\ &= \rho - c^\top x^k - c^\top p(x^k, \rho) \\ &\leq \rho - c^\top x^k + \frac{2\delta(\rho - c^\top x)}{q} \sqrt{n} \\ &= (1 + 2\delta(x^k, \rho) \frac{\sqrt{n}}{q}) (\rho - c^\top x^k) \\ &\leq (1 + 2\delta^{2^k} \frac{\sqrt{n}}{q}) (\rho - c^\top x^k) \\ &\leq (\rho - c^\top x^0) \prod_{i=0}^k (1 + 2\delta^{2^i} \frac{\sqrt{n}}{q}) \end{aligned}$$

Em particular, temos:

$$\begin{aligned} \rho - c^\top x(\rho) &\leq (\rho - c^\top x(\rho)) \prod_{i=0}^{\infty} \left(1 + 2\delta^{2^i} \frac{\sqrt{n}}{q}\right) \\ &\leq (\rho - c^\top x) \prod_{i=0}^{\infty} \left(1 + \frac{2\sqrt{n}}{q} \frac{\delta}{1-\delta}\right) \end{aligned}$$

Onde a última desigualdade vem do lema B.4 em [19]. ■

3.3.2 Complexidade

Nesta seção, apresentamos um limite superior para o número total de iterações de Newton. Apresentamos um limite superior tanto para as iterações internas, quanto para as externas. O produto desses limites superiores fornecem o limite superior para o total de iterações de Newton.

Teorema 6 *Depois de*

$$\frac{2}{\theta} \left(1 + \frac{n}{q}\right) \ln \frac{4(1 + \frac{n}{q})(\rho^0 - \rho^*)}{\epsilon}$$

Iterações, o algoritmo de centros pára com uma solução ϵ -ótima.

Demonstração: Seja ρ^k o limite superior na k -ésima iteração e x^k o ponto obtido ao final da k -ésima iteração. Usando os lemas 10 e 16 e o fato de que $q \geq 2\sqrt{n}$, obtemos:

$$\begin{aligned} \rho^{k-1} - \rho &\leq \left(1 + \frac{n}{q}\right) (\rho^{k-1} - c^\top x(\rho^{k-1})) \\ &\leq \left(1 + \frac{2\sqrt{n}}{q}\right) \left(1 + \frac{n}{q}\right) (\rho^{k-1} - c^\top x(\rho^{k-1})) \\ &\leq 2\left(1 + \frac{n}{q}\right) (\rho^{k-1} - c^\top x(\rho^{k-1})) \end{aligned} \tag{3.80}$$

Além disso,

$$\begin{aligned}
\frac{\rho^k - \rho^*}{\rho^{k-1} - \rho^*} &= \frac{(\rho^{k-1} + \theta(c^\top x^{k-1} - \rho^{k-1})) - \rho^*}{\rho^{k-1} - \rho^*} \\
&= \frac{\rho^{k-1} - \rho^*}{\rho^{k-1} - \rho^*} + \theta \frac{(c^\top x^{k-1} - \rho^{k-1})}{\rho^{k-1} - \rho^*} \\
&= 1 + \theta \frac{(c^\top x^{k-1} - \rho^{k-1})}{\rho^{k-1} - \rho^*} \\
&\leq 1 + \theta \frac{(c^\top x^{k-1} - \rho^{k-1})}{2(1 + \frac{n}{q})(\rho^{k-1} - c^\top x(\rho^{k-1}))} \\
&= 1 - \theta \frac{1}{2(1 + \frac{n}{q})} \\
&=: 1 - \theta' \tag{3.81}
\end{aligned}$$

onde a última desigualdade vem de 3.80. Portanto, depois de k iterações, obtemos:

$$\begin{aligned}
\rho^k - c^\top x^k &\leq \rho^k - \rho^* \\
&\leq (1 - \theta')(\rho^{k-1} - \rho^*) \\
&\leq (1 - \theta')^k(\rho^0 - \rho^*)
\end{aligned}$$

onde a última desigualdade vem de 3.81. Isso significa que:

$$\rho^K - c^\top x^K \leq \frac{\epsilon}{4(1 + \frac{n}{q})},$$

que é garantido, se for verificado

$$(1 - \theta')^K(\rho^0 - \rho^*) \leq \frac{\epsilon}{4(1 + \frac{n}{q})}.$$

Aplicando logaritmo a ambos os termos, obtemos:

$$\begin{aligned}
\ln(1 - \theta')^K(\rho^0 - \rho^*) &\leq \ln \frac{\epsilon}{4(1 + \frac{n}{q})} \\
\ln(1 - \theta')^K + \ln(\rho^0 - \rho^*) &\leq \ln \frac{\epsilon}{4(1 + \frac{n}{q})} \\
K \ln(1 - \theta') + \ln(\rho^0 - \rho^*) &\leq \ln \frac{\epsilon}{4(1 + \frac{n}{q})} \\
K \ln(1 - \theta') &\leq \ln \frac{\epsilon}{4(1 + \frac{n}{q})} - \ln(\rho^0 - \rho^*)
\end{aligned}$$

Multiplicando ambos os termos por -1 , obtemos:

$$\begin{aligned} -K \ln(1 - \theta') &\geq -\ln \frac{\epsilon}{4(1 + \frac{n}{q})} + \ln(\rho^0 - \rho^*) \\ -K \ln(1 - \theta') &\geq \ln \frac{4(1 + \frac{n}{q})(\rho^0 - \rho^*)}{\epsilon} \end{aligned} \quad (3.82)$$

Dado que $-\ln(1 - \theta') > \theta'$, então 3.82, certamente, será verdadeiro se K satisfizer a:

$$K > \frac{1}{\theta'} \ln \frac{4(1 + \frac{n}{q})(\rho^0 - \rho^*)}{\epsilon} = \frac{2}{\theta'} (1 + \frac{n}{q}) \ln \frac{4(1 + \frac{n}{q})(\rho^0 - \rho^*)}{\epsilon}$$

Se o algoritmo de centros pára, então, aplicando 3.81, temos:

$$\rho - c^\top x \leq \rho^k - \rho^0 \leq 2(1 + \frac{n}{q})(\rho^k - c^\top x) \leq \epsilon,$$

que significa que x é uma solução ϵ -ótima. Isto demonstra o teorema. ■

Lema 17 *Seja $\bar{\alpha} = (1 + \delta)^{-1}$, então*

$$\Delta f_\rho := f_\rho(x) - f_\rho(x + \bar{\alpha}p) \geq \delta - \ln(1 + \delta).$$

Demonstração: Sabemos que $0 < \delta < 1$ e $\alpha \geq 0$. Além disso, temos que:

$$g^\top p = -p^\top H p = -\delta^2 \quad (3.83)$$

A expansão de Taylor para a função f_ρ em relação a α é dada por:

$$f_\rho(x + \alpha p) = f_\rho(x) + \alpha g^\top p + \frac{1}{2} \alpha^2 p^\top H p + \sum_{k=3}^{\infty} t_k \quad (3.84)$$

onde t_k define o termo de k -ésima ordem na série de Taylor. Então podemos escrever t_k como:

$$t_k = \frac{(-\alpha)^k}{k} \langle p, \mathcal{A}A(x)^{-1} \rangle$$

logo,

$$|t_k| = \frac{(-\alpha)^k}{k} \langle p, \mathcal{A}A(x)^{-1} \rangle \leq \frac{\alpha^k \delta^k}{k} \quad (3.85)$$

aplicando 3.85 e 3.83 em 3.84, obtemos:

$$\begin{aligned}
\alpha(x + \alpha p, \rho) &\leq \alpha(x, \rho) + \alpha g^\top p + \frac{1}{2} \alpha^2 p^\top H p + \sum_{k=3}^{\infty} t_k \\
&= f_\rho(x) - \alpha \delta^2 + \frac{1}{2} \alpha^2 \delta^2 - \alpha \delta - \frac{\alpha^2 \delta^2}{2} - \ln(1 - \alpha \delta) \\
&= f_\rho(x) - \alpha \delta^2 - \alpha \delta - \ln(1 - \alpha \delta)
\end{aligned}$$

usamos como suposição que:

$$\alpha \delta < 1 \quad (3.86)$$

Portanto temos:

$$\begin{aligned}
f_\rho(x) - f_\rho(x + \alpha p) &\geq \alpha(\delta^2 + \delta) + \ln(1 - \alpha \delta) \\
\Delta f_\rho &\geq \alpha(\delta^2 + \delta) + \ln(1 - \alpha \delta) \quad (3.87)
\end{aligned}$$

O lado direito será máximo se $\alpha = \bar{\alpha} = (1 + \delta)^{-1}$. Este valor para α também satisfaz a condição 3.86. Substituindo $\bar{\alpha}$ em 3.87, obtemos:

$$\Delta f_\rho \geq \delta - \ln(1 + \delta)$$

■

Teorema 7 *Cada iteração externa exige, no mínimo*

$$\frac{11}{3} + 11q\theta \left(\frac{\theta}{1 - \theta} + \frac{2\sqrt{n}}{q + 2\sqrt{n}} \right)$$

iterações internas.

Demonstração: Definimos por ρ^k , o limite superior na iteração k e por ρ^{k+1} , o limite superior obtido a partir de ρ^k . O valor em cada iteração é definido por x . Portanto x é centralizado em relação a $x(\rho^*)$ e $\rho^{k+1} = \rho^k + \theta(c^\top x - \rho^k)$. Para cada iteração interna, sabemos, de acordo com o lema 17, que a função distância diminui pelo menos de:

$$\Delta f_\rho \geq \frac{1}{2} - \ln\left(1 + \frac{1}{2}\right) > \frac{1}{11} \quad (3.88)$$

Dado que $\delta \geq \frac{1}{2}$ em cada iteraçao. Seja N o numero de iteraçoes internas, entao devido a 3.88 e da definiçao de $x(\rho^{k+1})$, temos que:

$$\frac{1}{11}N \leq f_{\rho^{k+1}}(x) - f_{\rho^{k+1}}(x(\rho^{k+1})). \quad (3.89)$$

Definimos, por simplicidade:

$$\Phi(x, \rho^{k+1}) = f_{\rho^{k+1}}(x) - f_{\rho^{k+1}}(x(\rho^{k+1})).$$

De acordo com o teorema do valor medio, existe um $\hat{\rho}$ e (ρ^k, ρ^{k+1}) , tal que:

$$\Phi(x, \rho^{k+1}) = \Phi(x, \rho^k) + \left. \frac{d\Phi(x, \rho)}{d\rho} \right|_{\rho=\hat{\rho}} (\rho^{k+1} - \rho^k) \quad (3.90)$$

Analisando $\frac{d\Phi(x, \rho)}{d\rho}$, obtemos:

$$\frac{d\Phi(x, \rho)}{d\rho} = q \frac{1}{\rho - c^\top x}$$

E obtendo a derivada de $x(\rho)$ em relaao a ρ , temos:

$$\frac{d\Phi(x(\rho), \rho)}{d\rho} = q \frac{1}{\rho - c^\top x(\rho)}$$

Entao:

$$\begin{aligned} \left. \frac{d\Phi(x, \rho)}{d\rho} \right|_{\rho=\hat{\rho}} &= q \left(\frac{1}{\rho - c^\top x} - \frac{1}{\rho - c^\top x(\rho)} \right) \Big|_{\rho=\hat{\rho}} \\ &\leq q \left(\frac{1}{\rho^{k+1} - c^\top x} - \frac{1}{\rho^k - c^\top x(\rho^k)} \right). \end{aligned}$$

onde a ultima desigualdade vem do fato que $\rho^{k+1} > \hat{\rho}$ e do lema 8. Substituindo este resultado em 3.90, temos:

$$\begin{aligned} \Phi(x, \rho^{k+1}) &\leq \Phi(x, \rho^k) + q \left(\frac{1}{\rho^{k+1} - c^\top x} - \frac{1}{\rho^k - c^\top x(\rho^k)} \right) (\rho^k - \rho^{k+1}) \\ &= \Phi(x, \rho^k) + q\theta \left(\frac{1}{1-\theta} - \frac{\rho^k - c^\top x}{\rho^k - c^\top x(\rho^k)} \right) \\ &\leq \frac{1}{3} + q\theta \left(\frac{1}{1-\theta} - \frac{1}{1 + \frac{2\sqrt{n}}{q}} \right) \\ &= \frac{1}{3} + q\theta \left(\frac{\theta}{1-\theta} + \frac{2\sqrt{n}}{q + 2\sqrt{n}} \right) \end{aligned} \quad (3.91)$$

onde a última desigualdade vem do fato que $\Phi(x, \rho^k) \leq \frac{1}{3}$ de acordo com o lema 15 e:

$$\rho^k - c^\top x(\rho^k) \leq \left(1 + \frac{2\sqrt{n}}{q}\right) (\rho^k - c^\top x). \quad (3.92)$$

conferindo com o lema 16. Combinando 3.91 e 3.89, conclui-se a demonstração do lema. ■

3.4 Aspectos Computacionais

Nesta seção, apresentamos como obter um ponto inicial satisfazendo as condições de viabilidade e de proximidade para o algoritmo de centros analíticos aplicado para a programação semidefinida.

3.4.1 Obtenção da viabilidade

Para obtermos uma solução estritamente viável, escolhemos um ponto $x^0 \in \Re^m$ qualquer e tomamos $t^0 > -\min\{\lambda_{\min}(A(x^0)), 0\}$. Em seguida, passamos a resolver o problema dado por:

(Pt) :

$$\min \quad t \quad (3.93)$$

s.a :

$$A(x) + tI \succeq 0$$

$$t \geq 0$$

$$x \in \Re^m$$

A solução para o problema 3.93 é dada por $t = 0$. No entanto, estamos interessados apenas em obter uma solução viável para o problema original

3.2, portanto, quando uma solução satisfizer $\lambda_{\min}(A(x^0)) > 0$, a fase 1 do algoritmo encerra e obtemos uma solução inicial estritamente viável. Outro ponto importante da fase 1, está relacionada na atualização do limite superior. Ao invés de diminuirmos a cota de ρ ao longo da iterações, passamos a aumentá-la, ou seja:

$$\rho^{k+1} = \rho^k + \theta(\rho^k - c^\top x^k), \quad 0 < \theta < 1$$

Como a região inicial da fase 1 do algoritmo de centros é inviável para o problema original 3.93, a idéia com o aumento do limite superior é que seja feita uma expansão na região de busca até que um ponto da região viável para 3.93 seja obtido. Ou seja, o objetivo do algoritmo de fase 1 é o de minimizar as inviabilidades.

Após obtermos uma solução inicial viável, passamos para o algoritmo 3.2 de segunda fase.

3.4.2 Proximidade inicial

Outro ponto importante a ser satisfeito no algoritmo de centros é na obtenção de um ponto inicial satisfazendo ao critério de proximidade. Ou seja,

$$\delta_\rho(x^0) = (\nabla f_\rho(x^0)^\top H^{-1}(x^0) \nabla f_\rho(x^0))^{\frac{1}{2}} < \epsilon, \quad 0 < \epsilon < 1 \quad (3.94)$$

Para obter 3.94, fazemos uma perturbação na hessiana inicial de uma fator α^{-1} , onde $\alpha > 0$. Portanto, obtemos:

$$\delta_\rho(x^0) = (\nabla f_\rho(x^0)^\top \alpha H^{-1}(x^0) \nabla f_\rho(x^0))^{\frac{1}{2}} < \epsilon, \quad 0 < \epsilon < 1 \quad (3.95)$$

E 3.95 implica em:

$$\alpha < \frac{\epsilon^2}{(\nabla f_\rho(x^0)^\top H^{-1}(x^0) \nabla f_\rho(x^0))}, \quad 0 < \epsilon < 1 \quad (3.96)$$

Desse modo, podemos obter uma solução inicial que satisfaça ao critério de proximidade inicial dado por 3.94 com baixo custo computacional.

Capítulo 4

Uma nova barreira para PSDP

Apresentamos neste capítulo uma barreira aplicada ao problema primal com a matriz variável canalizada. Esta barreira foi desenvolvida para métodos de pontos interiores em Quiroz e Oliveira [51] e neste trabalho apresentamos a sua extensão para Programação Semidefinida. Apresentamos o algoritmo e uma importante aplicação de programação semidefinida: a obtenção da soma dos k maiores auto-valores de uma matriz semidefinida positiva, ver [1] e [26].

4.1 Definição do Problema e da Barreira Hipercúbica para Programação Semidefinida Positiva

Seja o problema dado por:

(PSD_c) :

$$\min \quad f(X) \quad (4.1)$$

s.a :

$$\mathcal{A}X = b$$

$$0 \preceq X \preceq I, \quad X \in S^n$$

onde $f : S^n \rightarrow R$ é uma função convexa contínua e duas vezes diferenciável, \mathcal{A} é o operador dado por:

$$\mathcal{A}X = \begin{pmatrix} \langle A_1, X \rangle \\ \vdots \\ \langle A_m, X \rangle \end{pmatrix} \quad (4.2)$$

I representa a matriz identidade n por n , b é um vetor m -dimensional e X é a matriz variável. Todos os resultados obtidos neste trabalho podem ser estendidos para conjuntos gerais, dados por $\alpha I \preceq X \preceq \beta I$, $\alpha \geq 0$ e $\beta > \alpha$.

Denotemos por:

$$P = \{X \in S^n : \mathcal{A}X = b, 0 \preceq X \preceq I\} \quad (4.3)$$

o conjunto viável do problema 4.1 e por

$$P^0 = \{X \in S^n : \mathcal{A}X = b, 0 \prec X \prec I\} \quad (4.4)$$

o seu interior relativo. Fazemos as seguintes suposições a respeito do problema 4.1:

1. O conjunto P^0 é não-vazio;
2. O núcleo da hessiana da função objetivo é um espaço vetorial;
3. As matrizes $A_i, \forall i = 1, \dots, m$ são linearmente independentes.

A primeira suposição é padrão para métodos de pontos interiores. A segunda é importante para demonstrar a convergência da trajetória central primal para o centro analítico do problema 4.1. A terceira é necessária para simplificar os argumentos. Uma vez que f é uma função contínua no conjunto compacto P , esta função atinge um ponto mínimo global em P e, devido à convexidade da função, todo mínimo local é global. Assim, o conjunto de soluções ótimas do problema 4.1 é um conjunto convexo não-vazio e limitado.

Lema 18 *Os auto-valores de X pertencentes ao conjunto P e P^0 satisfazem, respectivamente:*

$$0 \leq \lambda_i(X) \leq 1, \quad \forall i = 1, \dots, n$$

$$0 < \lambda_i(X) < 1, \quad \forall i = 1, \dots, n$$

Demonstração: Imediata. ■

A formulação dual de Wolfe para 4.1 é:

(PSD_c) :

$$\max \quad d(X, y, S, W) \tag{4.5}$$

s.a :

$$\mathcal{A}^\top y + S - W = \nabla f(X)$$

$$W, S \succeq 0 \quad X, S, W \in S^n, y \in R^m$$

onde $d(X, y, S, W) = f(X) - \langle y, AX - b \rangle + \langle W, X - I \rangle - \langle S, X \rangle$. Um ponto viável (X, y, S, W) para 4.5, com $W, S \succ 0$ é chamado de uma solução interior dual viável. A seguir, apresentamos uma função barreira para $[0, I]$. Esta função generaliza a que foi proposta para métodos de pontos interiores em [51].

$$B(X) = 2[\text{Tr}X \text{Ln}X + \text{Tr}(I-X) \text{Ln}(I-X)] - \text{Ln} \det X - \text{Ln} \det(I-X) \quad (4.6)$$

Esta barreira possui as seguintes propriedades:

- Se X se aproxima da fronteira, então $B(X) \rightarrow \infty$.
- $B(X) \geq 0, \forall X \in [0, I]$.
- $B \in C^\infty$ e, usando a definição 3 e tomando $U := I$, a derivada de segunda ordem de B é $2[I \odot X^{-1} + I \odot (I-X)^{-1}] + X^{-1} \odot X^{-1} + (I-X)^{-1} \odot (I-X)^{-1}$.

Portanto, $B(X)$ é uma função barreira estritamente convexa e infinitamente diferenciável em $[0, I]$.

4.2 Cálculo do Gradiente e Hessiana de $B(X)$

A seguir, apresentamos as derivadas de primeira e segunda ordem de $B(X)$.

A demonstração pode ser encontrada no apêndice A na seção A.5.

$$\begin{aligned} DB(X)[H] &= \text{Tr}\{2[\text{Ln}X(I-X)^{-1}]H\} - \\ &\quad \text{Tr}\{X^{-1}H\} + \text{Tr}\{(I-X)^{-1}H\} \end{aligned} \quad (4.7)$$

$$\begin{aligned} D^2B(X)[H, H] &= \text{Tr}\{2H[X^{-1} + (I-X)^{-1}]H\} + \text{Tr}\{X^{-1}HX^{-1}H\} + \\ &\quad \text{Tr}\{(I-X)^{-1}H(I-X)^{-1}H\} \end{aligned} \quad (4.8)$$

4.3 A Trajetória Central

Nesta seção, apresentamos um novo problema penalizado para resolver 4.1 e estudamos o comportamento da trajetória central (primal-dual), obtida pelas condições de otimalidade KKT. Uma das características que vamos explorar é que a trajetória primal é viável em relação ao problema 4.1, enquanto que a trajetória dual é inviável em relação ao problema 4.5 por um termo $\mu[\text{Ln}X(\mu) - \text{Ln}(I - X(\mu))]$ que converge para zero quando μ converge a zero.

Seja então:

(PSD_c) :

$$\min \phi_B(X, \mu) = f(X) + \mu B(X) \quad (4.9)$$

s.a :

$$AX = b$$

$$0 \prec X \prec I \quad X \in S^n$$

onde $B(X)$ é dado por 4.6 e μ é um parâmetro positivo. As derivadas de primeira e segunda ordem de ϕ_B são:

$$\begin{aligned} g = g(X, \mu) &= \nabla f(X) + 2\mu[\text{Ln}X - \text{Ln}(I - X)] \\ &\quad - \mu X^{-1} + \mu(I - X)^{-1} \end{aligned} \quad (4.10)$$

$$\begin{aligned} H = H(X, \mu) &= \nabla^2 f(X) + 2\mu[I \odot X^{-1} + I \odot (I - X)^{-1}] + \mu X^{-1} \odot X^{-1} \\ &\quad + \mu(I - X)^{-1} \odot (I - X)^{-1} \end{aligned} \quad (4.11)$$

Dado que $f(X)$ é convexa e a segunda parcela de 4.11 é uma matriz simétrica definida positiva, a função ϕ_B é estritamente convexa no interior relativo do conjunto viável e leva a valores infinitos na fronteira de P . Esta função atinge o valor mínimo neste domínio (para um μ fixo) em um único ponto, que é

definido por $X(\mu)$. As condições de otimalidade de primeira ordem para $X(\mu)$ são:

$$\mathcal{A}^\top y + S - W = \nabla f(X) + 2\mu[\text{Ln}X - \text{Ln}(I - X)] \quad (4.12)$$

$$\mathcal{A}X = b \quad (4.13)$$

$$XS = \mu I \quad (4.14)$$

$$(I - X)W = \mu I \quad (4.15)$$

$$S, W \in S_n^+ \quad (4.16)$$

$$y \in R^m \quad (4.17)$$

$$0 \prec X \prec I \quad (4.18)$$

A solução única deste sistema é definida por $(X(\mu), y(\mu), S(\mu), W(\mu))$ e é chamada de trajetória central. Como notado acima, esta trajetória é viável para o problema primal 4.1 e inviável para o problema dual 4.5.

Lema 19 *Seja $X(\mu_i) := \arg \min\{\phi_B(X, \mu_i) \text{ s.a.: } \mathcal{A}X = b, 0 \prec X \prec I\}$, $\forall i = 1, 2$, e seja $0 < \mu_2 < \mu_1$. Então, os resultados a seguir são verdadeiros:*

1. $\phi_B(X(\mu_2), \mu_2) < \phi_B(X(\mu_1), \mu_1)$;
2. $B(X(\mu_1)) \leq B(X(\mu_2))$;
3. $f(X(\mu_2)) \leq f(X(\mu_1))$;

Demonstração: Considerando o fato que $B \geq 0$, temos:

1. Temos que $X(\mu_1)$ e $X(\mu_2)$ são dados por:

$$X(\mu_1) := \arg \min\{\phi_B(X, \mu_1) \text{ s.a.: } \mathcal{A}X = b, 0 \prec X \prec I\}$$

$$X(\mu_2) := \arg \min\{\phi_B(X, \mu_2) \text{ s.a.: } \mathcal{A}X = b, 0 \prec X \prec I\}$$

Por definição, temos:

$$\phi_B(X(\mu_2), \mu_2) \leq \phi_B(X, \mu_2), \forall X \in P^0$$

Em particular, se tomarmos $X = X(\mu_1)$, então:

$$\phi_B(X(\mu_2), \mu_2) \leq \phi_B(X(\mu_1), \mu_2)$$

Por definição:

$$\phi_B(X(\mu_1), \mu_2) = f(X(\mu_1)) + \mu_2 B(X(\mu_1))$$

Como $0 < \mu_2 < \mu_1$, então:

$$f(X(\mu_1)) + \mu_2 B(X(\mu_1)) < f(X(\mu_1)) + \mu_1 B(X(\mu_1))$$

Por definição, temos:

$$f(X(\mu_1)) + \mu_1 B(X(\mu_1)) = \arg \min \{ \phi_B(X(\mu_1), \mu_1) \text{ s.a: } \mathcal{A}X = b, 0 \prec X \prec I \}$$

E, portanto, concluímos a demonstração do primeiro resultado do lema.

2. Temos que $\phi_B(X(\mu_1), \mu_2)$ e $\phi_B(X(\mu_2), \mu_1)$ são dados, respectivamente, por:

$$\phi_B(X(\mu_1), \mu_2) = f(X(\mu_1)) + \mu_2 B(X(\mu_1))$$

$$\phi_B(X(\mu_2), \mu_1) = f(X(\mu_2)) + \mu_1 B(X(\mu_2))$$

Temos que:

$$\mu_2 B(X(\mu_2)) \leq \phi_B(X(\mu_1), \mu_2) - f(X(\mu_2)) \quad (4.19)$$

$$\mu_1 B(X(\mu_1)) \leq \phi_B(X(\mu_2), \mu_1) - f(X(\mu_1)) \quad (4.20)$$

Desenvolvendo 4.19 e 4.20, respectivamente, temos:

$$\mu_2 B(X(\mu_2)) \leq f(X(\mu_1)) + \mu_2 B(X(\mu_1)) - f(X(\mu_2)) \quad (4.21)$$

$$\mu_1 B(X(\mu_1)) \leq f(X(\mu_2)) + \mu_1 B(X(\mu_2)) - f(X(\mu_1)) \quad (4.22)$$

Somando as expressões 4.21 e 4.22, temos:

$$\mu_1 B(X(\mu_1)) + \mu_2 B(X(\mu_2)) \leq \mu_1 B(X(\mu_2)) + \mu_2 B(X(\mu_1)) \quad (4.23)$$

Desenvolvendo 4.23, temos:

$$\begin{aligned} \mu_1 B(X(\mu_1)) - \mu_2 B(X(\mu_1)) &\leq \mu_1 B(X(\mu_2)) - \mu_2 B(X(\mu_2)) \\ (\mu_1 - \mu_2) B(X(\mu_1)) &\leq (\mu_1 - \mu_2) B(X(\mu_2)) \end{aligned}$$

Como $0 < \mu_2 < \mu_1$, então:

$$B(X(\mu_1)) \leq B(X(\mu_2)).$$

E, portanto, concluímos a demonstração do segundo resultado do lema.

3. Vamos demonstrar, agora, o último resultado.

Por definição, temos:

$$\phi_B(X(\mu_2), \mu_2) = f(X(\mu_2)) + \mu_2 B(X(\mu_2))$$

Sabemos, ainda, que:

$$f(X(\mu_2)) + \mu_2 B(X(\mu_2)) \leq f(X) + \mu_2 B(X), \forall X \in P^0 \quad (4.24)$$

em particular, façamos $X = X(\mu_1)$:

$$f(X(\mu_2)) + \mu_2 B(X(\mu_2)) \leq f(X(\mu_1)) + \mu_2 B(X(\mu_1))$$

$$f(X(\mu_2)) - f(X(\mu_1)) \leq \mu_2 (B(X(\mu_1)) - B(X(\mu_2)))$$

como $B(X(\mu_1)) \leq B(X(\mu_2))$ e $\mu_2 > 0$ logo $f(X(\mu_2)) \leq f(X(\mu_1))$.

■

Lema 20 *Se X^* for a solução ótima do problema 4.1 e $\mu > 0$, então:*

$$f(X^*) \leq f(X(\mu)) \leq \phi_B(X(\mu)), \forall X \in P^0.$$

Demonstração: É suficiente considerar que $f(X^*) \leq f(X(\mu)), \forall \mu > 0$ e $B \geq 0$. ■

Proposição 3 Para todo $r > 0$, o conjunto $\{(X(\mu), S(\mu), W(\mu)) | \mu < r\}$ é limitado.

Demonstração: Uma vez que $X(\mu) \in [0, I]$, temos que $X(\mu)$ é limitado, em particular, quando $\mu < r$. Agora, vamos provar que o subconjunto $\{(S(\mu), W(\mu)) | \mu < r\}$ também é limitado. Seja $0 < \mu < r$. Sabemos que $(X(\mu), S(\mu), W(\mu))$ resolve a equação:

$$\mathcal{A}^\top y + S - W = \nabla f(X) + 2\mu[LnX - Ln(I - X)]$$

então, definindo:

$$\begin{aligned} L(X, y, S, W) = f(X) &- Tr\{y(\mathcal{A}X - b)\} - TrSX \\ &+ Tr\{W(X - I)\} + 2\mu[XLnX - (I - X)Ln(I - X)] \end{aligned}$$

temos que:

$$\nabla_X L(X(\mu), y(\mu), S(\mu), W(\mu)) = 0 \quad (4.25)$$

onde ∇_X , define o gradiente de L em relação a variável X .

Dado que $L(\cdot, y(\mu), S(\mu), W(\mu))$ é uma função convexa em $[0, I]$ e usando 4.25, temos que:

$$X(\mu) \in \arg \min\{L(X, y(\mu), S(\mu), W(\mu)), X \in [0, 1]\}.$$

Supondo que o conjunto P^0 seja não-vazio, então passamos a calcular a diferença dada por:

$$f(X(\mu)) - L(X(\mu), y(\mu), S(\mu), W(\mu)) \quad (4.26)$$

que, por sua vez, é igual a:

$$\begin{aligned} TrS(\mu)X(\mu) - TrW(\mu)(X(\mu) - I) - 2\mu Tr[X(\mu)Ln(X(\mu)) \\ + (I - X(\mu))Ln(I - X(\mu))] \end{aligned} \quad (4.27)$$

Aplicando 4.14 e 4.15 à expressão 4.27, temos:

$$n\mu + n\mu - 2\mu Tr[X(\mu)Ln(X(\mu)) + (I - X(\mu))Ln(I - X(\mu))], \quad (4.28)$$

uma vez que os auto-valores da matriz $X(\mu)Ln(X(\mu)) + (I - X(\mu))Ln(I - X(\mu))$ são dados por

$$\lambda_i(X(\mu)) \ln(\lambda_i(X(\mu))) + (1 - \lambda_i(X(\mu))) \ln(1 - \lambda_i(X(\mu))) \quad \forall i = 1, \dots, n$$

tomando um $\lambda_k(X(\mu)) = \gamma$ particular, temos:

$$\psi(\gamma) = \gamma \ln(\gamma) + (1 - \gamma) \ln(1 - \gamma) \quad (4.29)$$

agora, aplicado a condição de otimalidade de primeira ordem em $\psi(\gamma)$, temos:

$$\psi'(\gamma) = 0$$

obtendo $\gamma = \frac{1}{2} = \lambda_k(X(\mu))$, uma vez que $0 < \lambda_k(X(\mu)) < 1$. Substituindo este resultado em 4.29, obtemos o seu ponto máximo, que é dado por $-\ln 2$ e, como a matriz possui ordem n , então obtemos a desigualdade em relação a 4.28 dada por:

$$\begin{aligned} &\leq n\mu + n\mu + 2n\mu \ln 2 \\ &= 2n\mu(1 + \ln 2) \end{aligned} \quad (4.30)$$

Portanto, associando 4.26 e 4.30, temos:

$$\begin{aligned} f(X(\mu)) - 2n\mu(1 + \ln 2) &\leq L(X(\mu), y(\mu), S(\mu), W(\mu)) \\ &\leq L(X^0, y(\mu), S(\mu), W(\mu)) \\ &\leq f(X^0) - TrS(\mu)X^0 + TrW(\mu)(X^0 - I) \\ &= f(X^0) - TrS(\mu)X^0 - TrW(\mu)(I - X^0) \end{aligned}$$

Assim,

$$f(X(\mu)) - 2n\mu(1 + \ln 2) \leq f(X^0) - TrS(\mu)X^0 - TrW(\mu)(I - X^0) \quad (4.31)$$

A última desigualdade ocorre porque X^0 é primal viável e $Tr[X^0 Ln(X^0) + (I - X^0)Ln(I - X^0)] \leq 0$. Uma vez que $0 < \lambda_i(X^0) < 1, \forall i = 1, \dots, n$, o valor definido por :

$$\xi = \min\{\lambda_i(X^0), (1 - \lambda_i(X^0)), i = 1, \dots, n\} \quad (4.32)$$

é positivo. Agora, seja f^* o valor ótimo do problema 4.1. De 4.31, temos que:

$$TrS(\mu)X^0 + TrW(\mu)(I - X^0) \leq f(X^0) - f(X(\mu)) + 2n\mu(1 + \ln 2)$$

usando o teorema 1, temos:

$$TrS(\mu)X^0 + TrW(\mu)(I - X^0) \geq \lambda_{\min}(X^0)TrS(\mu) + (1 - \lambda_{\min}(X^0))TrW(\mu)$$

aplicando o teorema 1, temos:

$$\begin{aligned} \lambda_{\min}(X^0)TrS(\mu) + (1 - \lambda_{\min}(X^0))TrW(\mu) \\ \geq \lambda_{\min}(X^0)\|S(\mu)\| + (1 - \lambda_{\min}(X^0))\|W(\mu)\| \end{aligned}$$

agora, usando 4.32, temos:

$$\begin{aligned} \lambda_{\min}(X^0)\|S(\mu)\| + (1 - \lambda_{\min}(X^0))\|W(\mu)\| &\geq \epsilon(\|S(\mu)\| + \|W(\mu)\|) \\ \epsilon(\|S(\mu)\| + \|W(\mu)\|) &\leq f(X^0) - f(X^*) + 2n\mu(1 + \ln 2) \\ \epsilon(\|S(\mu)\| + \|W(\mu)\|) &\leq f(X^0) - f(X^*) + 2nr(1 + \ln 2) \end{aligned}$$

onde $\|\cdot\|$ é a norma de Frobenius e a última desigualdade ocorre porque $\mu \leq r$. Portanto, o conjunto $\{(X(\mu), S(\mu), W(\mu)) | \mu < r\}$ é limitado. ■

Devemos observar que não podemos provar que a trajetória $\{y(\mu)\}$ é limitada, uma vez que a função $Tr\{X(\mu) - Ln(1 - X(\mu))\}$ não é limitada no intervalo $[0, 1]$. No restante deste capítulo, usaremos a seguinte notação:

$$(X^k, y^k, S^k, W^k) = (X(\mu_k), y(\mu_k), S(\mu_k), W(\mu_k))$$

O próximo teorema, é uma propriedade bem conhecida para barreiras positivas gerais.

Teorema 8 *Seja (μ_k) uma sequência de números reais positivos, tais que $\mu_k \rightarrow 0, k \rightarrow \infty$. Então:*

1. $\mu_k B(X^k, \mu_k) \rightarrow 0$, quando $k \rightarrow \infty$;
2. Qualquer ponto de acumulação da sequência $\{X^k\}$ é uma solução ótima para o problema 4.1;

Demonstração: Ver [6]. ■

Corolário 3 *Seja $(X^k), k \in M \subset N$ uma subsequência, tal que $X^k \rightarrow X^*$, onde X^* é a solução ótima do problema 4.1. Então:*

$$\mu_k [\ln \lambda_i(X^k) - \ln(1 - \lambda_i(X^k))] \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n$$

Demonstração: Pelo teorema anterior, temos que:

$$\begin{aligned} 2\mu_k [Tr(X^k Ln X^k) + Tr((I - X^k) Ln(I - X^k))] - \mu_k Ln \det X^k \\ - \mu_k Ln \det(I - X^k) \rightarrow 0, k \rightarrow \infty \end{aligned}$$

definindo $b(\lambda_i(X^k)) = (2\lambda_i(X^k) - 1)[\ln \lambda_i(X^k) - \ln(1 - \lambda_i(X^k))] \geq 0$, obtemos que:

$$\mu_k b(\lambda_i(X^k)) \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n$$

ou, de forma equivalente, temos:

$$\begin{aligned} 2\mu_k [\lambda_i(X^k) \ln \lambda_i(X^k) + (1 - \lambda_i(X^k)) \ln(1 - \lambda_i(X^k))] - \mu_k [\ln \lambda_i(X^k) \\ - \ln(1 - \lambda_i(X^k))] \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n \quad (4.33) \end{aligned}$$

A sequência $\gamma(\lambda_i(X^k)) = \lambda_i(X^k) \ln \lambda_i(X^k) + (1 - \lambda_i(X^k)) \ln(1 - \lambda_i(X^k))$ é limitada e uma vez que $\mu_k \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n$, temos que:

$$2\mu_k \gamma(\lambda_i(X^k)) \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n \quad (4.34)$$

subtraindo a expressão 4.34 de 4.33 e usando a propriedade da subtração de duas sequências convergentes, obtemos:

$$-\mu_k \ln \lambda_i(X^k) - \mu_k \ln(1 - \lambda_i(X^k)) \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n$$

Dado que $\lambda_i(X^k) \in [0, 1]$, temos que $-\mu_k \ln \lambda_i(X^k) > 0$ e $-\mu_k \ln(1 - \lambda_i(X^k)) > 0, \forall i = 1, \dots, n$. Temos, então que:

$$\mu_k \ln \lambda_i(X^k) \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n \quad (4.35)$$

$$-\mu_k \ln(1 - \lambda_i(X^k)) \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n \quad (4.36)$$

Adicionando 4.35 e 4.36, concluimos que:

$$\mu_k [\ln \lambda_i(X^k) - \ln(1 - \lambda_i(X^k))] \rightarrow 0, k \rightarrow \infty, \forall i = 1, \dots, n$$

■

4.4 Aplicação para $f(X) = TrCX$

Apresentamos, agora, a barreira $B(X)$ aplicada ao problema de programação semidefinida dado por:

(PSD_p) :

$$\min \quad TrCX \quad (4.37)$$

s.a :

$$AX = b$$

$$0 \preceq X \preceq I, \quad X \in S^n$$

Para este problema, mantemos as mesmas hipóteses dadas para o problema 4.1. A formulação dual deste problema é:

(PSD_d) :

$$\max \quad d(y, S, W) \quad (4.38)$$

s.a :

$$A^\top y + S - W = C$$

$$W, S \succeq 0 \quad S, W \in S^n, y \in R^m$$

onde $d(y, S, W) = \langle y, b \rangle - TrW$. Obteremos, agora, o intervalo (gap) de dualidade:

$$gap := C \bullet X - (\langle b, y \rangle - TrW) = Tr\{XS\} + Tr\{(I - X)W\} \quad (4.39)$$

Segue que $d(y, S, W) \leq z^* \leq TrCX$, onde z^* é definido como o valor ótimo de 4.37. Se X^* for a solução ótima de 4.37, então (y^*, S^*, W^*) é a solução ótima de 4.38, com $Tr\{CX^*\} = d(y^*, S^*, W^*) = z^*$. Essas soluções são caracterizadas pelas condições de Karush-Kuhn-Tucker (KKT):

$$A^\top y + S - W = C$$

$$AX = b$$

$$XS = 0$$

$$(I - X)W = 0$$

$$S, W \in S_+^n$$

$$y \in R^m$$

$$0 \prec X \prec I$$

Passamos a considerar, agora, a função barreira para o problema 4.37:

$$\phi(X, \mu) = \frac{1}{\mu}(TrCX) + 2[TrX LnX + Tr(I - X)Ln(I - X)] - Ln \det X - Ln \det(I - X) \quad (4.40)$$

onde μ é um parâmetro positivo. As derivadas de primeira e de segunda ordem de 4.40 são:

$$g = g(X, \mu) := \frac{1}{\mu}C + \{2[LnX - Ln(I - X)] - X^{-1} + (I - X)^{-1}\} \quad (4.41)$$

$$H = H(X, \mu) := \{2[I \odot X^{-1} + I \odot (I - X)^{-1}] + X^{-1} \odot X^{-1} + (I - X)^{-1} \odot (I - X)^{-1}\} \quad (4.42)$$

Uma vez que $\phi(X, \mu)$ é uma função estritamente convexa no interior relativo do conjunto estritamente viável e tende ao infinito na fronteira do conjunto P , esta função atinge o valor mínimo neste domínio (para um μ fixo) em um único ponto $X(\mu)$. As condições de otimalidade, necessária e suficientes, de primeira e segunda ordem, para $X(\mu)$ são:

$$\mathcal{A}^\top y + S - W = C + 2\mu[LnX - Ln(I - X)]$$

$$\mathcal{A}X = b$$

$$XS = \mu I$$

$$(I - X)W = \mu I$$

$$S, W \in S_n^+$$

$$y \in R^m$$

$$0 \prec X \prec I$$

Definimos por $(X(\mu), y(\mu), S(\mu), W(\mu))$, a única solução deste sistema. O intervalo de dualidade desta solução é dado por:

$$\begin{aligned} gap(\mu) &:= Tr\{CX(\mu)\} - d(X(\mu), y(\mu), S(\mu), W(\mu)) \\ &= 2\mu(n - TrX(\mu)LnX(\mu) + Tr\{X(\mu)Ln(I - X(\mu))\}) \end{aligned} \quad (4.43)$$

O lema a seguir mostra que a função objetivo primal diminui ao longo da trajetória primal e a função objetivo dual aumenta ao longo da trajetória

dual. Para esta demonstração, utilizaremos a função barreira logarítmica de regularização primal-dual para o problema 4.38:

$$\phi_B^d(X, y, W, S, \mu) = \frac{1}{\mu}(-\langle y, b \rangle + \text{Tr}W) - \text{Tr}LnS - \text{Tr}LnW - 2\text{Tr}Ln(I - X)$$

e o problema dual:

$$\min \quad \phi_B^d(X, y, W, S, \mu) \quad (4.44)$$

s.a :

$$\mathcal{A}^\top y + S - W = C = 2\mu \text{Tr}\{LnX - Ln(I - X)\}$$

Lema 21 *Se μ diminui, então a função objetivo $\text{Tr}CX$ do problema primal 4.37 é monotonamente decrescente e a função objetivo dual $d(y, S, W)$ 4.38 é monotonamente crescente.*

Demonstração: A primeira parte do lema é um resultado clássico de Fiacco e McCormick, ver [10] (Consultar também o lema na terceira parte). Para demonstrar a segunda parte, primeiro devemos observar que $\phi_B^d(X, y, W, S, \mu)$ é estritamente convexa. As condições de Karush-Kuhn-Tucker para obter um ponto mínimo são:

$$\mathcal{A}^\top y + S - W = C + 2\mu[LnX - Ln(I - X)]$$

$$\mu \mathcal{A}S^{-1} = b$$

$$(I - \mu S^{-1}) = \mu W^{-1}$$

$$2(I - X)^{-1} = 2\mu(X^{-1} + (I - X)^{-1})S^{-1}$$

Portanto, concluímos que o ponto $(X(\mu), y(\mu), W(\mu), S(\mu))$ é um minimizador de $\phi_B^d(X, y, W, S, \mu)$. Agora, podemos utilizar o resultado clássico de Fiacco e McCormick para demonstrar a segunda parte do lema. ■

4.4.1 O Algoritmo

Nesta seção, apresentamos o algoritmo para resolver o problema dado por:

$$\min \phi_B(X) = TrCX + \mu B(X)$$

s.a :

$$\mathcal{A}X = b$$

$$0 \prec X \prec I$$

A aproximação de Newton para este problema é dada por:

$$\min Tr\{\nabla\phi_B(X)\Delta X\} + \frac{1}{2}Tr\{\nabla^2\phi_B(X)\Delta X, \Delta X\} \quad (4.45)$$

s.a :

$$\mathcal{A}\Delta X = 0$$

As condições de otimalidade para este problema são:

$$\begin{aligned} \nabla\phi_B(X) + \nabla^2\phi_B(X)\Delta X + \sum_{i=1}^n A_i y_i &= 0 \\ \mathcal{A}\Delta X &= 0 \end{aligned}$$

Resolvendo este sistema, obtemos:

$$y = - \left[\begin{array}{c} \mathcal{A}(A_1 \nabla^2\phi_B(X)^{-1})^\top \\ \vdots \\ \mathcal{A}(A_m \nabla^2\phi_B(X)^{-1})^\top \end{array} \right]^{-1} \mathcal{A}(\nabla^2\phi_B(X)^{-1}\nabla\phi_B(X)) \quad (4.46)$$

e

$$\Delta X = -\nabla^2\phi_B(X)^{-1}(\nabla\phi_B(X) + \mathcal{A}^\top y). \quad (4.47)$$

Esta demonstração está no apêndice A, na seção A.6.

A seguir, apresentamos o algoritmo de passo longo seguidor da trajetória. Os escalares τ , θ e ϵ são parâmetros do algoritmo que devem ser especificados.

1. **Algoritmo:** ALGORITMO PRIMAL PARA BARREIRA HIPÉRCÚBICA
2. **Inicialização:**
3. $\tau < 1$;
4. $\theta \in (0, 1)$ (atualização da cota);
5. $\epsilon \in (0, 1)$ (proximidade);
6. $X^0 \in P^0 : \|\Delta X(X^0, \mu^0)\|_{H(X^0, \mu^0)} < \tau$;
7. $\mu_0 > 0$, valor da barreira inicial;
8. $k = 0$
9. $l = 0$
10. $X := X^0; \mu := \mu_0$
11. **enquanto** *gap* dado por 4.43 for maior que ϵ **faça**
12. $\mu^{l+1} := (1 - \theta)\mu^l$
13. **enquanto** $\|\Delta X(X^k, \mu^{l+1})\|_{H(X^k, \mu^{l+1})} > \tau$ **faça**
14. **calcular** ΔX **por** 4.47
15. $\lambda = \arg \min_{\lambda \geq 0} \{\phi_B(X^k + \lambda \Delta X) : X^k + \lambda \Delta X \in P^0\}$
16. $X^{k+1} = X^k + \lambda \Delta X$
17. $k = k + 1$
18. **fim**
19. $l = l + 1$
20. **fim**

Tabela 4.1: algoritmo para barreira Hipércúbica $B(X)$

O algoritmo exige uma solução inicial X^0 e um valor inicial μ^0 que satisfaçam o critério de proximidade $\|\Delta X(X^0, \mu^0)\|_{H(X^0, \mu^0)} < \tau$, onde $\tau < 1$.

4.4.2 Aplicação

Uma importante aplicação da barreira $B(X)$ está no problema de caracterização máxima de uma matriz simétrica $n \times n$ real, ver [48]. Este problema está relacionado à obtenção do número de Lovász, ver [1] e [26], que, por sua vez, é usado em otimização combinatória para obter limites para o Problema do Clique Máximo.

O Problema da Caracterização Máxima de uma matriz simétrica pode ser especificado como:

Dada uma matriz simétrica real n por n , A e $\kappa \in \{1, \dots, n\}$. Definimos os

auto-valores de A por $\lambda_1, \dots, \lambda_n$, satisfazendo:

$$\lambda_1 \geq \dots \geq \lambda_n \quad (4.48)$$

O problema consiste em determinar:

$$f_\kappa(A) = \sum_{i=1}^{\kappa} \lambda_i \quad (4.49)$$

Note que $f_1(A)$ é o maior auto-valor de A . Apresentamos, ainda, o conjunto

$\phi_{n,\kappa}$:

$$\phi_{n,\kappa} = \{U \in S_n : 0 \preceq U \preceq I, \text{Tr}(U) = \kappa\} \quad (4.50)$$

O conjunto $\phi_{n,\kappa}$ é um subconjunto convexo e compacto de S_n . Mostra-se que:

$$f_\kappa(A) = \max_{U \in \phi_{n,\kappa}} \langle A, U \rangle \quad (4.51)$$

Este, por sua vez, aplica-se ao problema que apresentamos neste capítulo.

4.5 Aspectos Computacionais

Nesta seção, apresentamos como obter um ponto inicial que satisfaça ao critério de proximidade dado por:

$$\|\Delta X(X^0, \mu^0)\|_{\nabla^2 \phi_B(X^0, \mu^0)} < \tau, \quad 0 < \tau < 1 \quad (4.52)$$

4.5.1 Proximidade inicial

A equação 4.52 é dada por:

$$(\text{Tr} \Delta X(X^0, \mu^0) \nabla^2 \phi_B(X^0, \mu^0) \Delta X(X^0, \mu^0))^{\frac{1}{2}} < \tau, \quad 0 < \tau < 1 \quad (4.53)$$

Para obter a proximidade desejada, fazemos uma perturbação na hessiana inicial por um parâmetro α^{-1} , onde $\alpha > 0$. Portanto, reescrevemos 4.53 com

o parâmetro de perturbação como:

$$(Tr \Delta X(X^0, \mu^0) \alpha^{-1} \nabla^2 \phi_B(X^0, \mu^0) \Delta X(X^0, \mu^0))^{\frac{1}{2}} < \tau, \quad 0 < \tau < 1 \quad (4.54)$$

Além disso, reescrevemos 4.47 utilizando o parâmetro de perturbação como:

$$\Delta X = -\alpha \nabla^2 \phi_B(X)^{-1} (\nabla \phi_B(X) + \mathcal{A}^\top y). \quad (4.55)$$

com y dado por 4.46.

Agora, aplicando 4.55 em 4.54, obtemos:

$$\alpha < \frac{\tau^2}{(Tr(\nabla \phi_B(X) + \mathcal{A}^\top y) \nabla^2 \phi_B^{-1}(X^0, \mu^0) (\nabla \phi_B(X) + \mathcal{A}^\top y))}, \quad 0 < \tau < 1$$

Deste forma, obtemos a proximidade inicial desejada com poucas operações e satisfazendo aos critérios que são necessários para garantir a convergência do método.

Capítulo 5

Problemas de Otimização Combinatória e SDP

Importantes problemas de otimização combinatória como os problemas de corte máximo, do clique máximo, da equipartição máxima, entre outros, são pertencentes à classe dos problemas NP-completos, ver [12]. A partir da observação de que a restrição $x_i \in \{0, 1\}$ poder ser escrita de forma equivalente como $x_i^2 - x_i = 0$, começou a se explorar a aproximação $x^\top x = X$ com a matriz X sendo semidefinida positiva e sem a restrição da matriz X possuir posto 1. Com isto, alguns problemas, para os quais as aproximações obtidas eram da ordem de 50%, passaram a possuir aproximações superiores a 80% com o uso da relaxação semidefinida, ver [13].

Neste capítulo, apresentamos alguns destes problemas e como obter as suas respectivas relaxações semidefinidas.

5.1 Relaxação de Programação Semidefinida para Problemas de Otimização Combinatória

As ferramentas usadas para obter relaxações semidefinidas para problemas de otimização combinatória seguem basicamente estes passos: reescrever o problema de otimização combinatória em um formato mais conveniente, usando notação matricial e aplicar a relaxação lagrangeana ou, de forma equivalente, aplicar a relaxação semidefinida. É importante reforçar que as duas técnicas são equivalentes como pode ser consultado em [32]. A seguir, apresentamos vários problemas de otimização combinatória e o processo para obter as suas respectivas relaxações semidefinidas. As três primeiras relaxações são baseadas no trabalho de [13].

5.1.1 Relaxação de Programação Semidefinida para o Problema do Corte Máximo

Seja $G = (V, E)$ um grafo completo, não-direcionado, com um conjunto de vértices $V = \{1, 2, \dots, n\}$ e um conjunto de arestas dado por:

$$E = \{(i, j) : i, j \in V, i < j\}.$$

Associamos um peso $C_{ij} = C_{ji}$ para cada aresta $(i, j) \in E$. O Problema do Corte Máximo consiste em encontrar uma partição (L, R) disjunta e completa de V , ou seja, uma partição tal que $L \cap R = \emptyset$ e $L \cup R = V$, de modo a maximizar o corte $c(L, R) = \sum_{i \in L, j \in R} C_{ij}$.

Introduzindo um vetor variável $u \in R^n$, podemos formular um programa quadrático não-convexo:

(CM) :

$$\max \frac{1}{2} \sum_{i < j} C_{ij} (1 - u_i u_j) \quad (5.1)$$

s.a :

$$u_i^2 = 1 \quad (1 \leq i \leq n)$$

Cada solução viável $u \in R^n$ deste problema corresponde a um corte (L, R) com $L = \{i \in V : u_i = -1\}$ e $R = \{i \in V : u_i = 1\}$. Se definirmos C como uma matriz simétrica com elementos $C_{ij} = C_{ji}$ ($(i, j) \in E$ e $C_{ii} = 0, 1 \leq i \leq n$), e a matriz simétrica, n por n , $A_0 = \text{Diag}(Ce) - C$, onde $e \in R^n$ define o vetor de uns e $\text{Diag}(Ce)$ é a matriz diagonal formada pelo vetor $Ce \in R^n$, podemos reescrever o problema quadrático acima como:

(CM_E) :

$$\max x^\top A_0 x \quad (5.2)$$

s.a :

$$x_i^2 = \frac{1}{4} \quad (1 \leq i \leq n)$$

Se $x \in R^n$ é uma solução viável do último programa quadrático, então podemos definir a matriz X simétrica n por n , semidefinida positiva, cujo (i, j) -ésimo elemento X_{ij} é dado por $X_{ij} = x_i x_j$ e $X_{ii} = \frac{1}{4}$ ($1 \leq i \leq n$), temos então $A_0 \bullet X = x^\top A_0 x$. Isto leva à seguinte relaxação de Programação Semidefinida do Problema de Corte Máximo:

(CM_R) :

$$\min -A_0 \bullet X \quad (5.3)$$

s.a :

$$E_{ii} \bullet X = \frac{1}{4} \quad (1 \leq i \leq n)$$

$$X \succeq 0$$

onde E_{ii} define a matriz simétrica n por n com o (ii) -ésimo elemento igual a 1 e todos os outros iguais a zero.

5.1.2 Relaxação de Programação Semidefinida para o Problema de Equipartição de Grafos

Seja $G = (V, E)$ um grafo completo, não-direcionado, com um conjunto de vértices $V = \{1, 2, \dots, n\}$ e um conjunto de arestas dado por:

$$E = \{(i, j) : i, j \in V, i < j\}.$$

Associamos um peso $C_{ij} = C_{ji}$ para cada aresta $(i, j) \in E$. O problema de equipartição de grafos consiste em encontrar uma partição uniforme (L, R) de V , isto é, uma partição (L, R) disjunta e completa de V com a mesma cardinalidade $|L| = |R| = \frac{n}{2}$ que minimiza o corte $c(L, R) = \sum_{i \in L, j \in R} C_{ij}$.

Introduzindo um vetor variável $u \in R^n$, podemos formular este problema como um programa quadrático não-convexo:

(GP) :

$$\min \frac{1}{2} \sum_{i < j} C_{ij} (1 - u_i u_j) \quad (5.4)$$

s.a :

$$(\sum_{i=1}^n u_i)^2 = 0$$

$$u_i^2 = 1 \quad (1 \leq i \leq n)$$

Semelhante ao que foi feito no problema de corte máximo, podemos derivar uma relaxação de programação semidefinida do problema de equipartição de grafos:

(GP_R) :

$$\min \quad A_0 \bullet X \quad (5.5)$$

s.a :

$$E_{ii} \bullet X = \frac{1}{4} \quad (1 \leq i \leq n)$$

$$E \bullet X = 0$$

$$X \succeq 0$$

onde A_0 e E_{ii} ($1 \leq i \leq n$) são as mesmas matrizes usadas no problema do corte máximo e E define a matriz n por n com todos os elementos iguais a 1.

5.1.3 Problema de k -Equipartições

Seja k e m inteiros e $G = (V, E)$ um grafo não-direcionado valorado com $n := km$ vértices, dado pela matriz de adjacências A . Definimos uma k -equipartição de V por uma partição de V em k subconjuntos (S_1, \dots, S_k) de cardinalidades iguais. Representamos por $Diag(x)$ o operador linear que forma uma matriz diagonal a partir do vetor x e o seu operador adjunto $diag(X)$ produz um vetor contendo os elementos da diagonal de X . Definimos por e_r (ou e se a dimensão estiver clara) o vetor consistindo de r uns. O problema de k -equipartições possui uma correspondência um para um com o conjunto

$$\mathcal{F}_k := \{Y : Ye_k = e_n, Y^\top e_n = me_k, y_{ij} \in \{0, 1\}\}$$

com a interpretação de que a coluna j de Y indica um membro no subconjunto S_j de uma k -partição.

Seja a matriz laplaceana $L := Diag(Ae_n) - A$ associada a G . O problema da k -equipartições ($k - GP$) é definido por:

(kGP) :

$$z^* := \min \frac{1}{2} \text{tr} Y^\top L Y \quad (5.6)$$

s.a :

$$Y \in \mathcal{F}_k$$

Este problema é NP-difícil. Existem diversos trabalhos que o abordam, em especial, destacamos o de Donath e Hoffman [9] com a introdução dos métodos de autovalores para obter relaxações para o (kGP), o qual foi, posteriormente aperfeiçoado por Rendl e Wolkowicz [52]. Alizadeh [1] mostrou que o modelo de autovalor de Donath e Hoffman pode ser interpretado como o dual de um programa semidefinido dado por:.

(kGP_A) :

$$z^* := \min \frac{1}{2} \text{tr} L X \quad (5.7)$$

s.a :

$$\text{diag}(X) = e_n$$

$$X e_n = m e_n$$

$$X \in S_n^+, \quad X \preceq mI$$

A seguir, apresentamos o modelo para programação semidefinida proposto por Karisch e Rendl [23]. A idéia básica consiste em linearizar $\text{tr} Y^\top L Y$ por $\text{tr} L X$, onde X corresponde a $Y Y^\top$. Definimos o conjunto Υ_k por:

$$\Upsilon_k := \{X : \exists Y \in \mathcal{F}_k, \text{ tal que } X = Y Y^\top\}.$$

Assim, podemos reescrever (kGP) por:

$$z^* := \min \left\{ \frac{1}{2} \text{tr} L X : X \in \text{conv}(\Upsilon_k) \right\}.$$

A relaxação proposta por Karischi e Rendl, ver [23], para (kGP) é:

(kGP_{KR}) :

$$z^* := \min \quad \frac{1}{2} \text{tr} LX \quad (5.8)$$

s.a :

$$\text{diag}(X) = e_n$$

$$Xe_n = me_n$$

$$X \in S_n^+$$

5.1.4 Relaxação de Programação Semidefinida para o Problema do Clique Máximo

Seja $G = (V, E)$ um grafo completo, não-direcionado, com um conjunto de vértices $V = \{1, 2, \dots, n\}$ e um conjunto de arestas $E = \{(i, j) : i, j \in V, i < j\}$. Um subconjunto C de V forma um clique de G se $(i, j) \in E$ para todo par de vértices distintos $i, j \in C$ tal que $i < j$. O problema do clique máximo consiste em encontrar um clique de cardinalidade máxima em G . É conhecido que o PPSD a seguir, fornece um limite superior para a cardinalidade do clique máximo de G , ver [26], capítulo 10, por exemplo.

($CMax$) :

$$\min \quad E \bullet X \quad (5.9)$$

s.a :

$$E_{ij} \bullet X = 0 \quad ((i, j) \notin E)$$

$$I \bullet X = 1$$

$$X \succeq 0$$

Neste modelo, E é a matriz n por n com todos os elementos iguais a 1 e E_{ij} é a matriz n por n com os (i, j) -ésimos elementos iguais a $\frac{1}{2}$ e todos os outros iguais a zero.

Note que esta formulação semidefinida possui $m = \frac{n(n-1)}{2} - |E| + 1$ restrições de igualdade, onde $|E|$ define a cardinalidade do conjunto E de arestas. Portanto, o número m de restrições de igualdade pode ser da ordem n^2 e muito maior que o tamanho n da matriz variável X . Por outro lado, todas as matrizes de restrições $E_{ij}((i, j) \notin E)$ e I são esparsas; $E_{ij}((i, j) \notin E)$ possui apenas dois elementos diferentes de zero, e I possui n elementos diferentes de zero.

5.1.5 Relaxação de Programação Semidefinida para o Problema de Atribuição Quadrática

Seja o conjunto $N = \{1, 2, \dots, n\}$ e as matrizes $n \times n$, $F = (f_{ij})$, $D = (d_{kl})$ e $C = (c_{ij})$, o Problema de Atribuição Quadrática (PAQ) é definido como:

$$\min_{p \in \Pi_N} \sum_{i=1}^n \sum_{j=1}^n f_{ij} d_{p(i)p(j)} + \sum_{i=1}^n c_{ip(i)}$$

onde Π_N é o conjunto de todas as permutações de N . Uma aplicação do (PAQ) está nos problemas de localização, onde a matriz $F = (f_{ij})$ é a matriz de fluxo, ou seja, f_{ij} é o fluxo de materiais da facilidade i à facilidade j , $D = (d_{kl})$ é a matriz distância, ou seja, d_{kl} representa a distância da localidade k para localidade l e $C = (c_{ij})$ é o custo de associar a facilidade i à localidade j . O custo de associar, simultaneamente, a facilidade i à localidade k e a facilidade j à localidade l é $f_{ij}d_{kl}$. O objetivo é encontrar uma associação de todas as facilidades às localidades (ou seja, uma permutação $p \in \Pi_N$), de tal

forma que o custo total da associação seja minimizado.

Existe uma correspondência um para um entre \prod_N e o conjunto de matrizes de permutação $X = (x_{ij})_{n \times n}$, onde $x_{ij} \in \{0, 1\}$. O problema de associação quadrática pode ser escrito como:

(PAQ) :

$$\min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{l=1}^n f_{ij} d_{kl} x_{ik} x_{jl} + \sum_{i,j=1}^n c_{ij} x_{ij} \quad (5.10)$$

s.a :

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= 1 & \forall_{i=1, \dots, n} \\ \sum_{i=1}^n x_{ij} &= 1 & \forall_{j=1, \dots, n} \\ x_{ij} &\in \{0, 1\}, i, j = 1, \dots, n \end{aligned}$$

Este problema pode ser formulado de forma equivalente trabalhando no espaço das matrizes de permutação, ou seja, onde cada elemento da matriz é zero ou um. O primeiro passo é escrever a matriz de permutação como uma matriz simétrica no espaço $n \times n$, $X_{ij} \in \{0, 1\}$, que pode ser escrito como $X_{ij}^2 - X_{ij} = 0$. Em seguida, adicionamos o conjunto de restrições de ortogonalidade dados por $XX^\top = X^\top X = I$ e o conjunto de restrições dado por $Xe = X^\top e = e$, onde e é definido como o vetor de uns em R^n . A interseção destas restrições, portanto, nos dá uma formulação equivalente para o problema 5.10. Para mais detalhes consultar [49][62][63].

(PAQ_ε) :

$$\min \operatorname{Tr}(FXDX^\top - 2CX^\top) \quad (5.11)$$

s.a :

$$XX^\top = X^\top X = I \quad (5.12)$$

$$Xe = X^\top e = e \quad (5.13)$$

$$X_{ij}^2 - X_{ij} = 0, \quad \forall i, j. \quad (5.14)$$

onde $F = (f_{ij})n \times n$, $D = (d_{ij})n \times n$ e $C = -\frac{1}{2}(c_{ij})n \times n$. Podemos substituir a restrição de associação 5.13 por $\|Xe - e\|^2 + \|X^\top e - e\|^2 = 0$, obtendo o novo modelo equivalente dado por:

(PAQ_o) :

$$\min \operatorname{Tr}(FXDX^\top - 2CX^\top) \quad (5.15)$$

s.a :

$$XX^\top = I \quad (5.16)$$

$$X^\top X = I \quad (5.17)$$

$$\|Xe - e\|^2 + \|X^\top e - e\|^2 = 0 \quad (5.18)$$

$$X_{ij}^2 - X_{ij} = 0, \quad \forall i, j. \quad (5.19)$$

Primeiramente, aplicamos a relaxação lagrangeana às restrições 5.18 e 5.19, associando os multiplicadores de lagrange u_0 e W_{ij} , respectivamente. Seja:

$$\begin{aligned} \mu_o = \min_{XX^\top = X^\top X = I} \max_{W, u_0} \{ & \operatorname{Tr}(FXDX^\top - 2CX^\top) + \sum_{ij} W_{ij}(X_{ij}^2 - X_{ij}) \\ & + u_0(\|Xe - e\|^2 + \|X^\top e - e\|^2) \}. \end{aligned} \quad (5.20)$$

O teorema de dualidade nos garante que:

$$\begin{aligned} \mu_o \geq \mu_L = \max_{W, u_0} \min_{XX^\top = X^\top X = I} \{ & \operatorname{Tr}(FXDX^\top - 2CX^\top) + \sum_{ij} W_{ij}(X_{ij}^2 - X_{ij}) \\ & + u_0(\|Xe - e\|^2 + \|X^\top e - e\|^2) \}. \end{aligned} \quad (5.21)$$

Agora, continuando com o processo de relaxação e homogenização da função objetivo, vamos multiplicá-la por um escalar x_0 restrito a 1, ou seja, $x_0 = 1$ e, com isso, aumentamos de uma unidade a dimensão do problema. A homogenização é feita para facilitar a transição para programação semidefinida. Logo:

$$\begin{aligned} \mu_o \geq \mu_L = \max_W \min_{XX^\top = X^\top X = I, x_0^2 = 1} \{ & Tr(FXD X^\top + W(X \circ X)^\top) \\ & + u_0(\|Xe\|^2 + \|X^\top e\|^2) - Trx_0(2C + W)X^\top \\ & - 2x_0u_0e^\top(X + X^\top)e + 2nu_0\}. \end{aligned} \quad (5.22)$$

Onde \circ é o produto de Hadamard (ver definição 2.4). Introduzindo um multiplicador de Lagrange w_0 à restrição em x_0 e multiplicadores de Lagrange S_b e S_o às restrições $X^\top X = I$ e $XX^\top = I$, respectivamente, obtemos o limite inferior μ_R , dado por:

$$\begin{aligned} \mu_o \geq \mu_L \geq \mu_R = \max_{W, S_b, S_o, u_0, w_0} \min_{X, x_0} \{ & TrFXDX^\top + u_0(\|Xe\|^2 + \|X^\top e\|^2) \\ & + TrW(X \circ X)^\top + w_0x_0^2 + TrS_bXX^\top + TrS_oXX^\top \\ & - Trx_0(2C + W)X^\top - 2x_0u_0e^\top(X + X^\top)e \\ & - w_0 - TrS_b - TrS_o + 2nu_0\}. \end{aligned} \quad (5.23)$$

Agora, agrupamos os termos quadráticos, lineares e constantes de 5.23. Relembrando a definição de $vec(\cdot)$ (ver 2.7), definimos por $x := vec(X)$, $y^\top := (x_0, x^\top)$ e $w^\top := (w_0, vec(W)^\top)$, então, reescrevemos 5.23 como:

$$\begin{aligned} \mu_R = \max_{w, S_b, S_o, u_0} \min_y \{ & y^\top [L_Q + Arrow(w) + B^0 Diag(S_b) + O^0 Diag(S_o) + u_0 D] y \\ & - w_0 - TrS_b - TrS_o \} \end{aligned} \quad (5.24)$$

Onde os termos L_Q , $Arrow(w)$, $B^0 Diag(S)$, $O^0 Diag(S)$ e D , são definidos como:

$$L_Q = \begin{bmatrix} 0 & -vec(C)^\top \\ -vec(C) & D \otimes F \end{bmatrix}$$

$$\text{Arrow}(w) := \begin{bmatrix} w_0 & -\frac{1}{2}w_{1:n^2}^\top \\ -\frac{1}{2}w_{1:n^2} & \text{Diag}(w_{1:n^2}) \end{bmatrix}$$

$$B^0\text{Diag}(S) := \begin{bmatrix} 0 & 0 \\ 0 & I \otimes S_b \end{bmatrix}$$

$$O^0\text{Diag}(S) := \begin{bmatrix} 0 & 0 \\ 0 & S_0 \otimes I \end{bmatrix}$$

$$D := \begin{bmatrix} n & -e^\top \otimes e^\top \\ -e \otimes e & I \otimes E \end{bmatrix} + \begin{bmatrix} n & -e^\top \otimes e^\top \\ -e \otimes e & E \otimes I \end{bmatrix}$$

Onde \otimes é definido como o produto de Kronecker (ver definição 2.5). Observe em 5.24, que o problema de minimização interna é limitado inferiormente, somente se a hessiana da forma quadrática for semidefinida positiva. Neste caso, a forma quadrática possui valor mínimo igual a zero. Assim, podemos definir o problema de programação semidefinida por:

(D_o) :

$$\max \quad -w_0 - \text{Tr}S_b - \text{Tr}S_0 \quad (5.25)$$

s.a :

$$L_Q + \text{Arrow}(w) + B^0\text{Diag}(S_b) + O^0\text{Diag}(S_0) + u_0D \succeq 0$$

Agora, obtemos a relaxação desejada de (QAP_o) como o dual Lagrangeano de (D_o) . Introduzimos a matriz variável dual $Y \succeq 0$ de dimensão $(n^2 + 1) \times (n^2 + 1)$ e obtemos o problema dual de (D_o) .

(SDP_o) :

$$\min \quad Tr L_Q Y \quad (5.26)$$

s.a :

$$b^0 \text{diag}(Y) = I, \quad o^0 \text{diag}(Y) = I$$

$$\text{arrow}(Y) = e_0, \quad Tr DY = 0$$

$$Y \succeq 0$$

Onde a matriz variável $Y \succeq 0$ é definida como:

$$Y = \begin{bmatrix} y_{00} & Y_0^\top \\ Y_0 & Z \end{bmatrix} = \begin{bmatrix} y_{00} & Y^{01} & \dots & Y^{0n} \\ y_{10} & Y^{11} & \dots & Y^{1n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n0} & Y^{n1} & \dots & Y^{nn} \end{bmatrix}$$

O operador $\text{arrow}(\cdot)$ é o operador adjunto de $\text{Arrow}(\cdot)$ e é definido como:

$$\text{arrow}(Y) := \text{diag}(Y) - (0, Y_{0,1:n^2})^\top \quad (5.27)$$

Os operadores bloco-0-diagonal e off-0-diagonal, ou seja, $b^0 \text{diag}(\cdot)$ e $o^0 \text{diag}(\cdot)$ são definidos por:

$$b^0 \text{diag}(Y) := \sum_{k=1}^n Y_{(k,\cdot),(k,\cdot)} \quad (5.28)$$

$$o^0 \text{diag}(Y) := \sum_{k=1}^n Y_{(\cdot,k),(\cdot,k)} \quad (5.29)$$

Eles são os operadores adjuntos de $B^0 \text{Diag}(\cdot)$ e $O^0 \text{Diag}(\cdot)$, respectivamente.

Esta demonstração mostra uma outra forma de se obter relaxação semidefinida para problemas de otimização combinatória, ou seja, usando relaxação Lagrangeana, enquanto que nos exemplos anteriores, usamos a relaxação geométrica. Para obter mais detalhes sobre a relação entre relaxação Lagrangeana e programação semidefinida, consultar [32].

Capítulo 6

Experiência Numérica

Neste capítulo, apresentamos resultados numéricos obtidos para os algoritmos propostos no capítulo 3, com problemas da biblioteca SDPLib, ver [7]. As implementações de todos os algoritmos apresentados foram feitas em MATLAB 7.0, sendo executados em um PC Intel Celeron 1.4GHz com 256 MB de memória RAM no ambiente Windows XP. Analisamos, ainda, as limitações dos métodos e apresentamos um exemplo completo para o problema de corte máximo no final deste capítulo conforme o modelo apresentado no capítulo 5.

6.1 Busca linear

Um dos pontos importantes para a obtenção de uma boa performance dos algoritmos está na escolha do método para busca linear. Optamos pelo método de Armijo para a barreira hiper-cúbica e pelo método de Wolfe para o algoritmo de centros. Observamos que as iterações necessárias para se encontrar uma solução eram, normalmente, inferiores a dez.

6.2 Precondicionamento do sistema de Newton

Em diversos pontos dos algoritmos apresentados, necessitamos resolver sistemas do tipo:

$$Hd = g, \quad (6.1)$$

onde H é uma matriz simétrica definida positiva n por n . É comum que este sistema seja mal-condicionado. Portanto, com o objetivo de afastar de zero o menor auto-valor da matriz H ao longo das iterações, fazemos uso de preconditionadores, como mostramos a seguir. Primeiro, calculamos:

$$\bar{H} = D^{-1}HD^{-1}, \quad \bar{g} = D^{-1}g, \quad (6.2)$$

onde $D^{-1} = \text{diag}\{H_{11}^{-\frac{1}{2}}, \dots, H_{nn}^{-\frac{1}{2}}\}$. Considerando 6.2, podemos reescrever 6.1 como:

$$D\bar{H}Dd = D\bar{g},$$

ou ainda, como:

$$\bar{H}\bar{d} = \bar{g}, \quad (6.3)$$

onde $\bar{d} = Dd$. Para resolver 6.3, aplicamos a fatorização de Cholesky da matriz \bar{H} dada por:

$$\bar{H} = LL^T,$$

e obtemos \bar{d} . Agora, calculamos d por:

$$d = D^{-1}\bar{d}.$$

6.3 Resultados Computacionais

Para testar a performance dos métodos que desenvolvemos, trabalhamos com um repositório de problemas disponível na internet chamado `sdplib`, ver [7].

Na tabela 6.1, apresentamos a interpretação para as características que foram avaliadas para cada problema. Na tabela 6.2, apresentamos as dimensões e as soluções de cada problema.

Nomeclatura	Interpretação
n	Dimensão das matrizes
m	Quantidade de matrizes de restrição
N. Iter.	Número de iterações externas
P.N.	Passos de Newton
T. Exec.	Tempo de execução em minutos
TOL	Tolerância
mcp	Problemas de corte máximo
qap	Problemas de associação quadrática
theta	Número de Lóvasz
gpp	Problemas de partição de grafos
gpK	Problemas de k-partições de grafos

Tabela 6.1: Interpretação das nomenclaturas

A seguir, apresentamos nas tabelas 6.3, 6.4 e 6.5 os resultados obtidos para os métodos de Newton com centros analíticos, BFGS com centros analíticos e para o pacote SDPT, que é considerado, até este momento, o melhor pacote existente para programação semidefinida, respectivamente.

Na tabela 6.6, apresentamos a interpretação das informações sobre o problema da k -equipartição de grafos. Estes problemas foram gerados pelo método descrito em [22]. Na tabela 6.7, apresentamos os resultados obtidos pela barreira hiper-cúbica que desenvolvemos. Na tabela 6.8, são apresentados os resultados obtidos pelo modelo proposto por Karisch e Rendl, ver [23].

Problema	n	m	Sol.
mcp100.dat-s	100	100	226.1574
mcp124-1.dat-s	124	124	141.9905
mcp124-2.dat-s	124	124	269.8802
mcp124-3.dat-s	124	124	467.7501
mcp124-4.dat-s	124	124	864.4119
mcp250-1.dat-s	250	250	317.2643
mcp250-2.dat-s	250	250	531.9301
mcp250-3.dat-s	250	250	981.1726
theta1.dat-s	104	50	23.000000
theta2.dat-s	498	100	32.8791
gpp100.dat-s	100	101	-44.9435
gpp124-1.dat-s	124	125	-7.3431
gpp124-2.dat-s	124	125	-46.8623
gpp124-3.dat-s	124	125	-153.014
gpp124-4.dat-s	124	125	-418.99
gpp250-1.dat-s	250	251	-15.445
gpp250-2.dat-s	250	251	-81.869
gpp250-3.dat-s	250	251	-303.50
gpp250-4.dat-s	250	251	-747.30
qap5.dat-s	136	26	-436.00
qap6.dat-s	229	37	-381.44
qap7.dat-s	358	50	-425.00

Tabela 6.2: Dados para problemas de otimização combinatória da biblioteca SDPLib

Problema	N. Iter.	P.N.	T.Exec.	Sol.	TOL.
mcp100.dat-s	10	250	1.02	226.62533	6.1948e-004
mcp124-1.dat-s	10	250	1.63	144.1237	6.7377e-004
mcp124-2.dat-s	10	250	1.83	270.2610	4.3202e-004
mcp124-3.dat-s	14	350	2.49	468.6002	6.5010e-004
mcp124-4.dat-s	11	275	2.10	865.2412	5.9682e-004
mcp250-1.dat-s	-	-	-	-	-
mcp250-2.dat-s	-	-	-	-	-
mcp250-3.dat-s	-	-	-	-	-
theta1.dat-s	13	321	0.404	23.59436	7.9532e-004
theta2.dat-s	13	325	4.39	36.9550	7.8651e-004
gpp100.dat-s	12	300	1.56	-44.54	5.2207e-004
gpp124-1.dat-s	15	375	3.21	-6.125	6.9412e-004
gpp124-2.dat-s	11	275	2.46	-46.207	7.05563e-004
gpp124-3.dat-s	10	250	2.268	-152.166	6.88918e-004
gpp124-4.dat-s	10	250	2.2679	-417.845	6.54601e-004
gpp250-1.dat-s	-	-	-	-	-
gpp250-2.dat-s	-	-	-	-	-
gpp250-3.dat-s	-	-	-	-	-
gpp250-4.dat-s	-	-	-	-	-
qap5.dat-s	61	1525	0.9440	-431.226	9.615190e-004
qap6.dat-s	71	1775	2.444	-347.3436	9.464234e-004
qap7.dat-s	60	1476	5.1489	-398.2043	0.00206932

Tabela 6.3: Resultados para problemas de otimização combinatória com Algoritmo de Centros Método de Newton

Problema	N. Iter.	P.N.	T.Exec.	Sol.	TOL.
mcp100.dat-s	15	108	0.20	226.5958	1e-06
mcp124-1.dat-s	14	93	0.25	143.1134	1e-06
mcp124-2.dat-s	17	238	0.62	269.8867	1e-06
mcp124-3.dat-s	16	474	1.2	467.7506	1e-06
mcp124-4.dat-s	18	312	0.91	919.6758	1e-06
mcp250-1.dat-s	19	571	7.63	317.2655	1e-06
mcp250-2.dat-s	18	861	12.42	531.93	1e-06
mcp250-3.dat-s	25	1512	22.37	981.20	1e-06
theta1.dat-s	33	478	0.37	23.00	1e-06
theta2.dat-s	21	191	1.86	36.91	1e-06
gpp100.dat-s	23	305	1.23	-44.94	1e-06
gpp124-1.dat-s	22	459	2.82	-7.3427	1e-06
gpp124-2.dat-s	16	121	0.77	-46.85	1e-06
gpp124-3.dat-s	33	1098	7.00	-153.00	1e-06
gpp124-4.dat-s	21	400	2.63	-418.98	1e-06
gpp250-1.dat-s	20	348	10.42	-15.4436	1e-06
gpp250-2.dat-s	16	331	10.18	-81.8679	1e-06
gpp250-3.dat-s	25	1426	45.71	-303.53	1e-06
gpp250-4.dat-s	21	597	19.46	-747.3267	1e-06
qap5.dat-s	372	5888	4.81	-435.9999	1e-09
qap6.dat-s	-	-	-	-	-
qap7.dat-s	-	-	-	-	-

Tabela 6.4: Resultados para problemas de otimização combinatória com Algoritmo de Centros Método de QuasiNewton BFGS

Problema	P.N.	T.Exec.	Sol.	TOL.
mcp100.dat-s	12	0.0260	226.157351	5.58362e-007
mcp124-1.dat-s	13	0.0250	141.990477	6.50464e-008
mcp124-2.dat-s	13	0.0280	269.880170	1.32798e-007
mcp124-3.dat-s	13	0.0302	467.750114	4.393250e-007
mcp124-4.dat-s	13	0.03177	864.411864	2.104151e-006
mcp250-1.dat-s	14	0.07317	317.264340	6.45273e-007
mcp250-2.dat-s	13	0.07968	531.930084	2.41634e-006
mcp250-3.dat-s	13	0.09296	981.172571	3.09153e-006
theta1.dat-s	12	0.00625	23.000000	8.54665e-008
theta2.dat-s	14	0.06067	36.879169	3.00376e-008
gpp100.dat-s	15	0.02395	-44.943547	2.626945e-008
gpp124-1.dat-s	16	0.045833	-7.3430736	6.376803e-007
gpp124-2.dat-s	15	0.039583	-46.862291	4.530134e-007
gpp124-3.dat-s	15	0.040104	-153.014121	6.121922e-008
gpp124-4.dat-s	16	0.0429687	-418.987591	1.458458e-006
gpp250-1.dat-s	16	0.1617187	-15.4449157	6.163295e-008
gpp250-2.dat-s	16	0.1648438	-81.8689548	9.322209e-008
gpp250-3.dat-s	15	0.1625000	-303.5393121	3.006484e-006
gpp250-4.dat-s	17	0.1729167	-747.3282905	1.532306e-005
qap5.dat-s	-	-	-	-
qap6.dat-s	-	-	-	-
qap7.dat-s	-	-	-	-

Tabela 6.5: Resultados para problemas de otimização combinatória com o pacote SDPT

Problema	n	m	numArestas
gpK1	8	4	6
gpK2	16	8	25
gpK3	50	25	273
gpK4	54	27	314
gpK5	60	30	395
gpK6	80	40	706
gpK7	100	50	1108
gpK8	100	50	1108
gpK9	100	50	1108
gpK10	140	70	2181

Tabela 6.6: Dados para problemas de k-partições

Problema	N. Iter.	P.N.	T.Exec.	Sol.	TOL.
gpK1	4	44	0.01145	4.420	1.2587e-004
gpK2	3	23	0.00963	20.7722	4.9290e-004
gpK3	4	34	0.08802	224.5981	4.1273e-004
gpK4	4	34	0.11093	211.8296	6.4132e-005
gpK5	4	34	0.15729	335.9654	6.1865e-004
gpK6	4	34	0.39500	571.2287	8.3591e-004
gpK7	5	45	1.1518	891.6193	5.0435e-005
gpK8	5	45	1.1492	1009.4884	7.00406e-005
gpK9	4	34	0.8684	1076.8420	9.85157e-004
gpK10	5	45	3.9403	1976.6238	1.128578e-004

Tabela 6.7: Resultados para problemas de k-partições de grafos com a barreira-hipercúbica

Problema	P.N.	T.Exec.	Sol.	TOL.
gpK1	15	0	4.000	2.3646e-010
gpK2	16	2.6041e-004	19.500	2.63339e-009
gpK3	21	7.8125e-004	211.044	1.70910e-009
gpK4	21	0	196.4940	2.23023e-008
gpK5	20	0.001042	314.5423	4.43026e-008
gpK6	21	7.8125e-04	536.6298	4.417984e-008
gpK7	21	0.0015625	838.9906	7.250605e-008
gpK8	21	7.81250e-04	944.6464	9.211294e-008
gpK9	21	7.81250e-004	1009.41950	1.6670833e-007
gpK10	22	0.002604166	1844.98901	9.8933924e-008

Tabela 6.8: Resultados para problemas de k-partições de grafos com o modelo proposto por Rendl e Karisch

Capítulo 7

Propostas e Conclusões

Neste trabalho, propusemos dois métodos para resolver problemas de programação semidefinida: o de centros e o método de Newton com a barreira hiper-cúbica. Demonstramos as propriedades de cada barreira e desenvolvemos uma técnica para obter uma solução inicial viável com garantia de proximidade em relação à trajetória central (requisito necessário para demonstrar a convergência do algoritmo) sem utilizar a metodologia do big-M. Além disso, fizemos vários testes com problemas da biblioteca `sdplib`. Mostramos que os métodos convergem e que, com mais investimento em estruturas de dados, podem tornar-se competitivos com os melhores métodos existentes.

Um ponto que deve ser explorado futuramente é a aplicação dessas barreiras para algoritmos primais-duais inviáveis juntamente com estruturas que explorem de forma mais adequada a esparsidade das matrizes que é comum na maioria dos problemas. Este parece ser um bom caminho para a obtenção de melhores performances.

Outro ponto importante é encontrar mais aplicações que se adequem à barreira hiper-cúbica e extrair vantagens de tais características.

Apêndice A

Notações

A.1 Operações

1. \succeq indica que a matriz é semidefinida positiva;
2. \succ indica que a matriz é definida positiva;
3. \bullet operador traço, ou seja,

$$A \bullet B = \text{Traço}(A^T B),$$

ou

$$\langle X, Y \rangle = \sum_{i=1}^m \sum_{j=1}^n X_{ij} Y_{ij}$$

o produto interno padrão de duas matrizes ($m \times n$).

A.2 Conjuntos

1. Re^n o espaço euclidiano de dimensão n ;
2. $M_{m \times n}$ ou $R^{m \times n}$ o espaço das matrizes $m \times n$;

3. S^n o espaço das matrizes simétricas de dimensão n ;
4. S_+^n o espaço das matrizes simétricas semidefinidas positivas de dimensão n ;
5. S_{++}^n o espaço das matrizes simétricas definidas positivas de dimensão n ;
6. $P = \{X : X \succeq 0\}$ o cone convexo das matrizes simétricas e semidefinidas positivas.

A.3 Vetores

1. x um vetor
2. x^\top o transposto do vetor x ;
3. x_i o i -ésimo componente do vetor x ;
4. x^k o k -ésimo vetor em uma sequência de vetores;
5. $\langle x, y \rangle = x^\top y$ o produto interno de dois vetores;

A.4 Elementos de Álgebra

1. Qualquer matriz simétrica S pode ser decomposta como:

$$S = V\Lambda V^\top$$

onde $VV^\top = I$ e Λ é a matriz diagonal formada pelos autovalores de S ;

2. Seja $A \in M_n$, então, temos que:

$$\text{Tr} A = \sum_{i=1}^n \lambda_i(A)$$

onde $\lambda_i(A)$ para $i = 1, \dots, n$ são os autovalores da matriz A ;

3. Sejam $A \in M_n$ e $B \in M_n$, então, temos que:

$$\text{Tr}(A + B) = \sum_{i=1}^n (\lambda_i(A) + \lambda_i(B))$$

4. Seja $S = V\Lambda V^\top$, onde $VV^\top = I$ e Λ é a matriz diagonal formada pelos autovalores de S , então a função dada por $f(S)$ é:

$$f(S) = Vf(\Lambda)V^\top$$

onde $f(\Lambda)$ é a matriz diagonal formada por $f(\lambda_i(S))$, para $i = 1, \dots, n$.

A.5 Derivadas de $\text{Tr}(X \text{Ln} X)$

Sejam as matrizes simétricas $n \times n$, $X \succ 0$ e H . Vamos decompô-las por:

$$X = P\Lambda(X)P^\top$$

$$H = Q_H\Lambda(H)Q_H^\top$$

Seja $F(X) = \text{Tr}(X \text{Ln} X)$. Então as derivadas de primeira e segunda ordem de $F(X)$ são:

$$\begin{aligned} DF(X)[H] &= \left. \frac{d}{dt} \right|_{t=0} \text{Tr}\{(X + tH)\text{Ln}(X + tH)\} \\ &= \left. \frac{d}{dt} \right|_{t=0} \text{Tr}\{(X + tH)\text{Ln}X(I + tX^{-1}H)\} \\ &= \left. \frac{d}{dt} \right|_{t=0} \text{Tr}\{(X + tH)\text{Ln}X\} + \text{Tr}\{(X + tH)\text{Ln}(I + tX^{-1}H)\} \\ &= \left. \frac{d}{dt} \right|_{t=0} \text{Tr}\{X \text{Ln}X\} + \text{Tr}\{tH \text{Ln}X\} + \text{Tr}\{X \text{Ln}(I + tX^{-1}H)\} \\ &\quad + \text{Tr}\{tH \text{Ln}(I + tX^{-1}H)\} \\ &= \left. \frac{d}{dt} \right|_{t=0} \text{Tr}\{tH \text{Ln}X\} + \text{Tr}\{X \text{Ln}(I + tX^{-1}H)\} + \text{Tr}\{tH \text{Ln}(I + tX^{-1}H)\} \end{aligned}$$

Derivando cada termo, temos:

$$\begin{aligned}
\left. \frac{d}{dt} \right|_{t=0} Tr\{tHLnX\} &= \left. \frac{d}{dt} \right|_{t=0} tTr\{HLnX\} \\
&= Tr\{HLnX\} \Big|_{t=0} \\
&= Tr\{HLnX\}
\end{aligned}$$

Seja $R = P^\top Q$, temos então, do teorema 2.9, que:

$$\begin{aligned}
\left. \frac{d}{dt} \right|_{t=0} Tr\{tHLn(I + tX^{-1}H)\} &= \left. \frac{d}{dt} \right|_{t=0} \sum_{i,j=1}^n t\lambda_i(H) \ln(1 + t\lambda_j(X^{-1}H)R_{ij}^2) \\
&= \sum_{i,j=1}^n (\lambda_i(H) \ln(1 + t\lambda_j(X^{-1}H))R_{ij}^2 \\
&\quad + t\lambda_i(H) \frac{1}{1 + \lambda_j(X^{-1}H)} \lambda_j(X^{-1}H)R_{ij}^2) \Big|_{t=0} \\
&= \sum_{i,j=1}^n \ln 1 R_{ij}^2 \\
&= 0
\end{aligned}$$

$$\begin{aligned}
\left. \frac{d}{dt} \right|_{t=0} Tr\{XLn(I + tX^{-1}H)\} &= \left. \frac{d}{dt} \right|_{t=0} \sum_{i,j=1}^n \lambda_i(X) \ln(1 + t\lambda_j(X^{-1}H))R_{ij}^2 \\
&= \sum_{i,j=1}^n \lambda_i(X) \frac{\lambda_j(X^{-1}H)}{1 + t\lambda_j(X^{-1}H)} R_{ij}^2 \Big|_{t=0} \\
&= \sum_{i,j=1}^n \lambda_i(X) \lambda_j(X^{-1}H) R_{ij}^2 \\
&= \sum_{i=1}^n \lambda_i(X) \sum_{j=1}^n \lambda_j(X^{-1}H) R_{ij}^2 \\
&= \sum_{i=1}^n \lambda_i(X) R_{i.} \Lambda(X^{-1}H) R_{i.}^\top \\
&= \sum_{i=1}^n \lambda_i(X) (R\Lambda(X^{-1}H)R^\top)_{ii} \\
&= Tr\{\Lambda(X)R\Lambda(X^{-1}H)R^\top\} \\
&= Tr\{\Lambda(X)P^\top Q\Lambda(X^{-1}H)Q^\top P\} \\
&= Tr\{P\Lambda(X)P^\top Q\Lambda(X^{-1}H)Q^\top\} \\
&= Tr\{XX^{-1}H\} \\
&= Tr\{IH\} = Tr\{H\}
\end{aligned}$$

Portanto,

$$DF(X)[H] = Tr\{I + LnX, H\} \quad (A.1)$$

Vamos calcular, agora, $DF(X)[H, H]$.

$$\begin{aligned} DF(X)[H, H] &= \left. \frac{d}{dt} \right|_{t=0} Tr\{Ln(X + tH)H\} \\ &= \left. \frac{d}{dt} \right|_{t=0} Tr\{LnX(I + tX^{-1}H)H\} \\ &= \left. \frac{d}{dt} \right|_{t=0} Tr\{LnXH\} + Tr\{Ln(I + tX^{-1}H)H\} \\ &= \left. \frac{d}{dt} \right|_{t=0} Tr\{Ln(I + tX^{-1}H)H\} \\ &= \left. \frac{d}{dt} \right|_{t=0} Tr\{HLn(I + tX^{-1}H)\} \\ &= \left. \frac{d}{dt} \right|_{t=0} \sum_{i,j=1}^n \lambda_i(H) \ln(1 + t\lambda_j(X^{-1}H)) R_{ij}^2 \\ &= \sum_{i,j=1}^n \lambda_i(H) \frac{1}{1 + t\lambda_j(X^{-1}H)} \lambda_j(X^{-1}H) R_{ij}^2 \Big|_{t=0} \\ &= \sum_{i,j=1}^n \lambda_i(H) \lambda_j(X^{-1}H) R_{ij}^2 \\ &= Tr\{HX^{-1}H\} \end{aligned}$$

A.6 Cálculo da Direção

Seja o sistema de equações dado por:

$$\nabla\phi_B(X) + \nabla^2\phi_B(X)\Delta X + \sum_{i=1}^m A_i y_i = 0 \quad (A.2)$$

$$\mathcal{A}\Delta X = 0 \quad (A.3)$$

Podemos reescrever A.2 como:

$$\Delta X = -\nabla^2\phi_B(X)^{-1}(\nabla\phi_B(X) + \sum_{i=1}^m A_i y_i) \quad (A.4)$$

Uma vez que $\mathcal{A}\Delta X$ é:

$$\mathcal{A}\Delta X = \begin{pmatrix} \langle A_1, \Delta X \rangle \\ \vdots \\ \langle A_m, \Delta X \rangle \end{pmatrix}$$

então, cada elemento de A.3, satisfaz:

$$\langle A_j, \Delta X \rangle = 0 \quad \forall j = 1, \dots, m \quad (\text{A.5})$$

Agora, substituindo A.4 em A.5, temos:

$$\begin{aligned} & - \langle A_j, \nabla^2 \phi_B(X)^{-1} (\nabla \phi_B(X) + \sum_{i=1}^m A_i y_i) \rangle = 0 \quad \forall j = 1, \dots, m \\ & \langle A_j, \nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X) + \nabla^2 \phi_B(X)^{-1} \sum_{i=1}^m A_i y_i \rangle = 0 \quad \forall j = 1, \dots, m \\ & \langle A_j, \nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X) \rangle + \langle A_j, \nabla^2 \phi_B(X)^{-1} \sum_{i=1}^m A_i y_i \rangle = 0 \quad \forall j = 1, \dots, n \\ & \langle A_j, \nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X) \rangle = - \langle A_j, \nabla^2 \phi_B(X)^{-1} \sum_{i=1}^m A_i y_i \rangle \quad \forall j = 1, \dots, m \end{aligned} \quad (\text{A.6})$$

Podemos reescrever A.6 como:

$$\langle A_j, \nabla^2 \phi_B(X)^{-1} \sum_{i=1}^m A_i y_i \rangle = - \langle A_j, \nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X) \rangle \quad \forall j = 1, \dots, m$$

que é equivalente a:

$$\begin{aligned} \sum_{i=1}^m \langle A_j, \nabla^2 \phi_B(X)^{-1} A_i \rangle y_i &= - \langle A_j, \nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X) \rangle \quad \forall j = 1, \dots, m \\ \mathcal{A}(\nabla^2 \phi_B(X)^{-1} A_j)^\top y &= - \langle A_j, (\nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X)) \rangle \quad \forall j = 1, \dots, m \end{aligned} \quad (\text{A.7})$$

aplicando a expressão A.7 para todo $j = 1, \dots, m$, temos:

$$\begin{bmatrix} \mathcal{A}(A_1 \nabla^2 \phi_B(X)^{-1})^\top \\ \vdots \\ \mathcal{A}(A_m \nabla^2 \phi_B(X)^{-1})^\top \end{bmatrix} y = - \mathcal{A}(\nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X)) \quad (\text{A.8})$$

como as matrizes A_i , $\forall i = 1, \dots, m$ são linearmente independentes e

$\nabla^2 \phi_B(X)^{-1}$ é definida positiva, temos:

$$y = - \begin{bmatrix} \mathcal{A}(A_1 \nabla^2 \phi_B(X)^{-1})^\top \\ \vdots \\ \mathcal{A}(A_m \nabla^2 \phi_B(X)^{-1})^\top \end{bmatrix}^{-1} \mathcal{A}(\nabla^2 \phi_B(X)^{-1} \nabla \phi_B(X)) \quad (\text{A.9})$$

podemos reescrever a expressão A.4 como:

$$\Delta X = -\nabla^2 \phi_B(X)^{-1}(\nabla \phi_B(X) + \mathcal{A}^\top y) \quad (\text{A.10})$$

agora, substituindo A.9 em A.10 temos a expressão da direção ΔX .

Apêndice B

Estruturas de Dados para Programação Semidefinida

B.1 Matrizes Esparsas

Os problemas no SDPLIB estão codificados no formato esparsa do SDPA. O formato esparsa SDPA foi desenvolvido para tratar problemas de programação semidefinida onde as matrizes F_i , $i = 1, \dots, m$, sejam bloco diagonais com blocos esparsos. Um arquivo no formato esparsa SDPA consiste de seis partes básicas:

1. Comentários. O arquivo pode iniciar com qualquer quantidade de comentários. Cada comentário deve iniciar com `'''` ou `''*`;
2. A primeira linha depois dos comentários contém m , o número de matrizes de restrições;
3. A segunda linha depois dos comentários contém $nblocos$, o número de blocos em uma matriz bloco-diagonal;

4. A terceira linha contém um vetor de números que dão os tamanhos dos blocos individuais.
5. A quarta linha contém o vetor c da função objetivo;
6. As linhas restantes contêm as entradas das matrizes de restrições, com um elemento por linha. O formato de cada linha é:

$\langle \text{matno} \rangle \langle \text{blcno} \rangle \langle i \rangle \langle j \rangle \langle \text{elemento} \rangle$,

onde matno é o número da matriz a qual o elemento pertence, blcno especifica o bloco dentro da matriz, i e j especificam a localização dentro do bloco, e elemento especifica o valor do elemento da matriz. É importante observar que todas as matrizes aqui consideradas são simétricas, apenas entradas da matriz triangular superior são dadas.

Por exemplo, considere o problema:

$$\min \quad 10x_1 + 20x_2$$

$$s.a : \quad x_1F_1 + x_2F_2 \succeq F_0$$

$$x_1, x_2 \in R^n$$

$$F_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

$$F_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 2 \\ 0 & 0 & 2 & 6 \end{bmatrix}$$

2					= dimensão do problema (Número de vértices)
2					= Número de blocos
{2,	2 }				= O tamanho de cada bloco
{10.0,	20.0}				= os coeficientes de c.
0	1	1	1	1.0	=(Matriz,Bloco,Linha,Coluna,Elemento)
0	1	2	2	2.0	
0	2	1	1	3.0	
0	2	2	2	4.0	
1	1	1	1	1.0	
1	1	2	2	1.0	
2	1	2	2	1.0	
2	2	1	1	5.0	
2	2	1	2	2.0	
2	2	2	2	6.0	

B.2 Exemplo do Problema de Corte Máximo

A seguir, apresentamos um exemplo completo de como escrever o problema de corte máximo no formato de programação semidefinida.

Seja o grafo não dirigido cuja matriz de adjacências é dada por:

$$C = \begin{bmatrix} 0 & 3 & 0 & 5 & 0 \\ 3 & 0 & 1 & 2 & 1 \\ 0 & 1 & 0 & 0 & 7 \\ 5 & 2 & 0 & 0 & 1 \\ 0 & 1 & 7 & 1 & 0 \end{bmatrix}.$$

Para os vértices que não possuem ligação, é associada uma aresta com peso zero.

A matriz laplaceana para este problema é dada por:

$$L = \begin{bmatrix} 0 & -3 & 0 & -5 & 0 \\ -3 & 7 & -1 & -2 & -1 \\ 0 & -1 & 8 & 0 & -7 \\ -5 & -2 & 0 & 8 & -1 \\ 0 & -1 & 7 & -1 & 9 \end{bmatrix}.$$

No formato SDPLIB, temos:

5					= dimensão do problema (Número de vértices)
1					= Número de blocos
5					= O tamanho de cada bloco
{+0.5,	+0.5,	+0.5,	+0.5,	+0.5}	= os coeficientes de c.
0	1	1	1	2	=(Matriz,Bloco,Linha,Coluna,Elemento)
0	1	1	2	-0.75	
0	1	1	4	-1.25	
0	1	2	2	1.75	
0	1	2	3	-0.25	
0	1	2	4	-0.5	
0	1	2	5	-0.25	
0	1	3	3	2	
0	1	3	5	-1.75	
0	1	4	4	2	
0	1	4	5	-0.25	
0	1	5	5	2.25	
1	1	1	1	1.0	
2	1	2	2	1.0	
3	1	3	3	1.0	
4	1	4	4	1.0	
5	1	5	5	1.0	

A solução ótima deste problema é:

$$S_1 = \{1, 5\}$$

$$S_2 = \{2, 3, 4\}$$

Bibliografia

- [1] Alizadeh, F.: Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization* 5(1), 13-51 (1995)
- [2] Alizadeh, F., Haeberly, J-P., Overton, M.L.: Primal-dual interior-point methods for semidefinite programming: convergence rates, stability and numerical results. *SIAM Journal on Optimization*, 8(3):746-768, 1998.
- [3] Andrade, L.M.: Breve introdução ao latex 2 ϵ . Universidade Federal da Paraíba, versão 2.1, abril 2000.
- [4] Anstrreicher, K. M., Fampa, M.: A long-step path following algorithm for semidefinite programming problems. In *Topics in Semidefinite and Interior-Point Methods*, volume 18 of *The fields Institute for Research in Mathematical Sciences, Communications Series*, pages 181-196, Providence, Rhode Island, 1998. American Mathematical Society.
- [5] Anstrreicher, K. M., Vial, J., -Ph.: On the convergence of an infeasible primal-dual interior-point method for convex programming. *Optimization Methods and Software*, 3:273-283, 1994.
- [6] Bazaara, M. S., Sherali, H. D., Shetty, C. M.: *Nonlinear Programming: Theory and Algorithms*. Wiley Text Books, 2nd Edition, 1993.

- [7] Borchers, B.: SDPLIB 1.2, A library of semidefinite programming test problems. The SDPLIB problems, (2000).
- [8] Burkard, R. E., Çela, E., Pardalos, P. M., Pitsoulis, L. S.: The quadratic assignment problem. Spezialforschungsbereich F 003, Optimierung und Kontrolle, Karl-Franzens-Universität Graz and Technische Universität Graz, Bericht Nr. 126 - Mai, 1998.
- [9] Donath, W.E. and Hoffman, A.J.: Lower bounds for the partitioning of graphs. IBM J. of. Res. Dev., 17:420-425, 1973.
- [10] Fiacco, A., McCormick, G.: Nonlinear programming: Sequential unconstrained minimization techniques, Wiley, New York, 1968; SIAM Classics in Applied Mathematics Series, reprint, (1990)
- [11] Fujisawa, K., Fukuda, M., Kojima, M. Nakata, K.: Numerical evaluation of SDPA (semidefinite programming algorithm). september, 1997, revised on june 1998.
- [12] Garey, M. R., Johnson, D. S.: Computers and Intractability. A guide to theory of np-completeness. W. H. Freeman and company, New York.
- [13] Goemans, M. X., Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM 42 1115-1145 (1995)
- [14] Gruber, G.: On Semidefinite Programming and Applications in Combinatorial Optimization. Ph.D. Thesis, Department of Mathematics, University of Technology, Graz. April, 2002.
- [15] Helmberg, C.: Semidefinite programming. European Journal of Operational Research, 137:461-482, march 2001.

- [16] Helmberg, C.: An interior point method for semidefinite programming and max-cut bounds. PhD thesis, Department of Mathematics, University of Technology, Graz. Austria, 1994.
- [17] Helmberg, C., Rendl, F., Vanderbei, R. J., Wolkowicz, H.: An interior-point method for semidefinite programming, *SIAM Journal on Optimization* 6, 342-361, 1996.
- [18] Hertog, D. den .: Interior Point Approach to Linear, Quadratic and Convex Programming, Ph.D. Thesis, Dept. of Applied Mathematics, Delft University of Technology, 1992.
- [19] Hertog, D. den .: Interior Point Approach to Linear, Quadratic and Convex Programming, Algorithms and Complexity, volume 277 of Mathematics and Its Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994.
- [20] Horn, R. A., Johnson, C.A.: Matrix Analysis. Cambridge University press, Cambridge, 1985.
- [21] Huard, P., "Resolution of Mathematical Programming with Nonlinear Constraints by the Method of Centers". In: *Nonlinear Programming*, J. Ndadie ed., North-Holland Publish Company, Amsterdam, Holland, 207-219, 1967.
- [22] Johnson, D.S., Aragon, C.R., McGeoch, L.A. and Schevon, C.: Optimization by simulated annealing: an experimental evaluation; part 1, graph partitioning. *Operations Research*, 37:865-892, 1989.
- [23] Karisch, S.E. and Rendl, F.: Semidefinite Programming and Graph Equipartition, in: *Topics in Semidefinite and Interior-Point Methods*

- (P. M. Pardalos, H. Wolkowicz, eds.), Fields Institute Communications 18, American Mathematical Society, 77-95, 1998.
- [24] Karmarkar, N.: A new polynomial-time algorithm for linear programming, *Combinatorica* 4, 373-395, 1984.
- [25] Klerk, E. de.: Interior point methods for semidefinite programming. PhD thesis, Faculty of Technical Mathematics and Informatics, TU Delft, NL-2600 GA Delft, The Netherlands, 1997.
- [26] Klerk, E. de.: Aspects of semidefinite programming: Interior point algorithms and selected applications. PhD thesis, Faculty of Technical Mathematics and Informatics, Kluwer Academic Publishers, 2002.
- [27] Kojima, M., Mizuno, S., Yoshise, A.: A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, 44:1-26, 1989.
- [28] Kojima, M., Shindoh, S., Hara, S.: Interior point methods for the monotone linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, 7(1):86-125, 1997.
- [29] Kojima, M., Shida, M., Shindoh, S.: Local convergence of predictor-corrector infeasible-interior-point algorithms for SDPs and SDLCs. *Mathematical Programming*, 50(3):331-342, 1991.
- [30] Kojima, M., Shida, M., Shindoh, S.: Global and local convergence of predictor-corrector infeasible-interior-point algorithms for semidefinite programs. *Research Reports on Information Sciences, Ser. B: Operations Research B-305*, Department of Information Sciences, Tokyo Institute of Technology, Oh-Okayama, Meguro-ku, Tokyo 152, Japan, 1995.

- [31] Kojima, M., Shida, M., Shindoh, S.: A predictor-corrector interior-point algorithm for semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction. *SIAM Journal on Optimization*, 9(2):444-465, 1999.
- [32] Lemaréchal, C., Oustry, F.: Semidefinite relaxations and Lagrangian duality with application to combinatorial optimization. *Rapport de recherche*, n 3710, juin 1999.
- [33] Lamport, L.: *Latex, user's guide and reference manual*. Addison-Wesley Publishing Company. Second edition, 1994.
- [34] Luenberger, D. G.: *Linear and Nonlinear Programming*. Addison-Wesley Publishing Co., Reading, second edition, 1984. Reprinted with corrections in May, 1989.
- [35] Magnus, J.R., Neudecker, H.: *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley and Sons, New York, 1988.
- [36] Mizuno, S., Todd, M. J. and Ye, Y.: On adaptative-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18(4):964-981, 1993.
- [37] Monteiro, R. D. C.: Primal-dual path following algorithms for semidefinite programming. *SIAM Journal on Optimization* 7, 663-678, 1997.
- [38] Monteiro, R. D. C.: Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions. *SIAM Journal on Optimization*, 8(3):797-812, 1998.

- [39] Monteiro, R. D. C., Adler, I.: Interior path following primal-dual algorithms: Part I: Linear programming. *Mathematical Programming*, 44(1):27-41, 1989.
- [40] Monteiro, R. D. C., Adler, I.: Interior path following primal-dual algorithms: Part II: Convex quadratic programming. *Mathematical Programming*, 44(1):43-66, 1989.
- [41] Monteiro, R. D. C., Zhang, Y.: A unified analysis for a class of long-step primal-dual path-following interior-point algorithms for semidefinite programming. *Mathematical Programming*, 81(3, Ser. A): 281-299, 1998.
- [42] Monteiro, R. D. C., Tsuchiya, T.: Polynomiality of primal-dual algorithms for semidefinite linear complementarity problems based on the Kojima-Shindoh-Hara family of directions. *Mathematical Programming*, 84(1):39-53, 1999.
- [43] Monteiro, R. D. C., Tsuchiya, T.: Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming. *SIAM Journal on Optimization*, 9(3):551-577, 1999.
- [44] Nesterov, Y.E., Nemirovski, A.S.: *Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications*, SIAM Studies in Applied Mathematics (SIAM, Philadelphia, PA, 1994).
- [45] Nesterov, Y.E., Todd, M.J.: Self-scaled barriers and interior-point methods for convex programming. *Mathematical Oper. Res.*, 22(1):1-42, 1997.
- [46] Nesterov, Y.E., Todd, M.J.: Primal-dual interior-point methods for self scaled cones. *SIAM Journal on Optimization*, 8(2):324-364, 1998.

- [47] Oliveira, P. R., Gregório, R., Monteiro, S.A: Relaxação lagrangeana, programação semidefinida e aplicações à combinatória. Mini-curso apresentado no XXXV Simpósio Brasileiro de Pesquisa Operacional, SOBRAPO, UFRN. 4 a 7 de novembro de 2003.
- [48] Overton, M.L., Womersley, R.S.: Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices. Submitted to Math Programming, june 1991.
- [49] Pardalos, P. M., Rendel, F., Wolkowicz, H.: The quadratic assignment problem: a survey and recent developments. DIMACS, Series in Discrete Mathematics and Theoretical Computer Science.
- [50] Potra, F. A., Sheng, R.: A superlinearity convergent primal-dual infeasible-interior-point algorithm for semidefinite programming. SIAM Journal on Optimization, 8(4):1007-1028, 1998.
- [51] Quiroz, E. A. P., Oliveira, P. R. . A new barrier for a class of semidefinite problems. RAIRO. Operations Research, França, v. 40, p. 303-323, 2006.
- [52] Rendl, F. and Wolkowicz, H.: A projection technique for partitioning the nodes of a graph. Ann. Oper. Res., 58:155-179, 1995.
- [53] Roos, C., Vial, J. -Ph.: A polynomial method of approximate centers for linear programming, Math. Programming 54, 295-305.
- [54] Tanabe, K.: A Centered Newton method for mathematical programming. In: System Modeling and Optimization, Springer-Verlag, NY, 197-206, (1988)
- [55] Tseng, P.: Analysis of infeasible path-following methods using the Alizadeh-Haeberley-Overton direction for the monotone semi-definite

- LCP. Techninal report, Departament of Mathematics, University of Washigton, Seattle, WA, 1996.
- [56] Todd, M. J., Ye, Y.: A centered projective algorithm for linear programming. *Math. Oper.Res.*, 15: 508-529 (1990).
- [57] Todd, M.J., Toh, K.C., Tütüncü, R. H.: On the Nesterov-Todd direction in semidefinite programming. *SIAM Journal on Optimization*, 8(3):709-796, 1998.
- [58] Vanderbei, R. J., Benson, Y.: *Semidefinite programming*. SIAM, march, 1996.
- [59] Vial, J.-Ph.: Computational experience with a primal-dual interior-point method for smooth convex programming. *Optimization Methods and Software*, 3:285-310, 1994.
- [60] Zhang, Y.: On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming. *SIAM Journal on Optimization*, 8(2):365-386, 1998.
- [61] Zhangwill, W, I.: *Nonlinear Programming. A Unified Approach*. Prentice-Hall, Englewood Cliffs, NJ, 1969.
- [62] Zhao, Q.: *Semidefinite programming for assignment and partitioning problem*. PhD thesis, University of Waterloo, Ontario, Canada, 1996.
- [63] Zhao, Q., Karisch, S. E., Rendl, F., Wolkowicz, H.: *Semidefinite programming relaxations for the quadratic assignment problem*. University of Waterloo, corr report, 95-27, march 1997.