

**Equações Polinomiais Matriciais:
Métodos de Resolução**

Fernando Eduardo Menzaque

TESE APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM
MATEMÁTICA APLICADA

Área de concentração: Análise Numérica
Orientador: Prof. Dr. Cyro de Carvalho Patarra

— São Paulo, Março de 1993 —

a Marcela y Alejandro

Agradecimentos

- Em primeiro lugar a DEUS fonte de toda sabedoria.
- Ao *Cyro*, pela sua esmerada orientação, pelas inúmeras horas de conversas, pelo seu apoio e incentivo, pela sua dedicação e infinita paciência e principalmente . . . pela sua amizade. Tenho certeza que sem ele a minha tarefa teria sido muito mais difícil.
- Aos professores *Dr. Julio Claeysen* e *Dr. León Sinay*, pelos seus valiosos conselhos e dicas precisas durante a redação final da tese.
- À *Elisa*, minha gratidão pelas críticas e discussões proveitosas, e pelo seu capricho na correção dos originais da tese.
- À *Wania*, pela sua inestimável colaboração na utilização das estações de trabalho.
- À *Angela* e ao *Alexandre*, nossos amigos brasileiros que nos socorreram quando chegamos no Brasil.
- Ao IME, e através dele a todos os docentes, funcionários e colegas que de uma ou outra maneira contribuíram para a concretização desta tese, em especial aos *professores* do MAP, pelo seu apoio e colaboração espontânea.
- Ao CCE pela colaboração durante o desenvolvimento dos programas.
- Ao CNPq pelo apoio financeiro.

- Al *Ing. Niell*, que con dedicación afecto guió mis primeros pasos por el Análisis Numérico.
- A *mis colegas* del Grupo de Análisis Numérico y Computación, mi agradecimiento por su apoyo y constante incentivo.
- A *mis colegas* de FAMAF, que me reemplazaron durante mi ausencia.
- Al FAMAF por la formación y por la ayuda financiera.
- A *Zulema y Domingo*, nuestra familia en São Paulo.
- A *Patricia y Pablo*, nuestros amigos cordobeses que nos brindaron su casa y su amistad.
- A *Marcela y Alejandro*, por su amor, apoyo, ayuda, incentivo, paciencia y comprensión. Sin ellos nada hubiera conseguido.

A todos, muito obrigado.

Finalmente, a mis padres ...

para quienes no tengo palabras, ...

pero ellos entienden.

Sumário

Agradecimentos	v
Resumo	xi
Abstract	xiii
Resumen	xv
Glossário de Símbolos	xvii
Capítulo I. Introdução	1
Capítulo II. Polinômios Matriciais	5
1. Modelos Matriciais	6
1.1. Análise dinâmica de vibrações	6
1.2. Previsão numérica do tempo	7
2. Definições	9
3. A derivada de Fréchet dos polinômios matriciais	15
3.1. Teoria de Perturbações	19
3.2. Singularidade da derivada de Fréchet	24
4. Divisão de polinômios matriciais	26

5.	Existência de solventes	31
5.1.	Cadeia de solventes e bi-diagonalização da matriz companheira	34
5.2.	Co-solventes e diagonalização da matriz companheira	40
6.	Interpolação	43
7.	Avaliação de polinômios matriciais	48
7.1.	Análise do erro de arredondamento	49
8.	Aplicações	52
Capítulo III. Métodos para o cálculo de solventes		59
1.	Método de Jordan	60
2.	Método de Newton	61
2.1.	Considerações sobre a aplicação do método	63
2.2.	Cálculo da correção Q_i . Polinômios de segundo grau	65
2.3.	Cálculo da correção Q_i . Caso Geral	66
2.4.	Raiz quadrada	67
3.	Método de Newton modificado	69
3.1.	Considerações sobre a aplicação do método	71
4.	Cálculo do solvente dominante I	72
5.	Cálculo do solvente dominante II	76
5.1.	Considerações sobre a aplicação do método	84
6.	Solventes de polinômios com coeficientes triangulares	84
6.1.	Considerações sobre a aplicabilidade do método	93
6.2.	Aplicação à previsão numérica do tempo	94
6.3.	Cálculo da raiz quadrada	97
7.	Polinômios com coeficientes triangulares por blocos	99
8.	Conclusões	104
Capítulo IV. Transformação simultânea de matrizes		107
1.	Triangularização simultânea	107
1.1.	Existência de autovetores comuns linearmente independentes	114
1.2.	Triangularização por transformações bi-unitárias	116
2.	Diagonalização simultânea	121
Capítulo V. Programas		125
1.	Descrição	125
2.	Listagens	133

SUMÁRIO	ix
Apêndice A. Álgebra Linear	145
1. Definições	145
1.1. Produto de Kronecker	146
2. Matrizes definidas por blocos	148
2.1. Funções de matrizes triangulares por blocos	149
Bibliografia	153
Índice	156

Resumo

Apresenta-se um estudo detalhado das equações polinomiais matriciais (EPM) e dos métodos utilizados para resolvê-las. Com este objetivo, são abordados os aspectos teóricos relacionados ao cálculo de soluções de uma EPM, procurando reunir os principais resultados sobre o assunto. A seguir, os algoritmos para resolver EPM são descritos detalhadamente, mencionando as particularidades inerentes à sua implementação, para serem posteriormente comparados.

Os resultados originais mais importantes estão relacionados ao desenvolvimento de um método direto para resolver uma equação polinomial matricial quando todos os coeficientes são matrizes triangulares superiores (EPMCTS). Proporciona-se uma caracterização das EPMCTS que admitem uma solução triangular superior, um algoritmo para o cálculo destas soluções e o programa que o implementa. Realiza-se um estudo da estabilidade das soluções de uma equação polinomial matricial em geral e das soluções proporcionadas pelo algoritmo em particular. Deduzem-se estimativas do erro.

O método é estendido para resolver equações polinomiais matriciais quando todos os coeficientes são matrizes triangulares superiores por blocos. No final do trabalho são discutidas algumas condições para a triangularização simultânea de um conjunto de matrizes.

Abstract

A detailed study about matrix polynomial equations (MPE) and the methods to solve them are presented. The theoretic subjects related to the solutions of a MPE are studied, in order to give the main results. The algorithms to solve a MPE are carefully described, emphasizing the particularities inherent to their implementation for a later comparison.

The more important original results are related to the development of a direct method to solve a matrix polynomial equation when all the coefficients are upper triangular matrices (MPEUTC). A characterization of MPEUTC that admits an upper triangular solution is presented. We also give an algorithm to compute these solutions and the program that implements the proposed method. A stability study of the polynomial matrix equation's solution and of the algorithm's solution are given. Error estimatives are derived.

The method is extended to solve polynomial matrix equations with block upper triangular coefficients. Some conditions for the simultaneously triangularization of a matrix set are discussed in the final part.

Resumen

En este trabajo se realiza un estudio detallado de las ecuaciones polinomiales matriciales (EPM) y de los métodos utilizados para resolverlas. Con este objetivo, son abordados los aspectos teóricos relacionados al cálculo de soluciones de una EPM, procurando reunir los principales resultados sobre el asunto. A continuación, se describen con detenimiento los algoritmos para resolver una EPM mencionando las particularidades inherentes a su implementación, para ser posteriormente comparados.

Los resultados originales más importantes están relacionados al desarrollo de un método directo para resolver una ecuación polinomial matricial cuando todos los coeficientes son matrices triangulares superiores (EPMCTS). Son caracterizadas las EPMCTS que tienen una solución triangular superior. Son descritos un algoritmo para el cálculo de estas soluciones y un programa que lo implementa. Se estudia la estabilidad de las soluciones de una ecuación polinomial matricial en general y de la calculada por el algoritmo en particular. Se calcula una estimativa del error cometido por el algoritmo.

El método es extendido para resolver ecuaciones polinomiales matriciales cuando todos los coeficientes son matrices triangulares superiores por bloques. Al final del trabajo son discutidas algunas condiciones para la triangularización simultánea de un conjunto de matrices.

Glossário de Símbolos

- A^* matriz transposta conjugada de A , 146
- $(a_{ij})_1^r$ matriz com entradas a_{ij} , 145
- $(A_{ij})_1^r$ matriz com blocos A_{ij} , 148
- $A \otimes B$ produto de Kronecker das matrizes A e B , 146
- $A \sim B$ as matrizes A e B são similares, 35
- A^t matriz transposta de A , 16
- A^T matriz transposta por blocos de A , 149
- $\text{Blcirc}(S_1, \dots, S_k)$ matriz circulante por blocos gerada pelos blocos S_1, \dots, S_k , 7
- \mathbb{C} números complexos, 145
- $[\xi_1, \dots, \xi_n]$ matriz com colunas ξ_1, \dots, ξ_n , 31
- $A \subseteq B$ o conjunto A está contido em B , 14
- $A \subset B$ o conjunto A está estritamente contido em B , 14

- $\mathbb{C}^{r \times s}$ matrizes complexas com r linhas e s colunas, **145**
- \mathbb{C}^r espaço dos vetores complexos de dimensão r , **145**
- δ_{ij} delta de Kronecker, **120**
- $\text{diag}(R_1, \dots, R_k)$ matriz diagonal por blocos gerada por R_1, \dots, R_k , **41**
- $\text{diag}(\lambda_1, \dots, \lambda_n)$ matriz diagonal com elementos $\lambda_1, \dots, \lambda_n$, **31**
- \emptyset conjunto vazio, **57**
- $fl(x)$ representante em ponto flutuante do número x , **49**
- $fl(*)$ operação $*$ realizada em ponto flutuante, **50**
- i unidade imaginária, **45**
- I matriz identidade, **10**
- $|\cdot|$ módulo de um complexo, **50**
- $\|\cdot\|_1$ norma vetorial ou norma matricial induzida pela norma vetorial, **145**
- $\|\cdot\|_2$ norma euclideana de um vetor, ou norma matricial induzida pela norma euclideana, **145**
- $\|\cdot\|_F$ norma de Frobenius, **146**
- $\|\cdot\|_\infty$ norma do máximo de um vetor, ou norma matricial induzida pela norma vetorial do máximo, **145**
- $P'(X_0)[H]$ derivada de Fréchet do polinômio matricial $P(X)$ no ponto X_0 , avaliada em H , **15**
- $P'(X_0)$ operador derivada de Fréchet do polinômio matricial $P(X_0)$ no ponto X_0 , **15**
- $P^*(X)$ operador derivada de Fréchet do polinômio matricial $P(X)$ no ponto X_0 , quando o espaço das matrizes $\mathbb{C}^{n \times n}$ for pensado como o espaço vetorial \mathbb{C}^{n^2} via a transformação $v(\cdot)$, **16**
- $P^*(X)v(H)$ derivada de Fréchet do polinômio matricial $P(X)$ no ponto X_0 , avaliada em H , **16**
- $P(\lambda)$ lambda matriz associada ao polinômio matricial $P(X)$, **10**
- $P(X)$ polinômio matricial na variável X , **9**

- $\hat{P}(X)$ polinômio matricial obtido a partir de $P(X)$ trocando a posição dos coeficientes com a variável, 9
- (Q, R) co-solvente, 14
- $\text{Im}(A)$ imagem da matriz A , 14
- $\sigma(A)$ espectro da matriz A , 31
- \mathcal{T}_n conjunto das matrizes triangulares de dimensão n , 111
- $\text{tr}(A)$ traço da matriz A , 146
- \mathcal{U}_n conjunto das matrizes unitárias de dimensão n , 111
- $v(\cdot)$ função que transforma uma matriz num vetor, 147
- $V(z_1, \dots, z_m)$ matriz de Vandermonde associada aos escalares z_1, \dots, z_m , 12
- $V(Z_1, \dots, Z_m)$ matriz de Vandermonde por blocos associada às matrizes Z_1, \dots, Z_m , 12
- $\langle \xi, \eta \rangle$ produto escalar entre os vetores ξ e η , 120
- $(\xi_i)_1^r$ vetor com coordenadas ξ_1, \dots, ξ_r , 145

Capítulo I

Introdução

O estudo das equações polinomiais matriciais (EPM) surge da necessidade de resolver equações diferenciais (e de diferenças) de ordem superior com coeficientes matriciais, sem apelar para o recurso de transformá-las em equações de primeira ordem. Esta transformação, embora teórica e praticamente muito empregada (ver Seções II-5 e II-8), acarreta grandes dificuldades para o seu tratamento numérico, devidas, principalmente, à introdução de maiores erros de arredondamento provocados pela maior dimensão das matrizes e dos vetores envolvidos.

A teoria das equações diferenciais lineares mostra que os subespaços de uma equação de evolução homogênea satisfazem que a soma de duas soluções que estão num mesmo subespaço é uma outra solução que pertence ao subespaço. Numericamente isto nem sempre acontece, é freqüente que nos problemas de evolução tendo algumas soluções limitadas e outras que crescem para o infinito, os erros de arredondamento provoquem a introdução de uma componente na direção das soluções ilimitadas, embora as condições iniciais pertençam a uma solução limitada. Em determinadas condições, este problema pode ser resolvido usando as soluções das equações polinomiais matriciais. Ver Seção II-8.

O presente trabalho tem por finalidade estudar um método direto para resolver equações polinomiais matriciais e efetuar a sua comparação com os algo-

ritmos já existentes. Com este propósito, na primeira parte, são abordados os aspectos teóricos relacionados ao cálculo de soluções de uma equação polinomial matricial, procurando-se reunir os principais resultados sobre o assunto. Assim, no Capítulo II são consideradas as EPM dando ênfase à existência das suas soluções, embora sejam tratados temas como interpolação, avaliação de polinômios matriciais e análise do erro de arredondamento na avaliação. Tais tópicos serão úteis para a definição e comparação dos métodos numéricos. Sempre que possível, faz-se uma analogia com a teoria das equações polinomiais escalares, com o intuito de ressaltar os pontos que dificultam a aplicação dos algoritmos empregados para o cálculo dos zeros de um polinômio escalar a uma EPM.

Os resultados originais deste capítulo são o Teorema 3.2 e o Teorema 3.3, duas extensões do resultado de Davis [9] sobre estabilidade de solventes, o Teorema 3.4 que fornece uma condição necessária e suficiente para a regularidade da derivada de Fréchet de um polinômio matricial e, finalmente, o Teorema 7.1 que proporciona uma estimativa para o erro na avaliação de um polinômio matricial estendendo um resultado de Davis [9]. Sempre que possível, construíram-se exemplos para ilustrar os conceitos apresentados.

O Capítulo III é dedicado aos métodos para a resolução das equações polinomiais matriciais. Cada um dos algoritmos é cuidadosamente descrito, ressaltando os requisitos indispensáveis para a sua aplicação e implementação. Apresenta-se um método original (ver Seção III-6) para calcular solventes de polinômios com coeficientes triangulares. Para este algoritmo é feito um estudo detalhado sobre as condições de aplicabilidade e estabilidade; também proporcionam-se uma estimativa do erro e o programa que foi utilizado nos testes. Os resultados originais encontram-se designados por Teorema 6.1, Teorema 6.2, Teorema 6.3 e Teorema 7.1. Na Seção III-7, estuda-se uma extensão do algoritmo acima que possibilita resolver equações polinomiais cujos coeficientes são matrizes triangulares por blocos. Na Seção 2.3, existe uma contribuição que permite reduzir o número de operações na implementação do algoritmo de Newton.

Levando-se em conta o bom desempenho apresentado pelo algoritmo para resolver equações com coeficientes triangulares, o Capítulo IV é dedicado ao estudo das condições para a existência de transformações que permitam a triangularização (ou diagonalização) simultânea de um conjunto de matrizes. Os resultados originais são o Teorema IV-1.3 e o Teorema IV-1.4, o quais estão relacionados à existência de transformações bi-unitárias que triangularizam simultaneamente um conjunto dado.

No último capítulo, são analisados alguns pontos a serem considerados na implementação do Algoritmo III-6.1, e do algoritmo empregado na triangularização simultânea de um conjunto de matrizes (ver Lema IV-1.1). No final do capítulo são fornecidas as listagens dos programas desenvolvidos.

No Apêndice A, são incluídos alguns resultados simples, e outros pouco conhecidos, sobre matrizes, que foram empregados em algum ponto do trabalho. No intuito de simplificar o enunciado dos resultados dos primeiros capítulos, neste apêndice também descrevem-se alguns símbolos e notações.

Neste trabalho foram consideradas equações sobre espaços de dimensão finita. Alguns resultados sobre equações com operadores polinomiais limitados e as suas aplicações em espaços de dimensão infinita podem ser encontrados em [23, 24, 25].

Como mostra o Teorema de Bézout generalizado (ver Teorema II-2.1), existe uma relação entre o problema de resolver uma equação polinomial matricial, e o problema de fatorar uma lambda matriz, porém, o segundo é bem mais abrangente e nem sempre conduz às soluções do primeiro. A fatoração de lambda matrizes é amplamente estudado em [17].

Finalmente, um esclarecimento sobre a numeração e citação de teoremas, lemas, etc. Estes itens possuem uma numeração independente dentro de cada capítulo, assim, o Teorema 2.1 refere-se ao primeiro teorema da Seção 2 do capítulo no qual se faz a citação. No momento em que for referenciado algum resultado de outro capítulo, ao número do resultado em questão lhe será acrescentado o número do capítulo ao qual pertence; assim, para referir-se ao Teorema 2.1 do Capítulo III é empregada a citação Teorema III-2.1. Os números das fórmulas serão colocados entre parênteses. A bibliografia, numerada segundo a ordem alfabética afim de facilitar a procura das referências pertinentes, é citada por números entre colchetes.

Capítulo II

Polinômios Matriciais

Neste capítulo, estudam-se aspectos teóricos dos polinômios matriciais da forma $P(X) = \sum_{l=0}^m A_{m-l}X^l$ e da equação $P(X) = 0$, onde todas as matrizes A_0, \dots, A_m e X são quadradas e de dimensão n .

Em primeiro lugar discutem-se algumas definições gerais sobre a equação e as suas soluções. A seguir, estuda-se a derivada de Fréchet de um polinômio matricial e a sua influência na estabilidade das soluções de $P(X) = 0$. Posteriormente, caracterizam-se as soluções da equação matricial nas quais a derivada de Fréchet de $P(X)$ é singular. Logo após uma generalização do conceito de divisão de polinômios, estudam-se a existência de soluções da equação $P(X) = 0$ e a sua relação com algumas propriedades da matriz companheira.

Na Seção 6, generaliza-se a noção de interpolação, e nas seções seguintes, estudam-se a avaliação de polinômios e a propagação dos erros de arredondamento. No final do capítulo, apresentam-se, sucintamente, algumas aplicações das soluções das equações polinomiais matriciais. Estes tópicos serão úteis na definição e comparação dos algoritmos apresentados no Capítulo III.

1. Modelos Matriciais

Os polinômios matriciais $P(X) = \sum_{l=0}^m A_{m-l} X^l$ surgem no estudo de variadas aplicações. Neste trabalho, a atenção é dirigida particularmente aos problemas que provém da análise dinâmica de vibrações e métodos numéricos de passo múltiplo para equações diferenciais, em particular, na previsão do tempo.

1.1. Análise dinâmica de vibrações

As equações de movimento de um sistema vibracional não conservativo podem ser obtidas através de várias técnicas, tais como as leis de Newton, equações de Lagrange, métodos lineares com grafos, e método de elementos finitos [38]. Todas elas conduzem à equação diferencial matricial de segunda ordem

$$M\mathbf{u}'' + C\mathbf{u}' + K\mathbf{u} = \mathbf{f}, \quad (1.1)$$

onde \mathbf{u} é um vetor n dimensional que representa os deslocamentos das massas do sistema, \mathbf{f} é o vetor das forças externas aplicadas no sistema, e M , C e K são as matrizes de massa, atrito e rigidez respectivamente. Em particular, a equação

$$M\mathbf{u}'' + (D + G)\mathbf{u}' + (K + H)\mathbf{u} = \mathbf{f}, \quad (1.2)$$

onde M , D e K são matrizes simétricas e G e H são matrizes antisimétricas para os efeitos giroscópicos e circulatorio, representa um dos modelos de maior complexidade na teoria das vibrações.

O estudo da equação (1.1) tem sido realizado, principalmente, através da determinação da resposta $\mathbf{u} = e^{i\lambda t}\mathbf{v}$ para uma excitação harmônica $\mathbf{f} = e^{i\lambda t}\boldsymbol{\xi}$, o que por sua vez motiva o estudo das lambda-matrizes. Porém, o estudo de modelos de certa complexidade (caso não simétrico) tem apresentado muitas dificuldades. Daí que o desenvolvimento de métodos diretos no domínio tempo, isto é, que não requerem o cálculo explícito de autovalores nem o uso de autovetores, tem começado a despertar a atenção dos pesquisadores [6, 43]. Com este propósito, o interesse deste trabalho orienta-se à construção de soluções matriciais exponenciais e ao estudo de solventes, em analogia ao caso escalar, e restringindo a classe dos coeficientes matriciais. Mais precisamente, a determinação de raízes da equação matricial

$$MX^2 + CX + K = 0,$$

quando M , C e K são triangularizáveis.

1.2. Previsão numérica do tempo

No estudo da previsão numérica do tempo, os modelos que provém das equações da água rasa são do maior interesse. Neste caso, a integração na variável vertical é realizada através de camadas descritas por equações da água rasa [2, 5, 7]. Por exemplo, um modelo linear unidimensional [2, 7] pode ser descrito como

$$\varphi' + \mathcal{L}\varphi = 0, \quad (1.3)$$

onde φ é o vetor de componentes u, v, ϕ , e o operador \mathcal{L} é definido por:

$$\mathcal{L} = \begin{pmatrix} U\partial x & -f & \partial x \\ f & U\partial x & 0 \\ \Phi\partial x & fU & U\partial x \end{pmatrix}. \quad (1.4)$$

Partindo o operador \mathcal{L} na soma de dois operadores \mathcal{S} e \mathcal{F} [2], obtém-se o seguinte esquema de diferenças para integrar a equação (1.4):

$$(I + \Delta t \mathcal{F})\varphi^{(n+1)} = -2\Delta t \mathcal{S}\varphi^{(n)} + (I - \Delta t \mathcal{F})\varphi^{(n-1)} \quad (1.5)$$

Uma das possibilidades para a escolha de \mathcal{F} e \mathcal{S} é considerar:

$$\mathcal{S} = U \begin{pmatrix} \partial x & 0 & 0 \\ 0 & \partial x & 0 \\ 0 & 0 & \partial x \end{pmatrix} \quad e \quad \mathcal{F} = \begin{pmatrix} 0 & -f & \partial x \\ f & 0 & 0 \\ \Phi\partial x & fU & 0 \end{pmatrix}. \quad (1.6)$$

Supondo que $x_0, \dots, x_k \in J$ são pontos equidistantes e que as funções u, v e ϕ são periódicas, usando diferenças divididas para aproximar o operador ∂_x , a equação de diferenças (1.5) pode ser discretizada da seguinte forma:

$$(I + \Delta t F)\varphi^{(n+1)} = -2\Delta t S\varphi^{(n)} + (I - \Delta t F)\varphi^{(n-1)}, \quad (1.7)$$

onde $\varphi^{(n)}$ é o vetor

$$\varphi^{(n)} = \left(u_1^{(n)}, v_1^{(n)}, \phi_1^{(n)}, u_2^{(n)}, v_2^{(n)}, \phi_2^{(n)}, \dots, u_k^{(n)}, v_k^{(n)}, \phi_k^{(n)} \right)^t,$$

quando

$$u_i^{(n)} = u(x_i, t_n) \quad v_i^{(n)} = v(x_i, t_n) \quad e \quad \phi_i^{(n)} = \phi(x_i, t_n)$$

e as matrizes S e F são bloco circulantes, definidas por:

$$S = \text{Blcirc}(S_1, S_2, 0, \dots, 0, S_k) \quad e \quad F = \text{Blcirc}(F_1, F_2, 0, \dots, 0, F_k).$$

Se as diferenças centradas forem utilizadas para discretizar o operador ∂_x , os blocos F_i e S_i são definidos por:

$$S_1 = 0 \quad S_2 = \frac{U}{2h}I \quad S_k = -\frac{U}{2h}I$$

$$F_1 = \begin{pmatrix} 0 & -f & 0 \\ f & 0 & 0 \\ 0 & fU & 0 \end{pmatrix} \quad F_2 = \frac{1}{2h} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \Phi & 0 & 0 \end{pmatrix} \quad F_k = -\frac{1}{2h} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \Phi & 0 & 0 \end{pmatrix}.$$

Se para aproximar ∂_x forem empregadas diferenças avançadas, F_i e S_i resultam:

$$S_1 = -\frac{U}{h}I \quad S_2 = \frac{U}{h}I \quad S_k = 0$$

$$F_1 = \begin{pmatrix} 0 & -f & -\frac{1}{h} \\ f & 0 & 0 \\ -\frac{\Phi}{h} & fU & 0 \end{pmatrix} \quad F_2 = \frac{1}{h} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \Phi & 0 & 0 \end{pmatrix} \quad F_k = 0.$$

Empregando diferenças avançadas ou centradas, as propriedades das matrizes circulantes em blocos [10], mostram que as matrizes S e F podem ser transformadas simultaneamente em matrizes diagonais por blocos, cujos blocos diagonais são dados respectivamente por:

$$\tilde{S}_i = S_1 + \omega^i S_2 + \omega^{-i} S_k \quad \text{e} \quad \tilde{F}_i = F_1 + \omega^i F_2 + \omega^{-i} F_k, \quad (1.8)$$

onde ω é a k -ésima raiz principal da unidade.

Levando-se em conta que as matrizes $I + \Delta t \tilde{F}_i$, $2\Delta t \tilde{S}_i$ e $-I + \Delta t \tilde{F}_i$ comutam, o Corolário IV-1.1 mostra que existe uma matriz unitária que transforma as matrizes $I + \Delta t \tilde{F}_i$, $2\Delta t \tilde{S}_i$ e $-I + \Delta t \tilde{F}_i$ nas matrizes triangulares Θ_i , Υ_i e Ω_i respectivamente. Assim sendo, resolver a equação (1.7) se reduz a resolver k equações de diferenças da forma:

$$\Theta_i \xi^{(n+1)} + \Upsilon_i \xi^{(n)} + \Omega_i \xi^{(n-1)} = 0, \quad (1.9)$$

onde Θ_i , Υ_i e $\Omega_i \in \mathbb{C}^{3 \times 3}$.

Este tipo de equações matriciais de diferenças de segunda ordem podem ser obtidas, na integração temporal do modelo DYNAMO [32] utilizando o método "leapfrog" ou diferenças centrais no tempo. Deve ser observado que a discretização do termo de advecção origina, em geral, matrizes não simétricas.

Será estabelecido que (1.9) pode ser transformada na equação polinomial matricial

$$P_i(Y) = \Theta_i Y^2 + \Upsilon_i Y + \Omega_i = 0. \quad (1.10)$$

2. Definições

Esta seção é dedicada ao resumo e à organização da terminologia empregada no estudo dos polinômios matriciais, em particular daqueles conceitos relacionados às soluções das equações envolvendo estes polinômios.

Chama-se **polinômio matricial** a uma função definida no espaço das matrizes quadradas de dimensão n , $n > 1$, cuja forma é:

$$P(X) = \sum_{l=0}^m A_{m-l} X^l,$$

onde $m > 1$ e os coeficientes $\{A_l ; l = 0, \dots, m\}$ são matrizes quadradas de dimensão n . Em geral, os polinômios matriciais serão denotados por letras maiúsculas. Se A_0 for uma matriz não nula, diz-se que o polinômio matricial tem grau m .¹ O número n é chamado de **ordem do polinômio matricial**.

Associada ao polinômio matricial $P(X)$ tem-se:

$$P(X) = 0, \quad (2.1)$$

conhecida como **equação polinomial matricial**. Frequentemente, esta denominação também é empregada para referir à equação

$$\hat{P}(X) = \sum_{l=0}^m X^l A_{m-l} = 0. \quad (2.2)$$

No presente trabalho, em geral, consideram-se somente equações do tipo (2.1), pois se a matriz E for uma solução de (2.2), então a matriz E^t é uma solução da equação

$$\sum_{l=0}^m A_{m-l}^t X^l = 0. \quad (2.3)$$

¹Esta restrição significa que somente serão consideradas somente equações não lineais. Quando $m = 1$ a teoria aqui apresentada continua sendo válida, porém, existem outras formas melhores de resolver sistemas lineares.

Deve-se observar, porém, que a existência de uma solução de (2.3) não implica na existência de uma solução de (2.1) por exemplo, se $P(X)$ for o polinômio matricial definido por:

$$P(X) = X^3 + \frac{1}{5} \begin{pmatrix} -8 & -6 \\ 4 & 3 \end{pmatrix} X^2 + \frac{1}{5} \begin{pmatrix} 2 & 4 \\ -1 & -2 \end{pmatrix} X + \frac{1}{5} \begin{pmatrix} -2 & -4 \\ 1 & 2 \end{pmatrix}$$

e E for a matriz $\frac{1}{5} \begin{pmatrix} +8 & +6 \\ -4 & -3 \end{pmatrix}$, então $\hat{P}(E) = 0$. No entanto, como veremos mais adiante, empregando o Algoritmo III-1.1, pode-se mostrar que não existe nenhuma matriz que seja solução da equação $P(X) = 0$.

Nas Seções 4 e 6, estudam-se algumas relações entre as soluções de (2.1) e as de (2.2). No intuito de simplificar a exposição, a partir deste momento, a notação $\hat{P}(X)$ referirá ao polinômio matricial obtido a partir de $P(X)$, trocando as posições dos coeficientes e da matriz variável.

Definição 2.1. Sejam λ um escalar e I a matriz identidade de ordem n , o polinômio $P(\lambda I)$ recebe o nome de **lambda matriz** associada a $P(X)$. Em geral, a lambda matriz $P(\lambda I)$ é denotada por $P(\lambda)$.

Observe-se que analogamente pode-se definir $\hat{P}(\lambda)$, sendo que $P(\lambda) = \hat{P}(\lambda)$.

O Teorema de Bézout generalizado [17] estende um conhecido resultado sobre polinômios escalares, o qual estabelece que, se z_0 for um número complexo arbitrário e $p(z)$ for um polinômio escalar de grau m com coeficientes complexos, então existe um polinômio $q(z)$ de grau $m - 1$ tal que $p(z) = q(z)(z - z_0) + p(z_0)$.

Teorema 2.1 (Teorema de Bézout generalizado). *Sejam $P(\lambda)$ uma lambda matriz e $Y \in \mathbb{C}^{n \times n}$ uma matriz arbitrária fixa, então existe uma lambda matriz $Q(\lambda)$ tal que*

$$P(\lambda) = Q(\lambda)(\lambda I - Y) + P(Y).$$

PROVA.

$$\begin{aligned} P(\lambda) &= \sum_{l=0}^m A_{m-l} \lambda^l \\ &= \left(\sum_{l=1}^m \lambda^{l-1} \sum_{k=l}^m A_{m-k} Y^{k-l} \right) (\lambda I - Y) + \sum_{l=0}^m A_{m-l} Y^l \\ &= \left(\sum_{l=1}^m \lambda^l \sum_{k=l}^m A_{m-k} Y^{k-l} \right) (\lambda I - Y) + P(Y). \quad \square \end{aligned}$$

Corolário 2.1. A matriz S é solução da equação $P(X) = 0$ se, e somente se,

$$P(\lambda) = Q(\lambda)(\lambda I - S), \quad (2.4)$$

onde $Q(\lambda)$ é definida por:

$$Q(\lambda) = \sum_{l=1}^m \lambda^{l-1} \sum_{k=l}^m A_{m-k} S^{k-l}.$$

Definição 2.2. Dado um polinômio $P(X)$, a matriz S é chamada **solvente** de $P(X)$, se $P(S) = 0$. Levando-se em conta que $P(\lambda) = Q(\lambda)(\lambda I - S)$, S é também chamada de **solvente à direita**. Se para todo autovalor λ_i de S tem-se $\det Q(\lambda_i) \neq 0$, $\lambda I - S$ é chamado **fator espectral à direita** de $P(\lambda)$.

Analogamente, se $\hat{P}(R) = 0$, a matriz R é chamada de **solvente** de $\hat{P}(X)$. Se R for um solvente de $\hat{P}(X)$, pode-se mostrar que, para alguma lambda matriz $\tilde{Q}(\lambda)$, $P(\lambda) = (\lambda I - R)\tilde{Q}(\lambda)$, assim sendo, R é chamada de **solvente à esquerda** de $P(X)$.

Como o determinante do produto de duas matrizes é igual ao produto dos determinantes, a equação (2.4) mostra um resultado importante: a reunião dos autovalores dos solventes de $P(X)$ está contida no conjunto das soluções da equação

$$\det P(\lambda) = \det \left(\sum_{l=0}^m A_{m-l} \lambda^l \right) = 0. \quad (2.5)$$

As raízes da equação (2.5) são conhecidas como **raízes latentes** de $P(\lambda)$. Se o polinômio $P(X)$ for **mônico**, isto é, quando $A_0 = I$, as raízes de (2.5) são os autovalores da matriz $C \in \mathbb{C}^{mn \times mn}$ definida por:

$$C = \begin{pmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ -A_m & -A_{m-1} & -A_{m-2} & \cdots & -A_1 \end{pmatrix}.$$

Como a matriz C tem dimensão mn , se $A_0 = I$, a lambda matriz $P(\lambda)$ tem exatamente mn raízes latentes. A matriz C é muito importante no estudo da existência de solventes de um polinômio matricial (ver Seção 5), e é conhecida como **matriz companheira por blocos** associada ao polinômio matricial $P(X)$.

Definição 2.3. Seja λ_i uma raiz latente de $P(\lambda)$. Neste caso, qualquer vetor, não nulo, no núcleo de $P(\lambda_i)$ é chamado de **vetor latente à direita** associado a λ_i .

Uma propriedade dos vetores e raízes latentes, que será de utilidade para estabelecer condições sobre a existência de solventes, é proporcionada pelo seguinte teorema.

Teorema 2.2 ([31]). *Se λ_i for uma raiz latente de $P(\lambda)$ e ξ_i for um vetor latente à direita associado a λ_i , então*

$$\begin{pmatrix} \xi_i \\ \lambda_i \xi_i \\ \vdots \\ \lambda_i^{m-1} \xi_i \end{pmatrix}$$

é um autovetor de C associado a λ_i .

PROVA. Aplicando as definições dos vetores e raízes latentes, o teorema resulta de uma simples verificação. \square

A **matriz de Vandermonde** associada aos escalares z_1, \dots, z_m definida por:

$$V(z_1, \dots, z_m) = \begin{pmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_m \\ \vdots & \vdots & & \vdots \\ z_1^{m-1} & z_2^{m-1} & \dots & z_m^{m-1} \end{pmatrix},$$

resulta ser de grande utilidade na teoria dos polinômios escalares. A matriz

$$V(X_1, \dots, X_m) = \begin{pmatrix} I & I & \dots & I \\ X_1 & X_2 & \dots & X_m \\ \vdots & \vdots & & \vdots \\ X_1^{m-1} & X_2^{m-1} & \dots & X_m^{m-1} \end{pmatrix}$$

desempenha na teoria dos polinômios matriciais, o papel da matriz de Vandermonde na teoria dos polinômios escalares. A matriz $V(X_1, \dots, X_m)$ é conhecida como **matriz de Vandermonde por blocos** associada a $X_1, \dots, X_m \in \mathbb{C}^{n \times n}$.

Existe uma grande diferença entre as matrizes de Vandermonde no caso escalar e no caso matricial. Sabe-se que

$$\det V(z_1, \dots, z_m) = \prod_{1 \leq i < j \leq m} (z_i - z_j)^2 \quad (2.6)$$

e portanto, a matriz $V(z_1, \dots, z_m)$ é inversível se, e somente se, todos os z_i forem distintos; como mostra o seguinte exemplo, o mesmo não ocorre no caso matricial.

Exemplo 2.1 ([11]). Sejam

$$X_1 = \begin{pmatrix} 2 & 0 \\ -2 & 1 \end{pmatrix}, \quad X_2 = \begin{pmatrix} 4 & 2 \\ 0 & 3 \end{pmatrix} \quad \text{e} \quad X_3 = \begin{pmatrix} 6 & 2 \\ 0 & 5 \end{pmatrix},$$

pode-se mostrar facilmente que $\det V(X_1, X_2) = \det(X_2 - X_1) = 0$ e que a matriz $V(X_1, X_2, X_3)$ é inversível.³

Esta característica da matriz de Vandermonde por blocos origina a seguinte definição.

Definição 2.4. Um conjunto de matrizes X_1, \dots, X_m é chamado **conjunto completo** quando a matriz $V(X_1, \dots, X_m)$ for não singular. Se S_1, \dots, S_m forem solventes de $P(X)$ para os quais $\det V(S_1, \dots, S_m) \neq 0$, o conjunto S_1, \dots, S_m recebe o nome de **conjunto completo de solventes**.

O seguinte exemplo mostra que, a propriedade descrita pelo Exemplo 2.1 é inerente ao conjunto e não à ordem na qual as matrizes são consideradas.

Exemplo 2.2. Se as matrizes X_1, X_2 e X_3 forem definidas por:

$$X_1 = \begin{pmatrix} 4 & -2 & 4 & 2 \\ 2 & 0 & 2 & 4 \\ 4 & 2 & 4 & -2 \\ 2 & 4 & 2 & 0 \end{pmatrix}, \quad X_2 = \begin{pmatrix} 3 & -1 & -3 & -3 \\ -1 & 3 & -3 & -3 \\ -3 & -3 & 3 & -1 \\ -3 & -3 & -1 & 3 \end{pmatrix}$$

e

$$X_3 = \begin{pmatrix} 4 & 2 & 0 & -6 \\ -4 & -2 & -4 & 2 \\ 0 & -6 & 4 & 2 \\ -4 & 2 & -4 & -2 \end{pmatrix},$$

pode-se mostrar que a matriz $V(X_1, X_2, X_3)$ é inversível enquanto que as matrizes $V(X_1, X_2)$, $V(X_1, X_3)$ e $V(X_2, X_3)$ são singulares. Isto significa que existem conjuntos completos nos quais nenhum subconjunto é completo.

Os exemplos precedentes também mostram que um conjunto completo pode conter subconjuntos completos ou não. Levando-se em conta os Exemplos 2.1 e 2.2, o conceito de conjunto completo parece pouco intuitivo, porém, nas Seções 6 e 8 mostra-se que os conjuntos completos surgem naturalmente.

²A notação empregada significa que o determinante da matriz $V(z_1, \dots, z_m)$ é o produto de todos os fatores $z_i - z_j$, possíveis, com $i < j$.

³A definição precisa de $\det V(X_1, X_2)$ é fornecida em Gantmacher [15]

O seguinte exemplo prova que a completitude de um conjunto de solventes é uma propriedade do conjunto e não do polinômio matricial considerado.

Exemplo 2.3. A equação $X^2 = X$ tem infinitas soluções⁴, chamadas de projetores, porém, nem todo par de soluções linearmente independentes formam um conjunto completo. Por exemplo, $S_1 = 0$ e $S_2 = I$ formam um conjunto completo, mas se S_1 e S_2 forem duas matrizes tais que

$$\{0\} \subset \text{Im}(S_1) \subseteq \text{Im}(S_2) \subseteq \mathbb{C}^n,$$

então S_1, S_2 não é um conjunto completo, pois $S_1 - S_2$ é singular.⁵

Os conjuntos completos serão utilizados exaustivamente nas Seções 6 e 8. Algumas condições suficientes para a inversibilidade da matriz de Vandermonde generalizada foram analisadas em [11] e [24].

Considere o conjunto de todas as matrizes similares ao solvente S , isto é, as matrizes $R \in \mathbb{C}^{n \times n}$ tais que existe uma matriz inversível Q satisfazendo

$$S = QRQ^{-1},$$

neste caso, para as matrizes Q e R vale

$$A_0QR^m + A_1QR^{m-1} + \dots + A_mQ = 0.$$

A equação acima origina a seguinte definição

Definição 2.5. Dadas duas matrizes $Q, R \in \mathbb{C}^{n \times n}$, o par (Q, R) é um **co-solvente** de $P(X)$ se $Q \neq 0$ e

$$A_0QR^m + A_1QR^{m-1} + \dots + A_mQ = 0.$$

Verifica-se facilmente que, o par (I, S) é um co-solvente de $P(X)$ se, e somente se, S for solvente de $P(X)$. Além disso, se (Q, R) for um co-solvente de $P(X)$ e U for uma matriz inversível, então (QU^{-1}, URU^{-1}) também é um co-solvente de $P(X)$.

Observe-se que sempre existem co-solventes do polinômio matricial $P(X)$. λ_i for uma raiz latente de $P(\lambda)$ associada ao vetor latente ξ_i , e se a matriz Q definida por $Q_i = [\xi_i, \dots, \xi_i]$, então $(Q_i, \lambda_i I)$ é um co-solvente de $P(X)$. Assim, sendo o conceito de co-solvente generaliza aquele de solvente. Lamentavelmente, co-solventes com propriedades adicionais são bem mais difíceis de calcular.

⁴Em [15] mostra-se que as soluções desta equação são todas as matrizes diagonalizáveis cujos autovalores são 0 e/ou 1.

⁵Para uma matriz $A \in \mathbb{C}^{n \times n}$ define-se $\text{Im}(A) = \{Ax : x \in \mathbb{C}^n\}$.

Algumas conseqüências da existência de co-solventes, como assim também, a utilização dos co-solventes para o cálculo de solventes, serão analisadas na Seção 5.2. Recentemente [27], o conceito de co-solventes foi estendido aos pares de matrizes (Q, R) onde $Q \in \mathbb{C}^{n \times r}$ e $R \in \mathbb{C}^{r \times n}$.

3. A derivada de Fréchet dos polinômios matriciais

Esta seção é dedicada ao estudo da derivada de Fréchet do polinômio matricial

$$P(X) = \sum_{l=0}^m A_{m-l} X^l. \quad (3.1)$$

Em particular, procura-se estabelecer uma relação entre a regularidade da derivada do polinômio matricial e o efeito de perturbações nos coeficientes do polinômio matricial sobre os solventes de $P(X)$ (ver Seção 3.1). A derivada de Fréchet de um polinômio matricial tem aplicações teóricas e práticas, pois é empregada para caracterizar a estabilidade dos solventes de $P(X)$ e, além disso, para definir alguns métodos destinados a calcular solventes do polinômio matricial (ver Seções III-2 e III-3).

Definição 3.1. A derivada de Fréchet do polinômio matricial $P(X)$ em X_0 , é o operador linear $P'(X_0)$ definido para toda matriz $H \in \mathbb{C}^{n \times n}$ por:

$$P'(X_0)[H] = \sum_{l=1}^m A_{m-l} \sum_{k=1}^l X_0^{k-1} H X_0^{l-k} = \sum_{l=1}^m \left(\sum_{k=l}^m A_{m-k} X_0^{k-l} \right) H X_0^{l-1}. \quad (3.2)$$

A derivada de Fréchet de $P(X)$ em X_0 é chamada **regular**, se o operador $P'(X_0)$ for bijetor.

Observe-se que $P'(X_0)$ é bijetor se, e somente se, $\inf_{\|H\|_F=1} \|P'(X_0)[H]\|_F > 0$, isto é, se $P'(X_0)$ for inversível.⁶

Definição 3.2. Um solvente S de $P(X)$ é chamado **solvente simples** se $P'(S)$ for regular.

Observação 3.1. Se X comutar com H tem-se que:

$$P'(X)[H] = P'(X)H, \quad \text{onde} \quad P'(X) = \sum_{l=1}^m l A_l X^{l-1},$$

isto é, se $XH = HX$ a derivada de Fréchet se reduz à derivada de polinômios escalares.

⁶ $\|A\|_F$ é a norma de Frobenius da matriz A (ver Apêndice A).

O operador $P'(X)$ é linear e, portanto, pode ser representado por uma matriz. O lema a seguir fornece esta representação em termos do produto de Kronecker \otimes (ver Seção A-1.1).

Lema 3.1. *Sejam $P(X)$ um polinômio matricial como em (3.1) e $P^*(X)$ a matriz definida por:*

$$P^*(X) = \sum_{l=1}^m \left((X^t)^{l-1} \otimes \sum_{k=l}^m A_{m-k} X^{k-l} \right),$$

onde X^t é a matriz transposta de X . Neste caso, a derivada de Fréchet de $P(X)$ satisfaz:

$$v(P'(X)[H]) = P^*(X)v(H).$$

Conseqüentemente, $P'(X)$ é regular se, e somente se, $P^*(X)$ for uma matriz não singular e, portanto

$$\inf_{\|H\|_F=1} \|P'(X)[H]\|_F = \min_{\|H\|_F=1} \|P^*(X)v(H)\|_2 > 0.$$

PROVA. A demonstração da primeira parte consiste em aplicar as definições da função $v(\cdot)$ (ver Seção A-1.1) e do produto de Kronecker à definição da derivada de Fréchet (3.2), pois:

$$\begin{aligned} v(P'(X)[H]) &= \sum_{l=1}^m v \left(\left(\sum_{k=l}^m A_{m-k} X^{k-l} \right) H X^{l-1} \right) \\ &= \sum_{l=1}^m \left((X^t)^{l-1} \otimes \sum_{k=l}^m A_{m-k} X^{k-l} \right) v(H). \end{aligned}$$

A prova da segunda parte consiste em notar que

$$\begin{aligned} \|P'(X)[H]\|_F &= \|v(P'(X)[H])\|_2 \\ &= \|P^*(X)v(H)\|_2. \end{aligned} \quad \square$$

Observação 3.2. Se $P'(X)$ for regular, a equação linear $P'(X)[H] = C$ admite uma única solução H , que satisfaz $v(H) = P^*(X)^{-1}v(C)$.

O objetivo do Teorema 3.1, que será enunciado a seguir, é mostrar que o operador $P'(X)$ definido por (3.2) é a derivada de Fréchet de $P(X)$ definida no contexto da análise funcional. A prova do teorema emprega as estimativas contidas na seguinte proposição.

Proposição 3.1 ([29]). *Dadas $A, B, C \in \mathbb{C}^{n \times n}$ e $k \in \mathbb{N}$, então*

$$\|A^k - B^k\|_F \leq \alpha_0 \|A - B\|_F \quad (3.3)$$

e

$$\left\| A^k - B^k - \sum_{l=0}^{k-1} C^{l-1} (A - B) C^{k-l} \right\|_F \leq \|A - B\|_F (\alpha_1 \|A - B\|_F + \alpha_2 \|A - C\|_F), \quad (3.4)$$

onde

$$\alpha_0 = \sum_{l=0}^{k-1} \|A\|_F^l \|B\|_F^{k-1-l},$$

$$\alpha_1 = k \sum_{l=0}^{k-2} \|C\|_F^l \|A\|_F^{k-2-l}$$

e

$$\alpha_2 = \sum_{l=2}^k \binom{k}{l} \|A - B\|_F^{l-2} \|A\|_F^{k-l}.$$

PROVA. A afirmação (3.3) decorre por indução em k . Para $k = 1$, a afirmação é trivial. Supondo, por indução, que (3.3) é verdadeira para k , tem-se:

$$\begin{aligned} \|A^{k+1} - B^{k+1}\|_F &= \frac{1}{2} \left\| (A^k + B^k)(A - B) + (A^k - B^k)(A + B) \right\|_F \\ &\leq \frac{1}{2} \|A - B\|_F \left(\|A\|_F^k + \|B\|_F^k \right. \\ &\quad \left. + (\|A\|_F + \|B\|_F) \sum_{l=0}^{k-1} \|A\|_F^l \|B\|_F^{k-1-l} \right) \\ &= \|A - B\|_F \left(\sum_{l=0}^k \|A\|_F^l \|B\|_F^{k-1-l} \right). \end{aligned}$$

Para demonstrar (3.4) seja $E = B - A$, logo:

$$B^k = (A + E)^k = A^k - \sum_{l=0}^{k-1} A^l (A - B) A^{k-1-l} + \sum_{l+j+i=k-2} A^l E A^j E A^i + \dots$$

Assim sendo,

$$A^k - B^k - \sum_{l=0}^k C^{l-1}(A - B)C^{k-l} = - \sum_{l=0}^{k-1} A^l(A - B)A^{k-1-l} - \sum_{l=0}^k C^{l-1}(A - B)C^{k-l} + \sum_{l+j+i=k-2} A^l E A^j E A^i + \dots,$$

e portanto,

$$\left\| A^k - B^k - \sum_{l=0}^k C^{l-1}(A - B)C^{k-l} \right\|_F \leq \beta_1 + \beta_2,$$

onde:

$$\begin{aligned} \beta_1 &= \left\| \sum_{l=1}^k A^{l-1}(A - B)A^{k-l} - \sum_{l=1}^k C^{l-1}(A - B)C^{k-l} \right\|_F \\ &\leq \|A - B\|_F \sum_{l=1}^k \|A\|_F^{l-1} \|A^{k-l} - C^{k-l}\|_F \\ &\quad + \|A - B\|_F \sum_{l=1}^k \|A^{l-1} - C^{l-1}\|_F \|C\|_F^{k-l} \\ &\leq \|A - B\|_F \|A - C\|_F \sum_{l=1}^k \left(\|A\|_F^{k-l} \sum_{j=0}^{l-2} \|A\|_F^j \|C\|_F^{l-2-j} \right. \\ &\quad \left. + \|C\|_F^{l-1} \sum_{j=0}^{k-1-l} \|A\|_F^j \|C\|_F^{k-1-l-j} \right) \\ &= \alpha_1 \|A - B\|_F \|A - C\|_F \end{aligned}$$

e

$$\begin{aligned} \beta_2 &= \sum_{l=2}^k \binom{k}{l} \|A - B\|_F^l \|A\|_F^{k-l} \\ &= \alpha_2 \|A - B\|_F^2. \end{aligned} \quad \square$$

A seguinte apresentação do resultado de Kratz e Stickel [29], permite uma boa caracterização da influência da derivada de Fréchet de $P(X)$, na estabilidade dos solventes do polinômio matricial. Ver Seção 3.1.

Teorema 3.1. *Seja $P(X)$ o polinômio matricial definido em (3.1), e sejam $X, Y, H \in \mathbb{C}^{n \times n}$ matrizes, e $\beta = \max \{\|X\|_F, \|Y\|_F, \|H\|_F\}$.*

Se $\max_{1 \leq l \leq m} \|A_l\|_F = \alpha$, então

$$\begin{aligned} \|P(X + H) - P(X) - P'(Y)[H]\|_F &\leq \\ &\begin{cases} 2^{m+1} \alpha \beta^{m-2} \|H\|_F (\|H\|_F + \|X - Y\|_F) & \text{se } \beta \geq 1 \\ 2^{m+1} \alpha \|H\|_F (\|H\|_F + \|X - Y\|_F) & \text{se } \beta < 1 \end{cases} \end{aligned} \quad (3.5)$$

e

$$P(X + H) = P(X) + P'(X)[H] + O(\|H\|_F^2) \quad \text{quando } \|H\|_F \rightarrow 0. \quad (3.6)$$

PROVA. A demonstração de (3.5) é uma consequência da Proposição 3.1. Se $\beta \geq 1$ tem-se:

$$\begin{aligned} \|P(X + H) - P(X) - P'(Y)[H]\|_F &= \\ &= \left\| \sum_{l=1}^m A_{m-l} \left((X + H)^l - X^l - \sum_{k=1}^l Y^{k-1} H Y^{l-k} \right) \right\|_F \\ &\leq \alpha \|H\|_F \sum_{l=2}^m \left[\|H\|_F \sum_{k=2}^l \binom{l}{k} \|H\|_F^{k-2} \|X\|_F^{l-k} \right. \\ &\quad \left. + \|X - Y\|_F l \sum_{k=0}^{l-2} \|Y\|_F^k \|X\|_F^{l-2-k} \right] \\ &\leq \alpha \beta^{m-2} \|H\|_F \left[\|H\|_F \sum_{l=2}^m \sum_{k=2}^l \binom{l}{k} + \|X - Y\|_F \sum_{l=2}^m l(l-1) \right] \\ &\leq 2^{m+1} \alpha \beta^{m-2} \|H\|_F (\|H\|_F + \|X - Y\|_F). \end{aligned}$$

A demonstração de (3.5) quando $\beta < 1$ decorre como acima, e a afirmação (3.6) segue-se de (3.5) substituindo Y por X . \square

3.1. Teoria de Perturbações

O cálculo de solventes é efetuado, em geral, através de algoritmos, envolvendo assim, implicitamente, o uso de computadores. Por causa da precisão finita com a qual os números são representados nos computadores, alguns erros, inevitáveis, são cometidos. Faz-se necessário, por isto, dispor de uma medida da qualidade dos resultados obtidos. O Teorema 3.2, abaixo, o qual estende um resultado de Davis [9] aos polinômios de grau arbitrário, fornece uma medida. Os resultados desta seção estão relacionados a erros na representação dos coeficientes do polinômio matricial, e nada tem a ver com os erros cometidos, por algum método particular, no cálculo de solventes.

Dado o polinômio

$$P(X) = \sum_{l=0}^m A_{m-l} X^l,$$

sejam S e $\tilde{S} = S + \Delta S$ solventes de $P(X)$ e do polinômio perturbado $\tilde{P}(X)$ respectivamente, onde

$$\tilde{P}(X) = \sum_{l=0}^m \tilde{A}_{m-l} X^l.$$

Sejam $P'(X)$ e $\tilde{P}'(X)$ as derivadas de Fréchet de $P(X)$ e $\tilde{P}(X)$, respectivamente, e sejam, por definição:

$$\Delta P(X) = \tilde{P}(X) - P(X) = \sum_{l=0}^m \Delta A_{m-l} X^l$$

e

$$\Delta P'(X) [H] = \tilde{P}'(X) [H] - P'(X) [H],$$

onde $\Delta A_i = \tilde{A}_i - A_i$.

O Teorema 3.2 fornece uma estimativa para $\|\tilde{S} - S\|_F$ quando $P'(S)$ for regular e $\|S\|_F \geq 1$.

Teorema 3.2. *Sejam $P(X)$ e $\tilde{P}(X)$ dois polinômios matriciais tais que, para algum ϵ positivo suficientemente pequeno, as perturbações ΔA_i satisfazem*

$$\|\Delta A_i\|_F \leq \epsilon \|A_i\|_F \quad (i = 0, \dots, m).$$

Se S e \tilde{S} forem solventes de $P(X)$ e $\tilde{P}(X)$, respectivamente, satisfazendo $\|\Delta S\|_F \leq \|S\|_F$ e, além disso, se S for um solvente simples de $P(X)$ para o qual $1 \leq \|S\|_F$ e

$$\|P'(S)^{-1}\|_F \|\Delta P(S)\|_F \leq \kappa < 1,$$

então $\|\Delta S\|_F \leq \gamma \epsilon$, onde:

$$\begin{aligned} \gamma &= \frac{2\alpha}{1 + \sqrt{1 - 4\alpha\beta\epsilon}},^7 \\ \alpha &= \frac{2}{1 - \kappa} \|P'(S)^{-1}\|_F \left(\sum_{l=1}^m \|A_{m-l}\|_F \|S\|_F^l \right), \\ \beta &= \frac{\delta(1 + \epsilon)}{1 - \kappa} \|P'(S)^{-1}\|_F \end{aligned}$$

⁷ $\gamma \approx \frac{2\alpha}{1+1-2\alpha\beta\epsilon} = \frac{\alpha}{1-\alpha\beta\epsilon} \approx \alpha(1 + \alpha\beta\epsilon)$.

e

$$\delta = 2^{m+1} \|S\|_F^{m-2} \max_{l=1, \dots, m} \{\|A_{m-l}\|_F\}.$$

PROVA. Se \tilde{S} for solvente de $\tilde{P}(X)$,

$$\begin{aligned} 0 &= \tilde{P}(\tilde{S}) \\ &= P(\tilde{S}) + \Delta P(\tilde{S}) \\ &= P(S) + P'(S)[\Delta S] + V_1(\Delta S) + \Delta P(S) + \Delta P'(S)[\Delta S] + V_2(\Delta S) \quad (3.7) \\ &= P'(S)[\Delta S] + \Delta P(S) + \Delta P'(S)[\Delta S] + V_1(\Delta S) + V_2(\Delta S). \end{aligned}$$

O Teorema 3.1 mostra que as matrizes $V_1(\Delta S)$ e $V_2(\Delta S)$ satisfazem as seguintes estimativas, $\|V_1(\Delta S)\|_F \leq \delta \|\Delta S\|_F^2$ e $\|V_2(\Delta S)\|_F \leq \delta \epsilon \|\Delta S\|_F^2$. Multiplicando (3.7) à esquerda por $P'(S)^{-1}$ tem-se:

$$-\Delta S = P'(S)^{-1} \Delta P'(S)[\Delta S] + P'(S)^{-1} (\Delta P(S) + V_1(\Delta S) + V_2(\Delta S)),$$

sendo $\|P'(S)^{-1}\|_F \|\Delta P'(S)\|_F \leq \kappa < 1$, segue-se que

$$\begin{aligned} \|\Delta S\|_F &\leq \frac{1}{1-\kappa} \|P'(S)^{-1}\|_F \|\Delta P(S) + V_1(\Delta S) + V_2(\Delta S)\|_F \\ &\leq \frac{1}{1-\kappa} \|P'(S)^{-1}\|_F [\|\Delta P(S)\|_F + \delta(1+\epsilon) \|\Delta S\|_F^2]. \end{aligned}$$

Uma vez que $\|\Delta A_l\|_F \leq \epsilon \|A_l\|_F$, e como

$$\|A_m\|_F \leq \sum_{l=1}^m \|A_{m-l}\|_F \|S\|_F^l,$$

obtem-se a seguinte desigualdade para $\|\Delta S\|_F$:

$$\|\Delta S\|_F \leq \frac{1}{1-\kappa} \|P'(S)^{-1}\|_F \left[2\epsilon \sum_{l=1}^m \|A_{m-l}\|_F \|S\|_F^l + \delta(1+\epsilon) \|\Delta S\|_F^2 \right],$$

portanto,

$$\|\Delta S\|_F \leq \alpha\epsilon + \beta \|\Delta S\|_F^2. \quad (3.8)$$

Sendo a desigualdade (3.8) satisfeita para $\|\Delta S\|_F \leq \epsilon\gamma$, o teorema fica provado. \square

O Teorema 3.2 fornece uma estimativa caso o erro relativo dos coeficientes seja pequeno, neste caso, se o polinômio $P(X)$ tiver um coeficiente nulo, o mesmo coeficiente deverá ser nulo no polinômio $\tilde{P}(X)$. O seguinte teorema levanta esta

restrição, e proporciona uma estimativa para $\|\Delta S\|_F$ quando o erro absoluto dos coeficientes for pequeno e $\|S\|_F \geq 1$.

Teorema 3.3. *Sejam $P(X)$ e $\tilde{P}(X)$ dois polinômios matriciais tais que, para algum ϵ positivo suficientemente pequeno, as perturbações ΔA_i satisfazem*

$$\|\Delta A_i\|_F \leq \epsilon \max_{l=1, \dots, m} \{\|A_{m-l}\|_F\} \quad (i = 0, \dots, m).$$

Se S e \tilde{S} forem solventes de $P(X)$ e $\tilde{P}(X)$, respectivamente, satisfazendo $\|\Delta S\|_F \leq \|S\|_F$ e, além disso, se S for um solvente simples de $P(X)$ para o qual $1 \leq \|S\|_F$ e

$$\|P'(S)^{-1}\|_F \|\Delta P(S)\|_F \leq \kappa < 1,$$

então $\|\Delta S\|_F \leq \gamma \epsilon$, onde:

$$\begin{aligned} \gamma &= \frac{2\alpha}{1 + \sqrt{1 - 4\alpha\beta\epsilon}}, \\ \alpha &= \frac{2}{1 - \kappa} \|P'(S)^{-1}\|_F \max_{l=1, \dots, m} \{\|A_{m-l}\|_F\} \left(\sum_{l=1}^m \|S\|_F^l \right), \\ \beta &= \frac{\delta(1 + \epsilon)}{1 - \kappa} \|P'(S)^{-1}\|_F, \end{aligned}$$

e

$$\delta = 2^{m+1} \|S\|_F^{m-2} \max_{l=1, \dots, m} \{\|A_{m-l}\|_F\}.$$

PROVA. A demonstração é análoga àquela do Teorema 3.2. □

Se, nos Teoremas 3.2 e 3.3, a hipótese $\|S\|_F \geq 1$ for substituída por $\|S\|_F \leq 1$, o Teorema 3.1 mostra que, definindo $\delta = 2^{m+1} \max_{l=1, \dots, m} \{\|A_{m-l}\|_F\}$, a conclusão continua sendo verdadeira. Já a hipótese $\|\Delta S\|_F \leq \|S\|_F$, embora razoável, pois $\tilde{P}(X) \rightarrow P(X)$ quando $\epsilon \rightarrow 0$, é um tanto restritiva quando $\|S\|_F$ for pequena, porém, esta hipótese é empregada como uma forma de garantir que \tilde{S} é o solvente mais próximo de S .

A singularidade da derivada de Fréchet representa um caso extremo de sensibilidade, no qual pequenos erros nos dados podem provocar grandes erros nos resultados. Isto não tem nada a ver com a precisão dos cálculos, é uma característica inerente aos polinômios matriciais. Os seguintes exemplos apresentam dois problemas típicos do que pode acontecer no caso extremo.

Exemplo 3.1 ([9]). Considere a equação

$$P_0(X) = \begin{pmatrix} 2 & 1 \\ 2 & 1 \end{pmatrix} (X^2 - 4I) = 0.$$

$P_0(X)$ tem infinitos solventes $S(x, y)$ que satisfazem

$$S(x, y)^2 = \begin{pmatrix} x & y \\ 8 - 2x & 4 - 2y \end{pmatrix} \quad x, y \in \mathbb{C},$$

logo, para qualquer solvente S tem-se

$$P^*(S) = \begin{pmatrix} 4s_{11} + 2s_{21} & 2s_{21} & 2s_{12} + s_{22} + s_{11} & s_{21} \\ 2s_{12} & 2s_{11} + s_{21} + 2s_{22} & s_{12} & 2s_{12} + 2s_{22} \\ 4s_{11} + 2s_{21} & 2s_{21} & 2s_{12} + s_{22} + s_{11} & s_{21} \\ 2s_{12} & 2s_{11} + s_{21} + 2s_{22} & s_{12} & 2s_{12} + 2s_{22} \end{pmatrix}$$

e, portanto, $P^*(S(x_1, x_2))$ é singular; entretanto, se

$$P_\epsilon(X) = P_0(X) + \epsilon \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

for uma perturbação de $P_0(X)$, a equação $P_\epsilon(X) = 0$ não tem solução. Se S_0 for um solvente de $P_0(X)$ e $S(\epsilon)$ for um solvente de $P_\epsilon(X)$ tal que $S(0) = S_0$ segue-se que:

$$P_\epsilon(S(\epsilon)) = P_0(S(\epsilon)) + \epsilon \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix},$$

donde, derivando em relação a ϵ em $\epsilon = 0$, tem-se que:

$$P'_\epsilon(S_0)[H] = - \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad (3.9)$$

onde $H = \left. \frac{d}{d\epsilon} S(\epsilon) \right|_{\epsilon=0}$. Sendo $P'_\epsilon(S_0)$ singular, o sistema (3.9) tem infinitas soluções ou nenhuma. Neste caso $P_\epsilon(X) = 0$ não tem solução.

Exemplo 3.2. O polinômio

$$P(X) = X^2 - \begin{pmatrix} 6 & -4 \\ 3 & -1 \end{pmatrix} X + \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix}$$

tem como solventes as matrizes

$$S_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{e} \quad S_2 = \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix};$$

além disso, a lambda matriz $P(\lambda)$ pode ser fatorada por:

$$\begin{aligned} P(\lambda) &= \left(\lambda I - \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \left(\lambda I - \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix} \right) \\ &= \left(\lambda I - \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix} \right) \left(\lambda I - \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right). \end{aligned}$$

Se $P_\epsilon(X)$ for uma perturbação de $P(X)$ dada por:

$$P_\epsilon(X) = X^2 - \left(\begin{pmatrix} 6 & -4 \\ 3 & -1 \end{pmatrix} + \epsilon \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \right) X + \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix} + 2\epsilon \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix},$$

a lambda matriz $P_\epsilon(\lambda)$ só pode ser fatorada por:

$$P_\epsilon(\lambda) = \left(\lambda I - \begin{pmatrix} 1 + \epsilon & -\epsilon \\ \epsilon & 1 - \epsilon \end{pmatrix} \right) \left(\lambda I - \begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix} \right).$$

Desta maneira, $P_\epsilon(X)$ e $P(X)$ possuem as mesmas raízes latentes, no entanto, nenhum solvete de $P_\epsilon(X)$ tem autovalores perto de $\lambda = 1$. De fato, o único solvete de $P_\epsilon(X)$ é $\begin{pmatrix} 5 & -4 \\ 3 & -2 \end{pmatrix}$ como pode ser comprovado pela aplicação do Algoritmo III-1.1.

Os Teoremas 3.2 e 3.3 permitem a obtenção das seguintes conclusões:

- (1) O cálculo do solvete S , quando $P(S)$ for singular, apresenta dificuldades para qualquer algoritmo, até mesmo para aqueles que não envolvem a avaliação da derivada de Fréchet.
- (2) Todo algoritmo empregado para calcular solvetes de $P(X)$ deveria detectar a possibilidade de uma derivada singular, como uma forma de estimar a qualidade da solução obtida.

3.2. Singularidade da derivada de Fréchet

As considerações feitas na Seção 3.1, mostram a necessidade de obter condições a serem satisfeitas por um solvete S de $P(X)$ para que o operador $P(S)$ seja regular. Estas condições serão estabelecidas no Corolário 3.1.

Dado o polinômio matricial

$$P(X) = \sum_{l=0}^m A_{m-l} X^l$$

e um solvente S de $P(X)$, seja $R(X)$ a matriz definida para cada $X \in \mathbb{C}^{n \times n}$ por:

$$R(X) = \sum_{l=1}^m \left((X^t)^{l-1} \otimes \sum_{k=l}^m A_{m-k} S^{k-l} \right).$$

O teorema a seguir, estabelece quando a matriz $R(X)$ é singular.

Teorema 3.4. *Seja S um solvente de $P(X)$ e seja $Q(\lambda)$ a lambda matriz definida por:*

$$P(\lambda) = Q(\lambda)(\lambda I - S).$$

Se X_0 for uma matriz arbitrária, então a matriz $R(X_0)$ é inversível se, e somente se,

$$\det Q(\lambda_i) \neq 0,$$

para todo λ_i autovalor de X_0 .

PROVA. Se X_0 for uma matriz arbitrária, segue-se, do Teorema de Schur (ver Apêndice A), que existem uma matriz unitária U e uma matriz triangular superior T , tais que a matriz transposta de X_0 satisfaz:

$$X_0^t = U^* T U.$$

Logo, pelas propriedades do produto de Kronecker (Seção A-1.1), a matriz $R(X_0)$ é similar à matriz

$$C = \sum_{l=1}^m \left(T^{l-1} \otimes \sum_{k=l}^m A_{m-k} S^{k-l} \right).$$

O resultado é uma consequência do fato de C ser uma matriz triangular superior por blocos, cujos blocos diagonais são $Q(\lambda_i)$, onde $\lambda_1, \dots, \lambda_n$ são os autovalores de X_0 . Uma vez que

$$\det C = \prod_{i=1}^n \det Q(\lambda_i),$$

segue-se que $\det C \neq 0$ se, e somente se, $\det Q(\lambda_i) \neq 0$ para todo i . □

Como $P^*(S) = R(S)$, obtém-se o seguinte corolário.

Corolário 3.1. *Se S for um solvente de $P(X)$, então $P^*(S)$ é regular se, e somente se, $\lambda I - S$ for um fator espectral à direita de $P(\lambda)$ (ver Definição 2.2).*

4. Divisão de polinômios matriciais

Nesta seção apresenta-se uma definição da divisão entre polinômios matriciais [11]. Com ela obtém-se tanto uma extensão da divisão escalar ($n = 1$), quanto da divisão de lambda matrizes. Além disso, a partir das propriedades da divisão entre polinômios matriciais, é possível estabelecer uma primeira relação entre os solventes de $P(X)$ e os de $\hat{P}(X)$. Além disso, com o estudo da divisão entre polinômios matriciais, tenta-se mostrar donde surge uma das dificuldades encontradas na generalização dos algoritmos empregados para calcular raízes de polinômios escalares.

Teorema 4.1 ([11]). *Sejam $m, p \in \mathbb{N}$ tais que $m \geq p$, e sejam $P(X)$ e $W(X)$ os polinômios matriciais definidos por:*

$$P(X) = X^m + A_1X^{m-1} + \cdots + A_m \quad e \quad W(X) = X^p + B_1X^{p-1} + \cdots + B_p.$$

Neste caso, existe um único polinômio matricial $F(X)$ de grau $m - p$, e um único polinômio matricial $L(X)$ cujo grau não excede $p - 1$, que satisfazem:

$$P(X) = F(X)X^p + B_1F(X)X^{p-1} + \cdots + B_pF(X) + L(X). \quad (4.1)$$

PROVA. Levando-se em conta que $P(X)$ e $W(X)$ são mônicos, o coeficiente do termo principal de $F(X)$ deve ser a matriz identidade. Sejam

$$F(X) = X^{m-p} + F_1X^{m-p-1} + \cdots + F_{m-p}$$

e

$$L(X) = L_0X^{p-1} + \cdots + L_{p-1}.$$

Igualando os coeficientes em ambos os lados de (4.1), obtém-se sucessivamente

$$F_1, \dots, F_{m-p}, L_0, \dots, L_{p-1},$$

pois

$$\begin{aligned} A_1 &= F_1 + B_1 \\ A_i &= F_i + \sum_{k+l=i} B_k F_l && (i = 2, \dots, m-p), \\ A_i &= L_{i-m+p-1} + \sum_{k+l=i} B_k F_l && (i = m-p+1, \dots, m-1), \\ A_m &= L_{p-1} + B_p F_{m-p}. \end{aligned}$$

Logo, os coeficientes de $F(X)$ e de $L(X)$ podem ser determinados, um de cada vez, seguindo a ordem estabelecida acima. \square

A equação (4.1) é a **divisão do polinômio matricial** $P(X)$ à esquerda por $W(X)$; o polinômio $F(X)$ é chamado **cociente** e o polinômio $L(X)$ **resto**.

Os coeficientes dos polinômios $F(X)$ e $L(X)$ existem e são únicos, pois os polinômios matriciais $P(X)$ e $W(X)$ são mônicos. Se algum dos polinômios $P(X)$ ou $W(X)$ não for mônico, a existência e unicidade de $F(X)$ e $L(X)$ dependerá de que os coeficientes A_0, \dots, A_m e B_0, \dots, B_p satisfaçam algumas relações adicionais. Note-se, por exemplo, que se B_0 for uma matriz singular, da existência do polinômio $F(X)$ resulta que o sistema de equações

$$B_0 Y = A_0$$

é consistente, pois $B_0 F_0 = A_0$, porém, em geral, essa condição não é suficiente.

O seguinte resultado generaliza uma propriedade importante dos polinômios que relaciona o resto e o dividendo.

Corolário 4.1. *Usando a mesma notação do Teorema 4.1, se R for um solvente de $\hat{W}(X)$, então $\hat{L}(R) = \hat{P}(R)$.*

PROVA. Seja $\hat{Q}(X)$ o polinômio matricial definido por $\hat{Q}(X) = \hat{P}(X) - \hat{L}(X)$, neste caso, reordenando os termos de $\hat{Q}(X)$ pode-se mostrar que:

$$\hat{Q}(X) = X^{m-p} \hat{W}(X) + X^{m-p-1} \hat{W}(X) F_1 + \dots + \hat{W}(X) F_{m-p}.$$

Assim sendo, para todo R , solvente de $\hat{W}(X)$, tem-se que $\hat{Q}(R) = 0$ e, portanto, o corolário fica demonstrado. \square

Se $W(X)$ tiver grau um, isto é, $W(X) = X - R$, $\hat{W}(X)$ e $W(X)$ coincidem, logo R é um solvente de $W(X)$ a ambos lados. Assim sendo, o Teorema 4.1 mostra que:

$$P(X) = F(X)X - RF(X) + L_0,$$

onde $F(X)$ é o polinômio

$$F(X) = X^{m-1} + \sum_{l=0}^{m-2} F_{m-2-l} X^l,$$

e os coeficientes F_1, \dots, F_{m-1} são definidos por:

$$F_i = A_i - B_{i-1} R.$$

Levando-se em consideração que $L_0 = \hat{P}(R)$, segue-se que:

$$P(X) = F(X)X - RF(X) + \hat{P}(R). \quad (4.2)$$

Para o polinômio $\hat{P}(X)$ existe uma definição análoga à da divisão entre polinômios matriciais.

Teorema 4.2. *Sejam $m, p \in \mathbb{N}$ tais que $m \geq p$, e sejam $P(X)$ e $W(X)$ os polinômios matriciais definidos por:*

$$P(X) = X^m + A_1 X^{m-1} + \cdots + A_m \quad e \quad W(X) = X^p + B_1 X^{p-1} + \cdots + B_p.$$

Neste caso, existe um único polinômio matricial $H(X)$ de grau $m - p$, e um único polinômio matricial $N(X)$ cujo grau não excede $p - 1$, que satisfazem:

$$\hat{P}(X) = X^p \hat{H}(X) + X^{p-1} \hat{H}(X) B_1 + \cdots + \hat{H}(X) B_p + \hat{N}(X),$$

Corolário 4.2. *Se S for um solvente de $W(X)$, então $N(S) = P(S)$.*

Considerando novamente $p = 1$, e $W(X) = X - S$ tem-se:

$$\hat{P}(X) = X \hat{H}(X) - \hat{H}(X) S + P(S). \quad (4.3)$$

Caso X seja igual a λI , das equações (4.2) e (4.3) obtém-se o Teorema de Bézout generalizado

$$\begin{aligned} P(\lambda) &= (\lambda I - R)F(\lambda) + \hat{P}(R) \\ &= H(\lambda)(\lambda I - S) + P(S), \end{aligned}$$

onde R e S são matrizes arbitrárias. Agora, se S e R forem solventes de $P(X)$ e $\hat{P}(X)$, respectivamente,

$$\begin{aligned} P(X) &= F(X)X - RF(X) \\ \hat{P}(X) &= \hat{H}(X)X - S\hat{H}(X), \end{aligned} \quad (4.4)$$

e portanto,

$$P(\lambda) = (\lambda I - R)F(\lambda) = H(\lambda)(\lambda I - S).$$

O seguinte corolário fornece uma condição necessária e suficiente para que um solvente de $P(X)$ seja solvente do cociente da divisão de $P(X)$ à esquerda por $X - R$.

Corolário 4.3. *Sejam S e R solventes de $P(X)$ e $\hat{P}(X)$ respectivamente, e seja $F(X)$ o cociente da divisão de $P(X)$ à esquerda por $X - R$. Neste caso, se R e S não tiverem autovalores em comum, então $F(S) = 0$.*

PROVA. Se R for um solvente de $\hat{P}(X)$,

$$F(X)X - RF(X) = P(X).$$

Como S é um solvente de $P(X)$, segue-se que

$$F(S)S - RF(S) = 0. \quad (4.5)$$

Levando-se em conta que as matrizes S e R não possuem autovalores em comum, o Teorema A-1.3 mostra que a única solução de (4.5) é $F(S) = 0$. \square

Seja R um solvente de $\hat{P}(X)$, e seja $F(X)$ o cociente da divisão de $P(X)$ à esquerda por $X - R$, neste caso, o Corolário 4.3 fornece uma condição para que um solvente S de $P(X)$ seja solvente de $F(X)$. O Exemplo 4.1 mostra que a condição fornecida pelo corolário não é necessária.

Exemplo 4.1. Seja $P(X)$ o polinômio matricial definido por:

$$P(X) = X^3 + \begin{pmatrix} 12.5 & -3.5 \\ 7.5 & 1.5 \end{pmatrix} X^2 + \begin{pmatrix} 39.5 & -13.5 \\ 31.5 & -5.5 \end{pmatrix} X + \begin{pmatrix} 37 & -13 \\ 33 & -9 \end{pmatrix}.$$

Como a matriz $R = \begin{pmatrix} 5 & -2 \\ 3 & 0 \end{pmatrix}$ é solvente de $\hat{P}(X)$, segue-se, da equação (4.4), que $P(X) = F(X)X - RF(X)$ onde

$$F(X) = X^2 + \begin{pmatrix} 7.5 & -1.5 \\ 4.5 & 1.5 \end{pmatrix} X + \begin{pmatrix} 11 & -3 \\ 9 & -1 \end{pmatrix}.$$

Sendo $S = \begin{pmatrix} -4.5 & 0.5 \\ -2.5 & -1.5 \end{pmatrix}$ um solvente de $P(X)$, tem-se $F(S)S - RF(S) = 0$. Observe-se que S e R têm um autovalor em comum, porém, $F(S)$ é a matriz nula.

O seguinte corolário estabelece uma relação entre os solventes dos polinômios $P(X)$ e $\hat{P}(X)$, e mostra como calcular um solvente de $\hat{P}(X)$ a partir de um solvente de $P(X)$.

Corolário 4.4. *Sejam S e R solventes de $P(X)$ e $\hat{P}(X)$ respectivamente, e seja $F(X)$ o cociente da divisão de $P(X)$ por $W(X) = X - R$. Se $F(S)$ for inversível, então*

$$R = F(S)SF(S)^{-1}.$$

PROVA. Se R for um solvente de $\hat{P}(X)$, o Teorema 4.1 aplicado a $W(X) = X - R$ mostra que existe um único polinômio cociente $F(X)$. Levando-se em conta que S é um solvente de $P(X)$ tal que $F(S)$ é inversível, então o corolário decorre de (4.5). \square

Um outro resultado sobre polinômios escalares estabelece que um polinômio divide outro exatamente, se todas as raízes do divisor forem raízes do dividendo. O Corolário 4.5 é uma generalização deste fato para polinômios matriciais, nele, a notação $V(X_1, \dots, X_k)^T$ é utilizada para referir-se à matriz transposta por blocos da matriz de Vandermonde de X_1, \dots, X_k . Considerando a mesma notação do Teorema 4.1, tem-se:

Corolário 4.5. *Sejam R_1, \dots, R_p solventes dos polinômios $\hat{W}(X)$ e $\hat{P}(X)$. Se a matriz $V(R_1, \dots, R_p)^T$ for não singular, então $\hat{L}(X)$ é o polinômio nulo.*

PROVA. Se as matrizes R_1, \dots, R_p forem solventes de $\hat{P}(X)$, o Corolário 4.1 mostra que para $i = 1, \dots, p$, $\hat{L}(R_i) = 0$. Sendo que $V(R_1, \dots, R_p)^T$ é inversível, a única solução do sistema:

$$V(R_1, \dots, R_p)^T \begin{pmatrix} L_{p-1} \\ L_{p-2} \\ \vdots \\ L_0 \end{pmatrix} = \begin{pmatrix} \hat{L}(R_1) \\ \hat{L}(R_2) \\ \vdots \\ \hat{L}(R_p) \end{pmatrix} = 0$$

é a trivial. Assim sendo, $\hat{L}(X)$ é o polinômio nulo. \square

O Exemplo 4.2 mostra que o polinômio $L(X)$ é o polinômio nulo em condições menos restritivas que as enunciadas pelo Corolário 4.5.

Exemplo 4.2. Sejam $P(X)$ e $W(X)$ os polinômios matriciais definidos por:

$$P(X) = X^3 + \begin{pmatrix} -7.5 & -2.5 \\ -0.5 & -9.5 \end{pmatrix} X^2 + \begin{pmatrix} 17 & 15 \\ 20 & 30 \end{pmatrix} X + \begin{pmatrix} -15 & -17 \\ -6 & -26 \end{pmatrix}$$

e

$$W(X) = X^2 + \begin{pmatrix} -6 & 0 \\ 0 & -6 \end{pmatrix} X + \begin{pmatrix} 8 & 0 \\ -1 & 9 \end{pmatrix}.$$

Observe-se que as matrizes $R_1 = \begin{pmatrix} 2 & 0 \\ -1 & 3 \end{pmatrix}$ e $R_2 = \begin{pmatrix} 4 & 0 \\ 1 & 3 \end{pmatrix}$ são solventes de $\hat{P}(X)$ e $\hat{W}(X)$; como R_1 e R_2 possuem um autovalor em comum, $V(R_1, R_2)$ é uma

matriz singular, porém, pode-se mostrar que:

$$P(X) = F(X)X^2 + \begin{pmatrix} -6 & 0 \\ 0 & -6 \end{pmatrix} F(X)X + \begin{pmatrix} 8 & 0 \\ -1 & 9 \end{pmatrix} F(X),$$

onde o polinômio matricial $F(X)$ é definido por:

$$F(X) = X + \begin{pmatrix} 4.5 & -0.5 \\ 2.5 & 1.5 \end{pmatrix}.$$

5. Existência de solventes

Em geral, não existem condições necessárias e suficientes que determinem a existência de solventes, exceto para o problema mais simples do cálculo da raiz quadrada de uma matriz [8]. Nesta seção, apresentam-se algumas condições suficientes tanto para a existência de um único solvente quanto para a existência de conjuntos de solventes [11, 21, 26]. Em geral, as condições de existência estão relacionadas à verificação de certas propriedades da matriz companheira por blocos do polinômio $P(X)$.

O primeiro resultado relaciona a existência de solventes de $P(X)$ às propriedades da lambda matriz $P(\lambda)$.

Lema 5.1 ([30]). *Se $P(\lambda)$ for uma lambda matriz com n vetores latentes à direita ξ_1, \dots, ξ_n linearmente independentes, associados às raízes latentes $\lambda_1, \dots, \lambda_n$, e se as matrizes Q e Λ forem definidas por $Q = [\xi_1, \dots, \xi_n]$ e $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$, então a matriz $Q\Lambda Q^{-1}$ é um solvente de $P(X)$.*

PROVA. Pela própria definição de $P(X)$,

$$P(Q\Lambda Q^{-1}) = (A_0 Q\Lambda^m + A_1 Q\Lambda^{m-1} + \dots + A_m Q) Q^{-1}$$

e, de acordo com as definições de raízes e vetores latentes,

$$A_0 Q\Lambda^m + A_1 Q\Lambda^{m-1} + \dots + A_m Q = 0. \quad \square$$

No seguinte teorema, a notação $\sigma(A)$ é empregada para referir-se ao conjunto dos autovalores da matriz A .

Teorema 5.1 ([11]). *Seja $P(X)$ um polinômio matricial mônico tal que as raízes latentes de $P(\lambda)$ são distintas, então $P(X)$ tem um conjunto de solventes S_1, \dots, S_m satisfazendo*

$$\bigcup_{i=1}^m \sigma(S_i) = \sigma(C),$$

onde C é a matriz companheira por blocos de $P(X)$.

Seja ξ_i for um vetor latente à direita de $P(\lambda)$ associado à raiz latente λ_i , em virtude do Teorema 2.2, os vetores

$$\eta_i = \begin{pmatrix} \xi_i \\ \lambda_i \xi_i \\ \vdots \\ \lambda_i^{m-1} \xi_i \end{pmatrix} \quad (i = 1, \dots, mn)$$

são os autovetores de C . Uma vez que as raízes latentes são distintas, os vetores η_1, \dots, η_{mn} são linearmente independentes e a matriz $Q = [\eta_1, \dots, \eta_{mn}]$ é inversível. Neste caso, segue-se, do Lema A-2.2, que existe uma permutação das colunas de Q , $\tilde{Q} = (\tilde{Q}_{ij})_1^m$, tal que $\tilde{Q}_{ii} \in \mathbb{C}^{n \times n}$ é inversível para $i = 1, \dots, m$. Assim sendo, os nm vetores ξ_i podem ser agrupados em m conjuntos disjuntos, formados por n vetores linearmente independentes. Desta maneira, o resultado é consequência do Lema 5.1. \square

Note-se que os solventes fornecidos pelo Teorema 5.1 formam um conjunto completo, pois a matriz de Vandermonde dos solventes coincide com o produto

$$\tilde{Q} \text{diag} (\tilde{Q}_{11}^{-1}, \tilde{Q}_{12}^{-1}, \dots, \tilde{Q}_{1m}^{-1}).$$

Observação 5.1. Se o polinômio matricial $P(X)$ admitir um solvente diagonalizável S tal que $S = Q\Lambda Q^{-1}$, onde $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$, então as colunas de Q são vetores latentes de $P(\lambda)$.

Corolário 5.1. Se $P(\lambda)$ tiver mn raízes latentes, não necessariamente distintas, e o conjunto dos vetores latentes à direita satisfizer a condição de Haar (cada conjunto de n vetores é linearmente independente), então existem exatamente $\binom{mn}{n}$ solventes distintos de $P(X)$.

PROVA. Sejam ξ_1, \dots, ξ_{mn} os vetores latentes à direita de $P(\lambda)$ que satisfazem a condição de Haar, e seja $\xi_{i_1}, \dots, \xi_{i_n}$ um subconjunto de n vetores latentes associados às raízes latentes $\lambda_{i_1}, \dots, \lambda_{i_n}$. Segundo o Lema 5.1, se as matrizes Q e Λ forem definidas por:

$$Q = [\xi_{i_1}, \dots, \xi_{i_n}] \quad \text{e} \quad \Lambda = \text{diag}(\lambda_{i_1}, \dots, \lambda_{i_n}),$$

então $Q\Lambda Q^{-1}$ é um solvente de $P(X)$. Levando-se em conta que existem exatamente $\binom{mn}{n}$ conjuntos distintos de raízes latentes, o corolário fica demonstrado. \square

Corolário 5.2. Se $P(\lambda)$ tiver raízes latentes distintas, então $P(\lambda)$ pode ser fatorado como um produto de fatores lineares.

PROVA. A demonstração é por indução sobre o grau de $P(X)$. Se $m = 1$, o resultado é óbvio. Se o corolário for verdadeiro para polinômios de grau $m - 1$, levando-se em conta que $P(\lambda)$ tem raízes latentes distintas, o Lema 5.1 mostra que existe um solvente S de $P(X)$, o qual, pelo Teorema de Bézout, satisfaz:

$$P(\lambda) = Q(\lambda)(\lambda I - S). \quad (5.1)$$

Como $Q(X)$ tem grau $m - 1$, segue-se, da hipótese de indução, que $Q(\lambda)$ pode ser fatorada como produto de fatores lineares. Assim sendo, em virtude de (5.1), o resultado é verdadeiro para todo m . \square

Definição 5.1. Um conjunto de matrizes R_1, \dots, R_m é chamado **cadeia de solventes** de $P(X)$ se

$$P(\lambda) = A_0(\lambda I - R_1)(\lambda I - R_2) \cdots (\lambda I - R_m).$$

Observação 5.2. É importante ressaltar que, se R_1, \dots, R_m for uma cadeia de solventes, em geral só R_m é solvente de $P(X)$ (ver Exemplo 3.2).

A cadeia de solventes permite generalizar o resultado clássico dos polinômios escalares, que relaciona os coeficientes do polinômio às funções simétricas de suas raízes. O Teorema 5.2 fornece esta generalização, ainda que, no caso matricial, as funções não sejam, em geral, simétricas.

Teorema 5.2 ([11]). *Se R_1, \dots, R_m for uma cadeia de solventes de*

$$P(X) = X^m + A_1 X^{m-1} + \cdots + A_m,$$

então

$$A_1 = -(R_1 + R_2 + \cdots + R_m),$$

$$A_2 = \sum_{1 \leq i < j \leq m} R_i R_j,$$

$$A_3 = - \sum_{1 \leq i < j < k \leq m} R_i R_j R_k,$$

\vdots

$$A_m = (-1)^m R_1 R_2 \cdots R_m.$$

PROVA. A demonstração consiste em igualar os coeficientes de $P(\lambda)$ e os de

$$(\lambda I - R_1)(\lambda I - R_2) \cdots (\lambda I - R_m). \quad \square$$

5.1. Cadeia de solventes e bi-diagonalização da matriz companheira

Nesta seção, estuda-se a relação entre a existência de uma cadeia de solventes do polinômio matricial $P(X)$ definido por:

$$P(X) = X^m + A_1 X^{m-1} + \cdots + A_m,$$

e a possibilidade de bi-diagonalizar a matriz companheira associada a $P(X)$ [21], isto é, supõe-se a existência de uma matriz não singular $P \in \mathbb{C}^{mn \times mn}$ tal que

$$\begin{pmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ -A_m & -A_{m-1} & -A_{m-2} & \cdots & -A_1 \end{pmatrix} = P^{-1} \begin{pmatrix} R_m & I & & & \\ & \ddots & \ddots & & \\ & & R_2 & I & \\ & & & & R_1 \end{pmatrix} P.$$

Para simplificar o enunciado e a prova dos resultados desta seção, chama-se $Q_m(X)$ ao polinômio $P(X)$.⁸

Lema 5.2 ([21]). *Seja R_m um solvente de $Q_m(X)$ e sejam C_m e $J(R_m)$ as matrizes definidas por:*

$$C_m = \begin{pmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ -A_m & -A_{m-1} & -A_{m-2} & \cdots & -A_1 \end{pmatrix} \quad (5.2)$$

c

$$J(R_m) = \begin{pmatrix} I & 0 & 0 & \cdots & 0 \\ R_m & I & 0 & \cdots & 0 \\ R_m^2 & R_m & I & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ R_m^{m-1} & R_m^{m-2} & R_m^{m-3} & \cdots & I \end{pmatrix}, \quad (5.3)$$

⁸Nesta seção emprega-se a indução no grau dos polinômios matriciais.

Sendo R_1, \dots, R_{m-1} uma cadeia de solventes de $Q_{m-1}(X)$, segue-se, da hipótese de indução, que

$$C_{m-1} \sim F_{m-1} = \begin{pmatrix} R_{m-1} & I & & & \\ & \ddots & \ddots & & \\ & & R_2 & I & \\ & & & & R_1 \end{pmatrix}. \quad (5.6)$$

Levando-se em conta (5.5) e (5.6), a afirmação é verdadeira para m e, portanto, o teorema fica demonstrado. \square

A partir do Lema 5.2 e do Teorema 5.3, obtém-se que a matriz P_m que transforma C_m em F_m é dada por:

$$P_m = J_m J_{m-1} \cdots J_2, \quad (5.7)$$

onde

$$\begin{aligned} J_m &= J(R_m) \\ J_{m-1} &= \text{diag}(I, J(R_{m-1})) \\ J_{m-2} &= \text{diag}(I, I, J(R_{m-2})) \\ &\vdots \\ J_2 &= \text{diag}(I, \dots, I, J(R_2)). \end{aligned}$$

Como consequência do Teorema 5.3 surge a seguinte pergunta: a similaridade entre C_m e uma matriz F_m como em (5.4), será uma condição suficiente para que R_m seja solvente de $Q_m(X)$? A resposta é fornecida pelo Teorema 5.4 com o auxílio do seguinte lema:

Lema 5.3 ([21]). *Sejam C_m e F_m as matrizes definidas por (5.2) e (5.4) respectivamente, e seja $P_m = (P_{ij}^{(m)})_1^m \in \mathbb{C}^{mn \times mn}$ uma matriz inversível, cujos blocos $P_{ij}^{(m)}$ são matrizes de dimensão n e o bloco $P_{11}^{(m)}$ é não singular. Se*

$$C_m P_m = P_m F_m,$$

então

$$(1) \ S_m \text{ definido por } S_m = P_{11}^{(m)} R_m (P_{11}^{(m)})^{-1} \text{ é solvente de } P(X).$$

(2) A matriz $J(S_m)$ definida por (5.3) satisfaz:

$$J(S_m)^{-1} P_m = \left(\begin{array}{c|cccc} P_{11}^{(m)} & P_{12}^{(m)} & P_{13}^{(m)} & \dots & P_{1m}^{(m)} \\ 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{array} \begin{array}{c} \\ \\ \\ \\ P_{m-1} \end{array} \right).$$

PROVA.

(1). Sendo $C_m P_m = P_m F_m$, os blocos da primeira coluna de P_m satisfazem

$$\begin{aligned} P_{21}^{(m)} &= P_{11}^{(m)} R_m, \\ P_{31}^{(m)} &= P_{21}^{(m)} R_m = P_{11}^{(m)} R_m^2, \\ &\vdots \\ P_{m1}^{(m)} &= P_{m-1\ 1}^{(m)} R_m = \dots = P_{11}^{(m)} R_m^{m-1}, \\ P_{m1}^{(m)} R_m &= -A_m P_{11}^{(m)} - A_{m-1} P_{21}^{(m)} - \dots - A_1 P_{m1}^{(m)}, \end{aligned} \quad (5.8)$$

logo

$$P_{11}^{(m)} R_m^m + A_1 P_{11}^{(m)} R_m^{m-1} + \dots + A_m P_{11}^{(m)} = 0. \quad (5.9)$$

Multiplicando (5.9) à direita por $(P_{11}^{(m)})^{-1}$, obtém-se que S_m é um solvente de $Q_m(X)$. \square

(2). Da expressão de $J(S_m)^{-1}$ tem-se:

$$J(S_m)^{-1} P_m = \left(\begin{array}{c|cccc} P_{11}^{(m)} & P_{12}^{(m)} & P_{13}^{(m)} & \dots & P_{1m}^{(m)} \\ -S_m P_{11}^{(m)} + P_{21}^{(m)} & & & & \\ -S_m P_{21}^{(m)} + P_{31}^{(m)} & & & & \\ \vdots & & & & \\ -S_m P_{m-1\ 1}^{(m)} + P_{m1}^{(m)} & & & & \end{array} \begin{array}{c} \\ \\ \\ \\ P_{m-1} \end{array} \right).$$

Assim sendo, de (5.8), segue-se que:

$$\begin{aligned} -S_m P_{11}^{(m)} + P_{21}^{(m)} &= -P_{11}^{(m)} R_m + P_{11}^{(m)} R_m = 0, \\ -S_m P_{21}^{(m)} + P_{31}^{(m)} &= -P_{11}^{(m)} R_m^2 + P_{11}^{(m)} R_m^2 = 0, \\ &\vdots \\ -S_m P_{m-1\ 1}^{(m)} + P_{m1}^{(m)} &= -P_{11}^{(m)} R_m^{m-1} + P_{11}^{(m)} R_m^{m-1} = 0, \end{aligned}$$

a afirmação fica demonstrada. \square

O Teorema 5.4 estabelece condições nas quais a recíproca do Teorema 5.3 é verdadeira.

Teorema 5.4 ([21]). *Sejam C_m e F_m as matrizes definidas por (5.2) e (5.4) respectivamente, e seja $P_m = \left(P_{ij}^{(m)}\right)_1^m \in \mathbb{C}^{mn \times mn}$ uma matriz inversível, cujos blocos são $P_{ij}^{(m)} \in \mathbb{C}^{n \times n}$. Se*

- (1) $C_m P_m = P_m F_m$;
- (2) $P_{11}^{(m)}$ é não singular;
- (3) Para cada $k = m - 1, \dots, 1$, a matriz $P_k = \left(P_{ij}^{(k)}\right)_1^k \in \mathbb{C}^{kn \times kn}$ for definida por:

$$J(S_{k+1})^{-1} P_{k+1} = \left(\begin{array}{c|cccc} P_{11}^{(k+1)} & P_{12}^{(k+1)} & P_{13}^{(k+1)} & \dots & P_{1m}^{(k+1)} \\ \hline 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & P_k & \end{array} \right), \quad (5.10)$$

onde

$$S_{k+1} = P_{11}^{(k+1)} R_{k+1} \left(P_{11}^{(k+1)}\right)^{-1} \quad (5.11)$$

e $P_{11}^{(k)}$ é não singular,

então existe uma cadeia de solventes S_1, \dots, S_m de $Q_m(X)$, onde S_1 é definido por (5.11) para $k = 0$.

PROVA. A demonstração segue-se por indução sobre o grau do polinômio. Se $m = 2$, o Lema 5.3 mostra que $S_2 = P_{11}^{(2)} R_2 \left(P_{11}^{(2)}\right)^{-1}$ é uma solução da equação $X^2 + A_1 X + A_2 = 0$, e como consequência do Lema 5.2, tem-se:

$$J(S_2)^{-1} C_2 J(S_2) = \begin{pmatrix} S_2 & I \\ 0 & C_1 \end{pmatrix} \quad (5.12)$$

e

$$Q_2(\lambda) = (\lambda I - C_1)(\lambda I - S_2).$$

Em virtude da hipótese (1), verifica-se que:

$$P_2^{-1} C_2 P_2 = \begin{pmatrix} R_2 & I \\ 0 & R_1 \end{pmatrix}. \quad (5.13)$$

Em conseqüência do Lema 5.3, tem-se

$$J(S_m)^{-1} P_m = \left(\begin{array}{c|cccc} P_{11}^{(m)} & P_{12}^{(m)} & P_{13}^{(m)} & \cdots & P_{1m}^{(m)} \\ \hline 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{array} \right) P_{m-1}. \quad (5.16)$$

Como $J(S_m)^{-1} P_m$ e $P_{11}^{(m)}$ são inversíveis, segue-se que P_{m-1} é não singular. Além disso, aplicando as hipóteses do teorema e a equação (5.15) tem-se:

- (a) $C_{m-1} P_{m-1} = P_{m-1} F_{m-1}$;
- (b) $P_{11}^{(m-1)}$ é não singular;
- (c) Para cada $k = m - 2, \dots, 1$, a matriz P_k satisfaz que $P_{11}^{(k)}$ é não singular.

Assim, usando a hipótese de indução, existe uma cadeia de solventes R_1, \dots, R_{m-1} de $Q_{m-1}(X)$. Combinando a cadeia de solventes de $Q_{m-1}(X)$ com a equação (5.14), o teorema fica demonstrado para todo m . \square

5.2. Co-solventes e diagonalização da matriz companheira

Esta seção é dedicada ao estudo da existência de co-solventes do polinômio matricial $P(X)$ definido por:

$$P(X) = X^m + A_1 X^{m-1} + \cdots + A_m.$$

Em especial, mostra-se de que maneira a existência de um conjunto completo de co-solventes afeta as propriedades da matriz companheira por blocos de $P(X)$ [26]. No final da seção, obtém-se uma condição suficiente para a existência de conjuntos completos de solventes.

Lembrando a Definição 2.5, o par de matrizes (Q, R) é um co-solvente de $P(X)$ se Q não for a matriz nula e

$$QR^m + A_1 QR^{m-1} + \cdots + A_m Q = 0.$$

O Teorema 5.5 fornece uma condição suficiente para a existência de solventes e co-solventes do polinômio matricial $P(X)$.

Teorema 5.5 ([26]). *Seja C a matriz companheira por blocos de $P(X)$ e seja D a matriz diagonal por blocos definida por $D = \text{diag}(R_1, \dots, R_k)$, onde $R_i \in \mathbb{C}^{n_i \times n_i}$*

para $i = 1, \dots, k$. Se existir uma matriz inversível P , com blocos $P_{ij} \in \mathbb{C}^{n \times n_i}$, satisfazendo:

$$CP = PD, \quad (5.17)$$

e se para algum $1 \leq l \leq k$, acontecer que $n_l = n$, então (P_{1l}, R_l) é um co-solvente de $P(X)$. Além disso, se P_{1l} for não singular, então $S = P_{2l}P_{1l}^{-1}$ é um solvente de $P(X)$.

PROVA. Considerando que $CP = PD$, ao equacionar a l -ésima coluna, obtém-se que a matriz R_l satisfaz:

$$\begin{aligned} P_{ml}R_l &= - \sum_{j=1}^m A_{m-j+1}P_{jl} \\ P_{il}R_l &= P_{i+1l} \end{aligned} \quad (i = 1, \dots, m-1),$$

e portanto,

$$\begin{aligned} P_{il} &= P_{1l}R_l^{i-1} \quad (i = 1, \dots, m), \\ P_{1l}R_l^m &= - \sum_{j=1}^m A_{m-j+1}P_{1l}R_l^{j-1}. \end{aligned} \quad (5.18)$$

Uma vez que P é inversível, a matriz P_{1l} é não nula. Assim sendo, a equação (5.18) mostra que (P_{1l}, R_l) é um co-solvente de $P(X)$.

Agora, se P_{1l} for inversível, $(I, P_{1l}R_lP_{1l}^{-1}) = (I, P_{2l}P_{1l}^{-1})$ também é um co-solvente de $P(X)$ e, portanto, $S = P_{2l}P_{1l}^{-1}$ é um solvente de $P(X)$. \square

Definição 5.2. O conjunto $\{(Q_i, R_i) \ ; \ i = 1, \dots, m\}$ de co-solventes de $P(X)$ é chamado de conjunto completo de co-solventes, se a matriz

$$W = \begin{pmatrix} Q_1 & Q_2 & \cdots & Q_m \\ Q_1R_1 & Q_2R_2 & \cdots & Q_mR_m \\ \vdots & \vdots & \ddots & \vdots \\ Q_1R_1^{m-1} & Q_2R_2^{m-1} & \cdots & Q_mR_m^{m-1} \end{pmatrix}$$

for inversível.

Observe-se que, se todas as matrizes Q_i forem inversíveis e se as matrizes S_i forem definidas por:

$$S_i = Q_iR_iQ_i^{-1} \quad (i = 1, \dots, m),$$

então o conjunto S_1, \dots, S_m é um conjunto completo de solventes de $P(X)$.

O teorema a seguir, caracteriza os polinômios que admitem um conjunto completo de co-solventes.

Teorema 5.6 ([26]). *O polinômio $P(X)$ admite um conjunto completo de co-solventes se, e somente se, a matriz companheira por blocos C de $P(X)$ for similar a uma matriz diagonal por blocos, isto é, se, e somente se, existirem:*

- (1) *uma matriz diagonal por blocos $D = \text{diag}(R_1, \dots, R_m)$ onde $R_i \in \mathbb{C}^{n \times n}$,*
- (2) *uma matriz inversível $P = (P_{ij})_1^n$, com blocos $P_{ij} \in \mathbb{C}^{n \times n}$*

satisfazendo:

$$CP = PD. \quad (5.19)$$

Nestas condições, $\{(P_{1j}, R_j) \ ; \ j = 1, \dots, m\}$ é um conjunto completo de co-solventes de $P(X)$.

PROVA. Levando-se em conta que as condições do Teorema 5.5 estão satisfeitas, então, para cada $j = 1, \dots, m$, (P_{1j}, R_j) é um co-solvente de $P(X)$. Para mostrar que o conjunto de co-solventes é completo, observe-se que as igualdades em (5.18) valem para todo l e, portanto, as matrizes P e W coincidem.

Reciprocamente, se $\{(P_{1j}, R_j) \ ; \ j = 1, \dots, m\}$ for um conjunto completo de co-solventes de $P(X)$, a matriz W associada ao conjunto é inversível e, além disso, verifica-se facilmente que:

$$CW = W \text{diag}(R_1, \dots, R_m). \quad \square$$

Corolário 5.3. *Seja $P(X)$ um polinômio cuja matriz companheira por blocos C satisfaz as condições do Teorema 5.6. Se, para $j = 1, \dots, m$, P_{1j} for inversível então $P(X)$ admite um conjunto completo de solventes definido por:*

$$S_j = P_{2j}P_{1j}^{-1} \quad (j = 1, \dots, m).$$

PROVA. Supondo que P_{1j} é inversível, de acordo com o Teorema 5.5,

$$S_j = P_{2j}P_{1j}^{-1}$$

é solvente de $P(X)$. Levando-se em conta que

$$P = V(S_1, \dots, S_m) \text{diag}(P_{11}, \dots, P_{1m}),$$

onde P e $\text{diag}(P_{11}, \dots, P_{1m})$ são inversíveis, segue-se que a matriz de Vandermonde associada a S_1, \dots, S_m é não singular. Logo, S_1, \dots, S_m é um conjunto completo de solventes. \square

Corolário 5.4. *Se a matriz companheira por blocos C associada a $P(X)$ for similar a uma matriz diagonal, então $P(X)$ admite um conjunto completo de solventes.*

PROVA. Seja $P \in \mathbb{C}^{nm \times nm}$ uma matriz inversível tal que

$$CP = P \operatorname{diag}(\lambda_1, \dots, \lambda_{mn}).$$

Pode-se supor, sem perda da generalidade (Lema A-2.2), que os blocos diagonais de P são inversíveis. Como a prova do Teorema 5.5 mostra que:

$$P_{ij} = P_{1j} R_j^{i-1},$$

onde R_j é uma matriz diagonal, e uma vez que, para $j = 1, \dots, m$, P_{jj} é inversível, as matrizes P_{1j} são não singulares, assim sendo, o resultado é uma consequência do Teorema 5.6. \square

Observe-se que o Corolário 5.4 estende o resultado do Teorema 5.1 pois, se todas as raízes latentes de $P(\lambda)$ forem distintas, então C é diagonalizável.

6. Interpolação

Esta seção é dedicada ao estudo da existência de polinômios matriciais interpolantes, e com este objetivo, generaliza-se o conceito de polinômios fundamentais. No final da seção, os polinômios matriciais fundamentais são utilizados para melhorar alguns resultados obtidos na Seção 4. Os resultados contidos nesta seção são essenciais para a demonstração da convergência do Algoritmo III-5.1.

Dados $x_1, \dots, x_m \in \mathbb{C}$ distintos, os polinômios $m_i(x)$ são os **polinômios fundamentais** da teoria de interpolação se satisfizerem $m_i(x_j) = \delta_{ij}$. Estes polinômios são muito empregados, pois formam uma base do subespaço dos polinômios de grau $m - 1$.

Definição 6.1. Os polinômios matriciais $M_1(X), \dots, M_m(X)$, de grau $m - 1$, são chamados de **polinômios matriciais fundamentais** associados às matrizes $X_1, \dots, X_m \in \mathbb{C}^{n \times n}$, se

$$M_i(X_j) = \delta_{ij} I \quad (i, j = 1, \dots, m).$$

No espaço dos polinômios matriciais, os polinômios fundamentais não existem para qualquer conjunto de matrizes X_1, \dots, X_m ; o Teorema 6.1 fornece uma condição sobre as matrizes X_1, \dots, X_m que garante a existência e unicidade dos polinômios matriciais fundamentais, enquanto que o Teorema 6.2 estabelece uma propriedade adicional para os polinômios $M_i(X)$.

Teorema 6.1 ([11]). *Dados $m \geq 2$, $m \in \mathbb{N}$, e um conjunto de pares de matrizes quadradas $\{X_i, Y_i ; i = 1, \dots, m\}$ de dimensão n , existe um único polinômio matricial de grau $m - 1$*

$$P_1(X) = \sum_{l=0}^{m-1} A_{m-1-l} X^l$$

e um único polinômio matricial mônico de grau m

$$P_2(X) = \sum_{l=0}^m B_{m-l} X^l$$

tais que

$$\begin{aligned} P_1(X_i) &= Y_i & (i = 1, \dots, m), \\ P_2(X_i) &= Y_i & (i = 1, \dots, m), \end{aligned}$$

se, e somente se, $V(X_1, \dots, X_m)$ for não singular.

PROVA. A demonstração segue-se da definição da matriz de Vandermonde por blocos, pois as equações

$$P_1(X_i) = Y_i \quad (i = 1, \dots, m),$$

podem ser escritas como

$$(A_{m-1}, \dots, A_0)V(X_1, \dots, X_m) = (Y_1, \dots, Y_m). \quad (6.1)$$

Levando-se em conta que $V(X_1, \dots, X_m)$ é inversível, os coeficientes A_0, \dots, A_{m-1} podem ser determinados a partir do sistema linear (6.1).

Um raciocínio análogo mostra a existência do polinômio matricial $P_2(X)$. \square

Observe-se que a condição do teorema acima não impõe restrições às matrizes Y_1, \dots, Y_m .

Se S_1, \dots, S_m for um conjunto completo de solventes de $P(X)$, então o Teorema 6.1 mostra que, para cada $i = 1, \dots, m$, existe um único polinômio matricial $M_i(X) = \sum_{l=0}^{m-1} A_{m-1-l}^{(i)} X^l$ tal que, para $j = 1, \dots, m$, $M_i(S_j) = \delta_{ij}I$. Contrariamente ao expresso em [11], podem existir outros solventes de $P(X)$ que não sejam ao mesmo tempo solventes de $M_i(X)$, como mostra o seguinte exemplo.

Exemplo 6.1. O polinômio matricial

$$X^2 - \begin{pmatrix} 1 & 0 \\ 2 & -1 \end{pmatrix}$$

tem como conjunto completo de solventes as matrizes

$$S_1 = \begin{pmatrix} 1 & 0 \\ 1 - i & i \end{pmatrix} \quad \text{e} \quad S_2 = \begin{pmatrix} -1 & 0 \\ -1 + i & -i \end{pmatrix},$$

cujos polinômios matriciais fundamentais associados são:

$$M_1(X) = \begin{pmatrix} -0.25 + 0.25i & 0.75 - 0.25i \\ -0.25 + 0.75i & -0.75 - 0.75i \end{pmatrix} X + \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}$$

e

$$M_2(X) = \begin{pmatrix} 0.25 - 0.25i & -0.75 + 0.25i \\ 0.25 - 0.75i & -0.75 + 0.75i \end{pmatrix} X + \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}.$$

Entretanto, $S_3 = \begin{pmatrix} 1 & 0 \\ 1 + i & -i \end{pmatrix}$ é um solvente de $P(X)$ mas não é de $M_1(X)$ nem de $M_2(X)$.

De fato, entre as matrizes S_1, \dots, S_m a única que não é solvente de $P(X)$ e de $M_i(X)$ é S_i , mas isto resulta óbvio a partir da definição dos polinômios matriciais fundamentais.

Teorema 6.2 ([11]). *Se $X_1, \dots, X_m \in \mathbb{C}^{n \times n}$ for um conjunto completo, então existem os polinômios matriciais fundamentais*

$$M_i(X) = \sum_{l=0}^{m-1} A_{m-1-l}^{(i)} X^l \quad (i = 1, \dots, m)$$

associados às matrizes X_1, \dots, X_m . Além disso, se $V(X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_m)$ for inversível, então $A_0^{(i)}$ é não singular.

PROVA. Sendo $V(X_1, \dots, X_m)$ inversível, o Teorema 6.1 mostra que existem os polinômios matriciais fundamentais $M_1(X), \dots, M_m(X)$ associados às matrizes X_1, \dots, X_m . Fixando $1 \leq i \leq m$ e aplicando novamente o Teorema 6.1, desta vez às matrizes $X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_m$, existe um único polinômio matricial mônico $N_i(X)$ de grau $m - 1$,

$$N_i(X) = X^{m-1} + \sum_{l=1}^{m-2} N_{m-2-l}^{(i)} X^l,$$

tal que $N_i(S_j) = 0$ para $j \neq i$. Seja $Q_i(X)$ o polinômio matricial definido por:

$$Q_i(X) = N_i(S_i)M_i(X).$$

Levando-se em conta que $Q_i(X)$ e $N_i(X)$ têm grau $m - 1$ e que

$$Q_i(S_j) = N_i(S_j) \quad (j = 1, \dots, m),$$

a unicidade do polinômio interpolante estabelecida pelo Teorema 6.1 mostra que os polinômios $Q_i(X)$ e $N_i(X)$ são idênticos. Assim sendo,

$$N_i(X) = N_i(S_i) M_i(X); \quad (6.2)$$

igualando os coeficientes em (6.2) tem-se que:

$$I = N_i(S_i) A_0^{(i)},$$

e portanto, $A_0^{(i)}$ é não singular. □

O Teorema 6.3 mostra que os polinômios matriciais fundamentais, como os polinômios fundamentais no caso escalar, geram o subespaço dos polinômios matriciais de grau $m - 1$.

Teorema 6.3 ([11]). *Seja $X_1, \dots, X_m \in \mathbb{C}^{n \times n}$ um conjunto completo de matrizes e sejam $M_1(X), \dots, M_m(X)$ os polinômios matriciais fundamentais associados às matrizes X_1, \dots, X_m , então para qualquer polinômio matricial*

$$Q(X) = \sum_{l=0}^{m-1} B_{m-1-l} X^l$$

tem-se:

$$Q(X) = \sum_{i=1}^m Q(X_i) M_i(X).$$

PROVA. Seja $N(X)$ o polinômio matricial definido por:

$$N(X) = \sum_{i=0}^m Q(X_i) M_i(X).$$

Os polinômios matriciais $Q(X)$ e $N(X)$ têm grau $m - 1$ e, além disso,

$$N(X_i) = Q(X_i) \quad (i = 1, \dots, m).$$

Uma vez que a matriz $V(X_1, \dots, X_m)$ é não singular, o Teorema 6.1 mostra que o polinômio interpolante é único, desta maneira o teorema fica demonstrado.

Levando-se em conta que uma matriz λI é um polinômio matricial onde a variável é da forma λI , do Teorema 6.3 obtém-se imediatamente o seguinte corolário.

Corolário 6.1. *Seja $X_1, \dots, X_m \in \mathbb{C}^{n \times n}$ um conjunto completo de matrizes e sejam $M_1(X), \dots, M_m(X)$ os polinômios matriciais fundamentais associados às matrizes X_1, \dots, X_m , então para qualquer lambda matriz*

$$Q(\lambda) = \sum_{l=0}^{m-1} B_{m-1-l} \lambda^l,$$

tem-se que:

$$Q(\lambda) = \sum_{i=0}^m Q(X_i) M_i(\lambda).$$

Usando os polinômios matriciais fundamentais pode-se obter uma nova caracterização dos solventes de $\hat{P}(X)$ em função dos solventes de $P(X)$. Considere o polinômio matricial mônico

$$P(X) = X^m + A_1 X^{m-1} + \dots + A_m.$$

Teorema 6.4 ([11]). *Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ e sejam $M_1(X), \dots, M_m(X)$ os polinômios matriciais fundamentais associados às matrizes S_1, \dots, S_m . Se $V(S_1, \dots, S_{i-1}, S_{i+1}, \dots, S_m)$ for não singular, então*

$$M_i(X)X - S_i M_i(X) = A_0^{(i)} P(X). \quad (6.3)$$

PROVA. Seja $Q_i(X)$ o polinômio matricial definido por:

$$Q_i(X) = (A_0^{(i)})^{-1} (M_i(X)X - S_i M_i(X)).$$

Levando-se em conta que $V(S_1, \dots, S_{i-1}, S_{i+1}, \dots, S_m)$ é inversível, o polinômio $Q_i(X)$ está bem definido. Como

$$Q_i(S_j) = 0 \quad (j = 1, \dots, m),$$

em virtude da unicidade do polinômio interpolante, $Q_i(X)$ e $P(X)$ coincidem e, portanto, o resultado é verdadeiro. \square

Um resultado anterior (equação (4.4)) mostra que, se R_i for um solvente de $\hat{P}(X)$, então existe um único polinômio matricial $F_i(X)$, mônico e de grau $m - 1$, tal que

$$P(X) = F_i(X)X - R_i F_i(X). \quad (6.4)$$

Além disso, provou-se que, se S_i for um solvente de $P(X)$ tal que $F_i(S_i)$ é inversível, então

$$R_i = F(S_i)S_i F(S_i)^{-1}.$$

Comparando (6.3) e (6.4) tem-se o seguinte corolário:

Corolário 6.2. *Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ e sejam $M_1(X), \dots, M_m(X)$ os polinômios matriciais fundamentais associados às matrizes S_1, \dots, S_m . Se $V(S_1, \dots, S_{i-1}, S_{i+1}, \dots, S_m)$ for não singular, então*

$$R_i = \left(A_0^{(i)}\right)^{-1} S_i A_0^{(i)}$$

é um solvente de $\hat{P}(X)$.

PROVA. O Teorema 6.4 mostra que:

$$M_i(X)X - S_i M_i(X) = A_0^{(i)} P(X);$$

em virtude da unicidade do polinômio cociente, tem-se que:

$$F_i(X) = \left(A_0^{(i)}\right)^{-1} M_i(X),$$

e portanto, o corolário é verdadeiro. □

7. Avaliação de polinômios matriciais

A avaliação de polinômios matriciais não é um assunto muito estudado. Os algoritmos utilizados são extensões daqueles empregados no caso escalar. Nesta seção, consideram-se dois destes métodos, para os quais são fornecidas as suas correspondentes análises de erros.

O primeiro método para avaliar o polinômio matricial

$$P(X) = \sum_{l=0}^m A_{m-l} X^l$$

é definido por:

Algoritmo 7.1.

$$\begin{aligned} B_0 &= A_m \\ B_k &= A_{m-k} X^k + B_{k-1} \end{aligned} \quad (k = 1, \dots, m).$$

O segundo método, chamado de **algoritmo de Horner**, é definido por:

Algoritmo 7.2.

$$\begin{aligned} \tilde{B}_0 &= A_0 \\ \tilde{B}_k &= \tilde{B}_{k-1} X + A_k \end{aligned} \quad (k = 1, \dots, m).$$

A comparação entre os dois métodos pode ser realizada levando-se em conta dois aspectos principais: primeiro, considerando o número de operações (flops) necessárias e segundo, analisando a estabilidade. No que diz respeito ao número de flops, o método de Horner é o mais econômico pois requer m multiplicações matriciais, enquanto que, o Algoritmo 7.1 precisa realizar no mínimo de $2m - 1$ multiplicações matriciais. Além disso, os resultados parciais do Algoritmo 7.2 (as matrizes \tilde{B}_k) podem ser utilizados para calcular a derivada de Fréchet de $P(X)$, diminuindo assim o custo computacional no cálculo da derivada. O estudo da estabilidade consiste em estimar, a priori, os erros introduzidos pelo arredondamento nas operações. É conhecido que, no caso escalar, o Algoritmo 7.2 é mais estável do que o Algoritmo 7.1; no caso matricial, o resultado é igualmente significativo.

algoritmo	espaço adicional	número de flops
7.1	$3n^2$	$m(m-1)n^3$
7.2	$2n^2$	mn^3

TABELA 7.1

A literatura [35, 41] apresenta algumas modificações do método de Horner que efetuam a avaliação de um polinômio matricial com um número menor de operações, porém, estes algoritmos são restritos a polinômios cujos coeficientes são múltiplos da identidade ($A_i = \lambda_i I$ com $\lambda_i \in \mathbb{C}$). Até agora, não se conhece nenhum algoritmo mais eficiente que o algoritmo de Horner para avaliar polinômios matriciais, isto é, nenhum método realiza a avaliação de um polinômio matricial $P(X)$ arbitrário, com menos de m multiplicações matriciais.

7.1. Análise do erro de arredondamento

Nesta seção, descreve-se a propagação dos erros de arredondamento na avaliação de um polinômio matricial tanto pelo Algoritmo 7.1 quanto pelo Algoritmo 7.2.

Se o símbolo $fl(x)$ denotar o representante em ponto flutuante do número x , a unidade de arredondamento μ é definida como o maior número y tal que

$$fl(1 + y) = 1.$$

Se ϵ for a precisão da máquina em ponto flutuante, o valor de μ é $\frac{1}{2}\epsilon$. Em aritmética real tem-se:

$$fl(x_1 * x_2) = (x_1 * x_2)(1 + \delta) \quad |\delta| \leq \mu,$$

onde $*$ representa alguma das operações soma, subtração, multiplicação ou divisão, e $fl(*)$ denota a operação $*$ realizada em ponto flutuante. Em aritmética complexa [42] tem-se um resultado análogo:

$$\begin{aligned} fl(z_1 \pm z_2) &= (z_1 \pm z_2)(1 + \delta) & |\delta| &\leq \mu, \\ fl(z_1 \cdot z_2) &= (z_1 \cdot z_2)(1 + \delta) & |\delta| &\leq 2\sqrt{2}\mu + O(\mu^2), \\ fl(z_1 \div z_2) &= (z_1 \div z_2)(1 + \delta) & |\delta| &\leq 5\sqrt{2}\mu + O(\mu^2). \end{aligned}$$

Uma análise do erro no estilo de Wilkinson [45] permite mostrar o seguinte resultado para o erro de arredondamento cometido na soma e na multiplicação de matrizes.

Lema 7.1 ([9]). *Se $A, B \in \mathbb{C}^{n \times n}$, então*

$$fl(A + B) = A + B + E_1, \quad \|E_1\|_F \leq \mu \|A + B\|_F$$

e

$$fl(AB) = AB + E_2, \quad \|E_2\|_F \leq 2\sqrt{2}n\mu \|A\|_F \|B\|_F + O(\mu^2).$$

PROVA. A afirmação sobre o erro cometido na soma de matrizes é uma consequência da estimativa da soma de dois números complexos.

A análise do erro cometido no produto é um pouco mais complicada. Em primeiro lugar, considere o problema de calcular

$$\eta = A\xi,$$

onde $A = (a_{ij})_1^n$ é uma matriz de dimensão n e $\eta = (\eta_i)_1^n$ e $\xi = (\xi_i)_1^n$ são dois vetores coluna. Logo,

$$\eta_i = \sum_{j=1}^n a_{ij}\xi_j + \zeta_i$$

onde

$$\begin{aligned} |\zeta_i| &\leq 2\sqrt{2}\mu \left| na_{i1}\xi_1 + \sum_{j=2}^n (n-j+2) a_{ij}\xi_j \right| \\ &\leq 2\sqrt{2}n\mu \left(\sum_{j=1}^n |a_{ij}| |\xi_j| \right). \end{aligned}$$

Assim sendo, tem-se a seguinte estimativa para a norma euclídeana do vetor ζ :

$$\|\zeta\|_2^2 \leq 8n^2 \mu^2 \|A\|_F^2 \|\xi\|_2^2. \quad (7.1)$$

Aplicando (7.1) a cada coluna da matriz B , obtém-se a estimativa para o erro na multiplicação de matrizes. \square

Com o auxílio do Lema 7.1, pode-se mostrar o seguinte teorema sobre o erro de arredondamento na avaliação do $P(X)$.

Teorema 7.1. *Se o polinômio matricial $P(X)$ for avaliado empregando o Algoritmo 7.1 ou o Algoritmo 7.2, então*

$$fl(P(X)) = P(X) + E,$$

onde

$$\|E\|_F \leq \mu \left((2\sqrt{2}n + 1) \sum_{l=1}^m l \|A_{m-l}\|_F \|X\|_F^l + \sum_{l=0}^{m-1} \|A_{m-l}\|_F \|X\|_F^l \right) + O(\mu^2).$$

PROVA. Se $P(X)$ for avaliado pelo Algoritmo 7.1, sejam $\tilde{B}_1, \dots, \tilde{B}_m$ as matrizes definidas por:

$$\begin{aligned} \tilde{B}_0 &= A_0 \\ \tilde{B}_k &= \tilde{B}_{k-1}X + A_k \end{aligned} \quad (k = 1, \dots, m),$$

então

$$fl(\tilde{B}_k) = \tilde{B}_k + E_k + O(\mu^2),$$

onde

$$\|E_k\| \leq \mu \left((2\sqrt{2}n + 1) \sum_{l=1}^k l \|A_{m-l}\| \|X\|^l + \sum_{l=0}^{k-1} \|A_{m-l}\| \|X\|^l \right) + O(\mu^2).$$

A prova é obtida por indução em k . Para $k = 1$ o resultado é verdadeiro; se a afirmação for verdadeira para k , o Lema 7.1 mostra que:

$$\begin{aligned} fl(\tilde{B}_{k+1}) &= fl(fl(fl(\tilde{B}_k)X) + A_{k+1}) \\ &= fl(fl((\tilde{B}_k + E_k)X) + A_{k+1}) \\ &= \tilde{B}_{k+1} + E'_1 + E'_2 + E_kX, \end{aligned}$$

onde

$$\begin{aligned} \|E'_1\| &\leq 2\sqrt{2}n\mu \|\tilde{B}_k\| \|X\| \\ \|E'_2\| &\leq \mu (\|\tilde{B}_{(k)}\| \|X\| + \|A_{k+1}\|). \end{aligned}$$

Usando uma estimativa para $\|\tilde{B}_k\|$ o teorema resulta verdadeiro. \square

Mesmo que a estimativa do erro seja a mesma para os dois algoritmos, em função do número de flops envolvidos, é de se esperar que em geral o erro produzido pelo Algoritmo 7.2 seja bem menor do que aquele do Algoritmo 7.1.

8. Aplicações

Esta seção é dedicada a mostrar as aplicações dos conceitos definidos nas seções precedentes. Com este objetivo, os solventes, co-solventes e cadeias de solventes são empregados para descrever as soluções de equações matriciais de diferenças e diferenciais [21, 25, 26].

Associada à equação de diferenças

$$\sum_{l=0}^m A_{m-l} X_{k+l} = 0, \quad (8.1)$$

tem-se o polinômio matricial $P(X)$ definido por:

$$P(X) = \sum_{l=0}^m A_{m-l} X^l. \quad (8.2)$$

O Teorema 8.1 mostra que os conjuntos completos de solventes ou de co-solventes podem ser utilizados para definir um conjunto gerador do espaço de soluções da equação de diferenças (8.1).

Teorema 8.1. *Se S_1, \dots, S_m for um conjunto completo de solventes do polinômio matricial $P(X)$, então as seqüências*

$$\{X_k^{(i)} = S_i^k ; k = 0, 1, \dots\} \quad (i = 1, \dots, m)$$

geram as soluções da equação (8.1).

Se $(Q_1, R_1), \dots, (Q_m, R_m)$ for um conjunto completo de co-solventes de $P(X)$, então as seqüências

$$\{X_k^{(i)} = Q_i R_i^k ; k = 0, 1, \dots\} \quad (i = 1, \dots, m),$$

geram as soluções da equação (8.1).

PROVA. Se S_1, \dots, S_m for um conjunto completo de solventes, segue-se, para cada $i = 1, \dots, m$, que a seqüência:

$$\{X_k^{(i)} = S_i^k ; k = 0, 1, \dots\}$$

é uma solução da equação (8.1), pois

$$\sum_{l=0}^m A_{m-l} X_{k+l}^{(i)} = \left(\sum_{l=0}^m A_{m-l} S_i^l \right) S_i^k = 0.$$

Seja $\{X_k ; k = 0, 1, \dots\}$ uma solução particular da equação (8.1) e sejam $\Omega_1, \dots, \Omega_m$ matrizes tais que

$$V(S_1, \dots, S_m) \begin{pmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_m \end{pmatrix} = \begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{m-1} \end{pmatrix}. \quad (8.3)$$

Sendo $V(S_1, \dots, S_m)$ inversível, existe uma única solução de (8.3) e, portanto, a seqüência $\{Y_k = S_1^k \Omega_1 + \dots + S_m^k \Omega_m ; k = 0, 1, \dots\}$ é uma solução da equação de diferenças (8.1) com condições iniciais:

$$Y_k = X_k \quad (k = 0, \dots, m-1),$$

logo, o resultado é uma conseqüência da unicidade da solução da equação de diferenças (8.1).

A prova da segunda afirmação é semelhante àquela da primeira parte. \square

O Teorema 8.2 é o análogo do Teorema 8.1 para a equação diferencial

$$\sum_{l=0}^m A_{m-l} Y^{(l)} = 0, \quad (8.4)$$

onde $Y = Y(t)$ e $Y^{(l)}$ é a derivada de ordem l de Y em relação a t .

Teorema 8.2. *Seja $P(X)$ o polinômio matricial definido em (8.2).*

(1) *Se S_1, \dots, S_m for um conjunto completo de solventes de $P(X)$, então as funções*

$$X_i(t) = e^{tS_i} \quad (i = 1, \dots, m),$$

geram as soluções da equação (8.4).

(2) Se $(Q_1, R_1), \dots, (Q_m, R_m)$ for um conjunto completo de co-solventes de $P(X)$, então as funções

$$X_i(t) = Q_i e^{tR_i} \quad (i = 1, \dots, m),$$

formam uma base de soluções da equação (8.4).

PROVA. Sendo $\frac{d}{dt} Q_i e^{tR_i} = Q_i R_i e^{tR_i}$ a demonstração é completamente análoga à prova do Teorema 8.2. \square

O Teorema 8.3 mostra como empregar uma cadeia de solventes do polinômio matricial $P(X)$ para obter uma solução da equação diferencial não homogênea

$$\sum_{l=0}^m A_{m-l} Y^{(l)} = F(t) \quad (8.5)$$

quando $A_0 = I$.

Teorema 8.3 ([21]). *Seja $P(X)$ o polinômio matricial mônico associado à equação (8.5), e seja R_1, \dots, R_m uma cadeia de solventes de $P(X)$, então a solução da equação (8.5) com condições iniciais*

$$Y^{(l)}(0) = Y_l \quad (l = 0, \dots, m-1),$$

é dada por:

$$\begin{aligned} Y(t) = & e^{tR_m} B_1 + \int_0^t e^{(t-t_1)R_m} e^{t_1 R_{m-1}} B_2 dt_1 + \dots \\ & + \int_0^t \int_0^{t_1} \dots \int_0^{t_{m-2}} e^{(t-t_1)R_m} e^{(t_1-t_2)R_{m-1}} \dots e^{(t_{m-2}-t_{m-1})R_2} \\ & e^{t_{m-1}R_1} B_m dt_{m-1} dt_{m-2} \dots dt_1 \\ & + \int_0^t \int_0^{t_1} \dots \int_0^{t_{m-1}} e^{(t-t_1)R_m} e^{(t_1-t_2)R_{m-1}} \dots e^{(t_{m-1}-t_m)R_1} \\ & F(t_m) dt_m dt_{m-1} \dots dt_1, \end{aligned}$$

onde

$$\begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{pmatrix} = P_m^{-1} \begin{pmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{m-1} \end{pmatrix}$$

e P_m é a matriz (5.7) fornecida pelo Teorema 5.3.

PROVA. A demonstração consiste em aplicar o Teorema 5.3. Usando a transformação $\{Y^{(l)}(t) = V_{l+1}(t) \ ; \ l = 0, \dots, m-1\}$, e definindo

$$V(t) = \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_m \end{pmatrix},$$

a equação (8.5) é equivalente ao sistema:

$$V'(t) = \begin{pmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ -A_m & -A_{m-1} & -A_{m-2} & \cdots & -A_1 \end{pmatrix} V(t) + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ F(t) \end{pmatrix}, \quad (8.6)$$

com condições iniciais $\{V_l(0) = Y_{l-1} \ ; \ l = 1, \dots, m\}$. Sendo R_1, \dots, R_m uma cadeia de solventes, o Teorema 5.3 mostra que a matriz companheira por blocos de $P(X)$ é similar a uma matriz bi-diagonal por blocos, isto é, existe uma matriz inversível P_m tal que

$$\begin{pmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ -A_m & -A_{m-1} & -A_{m-2} & \cdots & -A_1 \end{pmatrix} = P_m \begin{pmatrix} R_m & I & & & \\ & \ddots & \ddots & & \\ & & R_2 & I & \\ & & & & R_1 \end{pmatrix} P_m^{-1}.$$

Definindo

$$Z(t) = P_m^{-1}V(t), \quad B = P_m^{-1}V(0) \quad \text{e} \quad W(t) = P_m \begin{pmatrix} 0 \\ 0 \\ \vdots \\ F(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ F(t) \end{pmatrix},$$

o sistema (8.6) pode ser escrito da seguinte forma:

$$\begin{aligned} Z_1^{(1)} &= R_m Z_1 + Z_2 \\ Z_2^{(1)} &= R_{m-1} Z_2 + Z_3 \\ &\vdots \\ Z_m^{(1)} &= R_1 Z_m + F(t). \end{aligned} \quad (8.7)$$

Como a última equação não depende das anteriores, as equações em (8.7) podem ser resolvidas de baixo para cima. Como a matriz P_m é triangular superior por blocos e $P_{11}^{(m)} = I$, segue-se que $Z_1(t) = V_1(t)$. Assim sendo, o resultado fica demonstrado. \square

O Teorema 8.4 emprega uma cadeia de solventes de $P(X)$ para fornecer uma solução da equação de diferenças não homogênea

$$\sum_{l=0}^m A_{m-l} X_{k+l} = F_k. \quad (8.8)$$

Teorema 8.4 ([21]). *Se R_1, \dots, R_m for uma cadeia de solventes de $P(X)$, então a solução da equação (8.8) com condições iniciais*

$$X_k = Y_k \quad (k = 0, \dots, m-1),$$

é dada por:

$$\begin{aligned} X_k = & R_m^k B_1 + \sum_{k_1=0}^{k-1} R_m^{k-k_1-1} R_{m-1}^{k_1} B_2 + \dots \\ & + \sum_{k_1=0}^{k-1} \sum_{k_2=0}^{k_1-1} \dots \sum_{k_{m-1}=0}^{k_{m-2}-1} R_m^{k-k_1-1} R_{m-1}^{k_1-k_2-1} \dots R_2^{k_{m-2}-k_{m-1}-1} R_1^{k_{m-1}} \\ & + \sum_{k_1=0}^{k-1} \sum_{k_2=0}^{k_1-1} \dots \sum_{k_m=0}^{k_{m-1}-1} R_m^{k-k_1-1} R_{m-1}^{k_1-k_2-1} \dots R_1^{k_{m-1}-k_m-1} Y_{k_m}, \end{aligned}$$

onde

$$\begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{pmatrix} = P_m^{-1} \begin{pmatrix} Y_0 \\ Y_1 \\ \vdots \\ Y_{m-1} \end{pmatrix}$$

e P_m é a matriz fornecida em (5.7).

PROVA. A demonstração é análoga àquela do Teorema 8.3. \square

O Teorema 8.5 apresenta um outro problema cuja solução pode ser fornecida a partir do conhecimento dos co-solventes de um polinômio matricial.

Teorema 8.5 ([25]). *Dada a equação diferencial*

$$\sum_{l=0}^m A_{m-l} Y^{(l)} = 0 \quad (8.9)$$

considere o problema de valores de contorno

$$BY(b) - Y(0)C = E$$

onde b é um real positivo e B , C e E são matrizes quadradas. Se (Q, R) for um co-solvente de $P(X)$ tal que

$$\sigma(BQe^{bR}) \cap \sigma(QC) = \emptyset,$$

então

$$Y(t) = Qe^{tR}D$$

é a solução de (8.9), onde D é a única solução de

$$BQe^{bR}Z - ZQC = E. \quad (8.10)$$

PROVA. Considerando o Teorema 8.2, mostrar que a função

$$Y(t) = Qe^{tR}D$$

é a solução da equação (8.9) que satisfaz:

$$BY(b) - Y(0)C = E,$$

é equivalente a mostrar que a matriz D satisfaz

$$BQe^{bR}D - DQC = E.$$

Desde que BQe^{bR} e QC não tenham autovalores em comum, o Teorema A-1.3 mostra que a solução de (8.10) é única e, portanto, o teorema fica demonstrado. \square

Observe-se que os solventes de um polinômio matricial também podem ser empregados para resolver a equação diferencial

$$\sum_{l=0}^m A_{m-l} \xi^{(l)} = 0, \quad (8.11)$$

onde ξ é uma função vetorial, pois se S for um solvente de $P(X)$ com

$$P(X) = \sum_{l=0}^m A_{m-l} X^l,$$

então a função $e^{St}\eta$ é uma solução de (8.11) para todo $\eta \in \mathbb{C}^{n \times n}$. Se todos os autovalores de S forem em módulo menores do que um, a existência de raízes latentes de $P(X)$ com módulo maior que um, não afeta as propriedades das soluções $e^{St}\eta$.

Capítulo III

Métodos para o cálculo de solventes

O Capítulo III é dedicado ao estudo de alguns dos métodos usados para a resolução da equação

$$P(X) = \sum_{l=0}^m A_{m-l} X^l = 0.$$

Os algoritmos descritos neste capítulo são: método de Jordan, método de Newton, método de Newton modificado, métodos para solventes dominantes, algoritmos para polinômios com coeficientes triangulares e triangulares por blocos. Para alguns métodos, estudam-se variantes com o objetivo de melhorar as suas propriedades. Para outros, analisam-se casos particulares com o intuito de descrever melhor o comportamento do algoritmo considerado. No final do Capítulo, os métodos apresentados são comparados levando-se em consideração o número de operações, condições de convergência ou de aplicação e, principalmente, a possibilidade de verificar estas condições quando os algoritmos forem aplicados para resolver um determinado problema.

1. Método de Jordan

Nesta seção, apresenta-se uma forma de calcular solventes e co-solventes do polinômio matricial

$$P(X) = \sum_{l=0}^m A_{m-l} X^l = 0$$

que utiliza a decomposição na forma de Jordan de um solvente S de $P(X)$ [15].

Algoritmo 1.1. Seja S um solvente de $P(X)$, seja J a sua forma canônica de Jordan e seja Q uma matriz inversível tal que $S = QJQ^{-1}$. Assim sendo, a matriz Q é uma solução da equação

$$\sum_{l=0}^m A_{m-l} Q J^l = 0. \quad (1.1)$$

Se a matriz J for conhecida, o sistema (1.1) é equivalente à equação

$$\left(\sum_{l=0}^m (J^t)^l \otimes A_{m-l} \right) v(Q) = C v(Q) = 0. \quad (1.2)$$

Uma vez que J é uma matriz triangular superior, a matriz de coeficientes da equação (1.2), C , é triangular inferior por blocos. Levando-se em conta que os elementos da diagonal da matriz J são raízes latentes de $P(\lambda)$, os blocos da diagonal de C são matrizes singulares. Desta maneira, o sistema (1.1) tem solução e, portanto, o Algoritmo 1.1 é mais útil para calcular co-solventes que para fornecer solventes. Se alguma solução do sistema (1.1) for inversível, então QJQ^{-1} é um solvente de $P(X)$. A singularidade da matriz C pode ocasionar erros consideráveis no cálculo numérico das soluções de (1.2) devido ao mau condicionamento (ver Wilkinson [45]).

Exemplo 1.1. Considere o polinômio matricial

$$P(X) = \begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix} X^2 + \begin{pmatrix} 3 & -3 \\ -2 & 18 \end{pmatrix} X + \begin{pmatrix} 4 & -6 \\ -4 & 24 \end{pmatrix},$$

pode-se mostrar que as raízes latentes de $P(X)$ são $\lambda_1 = -0.624022 + 0.948946 i$, $\lambda_2 = \bar{\lambda}_1$, $\lambda_3 = -1.62598 + 1.41694 i$, $\lambda_4 = \bar{\lambda}_3$. Para determinar se existe um solvente com autovalores λ_1 e λ_3 , é suficiente estudar as soluções dos sistemas

$$\begin{pmatrix} 1.10574 + 0.478186 i & -4.12793 - 2.84684 i \\ -2.75196 - 1.89789 i & 9.70103 + 9.97508 i \end{pmatrix} \begin{pmatrix} q_{11} \\ q_{21} \end{pmatrix} = 0$$

$$e \quad \begin{pmatrix} 0.394258 - 4.96482 i & -1.12207 - 4.25081 i \\ -0.748044 - 2.83387 i & -1.45103 - 2.14203 i \end{pmatrix} \begin{pmatrix} q_{12} \\ q_{22} \end{pmatrix} = 0.$$

Pode-se observar que, uma solução dos sistemas acima deve satisfazer

$$q_{21} = (0.235669 - 0.0466885 i) q_{11} \quad e \quad q_{22} = (-1.069 - 0.374928 i) q_{12},$$

logo, escolhendo adequadamente q_{11} e q_{12} resulta que a matriz S definida por:

$$S = \begin{pmatrix} -0.75 + 1.10103 i & 0.637386 - 0.519062 i \\ 0.191692 - 0.115347 i & 1.5 + 1.26485 i \end{pmatrix},$$

é um solvente de $P(X)$.

Este algoritmo é numericamente “instável”, pois envolve a solução de um sistema linear cuja matriz de coeficientes é singular. O método é ainda mais perigoso quando o polinômio matricial possui raízes latentes muito próximas; no exemplo acima, isto se refletiria nos cocientes $\frac{q_{11}}{q_{21}}$ e $\frac{q_{12}}{q_{22}}$, os quais seriam muito parecidos.

2. Método de Newton

O método de Newton é um conhecido algoritmo para calcular raízes de equações algébricas em geral, e polinomiais em particular. Este método foi generalizado em [28] para calcular zeros de operadores entre espaços de Banach, no entanto, a demonstração de sua convergência é bem mais simples quando são consideradas somente equações polinomiais matriciais [29].

O método de Newton define uma seqüência de matrizes $\{X_1, X_2, \dots\}$ a partir de uma aproximação inicial X_0 , usando o seguinte algoritmo:

Algoritmo 2.1. Dado X_0 , define-se $\{X_1, X_2, \dots\}$ por:

$$X_{i+1} = X_i - Q_i \quad \text{com} \quad P'(X_i)[Q_i] = P(X_i), \quad (2.1)$$

onde $P'(X_i)$ é a derivada de Fréchet do polinômio $P(X)$ em X_i (ver Seção II-3).

O Teorema 2.1 estabelece as condições nas quais o Algoritmo 2.1 converge. Estas condições, analogamente ao que acontece no caso escalar, requerem que a derivada seja um operador inversível nas proximidades do solvente, e que a aproximação inicial esteja suficientemente próxima da solução. Com estas hipóteses obtém-se uma seqüência que converge quadraticamente ao solvente.

Teorema 2.1. *Seja $S \in \mathbb{C}^{n \times n}$ um solvente simples de $P(X)$. Se $X_0 \in \mathbb{C}^{n \times n}$ estiver suficientemente próximo de S , então a seqüência $\{X_0, X_1, \dots\}$ definida pelo Algoritmo 2.1 converge a S . Mais precisamente, seja $\epsilon_0 = \|X_0 - S\|_F$ e seja δ um número positivo suficientemente pequeno tal que*

$$0 < \alpha = \inf \left\{ \|P'(X)[H]\|_F : \|H\|_F = 1, \|X - S\|_F \leq \delta \right\}.$$

Neste caso, se

$$\epsilon_0 < \epsilon = \min \{ \delta, \alpha \beta^{-1} \}, \quad (2.2)$$

onde

$$\beta = 2^{m+1} \gamma (1 + \|S\|_F)^{m-2} \quad e \quad \gamma = \max_{l=0, \dots, m-1} \{ \|A_{m-l}\|_F \},$$

então

$$\lim_{i \rightarrow \infty} X_i = S. \quad (2.3)$$

Além disso, para $i = 1, 2, \dots$, tem-se que

$$\|X_{i+1} - S\|_F \leq \frac{\beta}{\alpha} \|X_i - S\|_F^2. \quad (2.4)$$

PROVA. Dado o solvente S de $P(X)$, define-se $b = 1 + \|S\|_F$ e $\mathcal{B} = \overline{B(S, \epsilon_0)}$.¹ Para todo $X \in \mathcal{B}$ tem-se $\|X\|_F \leq b$ e, além disso, para todo $H \in \mathbb{C}^{n \times n}$,

$$\|P'(X)[H]\|_F \geq \alpha \|H\|_F.$$

Como $\alpha > 0$, a derivada de Fréchet de $P(X)$ é regular em \mathcal{B} e, portanto, a função $Q(X) = P'(X)^{-1}[P(X)]$ está definida em \mathcal{B} . Para cada $X \in \mathcal{B}$ seja $\Phi(X)$ a função definida por:

$$\Phi(X) = X - Q(X).$$

Levando-se em conta que S é solvente de $P(X)$, o Teorema II-3.1 e a equação (2.2) mostram que:

$$\begin{aligned} \|\Phi(X) - S\|_F &= \|X - S - P'(X)^{-1}[P(X)]\|_F \\ &\leq \frac{1}{\alpha} \|P'(X)[X - S] - P(X) + P(S)\|_F \\ &\leq \frac{\beta}{\alpha} \|X - S\|_F^2. \end{aligned} \quad (2.5)$$

¹ $\overline{B(S, \epsilon_0)}$ é a bola fechada de centro S e raio ϵ_0 .

A prova de (2.4) segue-se por indução em i . Por hipótese, $X_0 \in \mathcal{B}$ e, portanto, X_1 está definida. Seja $q = \frac{\beta}{\alpha}\epsilon_0$, sendo, por definição, $q < 1$, segue-se de (2.5), que

$$\|X_1 - S\|_F \leq \frac{\beta}{\alpha} \|X_0 - S\|_F^2 \leq \epsilon_0 q < \epsilon_0.$$

Supondo que a afirmação (2.4) é verdadeira para i , tem-se que $X_i \in \mathcal{B}$ e, portanto, X_{i+1} está definida. Em virtude de (2.5),

$$\|X_{i+1} - S\|_F = \|\Phi(X_i) - S\|_F \leq \frac{\beta}{\alpha} \|X_i - S\|_F^2.$$

Como

$$\frac{\beta}{\alpha} \|X_i - S\|_F^2 \leq \epsilon_0 q \|X_i - S\|_F \leq \epsilon_0 q^{i+1},$$

a afirmação (2.3) é uma consequência de (2.4), pois $q < 1$. □

A seguir, realizam-se alguns comentários sobre a implementação do algoritmo de Newton.

2.1. Considerações sobre a aplicação do método

Alguns pontos devem ser esclarecidos quanto à utilização do método de Newton. Em primeiro lugar, o Algoritmo 2.1 tem um ponto crítico na escolha de X_0 . No caso escalar, muitas vezes é possível obter uma boa aproximação inicial empregando um método de primeira ordem, como por exemplo, o método de bisseção. Porém, no caso matricial, métodos de primeira ordem não têm sido estudados. Em consequência disto, na implementação do algoritmo de Newton, alguns autores foram obrigados a proporcionar uma aproximação inicial empírica, sem a existência de regras precisas para esta escolha [9, 29]. Em geral, e por motivos desconhecidos, tem-se preferido iniciar a iteração com um X_0 múltiplo da identidade, isto é,

$$X_0 = kI.$$

Se o polinômio for mônico, em [29] propõe-se escolher k como sendo:

$$k = \max \left\{ 1, \sum_{l=0}^{m-1} \|A_{m-l}\|_F \right\},$$

e quando $m = 2$, em [9] propõe-se definir k por:

$$k = \frac{\|B\|_F + \sqrt{\|B\|_F^2 + 4\|A\|_F\|C\|_F}}{2\|A\|_F}.$$

Em ambos os casos, a constante k é uma estimativa da norma de qualquer solvente de $P(X)$. Sendo a norma de uma matriz uma estimativa do seu raio espectral, pode-se supor que o objetivo é estimar o raio espectral da matriz companheira por blocos C de $P(X)$. Assim sendo, qualquer outra estimativa do raio espectral de C poderia ter sido utilizada. Uma vez que não existe um critério para a escolha de k , nenhum teste neste sentido foi realizado. Além disso, não é possível garantir a convergência, nem mesmo quando o espectro da matriz C for conhecido.

Um outro aspecto delicado na aplicação do Método de Newton é o cálculo da correção Q_i . Aqui devem ser mencionadas duas dificuldades a serem cuidadosamente estudadas. A primeira é a avaliação de $P(X_i)$ (ver Seção II-7), e a segunda é a inversão de $P^*(X_i)$. A singularidade da derivada de Fréchet foi estudada na Seção II-3.2.

É pouco o que pode-se dizer em relação aos métodos para a resolução do sistema linear (2.1) que define a correção Q_i . Em cada caso, devem ser analisadas as características especiais do sistema, para assim escolher um método que possa aproveitar estes aspectos, pois trata-se de resolver um sistema de ordem n^2 . Quando nenhuma particularidade for conhecida, pode-se utilizar o método de Gauss, por exemplo, para o qual são necessárias multiplicações da ordem de $\frac{1}{3}n^6$. Uma forma de reduzir o número de operações realizadas quando $m = 2$ é estudada na Seção 2.2. Quando $m > 2$, pode-se reduzir o número de operações na triangularização da matriz de coeficientes do sistema (2.1), observando que a matriz $P^*(X)$ pode ser transformada em uma matriz triangular por blocos (Teorema II-3.4). Uma descrição detalhada deste método encontra-se na Seção 2.3.

O número de operações requeridas para a avaliação do polinômio e para o cálculo da matriz $P^*(X_i)$, dependerá necessariamente do método empregado. Se o método de Horner for utilizado, são necessários da ordem de mn^3 flops para avaliar $P(X_i)$. A partir dos resultados parciais do Algoritmo II-7.2, são requeridos $(m-1)n^4 + (m-2)n^3$ flops para calcular $P^*(X_i)$.

O espaço de memória requerido depende muito das características do polinômio considerado. Para um polinômio geral e usando o método de Gauss para resolver o sistema linear, são necessárias da ordem de $n^2(n^2 + m)$ palavras adicionais.

2.2. Cálculo da correção Q_i . Polinômios de segundo grau

Nesta seção, apresenta-se uma forma de construir e resolver o sistema linear que determina a correção Q_i quando $m = 2$ [9]. Se $P(X)$ for um polinômio de segundo grau, a equação considerada é:

$$P(X) = A_0X^2 + A_1X + A_2 = 0,$$

e a correção Q_i é a solução do sistema

$$(A_0X_i + A_1)Q_i + A_0Q_iX_i = P(X_i). \quad (2.6)$$

O método proposto para resolver (2.6) consiste em reduzir as matrizes $A_0X_i + A_1$ e A_0 simultaneamente à forma triangular superior, e a matriz X_i à forma triangular inferior. Desta maneira, o sistema (2.6) é transformado em um sistema triangular superior, o qual pode ser resolvido facilmente desde que a matriz de coeficientes seja não singular.

Algoritmo 2.2. Dada uma aproximação X_i , o algoritmo consiste nas seguintes etapas:

- (1) Calcular as matrizes unitárias U e V e as matrizes triangulares superiores T_1 e T_2 tais que:

$$U^*(A_0X_i + A_1)V = T_1 \quad \text{e} \quad U^*A_0V = T_2.$$

- (2) Calcular a matriz unitária W e a matriz triangular inferior L de modo que:

$$W^*X_iW = L.$$

- (3) Resolver o sistema linear

$$T_1Y_i + T_2Y_iL = U^*P(X_i)W.$$

- (4) Calcular

$$Q_i = VY_iW^*.$$

Para o cálculo das matrizes $U, V \in \mathcal{U}_n$ da etapa (1), é usada uma versão complexa do algoritmo QZ .² Este algoritmo é empregado para calcular **autovalores** e **autovetores generalizados** de duas matrizes arbitrárias A e B , isto é, calcular escalares λ_i e vetores ξ_i satisfazendo

$$A\xi_i = \lambda_i B\xi_i.$$

²O algoritmo QZ é amplamente estudado em [16, 33, 39, 44].

Na segunda etapa emprega-se o mesmo algoritmo QZ , desta vez aplicado às matrizes X_i^t e I , para produzir $S, R \in \mathcal{U}_n$ tais que $S^*X_i^tR = T_3$ e $S^*IR = \tilde{I}$. A matriz \tilde{I} é triangular superior e unitária e, portanto, é uma matriz diagonal cujos elementos são ± 1 ; assim sendo, $X_i^t = ST_3R^* = R\tilde{I}T_3R^*$ e, portanto, $L = (\tilde{I}T_3)^t$ e $W = R$.

O número de flops necessários em cada iteração do algoritmo é dado pela expressão $22n^3 + 9n^3\sigma + O(n^2)$, onde σ é o número médio de iterações do algoritmo QZ para anular um elemento. No que se refere ao espaço de memória, uma implementação bem clara requer da ordem de $11n^2 + n$ palavras, porém, este número pode cair para $9n^2 + 3n$ palavras reduzindo a legibilidade do programa. Para uma análise mais detalhada do método ver Davis [9].

2.3. Cálculo da correção Q_i . Caso Geral

Como já foi esclarecido, a correção Q_i é a solução do sistema linear

$$P^*(X_i)v(Q_i) = v(P(X_i)), \quad (2.7)$$

sendo a matriz $P^*(X_i)$ de ordem n^2 e definida por:

$$P^*(X_i) = \sum_{l=1}^m \left((X_i^t)^{l-1} \otimes B_l \right),$$

onde as matrizes B_l são os resultados parciais obtidos na aplicação do Algoritmo II-7.2, quando o polinômio $P(X)$ for avaliado em X_i .

Seja U uma matriz unitária a qual satisfaz

$$U^*X_i^tU = T_i,$$

onde T_i é uma matriz triangular superior, e seja Y a matriz definida por:

$$Y = \sum_{l=1}^m \left((T_i)^{l-1} \otimes B_l \right).$$

Observe-se que a matriz Y é triangular superior por blocos, e que o sistema (2.7) é equivalente ao sistema:

$$Yv(UQ_i) = v(UP(X_i)).$$

A partir desta observação, o seguinte Algoritmo reduz significativamente o número de operações no cálculo de Q_i .

Algoritmo 2.3. Dada a aproximação X_i ,

- (1) Calcular uma matriz unitária U tal que $U^* X_i^t U = T_i$ é uma matriz triangular superior.
- (2) Calcular $Y = \sum_{l=1}^m ((T_i)^{l-1} \otimes B_l)$.
- (3) Resolver o sistema linear

$$Y v(\tilde{Q}) = v(UP(X_i)).$$

- (4) Calcular $Q_i = U^* \tilde{Q}$.

O Algoritmo QZ , aplicado às matrizes X_i^t e I , fornece duas matrizes unitárias U e V tais que as matrizes $U^* X_i^t V$ e $U^* V$ são triangulares superiores. Uma vez que $U^* V$ é uma matriz unitária, também é uma matriz diagonal e, portanto, U é a matriz necessária na etapa (1). Para calcular as matrizes U e T usando as versões complexas dos Algoritmos 7.7-1 e 7.7-2 contidos em Golub e Van Loan [18], são necessários $\alpha n^3 + O(n^2)$ flops, onde α é uma constante que não depende de n .

Na segunda etapa são requeridos $(m-2) \frac{n^3}{6}$ flops para avaliar as potências de T . Conhecidas as matrizes B_l e as potências de T com $\frac{1}{2}(m-1)(n+1)n^3$ flops pode-se calcular Y .

Para desenvolver a terceira etapa é preciso resolver n sistemas lineares para o que são executados da ordem de $\frac{n^4}{3}$ flops, e para calcular os termos independentes é preciso realizar $\frac{n^3}{2} + O(n^2)$ flops. A última etapa, que consiste em multiplicar duas matrizes, requer n^3 flops.

Com isto, o total de operações necessárias é da ordem de $\frac{1}{2}mn^4 + O(mn^3)$ flops, o que representa um número bem menor em relação aos $\frac{n^6}{3}$ flops necessários só para resolver o sistema linear (2.7).

2.4. Raiz quadrada

Nesta seção, são expostos brevemente alguns métodos para resolver a equação

$$X^2 = C, \tag{2.8}$$

no intuito de ilustrar a aplicação de algumas variantes do método de Newton a uma equação relativamente simples.

Uma solução de (2.8) é chamada **raiz quadrada** de C . Uma condição necessária e suficiente para a existência de uma raiz quadrada é dada por Lancaster [8].

O método de Newton aplicado a (2.8) fornece a seguinte iteração:

$$X_{i+1} = X_i - Q_i, \quad \text{onde} \quad X_i Q_i + Q_i X_i = X_i^2 - C. \quad (2.9)$$

Algumas das modificações do método de Newton começam por uma aproximação inicial X_0 que comuta com C . Neste caso, a correção Q_i de (2.9) é substituída por:

$$Q_i = \frac{1}{2} (X_i - X_i^{-1} C).$$

Pode-se mostrar que, com esta definição da correção, todos os elementos da sequência $\{X_1, X_2, \dots\}$ definida por:

$$X_{i+1} = \frac{1}{2} (X_i + X_i^{-1} C) \quad (2.10)$$

comutam com C . O algoritmo (2.10) é conhecido pelo nome de **método de Heron**. Um caso particular deste método é definido quando a aproximação inicial for a matriz C . Neste caso, o algoritmo (2.10) é equivalente a

$$\begin{aligned} X_{i+1} &= \frac{1}{2} (X_i + Y_i^{-1}) \\ Y_{i+1} &= \frac{1}{2} (Y_i + X_i^{-1}), \end{aligned} \quad (2.11)$$

onde $X_0 = C$ e $Y_0 = I$. O algoritmo (2.11) também pode ser obtido como uma aplicação da função sinal matricial [3] e, portanto, é convergente desde que C seja definida positiva. Embora os algoritmos (2.10) e (2.11) sejam essencialmente iguais, a prática tem demonstrado que (2.10) é instável em relação ao crescimento do erro de arredondamento, mesmo em matrizes de ordem pequena [22]. O método (2.11) tem sido preferido ao (2.10) ainda que requeira mais espaço de memória e o número de operações envolvidas seja maior. A justificativa para este fato encontra-se na possibilidade de reescrever (2.11) na forma

$$S_{i+1} = \frac{1}{2} (S_i + S_i^{-1}),$$

onde

$$S_i = \begin{pmatrix} 0 & X_i \\ Y_i & 0 \end{pmatrix}.$$

Se a matriz C for definida positiva, o condicionamento da matriz S_i melhora com cada iteração. Como no algoritmo (2.10) a matriz C é empregada em todas as iterações, se o número de condição³ de C for grande, é provável que o erro de arredondamento em (2.10) seja maior do que em (2.11). Hoskins-Waiter.

³Dada uma norma matricial $\|\cdot\|$ define-se o número de condição da matriz A por $\|A\| \|A^{-1}\|$.

analisaram uma variante de (2.11) que melhora a convergência, quando o espectro de C é grande.

3. Método de Newton modificado

Nesta seção, considera-se uma conhecida modificação do método de Newton para equações escalares, a qual consiste em calcular a derivada da função em um único ponto. Este algoritmo, conhecido pelo nome de método de Newton modificado, parte de uma aproximação inicial X_0 suficientemente boa, e gera a seqüência $\{X_1, X_2, \dots\}$ da seguinte forma:

Algoritmo 3.1. Dado X_0 , define-se $\{X_1, X_2, \dots\}$ por:

$$X_{i+1} = X_i - Q_i \quad \text{com} \quad Q_i = P'(X_0)^{-1} P(X_i). \quad (3.1)$$

Assim como para o método de Newton, existe uma generalização do Algoritmo 3.1 que permite calcular zeros de operadores entre espaços de Banach [28], mas a prova aqui apresentada é bem mais simples.

O Teorema 3.1 mostra que a seqüência gerada pelo Algoritmo 3.1 converge para um solvente sem supor a sua existência. As condições estabelecidas pelo Teorema são análogas às do caso escalar.

Teorema 3.1 ([29]). *Seja $X_0 \in \mathbb{C}^{n \times n}$ uma matriz, para a qual $P'(X_0)$ é regular, isto é,*

$$0 < \alpha = \inf_{\|H\|_F=1} \|P'(X_0)[H]\|_F.$$

Neste caso, se $\epsilon_0 = \|P(X_0)\|_F$ satisfizer

$$\epsilon_0 \leq \epsilon = \min \left\{ \alpha/2, \beta^{-1} \alpha^2/4 \right\},$$

onde

$$\beta = 2^{m+2} \gamma (1 + \|X_0\|_F)^{m-2} \quad \text{e} \quad \gamma = \max_{l=0, \dots, m-1} \{\|A_{m-l}\|_F\},$$

então existe um solvente S de $P(X)$ tal que $S = \lim_{i \rightarrow \infty} X_i$ e

$$\|S - X_0\|_F \leq 2 \frac{\epsilon_0}{\alpha}.$$

Além disso, se $q = \frac{2\epsilon_0\beta}{\alpha^2} \leq 1/2$, a seqüência definida por (3.1) satisfaz

$$\|X_{i+1} - X_i\|_F \leq q \|X_i - X_{i-1}\|_F. \quad (3.2)$$

PROVA. Dada a aproximação inicial X_0 , seja $b = 1 + \|X_0\|_F$. Para todo $X \in \mathbb{C}^{n \times n}$, define-se $Q(X)$ como sendo a única solução do sistema linear

$$P'(X_0)[Q(X)] = P(X).$$

Observe-se que $Q(X)$ está definida para todo $X \in \mathbb{C}^{n \times n}$, pois $P'(X_0)$ é regular. Seja $\Phi(X)$ a função definida por:

$$\Phi(X) = X - Q(X).$$

Uma vez que $2\epsilon_0 < \alpha$, se as matrizes $Y_1, Y_2 \in \mathbb{C}^{n \times n}$ satisfizerem

$$\|X_0 - Y_1\|_F \leq 2\frac{\epsilon_0}{\alpha}, \quad \|X_0 - Y_2\|_F \leq 2\frac{\epsilon_0}{\alpha} \quad \text{e} \quad \|Y_1 - Y_2\|_F \leq b,$$

tem-se $\|Y_k\|_F \leq b$, para $k = 1, 2$. Segue-se, do Teorema II-3.1, que:

$$\begin{aligned} \|\Phi(Y_1) - \Phi(Y_2)\|_F &= \|Y_1 - Y_2 - P'(X_0)^{-1}[P(Y_1) - P(Y_2)]\|_F \\ &\leq \frac{1}{\alpha} \|P(Y_1) - P(Y_2) - P'(X_0)[Y_1 - Y_2]\|_F \\ &\leq \frac{\beta}{\alpha} \|Y_1 - Y_2\|_F \max\{\|Y_1 - Y_2\|_F, \|Y_1 - X_0\|_F\}. \end{aligned} \quad (3.3)$$

O primeiro passo da demonstração consiste em provar que todos os elementos da seqüência definida por (3.1) satisfazem

$$\|X_0 - X_i\|_F \leq 2\frac{\epsilon_0}{\alpha} \quad \text{e} \quad \|X_{i+1} - X_i\|_F \leq b. \quad (3.4)$$

A prova de (3.4) decorre por indução. Por hipótese, a afirmação é verdadeira para $i = 1$, pois:

$$\begin{aligned} \|X_1 - X_0\|_F &= \|P'(X_0)^{-1}[P(X_0)]\|_F \\ &\leq \|P'(X_0)^{-1}\|_F \|P(X_0)\|_F \\ &\leq \frac{\epsilon_0}{\alpha}. \end{aligned}$$

Se (3.4) for verdadeira para i , pode-se aplicar (3.3) às matrizes $Y_1 = X_i$ e

$Y_2 = X_{i-1}$. Assim sendo,

$$\begin{aligned}\|X_{i+1} - X_0\|_F &\leq \|\Phi(X_i) - \Phi(X_0)\|_F + \|X_1 - X_0\|_F \\ &\leq \frac{\beta}{\alpha} \|X_i - X_0\|_F^2 + \frac{\epsilon_0}{\alpha} \\ &\leq \frac{\beta}{\alpha} \left(\frac{\epsilon_0}{\alpha}\right)^2 + \frac{\epsilon_0}{\alpha} \\ &\leq 2\frac{\epsilon_0}{\alpha}\end{aligned}$$

e

$$\begin{aligned}\|X_{i+1} - X_i\|_F &= \|\Phi(X_i) - \Phi(X_{i-1})\|_F \\ &\leq \frac{\beta}{\alpha} \|X_i - X_{i-1}\|_F \max\{\|X_i - X_{i-1}\|_F, \|X_i - X_0\|_F\} \\ &\leq q \|X_i - X_{i-1}\|_F \\ &\leq b.\end{aligned}$$

Logo, a seqüência $\{X_0, X_1, \dots\}$ é uma seqüência de Cauchy e portanto converge; seja então $S = \lim_{i \rightarrow \infty} X_i$. Como $P(X_0)$ é regular, segue-se que S é um solvente de $P(X)$. Além disso, a equação (3.4) mostra que:

$$\|S - X_0\|_F \leq 2\frac{\epsilon_0}{c_0}$$

e

$$\begin{aligned}\|X_{i+1} - S\|_F &= \|\Phi(X_i) - \Phi(S)\|_F \\ &\leq \frac{c}{c_0} \|X_i - S\|_F \max\{\|X_i - S\|_F, \|X_i - X_0\|_F\} \\ &\leq q \|X_i - S\|_F \\ &\leq q^{i+1} \|X_0 - S\|_F.\end{aligned}$$

□

As hipóteses do Teorema acima são muito restritivas, tanto que representam uma condição suficiente para a existência de uma solução da equação $P(X) = 0$.

3.1. Considerações sobre a aplicação do método

O Algoritmo 3.1 converge só linearmente, mas, a cada passo, somente uma equação linear é resolvida. A matriz de coeficientes $P^*(X_0) \in \mathbb{C}^{n^2 \times n^2}$ é calculada no início e não é modificada. Entretanto, este método apresenta algumas desvantagens quando comparado com o Algoritmo 2.1. A primeira, e a mais importante,

é que as condições de convergência são muito restritivas, tanto que garantem a existência de um solvente. A segunda desvantagem, como ocorre no caso escalar, é que podem existir problemas nos quais o método de Newton modificado não converge e o algoritmo de Newton sim.

Como cada iteração consiste na solução de um sistema linear, o número de operações desenvolvidas depende do método escolhido. Comparado com o Algoritmo 2.1 nas mesmas condições expostas na Seção 2.1, o método de Newton modificado é mais econômico (em número de operações), pois a matriz de coeficientes pode ser triangularizada no começo do procedimento e, a cada iteração, somente será necessário resolver um sistema linear com uma matriz de coeficientes triangular.

As considerações referentes ao espaço de memória e à aproximação inicial são idênticas às do método de Newton.

4. Cálculo do solvente dominante I. Algoritmo de Bernoulli.

Esta seção é dedicada à generalização do método de Bernoulli utilizado no cálculo da raiz de um polinômio escalar, cujo módulo é maior do que o módulo das outras raízes [12, 17, 19]. Na teoria dos polinômios escalares, é freqüente comparar as raízes de um polinômio através de seus módulos. No caso dos polinômios matriciais, a comparação é feita entre os autovalores dos solventes.

Definição 4.1. Dadas as matrizes A e B , diz-se que a matriz A **domina** a matriz B , se todos os autovalores de A forem estritamente maiores, em módulo, do que os autovalores de B .

Definição 4.2. Um solvente **dominante** é aquele que domina qualquer outro solvente.

Observação 4.1. Se o solvente dominante existir ele é necessariamente escalar.

Dado o polinômio matricial $P(X)$ definido por:

$$P(X) = \sum_{l=0}^m A_{m-l} X^l, \quad (4.1)$$

e dado S um solvente de $P(X)$, segue-se, do Teorema II-8.1, que a seqüência $\{X_i = S^i ; i = 0, \dots\}$ é uma solução particular da equação de diferenças

$$A_0 X_{k+1} + A_1 X_k + \dots + A_m X_{k-m+1} = 0. \quad (4.2)$$

O algoritmo de Bernoulli generalizado, percorre o caminho inverso, fornecendo um solvente de $P(X)$ a partir de uma solução particular da equação de diferenças (4.2).

Algoritmo 4.1. Se a seqüência $\{X_0, X_1, \dots\}$ for a solução da equação de diferenças

$$\begin{aligned} A_0 X_{k+1} + A_1 X_k + \dots + A_m X_{k-m+1} &= 0, \\ X_0 = X_1 = \dots = X_{m-2} &= 0, \quad X_{m-1} = I, \end{aligned} \quad (4.3)$$

então define-se S por:

$$S = \lim_{i \rightarrow \infty} X_i (X_{i-1})^{-1}.$$

O Teorema 4.1 estabelece quais são as condições de convergência da seqüência $X_k (X_{k-1})^{-1}$ para um solvente de $P(X)$. Estas condições são fortes, pois devem assegurar a inversibilidade de X_i . Nesta seção, $\|\cdot\|_\infty$ denota a norma matricial induzida pela norma vetorial do máximo (ver Apêndice A).

Antes de proceder à prova da convergência do Algoritmo 4.1, é preciso mostrar o seguinte resultado.

Lema 4.1 ([12]). *Se a matriz A dominar a matriz B , então para qualquer matriz C ,*

$$\lim_{i \rightarrow \infty} A^{-i} C B^i = 0.$$

PROVA. Dado $\epsilon > 0$, existe uma matriz $P_B(\epsilon)$ tal que

$$B = P_B(\epsilon) J_B(\epsilon) P_B(\epsilon)^{-1},$$

onde $J_B(\epsilon)$ é uma matriz diagonal por blocos. Como os blocos de $J_B(\epsilon)$ são da forma

$$\begin{pmatrix} \lambda_B & \epsilon & & \\ & \lambda_B & \ddots & \\ & & \ddots & \epsilon \\ & & & \lambda_B \end{pmatrix},$$

onde λ_B é um autovalor de B , tem-se:

$$\|B^i\|_\infty \leq \|P_B(\epsilon)\|_\infty \|P_B(\epsilon)^{-1}\|_\infty (\epsilon + \max\{|\lambda_B|\})^i. \quad (4.4)$$

Analogamente,

$$\|A^{-i}\|_{\infty} \leq \|P_{A^{-1}}(\epsilon)\|_{\infty} \|P_{A^{-1}}(\epsilon)^{-1}\|_{\infty} \left(\epsilon + \frac{1}{\min\{|\lambda_A|\}} \right)^i. \quad (4.5)$$

Combinando as equações (4.4) e (4.5), obtém-se

$$\|A^{-i}CB^i\|_{\infty} \leq \alpha \|C\|_{\infty} \left[(\epsilon + \max\{|\lambda_B|\}) \left(\epsilon + \frac{1}{\min\{|\lambda_A|\}} \right) \right]^i, \quad (4.6)$$

onde α é uma função de ϵ que não depende de i . Como a matriz A domina a matriz B ,

$$\frac{\max(|\lambda_B|)}{\min(|\lambda_A|)} < 1;$$

assim, para ϵ suficientemente pequeno a quantidade entre colchetes na equação (4.6) ainda é menor do que um. O resultado segue-se fazendo i tender a infinito. \square

O Lema 4.1 é puramente algébrico, mas resulta de fundamental importância na demonstração do Teorema 4.1.

Teorema 4.1 ([12]). *Seja $P(X)$ o polinômio matricial definido por (4.1) e seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ tal que S_1 é um solvente dominante e a matriz $V(S_2, \dots, S_m)$ é não singular. Se a seqüência $\{X_0, X_1, \dots\}$ for a solução da equação de diferenças (4.3), então*

$$\lim_{k \rightarrow \infty} X_k (X_{k-1})^{-1} = S_1. \quad (4.7)$$

PROVA. O Teorema II-8.1 mostra que se S_1, \dots, S_m for um conjunto completo de solventes de $P(X)$, toda solução $\{X_0, X_1, \dots\}$ da equação de diferenças (4.3) satisfaz

$$X_k = \sum_{i=1}^m S_i^k \Omega_i. \quad (4.8)$$

Combinando (4.8) com as condições iniciais

$$X_0 = X_1 = \dots = X_{m-2} = 0 \quad \text{e} \quad X_{m-1} = I,$$

obtém-se que as matrizes $\Omega_1, \dots, \Omega_m$ são as soluções do seguinte sistema linear:

$$V(S_1, \dots, S_m) \begin{pmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_m \end{pmatrix} = \begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{m-1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ I \end{pmatrix}. \quad (4.9)$$

Isolando a última linha de (4.9), tem-se que:

$$\begin{pmatrix} I \\ S_1 \\ \vdots \\ S_1^{m-2} \end{pmatrix} \Omega_1 + V(S_2, \dots, S_m) \begin{pmatrix} \Omega_2 \\ \Omega_3 \\ \vdots \\ \Omega_m \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (4.10)$$

e

$$S_1^{m-1} \Omega_1 + (S_2^{m-1}, \dots, S_m^{m-1}) \begin{pmatrix} \Omega_2 \\ \Omega_3 \\ \vdots \\ \Omega_m \end{pmatrix} = I. \quad (4.11)$$

Combinando as equações (4.10) e (4.11) com a inversibilidade de $V(S_2, \dots, S_m)$, tem-se:

$$\left[S_1^{m-1} - (S_2^{m-1}, \dots, S_m^{m-1}) V(S_2, \dots, S_m)^{-1} \begin{pmatrix} I \\ S_1 \\ \vdots \\ S_1^{m-2} \end{pmatrix} \right] \Omega_1 = I,$$

e portanto, Ω_1 é não singular. Seja $\{W_0, W_1, \dots\}$ a seqüência definida por:

$$W_k = X_k - S_1^k \Omega_1;$$

segue-se, do Lema 4.1, que:

$$\lim_{k \rightarrow \infty} W_k \Omega_1^{-1} S_1^{1-k} = 0.$$

Assim sendo, para k suficientemente grande, $I + W_k \Omega_1^{-1} S_1^{1-k}$ é inversível e

$$\lim_{k \rightarrow \infty} X_k (X_{k-1})^{-1} = S_1,$$

pois

$$\begin{aligned} X_k X_{k-1}^{-1} &= \left(\sum_{i=1}^m S_i^k \Omega_i \right) \left(\sum_{i=1}^m S_i^{k-1} \Omega_i \right)^{-1} \\ &= (S_1 + W_k \Omega_1^{-1} S_1^{1-k}) (I + W_{k-1} \Omega_1^{-1} S_1^{1-k})^{-1}. \quad \square \end{aligned}$$

O Teorema poderia ter sido provado com hipóteses menos restritivas, deixando de lado a condição sobre $V(S_2, \dots, S_m)$ e impondo condições iniciais de modo a garantir que a solução de (4.3) seja da forma

$$X_k = S_1^k \Omega_1 + \dots + S_m^k \Omega_m,$$

com Ω_1 não singular. É importante observar que a hipótese sobre $V(S_2, \dots, S_m)$ significa uma restrição considerável sobre o conjunto de solventes, como mostra o Exemplo II-2.2.

Uma prova semelhante à do Teorema 4.1 mostra que a seqüência $\{(X_{k-1})^{-1}X_k; k = 1, 2, \dots\}$, converge, mas não para S_1 e sim para $\Omega_1^{-1}S_1\Omega_1$.

5. Cálculo do solvente dominante II. Algoritmo de Dennis, Traub e Weber

Nesta seção, estuda-se outro método para calcular o solvente dominante da equação

$$P(X) = X^m + A_1X^{m-1} + \dots + A_m = 0, \quad (5.1)$$

o qual é uma generalização do algoritmo de Traub [40], empregado para calcular o zero de maior valor absoluto de um polinômio escalar.

Algoritmo 5.1 ([12]). Considere o polinômio matricial $P(X)$ definido por (5.1) e as seguintes etapas:

- (1) Dado $L > 0$ e $G^{(0)}(X) = X^{m-1}$, define-se $G^{(k)}(X)$ para $k = 0, \dots, L-1$ da seguinte maneira:

$$G^{(k+1)}(X) = G^{(k)}(X)X - \Gamma_0^{(k)}P(X),$$

onde

$$G^{(k)}(X) = \sum_{l=0}^{m-1} \Gamma_{m-1-l}^{(k)} X^l.$$

- (2) Sejam $\Phi^{(L)}(X)$ a função definida, onde for possível, por:

$$\Phi^{(L)}(X) = G^{(L)}(X)G^{(L-1)}(X)^{-1}, \quad (5.2)$$

e

$$X_0 = \Gamma_0^{(L)}(\Gamma_0^{(L-1)})^{-1}.$$

Neste caso, gera-se a seqüência $\{X_1, X_2, \dots\}$ por:

$$X_{i+1} = \Phi^{(L)}(X_i). \quad (5.3)$$

As condições para a convergência do método são muito fortes; o Teorema 5.1 não só exige a existência de um solvente dominante, como também a necessidade de que o solvente dominante forme parte de um conjunto completo de solventes (ver Seção II-6). A maior vantagem deste algoritmo iterativo é a de fornecer uma aproximação inicial, o que não acontece em geral com os métodos iterativos.

Teorema 5.1 ([12]). *Seja $P(X)$ um polinômio matricial de grau m . Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ tal que S_1 é um solvente dominante e a matriz $V(S_2, \dots, S_m)$ é não singular. Se os polinômios $\overline{G}^{(L)}(X)$ e $\overline{M}_1(X)$ forem definidos por:*

$$\overline{G}^{(L)}(X) = (\Gamma_0^{(L)})^{-1} G^{(L)}(X),$$

e

$$\overline{M}_1(X) = (A_0^{(1)})^{-1} M_1(X),$$

então

- (1) $\lim_{L \rightarrow \infty} \overline{G}^{(L)}(X) = \overline{M}_1(X)$,
- (2) para L suficientemente grande, $\lim_{i \rightarrow \infty} X_i = S_1$.

Por motivos didáticos, a demonstração do Teorema foi dividida em várias partes. O roteiro da prova é o seguinte:

- Aplicando as propriedades dos polinômios matriciais fundamentais, define-se um conjunto compacto \mathcal{B} , de interior não vazio.
- Demonstra-se que para L suficientemente grande, a função $\Phi^{(L)}$ está definida no conjunto \mathcal{B} .
- Mostra-se que se $X_0 \in \mathcal{B}$, a segunda etapa do algoritmo converge.
- Verifica-se que se a primeira etapa for executada para L suficientemente grande, então a aproximação inicial X_0 está em \mathcal{B} .

Até o fim da seção, e por causa do Lema 4.1, emprega-se a norma matricial induzida pela norma vetorial do máximo $\| \cdot \|_{\infty}$, mas para qualquer outra norma matricial os resultados continuam válidos.

O Lema 5.1 mostra que todo solvente inversível de $P(X)$ é um ponto fixo de $\Phi^{(L)}(X)$.

Lema 5.1. *Se S for um solvente não singular de $P(X)$ e L for um inteiro positivo fixo, então*

$$\Phi^{(L)}(S) = S.$$

PROVA. A demonstração consiste em observar que, pela definição de $G^{(k)}(X)$, qualquer solvente de $P(X)$ satisfaz

$$G^{(k)}(S) = S^k \quad (k = 1, 2, \dots). \quad (5.4)$$

Como S é inversível, $\Phi^{(L)}(S)$ está definida e satisfaz

$$\Phi^{(L)}(S) = S^L S^{-L+1} = S.$$

Assim sendo, o Lema fica demonstrado. \square

No que resta desta seção, denota-se por $M_1(X), \dots, M_m(X)$ aos polinômios matriciais fundamentais associados a S_1, \dots, S_m (ver Seção II-6).

Nos seguintes lemas, as hipóteses foram restringidas ao mínimo, ainda que eles só serão empregados na demonstração do Teorema 5.1, no intuito de pôr em evidência onde cada hipótese é empregada.

O Lema 5.2 mostra que o polinômio $M_1(X)$ é inversível numa vizinhança da matriz dominante S_1 .

Lema 5.2. *Se S_1, \dots, S_m for um conjunto completo, então existe uma bola compacta \mathcal{B} de interior não vazio, centrada em S_1 , tal que para todo $X \in \mathcal{B}$ tem-se:*

- (1) $\|I - M_1(X)\|_\infty \leq \kappa < 1$,
- (2) para $j \neq 1$, $\|M_j(X)\|_\infty \leq 1$,
- (3) $M_1(X)$ é inversível, e

$$\|M_1(X)^{-1}\|_\infty \leq \frac{1}{1 - \|I - M_1(X)\|_\infty}.$$

PROVA. As afirmações (1) e (2) decorrem da definição de polinômios matriciais fundamentais e da sua continuidade.

Para cada $X \in \mathcal{B}$, tem-se $\|I - M_1(X)\|_F \leq \kappa < 1$ e, portanto, o raio espectral de $I - M_1(X)$ é menor do que um. Assim sendo,

$$M_1(X)^{-1} = \lim_{k \rightarrow \infty} \sum_{i=0}^k (I - M_1(X))^i,$$

e portanto,

$$\|M_1(X)^{-1}\|_\infty \leq \frac{1}{1 - \|I - M_1(X)\|_\infty}. \quad \square$$

Observe-se que neste lema não foram empregadas nem a inversibilidade da matriz $V(S_2, \dots, S_m)$, nem a dominância do solvente S_1 .

O Lema 5.3 mostra que $\Phi^{(L)}(X)$ está definida numa vizinhança de S_1 .

Lema 5.3. *Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ tal que S_1 é um solvente dominante. Se \mathcal{B} for o conjunto compacto fornecido pelo Lema 5.2, então existe um número L' tal que para todo $X \in \mathcal{B}$, a função $\Phi^{(L)}(X)$ de (5.2), está definida para todo $L \geq L'$.*

PROVA. Para $X \in \mathcal{B}$, define-se $W_L(X)$ por:

$$W_L(X) = \sum_{j=2}^m S_1^{-L} S_j^L V_j(X),$$

onde

$$V_j(X) = M_j(X) M_1(X)^{-1}.$$

Segue-se, do Teorema II-6.3, que:

$$G^{(L-1)}(X) = \sum_{j=1}^m G^{(L-1)}(S_j) M_j(X),$$

e portanto, da definição de $W_L(X)$ e de (5.4) tem-se:

$$\begin{aligned} G^{(L-1)}(X) &= \sum_{j=1}^m S_j^{L-1} M_j(X) \\ &= S_1^{L-1} \left(I + \sum_{j=2}^m S_1^{-L+1} S_j^{L-1} V_j(X) \right) M_1(X) \\ &= S_1^{L-1} (I + W_{L-1}(X)) M_1(X). \end{aligned} \tag{5.5}$$

Note-se que $\lim_{L \rightarrow \infty} W_L(X) = 0$ uniformemente em \mathcal{B} , pois do Lema 5.2 segue-se que:

$$\|V_j(X)\|_\infty \leq \tau \sigma^L \|M_j(X) M_1(X)^{-1}\|_\infty \leq \frac{1}{1 - \kappa} < \infty. \tag{5.6}$$

Assim sendo, para L suficientemente grande, $I + W_L(X)$ é inversível. Sendo S_1 uma matriz não singular e $M_1(X)$ inversível para todo $X \in \mathcal{B}$, da equação (5.5) segue-se que $G^{(L-1)}(X)$ é inversível para todo $X \in \mathcal{B}$ e, portanto, o lema fica demonstrado. \square

O seguinte lema é um passo intermediário na prova da convergência da segunda etapa do algoritmo, quando X_0 estiver em \mathcal{B} .

Lema 5.4. *Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ tal que S_1 é dominante e seja \mathcal{B} o conjunto compacto fornecido pelo Lema 5.2, então para todo $X \in \mathcal{B}$ e para $j \neq 1$,*

$$\|S_j^L V_j(X) S_1^{-L}\|_\infty \leq \tau \sigma^L \|M_j(X) M_1^{-1}(X)\|_\infty \leq \frac{\tau \sigma^L}{1 - \kappa}, \quad (5.7)$$

onde $0 \leq \sigma < 1$, e τ é uma constante que não depende de L nem de X .

PROVA. Em virtude do Lema 4.1,

$$\|S_j^L V_j(X) S_1^{-L}\|_\infty \leq \tau \sigma_j^L \|M_j(X) M_1^{-1}(X)\|_\infty,$$

onde, para $j \neq 1$,

$$\sigma_j = \left(\epsilon + \max \{ |\lambda_{S_j}| \} \right) \left(\epsilon + \frac{1}{\min \{ |\lambda_{S_1}| \}} \right).$$

Escolhendo $\epsilon > 0$ tal que

$$\sigma = \min_{i=2, \dots, m} \{ \sigma_i \} < 1$$

e usando (5.6) obtém-se o resultado. \square

O Lema 5.5 mostra que a etapa 2 do Algoritmo 5.1 converge se X_0 estiver suficientemente próximo de S_1 .

Lema 5.5. *Seja S_1, \dots, S_m um conjunto completo de solventes de $P(X)$ tal que S_1 é um solvente dominante e seja \mathcal{B} o compacto fornecido pelo Lema 5.2. Se $X_0 \in \mathcal{B}$ e L for suficientemente grande, então a seqüência $\{X_0, X_1, \dots\}$ gerada pelo Algoritmo 5.1 converge para S_1 .*

PROVA. Seja L' como definido no Lema 5.3 e seja $L \geq L'$. Para cada $X \in \mathcal{B}$ define-se $E^{(L)}(X)$ por:

$$E^{(L)}(X) = \Phi^{(L)}(X) - S_1.$$

Em virtude da definição de $V_j(X)$ e da equação (5.5), tem-se que:

$$\begin{aligned} \Phi^{(L)}(X) &= G^{(L)}(X) \left(G^{(L-1)}(X) \right)^{-1} \\ &= \left(\sum_{j=1}^m S_j^L V_j(X) \right) \left(\sum_{j=1}^m S_j^{L-1} V_j(X) \right)^{-1}, \end{aligned} \quad (5.8)$$

e portanto,

$$E^{(L)}(X) \sum_{j=1}^m S_j^{L-1} V_j(X) = \sum_{j=2}^m (S_j - S_1) S_j^{L-1} V_j(X).$$

Se $T_j^{(L)}(X)$ for definido por:

$$T_j^{(L)}(X) = S_j^{L-1} V_j(X) S_1^{1-L},$$

em consequência do Lema 5.4 obtém-se que:

$$\|T_j^{(L)}(X)\|_{\infty} \leq \frac{\tau \sigma^{L-1}}{1 - \kappa}.$$

Como $\sigma < 1$, pode-se escolher L suficientemente grande de maneira a obter que para todo $X \in \mathcal{B}$,

$$\sum_{j=2}^m \|T_j^{(L)}(X)\|_{\infty} \leq \gamma < 1.$$

Desde que

$$E^{(L)}(X) \left[I + \sum_{j=2}^m T_j^{(L)}(X) \right] = \sum_{j=2}^m (S_j - S_1) T_j^{(L)}(X),$$

e em consequência da estimativa (5.7) segue-se, para todo $X \in \mathcal{B}$, que:

$$\|E^{(L)}(X)\|_{\infty} \leq \frac{\tau \sigma^{L-1}}{1 - \gamma} \sum_{j=2}^m \|S_j - S_1\|_{\infty} \|M_j(X)\|_{\infty} \|M_1^{-1}(X)\|_{\infty}.$$

Levando-se em consideração a continuidade dos polinômios matriciais e a compacidade de \mathcal{B} , o Teorema II-3.1 mostra que:

$$\|M_j(X)\|_{\infty} \leq \delta \|X - S_1\|_{\infty},$$

para todo $X \in \mathcal{B}$ e para $j \neq 1$, onde δ não depende de X . Finalmente, para todo $X \in \mathcal{B}$ tem-se:

$$\|\Phi^{(L)}(X) - S_1\|_{\infty} \leq \alpha \sigma^{L-1} \|X - S_1\|_{\infty}, \quad (5.9)$$

com

$$\alpha = \frac{\tau \delta}{(1 - \gamma)(1 - \kappa)} \sum_{j=2}^m \|S_j - S_1\|_{\infty}.$$

Como α não depende de L , pode-se escolher L suficientemente grande para que $\alpha \sigma^{L-1} < 1$; assim sendo, a desigualdade (5.9) mostra que a seqüência $\{X_i ; i = 1, 2, \dots\}$, definida por $X_{i+1} = \Phi^{(L)}(X_i)$, converge para S_1 . \square

PROVA DO TEOREMA 5.1.

PROVA DE (1). Uma vez que a matriz $V(S_2, \dots, S_m)$ é inversível, segue-se, do Teorema II-6.2, que a matriz $A_0^{(1)}$ é também inversível e, portanto, $\overline{M}_1(X)$ está definido em \mathcal{B} . Analogamente à prova do Lema 5.3, define-se

$$V_j = A_0^{(j)} \left(A_0^{(1)} \right)^{-1} \quad \text{e} \quad W_L = \sum_{j=2}^m S_1^{-L} S_j^L V_j.$$

Note-se que a matriz V_j está definida e que existe $\kappa < \infty$ tal que, para todo $j > 1$,

$$\|V_j\|_\infty \leq \kappa.$$

Como $\|S_1^{-L} S_j^L\|_\infty \rightarrow 0$, então $\|W_L\|_\infty \rightarrow 0$, e assim, para L suficientemente grande $I - W_L$ é inversível. Igualando os coeficientes em (5.5) segue-se que:

$$\begin{aligned} \Gamma_0^{(L)} &= \sum_{j=1}^m S_j^L A_0^{(j)} \\ &= S_1^{L-1} \left(I + \sum_{j=2}^m S_1^{-L+1} S_j^{L-1} V_j \right) A_0^{(1)} \\ &= S_1^{L-1} (I + W_{L-1}) A_0^{(1)}, \end{aligned}$$

e portanto, para L suficientemente grande $\Gamma_0^{(L)}$ é inversível; logo, a função $\overline{G}^{(L)}(X)$ está definida. Em virtude do Lema 4.1 tem-se que:

$$\begin{aligned} \lim_{L \rightarrow \infty} (S_1^L)^{-1} \Gamma_0^{(L)} &= A_0^{(1)} + \lim_{L \rightarrow \infty} \sum_{j=2}^m S_1^{-L} S_j^L A_0^{(j)} \\ &= A_0^{(1)}, \end{aligned} \tag{5.10}$$

portanto:

$$\begin{aligned} \overline{G}^{(L)}(X) &= \left(\sum_{i=1}^m S_i^L A_0^{(i)} \right)^{-1} S_1^L S_1^{-L} \left(\sum_{i=1}^m S_i^L M_i(X) \right) \\ &= \left(\sum_{i=1}^m S_1^{-L} S_i^L A_0^{(i)} \right)^{-1} \left(\sum_{i=1}^m S_1^{-L} S_i^L M_i(X) \right) \\ &= \left(A_0^{(1)} + \sum_{i=2}^m S_1^{-L} S_i^L A_0^{(i)} \right)^{-1} \left(M_1(X) + \sum_{i=2}^m S_1^{-L} S_i^L M_i(X) \right). \end{aligned} \tag{5.11}$$

Como $\|S_1^{-L} S_j^L\|_\infty \rightarrow 0$, das equações (5.10) e (5.11) segue-se que:

$$\lim_{L \rightarrow \infty} \overline{G}^{(L)}(X) = \overline{M}_1(X). \quad \square$$

PROVA DE (2). Uma vez que $\Gamma_0^{(L)}$ é inversível,

$$E^{(L)} = \Gamma_0^{(L)} \left(\Gamma_0^{(L-1)} \right)^{-1}$$

está definido e satisfaz:

$$E^{(L)} \sum_{j=1}^m S_j^{L-1} V_j = \sum_{j=2}^m (S_j - S_1) S_j^{L-1} V_j.$$

Se $T_j^{(L)}$ for definido por:

$$T_j^{(L)} = S_j^{L-1} V_j S_1^{1-L},$$

o Lema 4.1 mostra que $\|T_j^{(L)}\|_\infty \rightarrow 0$, e assim, para L suficientemente grande tem-se:

$$\sum_{j=2}^m \|T_j^{(L)}\|_\infty \leq \gamma < 1,$$

e portanto, as matrizes $I - T_j^{(L)}$ são inversíveis. Desde que

$$E^{(L)} \left[I + \sum_{j=2}^m T_j^{(L)} \right] = \sum_{j=2}^m (S_j - S_1) T_j^{(L)},$$

obtém-se a seguinte estimativa para a norma infinito de $E^{(L)}$:

$$\|E^{(L)}\|_\infty \leq \frac{1}{1 - \gamma} \sum_{j=2}^m \|S_j - S_1\|_\infty \|T_j(X)\|_\infty,$$

logo, $\|E^{(L)}\|_\infty \rightarrow 0$. Desta maneira, para L suficientemente grande

$$\Gamma_0^{(L)} \left(\Gamma_0^{(L-1)} \right)^{-1} \in \mathcal{B}. \quad \square$$

A equação (5.9) fornece a velocidade de convergência da etapa 2 do Algoritmo 5.1, pois dela obtém-se o seguinte resultado.

Corolário 5.1. Para todo $X \in \mathcal{B}$, $\|\Phi^{(L)}(X) - S_1\|_\infty \leq c\sigma^{L-1} \|X - S_1\|_\infty$, onde $0 \leq \sigma < 1$ e c é uma constante que não depende de L .

Este corolário mostra que, embora a convergência da primeira etapa do algoritmo seja só linear, o erro assintótico pode ser feito tão pequeno quanto se quiser, aumentando o número de iterações na primeira etapa; de fato, a constante assintótica do erro desta etapa depende de

$$\frac{\max_{2 \leq j \leq m} \{|\lambda_{s_j}|\}}{\min \{|\lambda_{s_1}|\}}.$$

5.1. Considerações sobre a aplicação do método

O objetivo da segunda etapa do Algoritmo 5.1 é precisamente acelerar a convergência, no entanto, a primeira etapa poderia demandar muitas iterações até atingir um L satisfatório. A maior dificuldade na implementação do método é saber quando passar da primeira para a segunda etapa, principalmente porque se a etapa 1 for concluída para um L pequeno, a etapa 2 não converge. Assim sendo, neste caso, deve voltar-se à primeira etapa, onde os resultados da segunda não podem ser aproveitados. Porém, se L for muito grande, a velocidade de convergência fica sensivelmente reduzida.

O número de operações desenvolvidas para cada passo não é muito significativo. Para cada iteração da primeira etapa são necessários mn^3 flops para calcular $G^{(k+1)}(X)$ a partir de $G^{(k)}(X)$. Já a segunda etapa resulta um pouco mais trabalhosa, sendo preciso avaliar dois polinômios matriciais e resolver um sistema linear. Assim, se o método de Gauss for empregado para resolver o sistema linear, cada iteração da etapa 2 envolve $2mn^3 + \frac{n^3}{3}$ flops.

O espaço de memória requerida para a implementação do Algoritmo 5.1 é de $(2 + m)n^2$ palavras.

6. Solventes de polinômios com coeficientes triangulares

Nesta seção, desenvolve-se um método direto para resolver a equação

$$P(X) = \sum_{l=0}^m A_{m-l} X^l = 0,$$

quando todos os coeficientes $A_{m-l} = (a_{ij}^{(m-l)})_1^n$ forem matrizes triangulares superiores. O algoritmo fundamenta-se numa propriedade das matrizes triangulares a qual estabelece que: se $X = (x_{ij})_1^n$ for uma matriz triangular superior e $k > 1$,

ainda que X^k não seja linear em X , $(X^k)_{ij}$ é linear em x_{ij} (ver Seção A-2.1). O procedimento empregado para calcular um solvente de $P(X)$ é uma adaptação do algoritmo de Horner (ver Seção II-7).

Como o método a ser apresentado é de natureza extremamente prática apresentar-se-á através do Exemplo 6.1, deixando as considerações de natureza formal para depois. O seguinte exemplo, é um caso particular de uma das equações descritas no Exemplo IV-1.1.

Exemplo 6.1. Dado o polinômio matricial

$$P(X) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} X^2 + \begin{pmatrix} 2 & -2 & 0 \\ -2 & 4 & -2 \\ 0 & -1 & 1 \end{pmatrix} X + \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix},$$

pode-se mostrar que, se as matrizes U e V forem definidas por:

$$U = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & 0 & -\frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \end{pmatrix} \quad e \quad V = \begin{pmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix},$$

multiplicando os coeficientes de $P(X)$ à esquerda por V e à direita por U , e definindo $Y = U^* X U$, segue-se que a equação $P(X) = 0$ é equivalente a:

$$\begin{pmatrix} \sqrt{2} & 0 & -1 \\ 1 & 0 & 0 \\ & & \sqrt{2} \end{pmatrix} Y^2 + \begin{pmatrix} 0 & 0 & -2 \\ 2 & 0 & 0 \\ & & 4\sqrt{2} \end{pmatrix} Y + \begin{pmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ & & 2\sqrt{2} \end{pmatrix} = 0. \quad (6.2)$$

Se Y for a matriz triangular

$$\begin{pmatrix} y_{11} & y_{12} & y_{13} \\ & y_{22} & y_{23} \\ & & y_{33} \end{pmatrix},$$

reordenando os termos em (6.2), o primeiro membro fica:

$$\begin{pmatrix} \sqrt{2}y_{11}^2 & \sqrt{2}(y_{11} + y_{22})y_{12} & -1 + \sqrt{2}y_{11}y_{13} + \sqrt{2}y_{12}y_{23} + y_{33}(-2 + \sqrt{2}y_{13}) + y_{33}^2 \\ 0 & (1 + y_{22})^2 & y_{23}(2 + y_{22} + y_{33}) \\ 0 & 0 & \sqrt{2}(2 + 4y_{33} + y_{33}^2) \end{pmatrix}. \quad (6.3)$$

A equação (6.3) mostra que, quando Y for uma solução de (6.2), os elementos y_{ii} , para $i = 1, 2, 3$, devem satisfazer

$$\begin{aligned}\sqrt{2} y_{11}^2 &= 0, \\ (1 + y_{22})^2 &= 0, \\ (2 + 4y_{33} + y_{33}^2) &= 0,\end{aligned}\tag{6.4}$$

e, portanto, $y_{11} = 0$ e $y_{22} = -1$. Existem duas possibilidades para a escolha de y_{33} , $y_{33} = -2 + \sqrt{2}$ e $y_{33} = -2 - \sqrt{2}$. Se a escolha para y_{33} for $-2 + \sqrt{2}$, as equações que ficam na primeira supra diagonal de (6.3) são

$$\begin{aligned}-\sqrt{2} y_{12} &= 0, \\ (-1 + \sqrt{2}) y_{23} &= 0.\end{aligned}\tag{6.5}$$

Calculados os elementos y_{12} e y_{23} , resta somente determinar o valor de y_{13} , o qual deve satisfazer a equação

$$(2 - 2\sqrt{2}) y_{13} = 3 - 2\sqrt{2}.\tag{6.6}$$

Observe-se que as equações (6.4), (6.5) e (6.6) puderam ser resolvidas, pois dependiam de um único parâmetro.

O seguinte algoritmo é a generalização das etapas desenvolvidas no Exemplo 6.1 para calcular solventes de um polinômio com coeficientes triangulares superiores.

Algoritmo 6.1. Se z_1, \dots, z_n forem números complexos, então define-se uma matriz triangular superior $S = (s_{ij})_1^n$ por:

$$\begin{array}{l} \text{Para } i = 1, \dots, n \\ \quad \left| \begin{array}{l} s_{ii} = z_i \\ y_{ii}^{(0)} = a_{ii}^{(0)} \\ \text{Para } k = 1, \dots, m \\ \quad \left| y_{ii}^{(k)} = y_{ii}^{(k-1)} s_{ii} + a_{ii}^{(k)} \end{array} \right. \\ \text{Para } h = 1, \dots, n-1 \\ \quad \left| \begin{array}{l} \text{Para } i = 1, \dots, n-h \\ \quad \left| \begin{array}{l} \text{Definir } j = i+h \\ \text{Calcular } c_{ij} = c_{ij}^{(m)} \text{ por} \\ \quad \left| c_{ij}^{(0)} = 0 \end{array} \right. \end{array} \right. \end{array} \right.$$

$$\begin{array}{|l}
 \text{Para } k = 1, \dots, m \\
 \quad | \quad c_{ij}^{(k)} = y_{ii}^{(k-1)} + c_{ij}^{(k-1)} s_{jj} \\
 \text{Calcular } d_{ij} = d_{ij}^{(m)} \text{ por} \\
 \quad | \quad d_{ij}^{(0)} = a_{ij}^{(0)} \\
 \quad | \quad \text{Para } k = 1, \dots, m \\
 \quad \quad | \quad d_{ij}^{(k)} = d_{ij}^{(k-1)} s_{jj} + \sum_{l=i+1}^{j-1} y_{il}^{(k-1)} s_{lj} + a_{ij}^{(k)} \\
 s_{ij} = -d_{ij}/c_{ij} \\
 \text{Para } k = 1, \dots, m-1 \\
 \quad | \quad y_{ij}^{(k)} = d_{ij}^{(k)} + s_{ij} c_{ij}^{(k)}.
 \end{array}$$

Se todos os coeficientes do polinômio matricial $P(X)$ forem matrizes diagonais, todos os números c_{ij} são não nulos, assim sendo, neste caso sempre existe um solvente diagonal. No caso geral, a existência de solventes dependerá de que os coeficientes c_{ij} não se anulem.

As condições nas quais a matriz S , construída pelo algoritmo acima, é um solvente de $P(X)$ são fornecidas pelo Teorema 6.2.

O Corolário A-2.1 estabelece que se $X = (x_{ij})_1^n$ for uma matriz triangular superior e se $1 < r \in \mathbb{N}$, então X^r é uma matriz triangular superior. Além disso, $(X^r)_{ij}$ é linear em x_{ij} e depende no máximo dos x_{kl} mais próximos da diagonal principal do que x_{ij} , isto é, dos x_{kl} tais que $l - k < j - i$. A partir deste resultado pode-se provar o seguinte teorema:

Teorema 6.1. *Seja $P(X)$ o polinômio matricial definido em (6.1). Se $W = (w_{ij})_1^n$ for uma matriz triangular, então $P(W)$ é triangular e $P(W)_{ij}$ depende no máximo de w_{ij} e dos coeficientes w_{kl} mais próximos da diagonal principal do que w_{ij} . Além disso, para $i < j$, $P(W)_{ij}$ é linear em w_{ij} .*

PROVA. Como

$$(A^{(m-k)} W^k)_{ij} = \sum_{r=i}^j a_{ir}^{(m-k)} (W^k)_{rj}, \quad (6.7)$$

o elemento $P(W)_{ij}$ é dado por:

$$P(W)_{ij} = \sum_{k=0}^m a_{ii}^{(m-k)} (W^k)_{ij} + \sum_{k=0}^m \sum_{r=i+1}^j a_{ir}^{(m-k)} (W^k)_{rj}. \quad (6.8)$$

Sendo que, em (6.7), o único termo que envolve w_{ij} é $A_{ii}^{(m-k)} (W^k)_{ij}$ e como para $i < j$, $(W^k)_{ij}$ é linear em w_{ij} , o teorema fica demonstrado. \square

Para cada $i = 1, \dots, n$, seja $p_i(z)$ o polinômio escalar definido, a partir do polinômio $P(X)$, por:

$$p_i(z) = \sum_{l=0}^m a_{ii}^{(m-l)} z^l. \quad (6.9)$$

Com esta definição, para qualquer matriz triangular superior W , o resultado do Teorema 6.1 pode ser reescrito como:

$$P(W) = \begin{pmatrix} p_1(w_{11}) & c_{12}w_{12} + d_{12} & \cdots & c_{1n}w_{1n} + d_{1n} \\ & p_2(w_{22}) & & c_{2n}w_{2n} + d_{2n} \\ & & \ddots & \vdots \\ & & & p_n(w_{nn}) \end{pmatrix},$$

onde c_{ij} e d_{ij} dependem somente de elementos de W mais próximos da diagonal principal do que w_{ij} . Logo, para obter um solvente de $P(X)$ é suficiente, mas não necessário, que para todo $i < j$,

$$c_{ij} \neq 0. \quad (6.10)$$

Se a condição (6.10) for verdadeira para todo $i < j$, empregando o Algoritmo 6.1, é possível calcular todos os elementos de um solvente, uma supra diagonal de cada vez. Assim sendo, é muito importante estabelecer uma caracterização das constantes c_{ij} , com o intuito de determinar em que condições (6.10) é satisfeita.

Seja S um solvente triangular superior de $P(X)$ e, para cada $i = 1, \dots, n$, sejam $z_i^{(1)}, \dots, z_i^{(m)}$ os zeros do polinômio escalar $p_i(z)$. Como o espectro de qualquer solvente de $P(X)$ está contido no conjunto das raízes latentes de $P(\lambda)$ (ver Corolário II-2.1), s_{ii} deve ser uma raiz de $p_i(z)$, isto é, para algum k

$$s_{ii} = z_i = z_i^{(k)}. \quad (6.11)$$

O passo seguinte é caracterizar os c_{ij} em função dos elementos da diagonal principal de S . Se $1 < r \in \mathbb{N}$ e $x, y \in \mathbb{C}$, define-se $h^{(r)}(x, y)$ por:

$$h^{(r)}(x, y) = \begin{cases} \frac{x^r - y^r}{x - y} & \text{if } x \neq y \\ rx^{r-1} & \text{if } x = y \end{cases}.$$

Se W for uma matriz triangular superior, segue-se, do Corolário A-2.1, que $(W^r)_{ij}$ é linear em w_{ij} , isto é, $(W^r)_{ij} = \alpha_{ij}w_{ij} + \beta_{ij}$. O Lema 6.1 caracteriza os c_{ij} em função dos elementos da diagonal principal de W .

Lema 6.1. Se $W = (w_{ij})_1^n$ for uma matriz triangular superior, então

$$\begin{aligned}(W^r)_{ii} &= w_{ii}^r, \\ (W^r)_{ij} &= h^{(r)}(w_{ii}, w_{jj})w_{ij} + \beta_{ij}.\end{aligned}$$

PROVA. Consiste na repetição da prova do Teorema A-2.1 usando a definição da função $h^{(r)}(x, y)$. Detalhes sobre o β_{ij} são fornecidos em [18]. \square

Para cada $i = 1, \dots, n$ seja z_i uma raiz do polinômio $p_i(z)$ definido em (6.9). Se $\tilde{p}_i(z)$ for definido por:

$$\tilde{p}_i(z) = \begin{cases} \frac{p_i(z)}{z - z_i} & \text{if } z \neq z_i \\ p'_i(z) & \text{if } z = z_i \end{cases}, \quad (6.13)$$

tem-se a seguinte caracterização dos coeficientes c_{ij} :

Lema 6.2. Se S for um solvente triangular superior de $P(X)$ tal que para todo i , $s_{ii} = z_i$ é um zero de $p_i(z)$, então para $i < j$

$$c_{ij} = \tilde{p}_i(z_j).$$

PROVA. O Lema 6.1 e a equação (6.8) mostram que:

$$P(S)_{ij} = \sum_{k=1}^m a_{ii}^{(m-k)} h^{(k)}(z_i, z_j) s_{ij} + d_{ij},$$

onde d_{ij} não depende de s_{ij} . Assim sendo,

$$c_{ij} = \sum_{k=1}^m a_{ii}^{(m-k)} h^{(k)}(z_i, z_j),$$

e portanto, conclue-se, da definição de $h^{(r)}(x, y)$, que $c_{ij} = \tilde{p}_i(z_j)$. \square

O seguinte teorema estabelece uma condição sob a qual o Algoritmo 6.1 fornece um solvente de $P(X)$.

Teorema 6.2. Seja $P(X)$ o polinômio matricial definido por (6.1) e sejam $p_1(z), \dots, p_n(z)$ os polinômios escalares definidos em (6.9). Se existirem escalares z_1, \dots, z_n satisfazendo

$$\begin{aligned}p_i(z_i) &= 0 && (i = 1, \dots, n) \\ \tilde{p}_i(z_j) &\neq 0 && (1 \leq i < j \leq n),\end{aligned}$$

então existe um solvente triangular superior de $P(X)$, $S = (s_{ij})_1^n$, tal que $s_{ii} = z_i$.

PROVA. Se S for uma matriz triangular superior tal que $s_{ii} = z_i$, o Teorema 6.1 mostra que $P(S)_{ii} = 0$. Por hipótese, tem-se que:

$$\tilde{p}_i(z_j) \neq 0$$

para todo $i < j$ e, portanto, segue-se, do Teorema 6.1, que é possível definir os elementos s_{ij} por:

$$s_{ij} = -\frac{d_{ij}}{c_{ij}},$$

desde que os coeficientes d_{ij} sejam conhecidos. Segundo o Teorema 6.1, os coeficientes d_{ij} para $j - i = 1$ só dependem dos elementos da diagonal de S e, portanto, os elementos $s_{i, i+1}$ podem ser calculados. Como este processo pode ser desenvolvido até esgotar as diagonais de S , o teorema fica demonstrado. \square

Observação 6.1. Se todos os polinômios escalares $p_1(z), \dots, p_n(z)$ tiverem uma raiz comum \tilde{z} com multiplicidade 1, então existe um solvente triangular de $P(X)$.

A seguinte proposição fornece uma condição suficiente para que os coeficientes c_{ij} sejam não nulos.

Proposição 6.1. Se $P(X)$ for o polinômio matricial definido por (6.1) e se S for um solvente triangular superior de $P(X)$ tal que, para $i = 1, \dots, n$, $s_{ii} = z_i$, então $\tilde{p}_i(z_j)$ é um autovalor de $P^*(S)$, para todo i e j .

PROVA. Dado o solvente S de $P(X)$, seja $Q(\lambda)$ a lambda matriz que satisfaz

$$P(\lambda) = Q(\lambda)(\lambda I - S).$$

A prova do Teorema II-3.4 mostra que os autovalores da matrizes $Q(z_1), \dots, Q(z_n)$ são autovalores da matriz $P^*(S)$, e do Corolário 2.1, segue-se que:

$$Q(\lambda) = \sum_{l=1}^m \lambda^{l-1} \sum_{k=l}^m A_{m-k} S^{k-l}. \quad (6.14)$$

Como todas as matrizes envolvidas em (6.14) são triangulares superiores, os autovalores de $Q(z_i)$ são:

$$\sum_{l=1}^m z_i^{l-1} \sum_{k=l}^m a_{jj}^{(m-k)} z_j^{k-l} = \sum_{k=1}^m a_{ii}^{(m-k)} h^{(k)}(z_i, z_j),$$

e portanto, o resultado fica demonstrado. \square

Observação 6.2. A proposição acima estabelece que a regularidade da derivada de Fréchet de $P(X)$ é uma condição suficiente para que $c_{ij} \neq 0$ quando $i < j$.

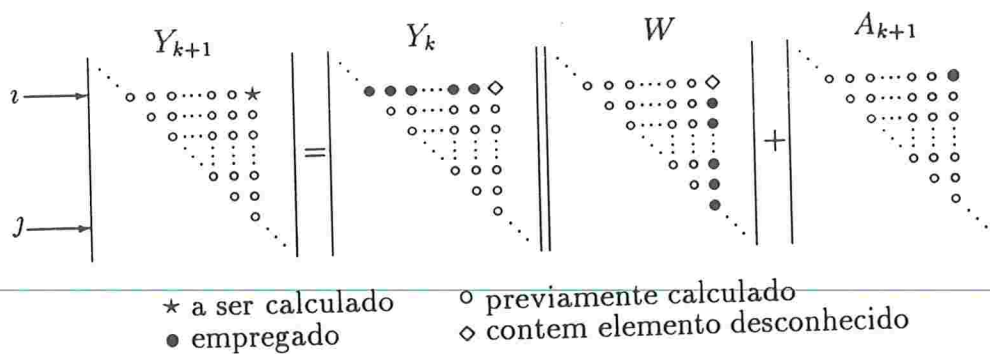


FIGURA 6.1. Elementos necessários para calcular $y_{ij}^{(k+1)}$

Uma vez que os coeficientes do polinômio matricial $P(X)$ são matrizes triangulares superiores, as operações no método de Horner podem ser reorganizadas para efetuar a avaliação de $P(X)$ por diagonais. O Algoritmo de Horner para avaliar o polinômio matricial $P(X)$ na matriz W é definido (ver Seção II-7) por:

$$\begin{aligned}
 Y_0 &= A_0 \\
 Y_k &= Y_{k-1}W + A_k \quad (k = 1, \dots, m).
 \end{aligned}$$

Em termos gráficos, cada etapa do Algoritmo II-7.2 pode ser descrito como mostra a Figura 6.1, isto é, o ij -ésimo elemento de Y_{k+1} depende do ij -ésimo elemento de A_{k+1} , da i -ésima linha de Y_k e da j -ésima coluna de W . Assim sendo, o Algoritmo 6.1 é uma forma de descrever o método de Horner, pois:

$$y_{ij}^{(k)} = \sum_{l=i}^j y_{il}^{(k-1)} w_{lj} + a_{ij}^{(k)}.$$

Como acima, sejam z_1, \dots, z_n os elementos diagonais de um solvente $S = (s_{ij})_1^n$, e seja $\tilde{S} = (\tilde{s}_{ij})_1^n$ a matriz triangular superior obtida empregando o Algoritmo 6.1 a partir dos números $\tilde{z}_1, \dots, \tilde{z}_n$, próximos dos elementos diagonais de S .

O seguinte teorema fornece uma estimativa para o erro nos coeficientes \tilde{s}_{ij} , em função dos erros $e_i = \tilde{z}_i - z_i$, para $i = 1, \dots, n$.

Teorema 6.3. *Seja $P(X)$ o polinômio matricial definido em (6.1), e sejam S e \tilde{S} as matrizes definidas acima, então para $i < j$, tem-se:*

$$|\tilde{s}_{ij} - s_{ij}| \leq \frac{\alpha}{\beta} \left(\gamma + |\tilde{s}_{ij}| \max_{1 \leq i < j \leq n} |\tilde{p}'_i(z_j)| + |\tilde{s}_{ij}| \frac{\max_{i=1, \dots, n} |p'_i(z_i)|}{\delta} \right) + O(\alpha^2),$$

onde os polinômios escalares $p_i(z)$ e $\tilde{p}_i(z)$ são os definidos em (6.9) e (6.13) respectivamente, e $p'_i(z)$ e $\tilde{p}'_i(z)$ são as suas derivadas e as constantes α , β , γ e δ são definidas por:

$$\alpha = \max_{i=1, \dots, n} |e_i|, \quad \beta = \min_{1 \leq i < j \leq n} |\tilde{p}_i(z_j)|,$$

$$\gamma = \max_{1 \leq i < j \leq n} |\tilde{p}'_i(z_j)| \quad e \quad \delta = \min_{1 \leq i < j \leq n} |z_i - z_j|.$$

PROVA. Sejam i e j dois inteiros fixos, tais que $i < j$, e sejam \tilde{d}_{ij} e \tilde{c}_{ij} as quantidades fornecidas pelo Algoritmo 6.1 a partir dos números $\tilde{z}_1, \dots, \tilde{z}_n$, então

$$s_{ij} - \tilde{s}_{ij} = \frac{\tilde{d}_{ij}}{\tilde{c}_{ij}} - \frac{d_{ij}}{c_{ij}} = \frac{\tilde{d}_{ij} - d_{ij}}{c_{ij}} + \frac{\tilde{d}_{ij}}{\tilde{c}_{ij}} \frac{c_{ij} - \tilde{c}_{ij}}{\tilde{c}_{ij}},$$

logo,

$$|s_{ij} - \tilde{s}_{ij}| \leq \frac{1}{\min_{1 \leq i < j \leq n} |c_{ij}|} \left(|\tilde{d}_{ij} - d_{ij}| + \tilde{s}_{ij} |c_{ij} - \tilde{c}_{ij}| \right).$$

Levando-se em conta o Lema 6.2 e a sua prova, tem-se que:

$$c_{ij} = \tilde{p}(z_j) \quad e \quad \tilde{c}_{ij} = \frac{p_i(\tilde{z}_j) - p_i(\tilde{z}_i)}{\tilde{z}_j - \tilde{z}_i},$$

e portanto,

$$|c_{ij} - \tilde{c}_{ij}| \leq \left(|\tilde{p}'_i(z_j)| + \frac{|p'_i(z_i)|}{\delta} \right) \alpha + O(\alpha^2).$$

Com o objetivo de simplificar o cálculo da estimativa de $|\tilde{d}_{ij} - d_{ij}|$, define-se para cada matriz $Y = (y_{ij})_1^n \in \mathbb{C}^{n \times n}$, a matriz $Y^{[i,j]} = (y_{kl}^{[i,j]})_1^{j-i+1}$ como sendo a matriz de dimensão $j - i + 1$ cujos elementos são

$$y_{kl}^{[i,j]} = y_{k+i \ l+i} \quad (k, l = 1, \dots, j - i + 1).$$

Sendo $Q(Z)$ o polinômio matricial definido por:

$$Q(Z) = \sum_{l=0}^m A_{m-l}^{[i,j]} Z^l,$$

do Teorema 7.1 segue-se que se S for um solvente de $P(X)$, então $S^{[i,j]}$ é um solvente de $Q(X)$.

Se $R = (r_{kl})_1^{j-i+1}$ e $\tilde{R} = (\tilde{r}_{kl})_1^{j-i+1}$ forem as matrizes cujos elementos satisfazem

$$r_{kl} = \begin{cases} 0 & \text{se } k = i \text{ e } l = j \\ s_{kl}^{[i,j]} & \text{outro caso} \end{cases} \quad \text{e} \quad \tilde{r}_{kl} = \begin{cases} 0 & \text{se } k = i \text{ e } l = j \\ \tilde{s}_{kl}^{[i,j]} & \text{outro caso} \end{cases},$$

pode-se mostrar que $Q(R)$ é uma matriz cujos elementos são todos nulos exceto o ij -ésimo, cujo valor é d_{ij} , enquanto que $Q(\tilde{R})_{ij} = \tilde{d}_{ij}$. Assim sendo, para qualquer norma matricial, tem-se:

$$|\tilde{d}_{ij} - d_{ij}| \leq \|Q(\tilde{R}) - Q(R)\| \leq \|P'(R)\| \|\tilde{R} - R\| + O(\|\tilde{R} - R\|^2),$$

e portanto, como a desigualdade acima também vale para o raio espectral, segue-se que

$$|\tilde{d}_{ij} - d_{ij}| \leq \gamma\alpha + O(\alpha^2).$$

Assim sendo, o resultado decorre juntando as estimativas de $|\tilde{d}_{ij} - d_{ij}|$ e de $|\tilde{c}_{ij} - c_{ij}|$. \square

Observe-se que se $\min_{1 \leq i < j \leq n} |\tilde{p}_i(z_j)|$ for muito pequeno, os zeros z_1, \dots, z_n têm que ser calculados de forma bem precisa. Porém, este fato não desqualifica o método apresentado, pois os números $\tilde{p}_i(z_j)$ são autovalores da derivada de Fréchet de $P(X)$ em S (ver Proposição 6.1), e como mostram os Teoremas II-3.2 e II-3.3, qualquer algoritmo pode apresentar problemas quando $P^*(S)$ for singular.

6.1. Considerações sobre a aplicabilidade do método

Uma vez descrito o Algoritmo, e caracterizadas as condições de convergência, é preciso analisar as condições para a sua utilização. Nas aplicações dificilmente surgem polinômios com coeficientes triangulares; no entanto, o algoritmo pode ser empregado desde que os coeficientes do polinômio possam ser triangularizados simultaneamente, seja por transformações de equivalência ou por transformações de similaridade. Uma análise sobre a triangularização simultânea de matrizes será desenvolvida no Capítulo IV.

O aspecto mais interessante do Algoritmo 6.1 é que a dificuldade de calcular solventes fica reduzida a calcular zeros de polinômios escalares. O número de operações envolvidas no cálculo de um solvente triangular, conhecidos os seus autovalores, é relativamente baixo já que é praticamente igual à quantidade de flops necessários para avaliar um polinômio matricial. A Tabela 6.1 descreve o número de operações em cada etapa do Algoritmo 6.1. Pode-se deduzir dela que, com o método proposto são realizados aproximadamente $\frac{1}{6}mn^3 + O(mn^2)$ flops.

	flops	divisões
c_{ij}	m	
d_{ij}	$(j - i)m$	
s_{ij}		1
$y_{ij}^{(k)}$	$m - 1$	
total	$(j - i + 2)m - 1$	1

TABELA 6.1. Operações requeridas para calcular s_{ij}

Para implementar o algoritmo é requerido um espaço de $\frac{3}{2}(m+1)(n+1)n$ palavras, porém, pode ser reduzido à ordem de $\frac{1}{2}(m+1)(n+1)n$ se a legibilidade do programa for deixada de lado.

Com a finalidade de se avaliar o algoritmo, efetuou-se a sua aplicação em numerosos exemplos-teste, os quais não foram aqui descritos, pois não representam uma contribuição ao entendimento do método. O mais importante a se destacar é o fato de que todos os solventes foram calculados com mais de 7 decimais corretos.

Alguns detalhes da implementação do Algoritmo 6.1 são analisados no Capítulo V.

6.2. Aplicação à previsão numérica do tempo

Na Seção II-1 mostrou-se que os modelos que provém das equações da água rasa podem ser resolvidas a partir do conhecimento das soluções de equações da forma

$$P_i(Y) = \Theta_i Y^2 + \Upsilon_i Y + \Omega_i = 0. \quad (6.15)$$

Empregando um manipulador algébrico pode-se mostrar que as matrizes $\Theta_i, \Upsilon_i, \Omega_i$ satisfazem as hipóteses do Teorema 6.2 e, portanto, o polinômio matricial $P_i(Y)$ tem um solvente triangular superior. A demonstração deste fato em geral, requer um esforço significativo que não acrescenta nada à compreensão dos fatos estudados. Assim, preferiu-se estudar a equação $\tilde{P}_i(Y) = T_0 Y^2 + T_1 Y + T_2 = 0$ próxima

de (1.10), quando as diferenças avançadas (ou centradas) forem empregadas para discretizar ∂_x .

Observe-se que, para h suficientemente pequeno, tem-se:

$$I + \Delta t \tilde{F}_i \approx A_0 = I + (1 - \omega^i) \Gamma \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ -\Phi & 0 & 0 \end{pmatrix},$$

$$2\Delta t \tilde{S}_i \approx A_1 = 2\Gamma U (1 - \omega^i) I$$

e

$$-I + \Delta t \tilde{F}_i \approx A_2 = -I + (1 - \omega^i) \Gamma \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ -\Phi & 0 & 0 \end{pmatrix},$$

onde $\Gamma = \Delta t h^{-1}$. Assim, definindo

$$V = \begin{pmatrix} 0 & \frac{1}{\sqrt{\Phi+1}} & \frac{\sqrt{\Phi}}{\sqrt{\Phi+1}} \\ 1 & 0 & 0 \\ 0 & \frac{\sqrt{\Phi}}{\sqrt{\Phi+1}} & -\frac{1}{\sqrt{\Phi+1}} \end{pmatrix}$$

segue-se que:

$$V^* A_0 V = T_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 - (1 - \omega^i) \Gamma \sqrt{\Phi} & (1 - \omega^i) \Gamma (1 - \Phi) \\ 0 & 0 & 1 + (1 - \omega^i) \Gamma \sqrt{\Phi} \end{pmatrix},$$

$$V^* A_1 V = T_1 = 2\Gamma U (1 - \omega^i) I$$

e

$$V^* A_2 V = T_2 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 - (1 - \omega^i) \Gamma \sqrt{\Phi} & (1 - \omega^i) \Gamma (1 - \Phi) \\ 0 & 0 & -1 + (1 - \omega^i) \Gamma \sqrt{\Phi} \end{pmatrix}.$$

Verifica-se que para $\Gamma = 0$, as raízes latentes de $\tilde{P}(Y)$ são as soluções de $\lambda^2 - 1 = 0$. Assim sendo, pela continuidade, para Γ suficientemente pequeno, os

polinômios escalares

$$p_1(z) = z^2 + 2\Gamma U (1 - \omega^i) z - 1,$$

$$p_2(z) = (1 - (1 - \omega^i) \Gamma \sqrt{\Phi}) z^2 + 2\Gamma U (1 - \omega^i) z - 1 - (1 - \omega^i) \Gamma \sqrt{\Phi}$$

e

$$p_3(z) = (1 + (1 - \omega^i) \Gamma \sqrt{\Phi}) z^2 + 2\Gamma U (1 - \omega^i) z - 1 + (1 - \omega^i) \Gamma \sqrt{\Phi}$$

tem raízes diferentes. Do Teorema III-6.2 segue-se que existe um solvente triangular superior de $\tilde{P}(Y)$. Como existe um fator espectral à direita de $\tilde{P}(\lambda)$, $\lambda I - S$, tem-se que $\tilde{P}^*(S)$ é não singular. Assim sendo, para Γ próximo de 0, sempre existe um solvente triangular superior de $P_i(Y)$.

A hipótese $\Gamma = 0$ não tem valor prático, porém, é de grande interesse analítico uma vez que podem ser calculados os solventes. Uma análise mais detalhada permite mostrar para quais valores de Γ existem solventes triangulares superiores, observe-se que os zeros dos polinômios $p_1(z)$, $p_2(z)$ e $p_3(z)$ são dados por:

$$\begin{aligned} \beta_1^+ &= \Gamma U (\omega^i - 1) + \sqrt{(\Gamma U (\omega^i - 1))^2 + 1} + O(h^{\frac{1}{3}}) \\ \beta_1^- &= \Gamma U (\omega^i - 1) - \sqrt{(\Gamma U (\omega^i - 1))^2 + 1} + O(h^{\frac{1}{3}}) \\ \beta_2^+ &= \frac{U (\omega^i - 1) + \sqrt{(U (\omega^i - 1))^2 + \Gamma^{-2} - \tilde{a}^2 \Phi}}{\Gamma^{-1} + \sqrt{\tilde{a}^2 \Phi}} + O(h^{\frac{1}{3}}) \\ \beta_2^- &= \frac{U (\omega^i - 1) - \sqrt{(U (\omega^i - 1))^2 + \Gamma^{-2} - \tilde{a}^2 \Phi}}{\Gamma^{-1} + \sqrt{\tilde{a}^2 \Phi}} + O(h^{\frac{1}{3}}) \\ \beta_3^+ &= \frac{U (\omega^i - 1) + \sqrt{(U (\omega^i - 1))^2 + \Gamma^{-2} - \tilde{a}^2 \Phi}}{\Gamma^{-1} - \sqrt{\tilde{a}^2 \Phi}} + O(h^{\frac{1}{3}}) \\ \beta_3^- &= \frac{U (\omega^i - 1) - \sqrt{(U (\omega^i - 1))^2 + \Gamma^{-2} - \tilde{a}^2 \Phi}}{\Gamma^{-1} - \sqrt{\tilde{a}^2 \Phi}} + O(h^{\frac{1}{3}}), \end{aligned} \quad (6.16)$$

onde β_j são as raízes de $p_j(z)$ e \tilde{a} é igual a $\frac{\omega^i - \omega^{-i}}{2}$ ou $\omega^i - 1$ segundo se trate de diferenças centradas ou avançadas. Como consequência do Teorema III-6. segue-se que, o solvente triangular existirá se Γ satisfizer que

$$\begin{aligned} \gamma_1 &= (\Gamma U (\omega^i - 1))^2 + 1 \neq 0 \\ \gamma_2 &= (U (\omega^i - 1))^2 + \Gamma^{-2} - \tilde{a}^2 \Phi \neq 0. \end{aligned} \quad (6.17)$$

Observe-se que $|(\Gamma U (\omega^i - 1))^2 + 1|$ é não nulo para todo ω tal que $|\omega| = 1$. Quando $\tilde{a} = \omega^i - 1$, $\gamma_2 = 0$ se, e somente se,

$$\Gamma^2 = -\frac{1}{2} \frac{\omega^{-i}}{(\Phi - U^2)(1 - \operatorname{Re}(\omega^i))},$$

e, portanto, γ_2 é não nulo para todo Γ positivo desde que $\omega^i \neq -1$, em cujo caso γ_2 é zero se

$$\Gamma^2 = -\frac{1}{4(U^2 - \Phi)}.$$

Se $\tilde{a} = \frac{\omega - \omega^{-1}}{2}$, para todo Γ positivo e para todo ω tal que $|\omega| = 1$, tem-se que $\gamma_2 \neq 0$, pois a única solução de $\gamma_2 = 0$ é dada por:

$$\Gamma^2 = -\frac{1}{2U^2 \operatorname{Re}(\omega) (\operatorname{Re}(\omega) - 1) + \Phi \operatorname{Im}(\omega) + 2iU^2 \operatorname{Im}(\omega) (1 - \operatorname{Re}(\omega))}.$$

Resumindo, o Algoritmo III-6.1 fornece solventes triangulares para quase todo valor de Γ , somente fica excluído $\Gamma^2 = -\frac{1}{4(U^2 - \Phi)}$. Além disso, nestas condições, o algoritmo proposto fornece no mínimo um conjunto completo de solventes. Observe-se que neste caso os solventes são empregados para calcular um conjunto gerador do espaço de soluções da equação de diferenças, e que a existência de autovalores com módulo maior do que um está relacionada à escolha de h e Δt e não à utilização de equações polinomiais matriciais como forma de resolver a equação de diferenças. Por motivos técnicos, neste caso, os solventes não foram empregados para estimar as raízes latentes do polinômio matricial, porém, isto pode ser feito se as constantes U e Φ forem conhecidas.

6.3. Cálculo da raiz quadrada

Nesta seção, apresenta-se uma aplicação do Algoritmo 6.1 ao cálculo da raiz quadrada de uma matriz $C \in \mathbb{C}^{n \times n}$. O Teorema A-1.1 mostra que existem uma matriz unitária U e uma matriz triangular superior T tais que

$$C = U^* T U.$$

Assim sendo, calcular a raiz quadrada da matriz C é equivalente a encontrar uma matriz triangular superior Y tal que

$$Y^2 - T = 0.$$

O seguinte algoritmo fornece uma raiz quadrada de T , desde que $\lambda = 0$ seja um autovalor de T com multiplicidade no máximo um [4].

Algoritmo 6.2.

```

Para  $i = 1, \dots, n$ 
   $s_{ii} = \sqrt{t_{ii}}$ 
Para  $i = 1, \dots, n$ 
  Para  $k = 1, \dots, n - 1$ 
    Definir  $j = i + k$ 
    Calcular  $y = t_{ij}$ 
    Para  $l = i + 1, \dots, j - 1$ 
       $y = y + s_{il}s_{lj}$ 
    Calcular  $s_{ij} = y (s_{ii} + s_{jj})^{-1}$ .
  
```

Observe-se que o Algoritmo 6.2 pode ser aplicado só quando $s_{ii} \neq -s_{jj}$. A existência de autovalores múltiplos não nulos não representa um impedimento para que a condição $s_{ii} + s_{jj} \neq 0$ seja satisfeita, pois se para algum $i \neq j$, $t_{ii} = t_{jj}$, é suficiente escolher $s_{ii} = s_{jj}$. Note-se que a condição $s_{ii} + s_{jj} \neq 0$ é idêntica àquela do Teorema 6.2 e representa a regularidade da derivada de Fréchet do polinômio

$$P(X) = X^2 - T.$$

O número de operações requeridas pelo Algoritmo 6.2 é aproximadamente igual ao necessário para multiplicar duas matrizes triangulares (da ordem de $\frac{1}{6}n^3$ flops), mais n raízes quadradas e $\frac{1}{2}n(n-1)$ divisões. Uma análise mais detalhada deste algoritmo é realizada por Björck e Hammarling [4], onde também pode-se encontrar uma descrição de um algoritmo para calcular uma raiz cúbica da matriz T .

Parlett [34] apresenta um método para calcular funções analíticas de matrizes; este algoritmo pode ser utilizado para calcular a raiz k -ésima de T , com aproximadamente $\frac{1}{3}n^3$ flops, porém, este método não pode ser estendido a polinômios matriciais em geral, pois falta a comutatividade, a qual é de importância fundamental na definição do algoritmo.

7. Polinômios com coeficientes triangulares por blocos

Nesta seção, desenvolve-se um algoritmo para resolver a equação

$$P(X) = \sum_{l=0}^m A_{m-l} X^l = 0, \quad (7.1)$$

quando todos os coeficientes $A_{m-l} = (A_{ij}^{(m-l)})_1^{n_1}$, forem matrizes triangulares superiores por blocos de dimensão n_2 .

O método aqui apresentado é uma generalização do Algoritmo 6.1, e como na Seção 6, o procedimento proposto é apresentado com o auxílio de um exemplo.

Exemplo 7.1. Considere a equação (ver Exemplo IV-1.1)

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} X^2 + \begin{pmatrix} 1 & -1 & 0 \\ -1 & 3 & -2 \\ 0 & -2 & 2 \end{pmatrix} X + \begin{pmatrix} 1 & -1 & 0 \\ -1 & 4 & -3 \\ 0 & -3 & 3 \end{pmatrix} = 0. \quad (7.2)$$

Pode-se mostrar que as matrizes acima não triangularizam simultaneamente, porém, se as matrizes U e V forem definidas por:

$$V = \begin{pmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 1 \end{pmatrix} \quad \text{e} \quad U = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & -1 & 1 \end{pmatrix},$$

multiplicando os coeficientes da equação (7.2) à esquerda por V e à direita por U , e definindo $Y = U^* X U$, segue-se que a equação acima é equivalente a:

$$P(Y) = \begin{pmatrix} 6 & 0 & -6 \\ 0 & 2 & 0 \\ 0 & 0 & 6 \end{pmatrix} Y^2 + \begin{pmatrix} 0 & 1 & -9 \\ 0 & 3 & -3 \\ 0 & -2 & 18 \end{pmatrix} Y + \begin{pmatrix} 0 & 2 & -12 \\ 0 & 4 & -6 \\ 0 & -4 & 24 \end{pmatrix} = 0.$$

É óbvio que um solvente $S = (s_{ij})_1^3$ com a mesma estrutura de blocos dos coeficientes da equação acima, isto é, $s_{21} = s_{31} = 0$, precisa satisfazer $6 s_{11}^2 = 0$ e que a matriz S_1 definida por:

$$S_1 = \begin{pmatrix} s_{22} & s_{23} \\ s_{32} & s_{33} \end{pmatrix},$$

seja solvente do polinômio matricial

$$P_1(Y_1) = \begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix} Y_1^2 + \begin{pmatrix} 3 & -3 \\ -2 & 18 \end{pmatrix} Y_1 + \begin{pmatrix} 4 & -6 \\ -4 & 24 \end{pmatrix}.$$

Como mostra o Exemplo 1.1, a matriz

$$S_1 = \begin{pmatrix} -0.75 + 1.10103 i & 0.637386 - 0.519062 i \\ 0.191692 - 0.115347 i & 1.5 + 1.26485 i \end{pmatrix}$$

é solvente de $P_1(Y_1)$. Substituindo-se em S , $s_{11} = 0$ e os valores obtidos para S_1 , segue-se que S é solvente de $P(X)$ se, e somente se, os elementos s_{12} e s_{13} satisfizerem o seguinte sistema linear:

$$\begin{pmatrix} -4.5 + 6.6018 i & 1.1505 - 0.692082 i \\ 3.82432 - 3.11437 i & -9.0 + 7.5891 i \end{pmatrix} \begin{pmatrix} s_{12} \\ s_{13} \end{pmatrix} = \begin{pmatrix} 0.475229 - 2.13916 i \\ -2.13739 + 11.9027 i \end{pmatrix}.$$

Como a matriz de coeficientes do sistema acima é inversível, o sistema tem uma única solução

$$\begin{pmatrix} s_{12} \\ s_{13} \end{pmatrix} = \begin{pmatrix} 0.0958461 - 0.0576734 i \\ -0.75 + 0.632425 i \end{pmatrix}.$$

O Algoritmo 7.1 desenvolve os passos descritos no exemplo acima para qualquer polinômio matricial com coeficientes triangulares por blocos. Para simplificar a notação, o método é apresentado somente para polinômios cujos blocos sejam todos quadrados e da mesma dimensão.

Assim como o Algoritmo 6.1, o método definido nesta seção é uma aplicação do Algoritmo de Horner (ver Seção II-7) e do Teorema A-2.1.

Algoritmo 7.1. Se Z_1, \dots, Z_{n_1} forem matrizes de dimensão n_2 , então define-se uma matriz triangular superior por blocos $S = (S_{ij})_1^{n_1}$ por:

$$\begin{array}{l} \text{Para } i = 1, \dots, n_1 \\ \left| \begin{array}{l} S_{ii} = Z_i \\ Y_{ii}^{(0)} = A_{ii}^{(0)} \\ R_i^{(0)} = I \\ \text{Para } k = 1, \dots, m \\ \left| \begin{array}{l} Y_{ii}^{(k)} = Y_{ii}^{(k-1)} S_{ii} + A_{ii}^{(k)} \\ R_i^{(k)} = R_i^{(k-1)} S_{ii} \end{array} \right. \end{array} \right. \\ \text{Para } h = 1, \dots, n_1 - 1 \\ \left| \begin{array}{l} \text{Para } i = 1, \dots, n_1 - h \\ \left| \text{Definir } j = i + h \end{array} \right. \end{array} \right.$$

	Calcular C_{ij} por $C_{ij} = 0$ Para $k = 0, \dots, m - 1$ $C_{ij} = C_{ij} + (R_j^{(m-1-k)})^t \otimes Y_{ii}^{(k)}$
	Calcular $D_{ij} = D_{ij}^{(m)}$ por $D_{ij}^{(0)} = A_{ij}^{(0)}$ Para $k = 1, \dots, m$ $D_{ij}^{(k)} = D_{ij}^{(k-1)} S_{jj} + \sum_{l=i+1}^{j-1} Y_{il}^{(k-1)} S_{lj} + A_{ij}^{(k)}$
	$v(S_{ij}) = -C_{ij}^{-1} v(D_{ij})$ $Y_{ij}^{(0)} = A_{ij}^{(0)}$ $F_{ij} = A_{ij}^{(0)}$ Para $k = 1, \dots, m - 1$ $F_{ij} = Y_{ij}^{(k-1)} S_{ij} + F_{ij} S_{jj}$ $Y_{ij}^{(k)} = D_{ij}^{(k)} + F_{ij}$

O Teorema 7.2 fornece condições nas quais a matriz construída pelo algoritmo acima é um solvente de $P(X)$.

A prova da convergência do Algoritmo 7.1 decorre como aquela do Algoritmo 6.1. Na presença de coeficientes triangulares superiores por blocos, o Teorema 6.1 é substituído pelo seguinte resultado:

Teorema 7.1. *Seja $P(X)$ o polinômio matricial definido em (7.1). Nestas condições, se $W = (W_{ij})_1^{n_1}$ for uma matriz triangular por blocos, então $P(W)$ é triangular por blocos e o bloco $P(W)_{ij}$ só depende dos blocos de W mais próximos da diagonal principal do que W_{ij} . Além disso, para $i < j$, $P(W)_{ij}$ é linear em W_{ij} .*

PROVA. A demonstração é análoga àquela do Teorema 6.1. □

Para cada $i = 1, \dots, n_1$, seja $P_i(Z)$ o polinômio matricial de grau no máximo m e ordem n_2 definido, a partir do polinômio $P(X)$, por:

$$P_i(Z) = \sum_{l=0}^m A_{ii}^{(m-l)} Z^l. \tag{7.3}$$

Com esta definição, o Teorema 7.1 mostra que para uma matriz triangular superior

por blocos W :

$$P(W) = \begin{pmatrix} P_1(W_{11}) & P(W)_{12} & \cdots & P(W)_{1n_1} \\ & P_2(W_{22}) & & P(W)_{2n_1} \\ & & \ddots & \vdots \\ & & & P_{n_1}(W_{n_1n_1}) \end{pmatrix}, \quad (7.4)$$

onde $P(W)_{ij}$ satisfaz

$$v(P(W)_{ij}) = C_{ij}v(W_{ij}) + v(D_{ij}),$$

e as matrizes C_{ij} e D_{ij} dependem somente dos blocos de W mais próximos da diagonal principal do que W_{ij} . Logo, para obter-se um solvente de $P(X)$ é suficiente, mas não necessário, que para todo $i = 1, \dots, n_1$, $P_i(W_{ii}) = 0$ e, para $i < j$,

$$\det(C_{ij}) \neq 0. \quad (7.5)$$

Se a condição (7.5) for verdadeira para todo $i < j$, o Algoritmo 7.1 permite calcular todos os blocos de um solvente, uma supra diagonal de cada vez. Assim sendo, como no Algoritmo 6.1, é muito importante uma boa caracterização das matrizes C_{ij} .

Lema 7.1. *Se S for um solvente triangular superior por blocos de $P(X)$ tal que, para $i = 1, \dots, n_1$, a matriz S_{ii} é um solvente de $P_i(Z)$, então para $i < j$*

$$C_{ij} = \sum_{l=1}^m \left((S_{jj}^t)^{l-1} \otimes \sum_{k=l}^m A_{ii}^{(m-k)} S_{ii}^{k-l} \right).$$

PROVA. Considerando a Observação A-2.1, segue-se que para todo $l \geq 1$:

$$(S^l)_{ij} = \sum_{k=0}^{l-1} S_{ii}^k S_{ij} S_{jj}^{l-1-k} + H_{ij},$$

onde H_{ij} não depende de S_{ij} . Assim sendo,

$$P(S)_{ij} = \sum_{l=1}^m A_{ii}^{(m-l)} \sum_{k=0}^{l-1} S_{ii}^k S_{ij} S_{jj}^{l-1-k} + \tilde{H}_{ij},$$

onde \tilde{H}_{ij} não depende de S_{ij} , logo, aplicando a definição do produto de Kronecker, tem-se:

$$C_{ij} = \sum_{l=1}^m \left((S_{jj}^t)^{l-1} \otimes \sum_{k=l}^m A_{ii}^{(m-k)} S_{ii}^{k-l} \right). \quad \square$$

Para cada $i = 1, \dots, n_1$, define-se $R_i(X)$ por:

$$R_i(Z) = \sum_{l=1}^m \left((Z^l)^{l-1} \otimes \sum_{k=l}^m A_{ii}^{(m-k)} Z_i^{k-l} \right),$$

e os polinômios matriciais $Q_i(Z)$ tais que:

$$P_i(\lambda) = Q_i(\lambda) (\lambda I - Z_i).$$

Observação 7.1. O Teorema II-3.4 mostra que $R_i(Z_j)$ é uma matriz inversível se, e somente se, $\det(Q_i(\lambda_j)) \neq 0$ para todo λ_j autovalor de Z_j .

O seguinte teorema fornece uma condição suficiente para a existência de solventes triangulares por blocos.

Teorema 7.2. *Sejam $P(X)$ e $P_1(Z), \dots, P_{n_1}(Z)$ os polinômios matriciais definidos por (7.1) e (7.3) respectivamente. Se existirem matrizes $Z_1, \dots, Z_{n_1} \in \mathbb{C}^{n_2 \times n_2}$ que satisfazem*

$$\begin{aligned} P_i(Z_i) &= 0 & (i = 1, \dots, n_1) \\ R_i(Z_j) &\text{ é inversível,} & (1 \leq i < j \leq n_1), \end{aligned}$$

então, existe um solvente triangular superior por blocos de $P(X)$, $S = (S_{ij})_1^n$, tal que, para $i = 1, \dots, n_1$, $S_{ii} = Z_i$.

PROVA. A demonstração decorre do Teorema 7.1 e do Lema 7.1. \square

Corolário 7.1. *Sejam $P(X)$, $P_1(Z), \dots, P_{n_1}(Z)$ e $Q_1(Z), \dots, Q_{n_1}(Z)$ os polinômios matriciais definidos acima. Se existirem matrizes $Z_1, \dots, Z_{n_1} \in \mathbb{C}^{n_2 \times n_2}$ que satisfazem*

$$\begin{aligned} P_i(Z_i) &= 0 & (i = 1, \dots, n_1) \\ \det(Q_i(\lambda_j)) &\neq 0 & (1 \leq i < j \leq n_1), \end{aligned}$$

para todo autovalor λ_j de Z_j , então existe um solvente triangular superior por blocos de $P(X)$, $S = (S_{ij})_1^n$, tal que $S_{ii} = Z_i$.

Observe-se que as matrizes

$$\sum_{k=l}^m A_{ii}^{(m-k)} S_{ii}^{k-l},$$

empregadas na definição de $R_i(Z)$, são os resultados parciais obtidos na avaliação do polinômio matricial $P_i(Z)$ em S_{ii} pelo Algoritmo de Horner (Algoritmo II-7.2).

8. Conclusões

Uma comparação objetiva, de métodos de natureza tão diferente como os apresentados neste capítulo, é uma tarefa muito difícil. Porém, nesta seção, tenta-se resumir algumas considerações que se encontram ao longo de todo o capítulo, e que provêm do estudo cuidadoso dos algoritmos descritos.

O primeiro ponto a ser levantado, refere-se à existência de solventes de um polinômio matricial. Quase todos os métodos supõem a existência de um solvente, o que nem sempre é verdadeiro, como mostra-se na Seção II-2. Mesmo existindo um solvente, é muito difícil estimar uma região onde ele possa ser encontrado (ver Teorema 2.1), portanto, a maior parte dos métodos empregados para calcular solventes envolvem condições muito restritivas sobre os polinômios e sobre os solventes.

Estas condições são expressas das mais diversas formas:

- Existência de uma aproximação inicial suficientemente boa.
- Existência de um conjunto completo de solventes.
- Restrições sobre os coeficientes.

Os métodos podem ser agrupados em diretos ou iterativos, e estes últimos em dois subgrupos: os que fornecem uma aproximação inicial e os que não.

Os Algoritmos 4.1 e 5.1 proporcionam uma aproximação inicial restringindo o conjunto dos polinômios aos quais o método é aplicado. Esta restrição consiste em supor a existência de um conjunto completo de solventes.

Alguns exemplos do Capítulo II são dedicados aos conjuntos completos e às suas propriedades. Estes exemplos mostram que as condições de convergência dos Algoritmos 4.1 e 5.1 são muito restritivas. Mais prejudicial, que as restrições para a aplicação do método, é o fato de que a existência de conjuntos completos não pode ser determinada a priori. Além disso, estes métodos somente calculam o solvente dominante, diminuindo assim as suas possibilidades de aplicação, pois, como segue-se da Seção II-8, em geral são empregados solventes

com determinados autovalores ou, na sua ausência, conjuntos completos de solventes. É importante ressaltar que, no caso dos polinômios matriciais, a deflação não pode ser empregada (ver Observação II-5.2).

Os algoritmos de Newton e Newton modificado permitem calcular qualquer solvente desde que seja estável e a aproximação inicial seja suficientemente boa. A singularidade da derivada de Fréchet do polinômio matricial é uma dificuldade comum a todos os métodos e, portanto, não deve ser um fator decisivo na escolha do algoritmo.

Sendo que, por enquanto, não existem métodos para determinar regiões contendo um único solvente, a escolha da aproximação inicial representa o maior desafio a ser encarado na aplicação dos Algoritmos 2.1 e 3.1. Nos artigos de Kratz e Stickel [29] e de Davis [9] são sugeridas algumas possibilidades para a escolha da aproximação inicial, porém, não é feito um estudo para determinar se estas aproximações satisfazem as condições dos respectivos teoremas de convergência.

O terceiro grupo, o dos métodos diretos, apresenta uma dificuldade um pouco diferente, calcular raízes de polinômios escalares. O Algoritmo 1.1, ainda que de aplicação geral, é pouco prático quando a dimensão das matrizes envolvidas for grande, pois é preciso o conhecimento prévio dos autovalores do solvente procurado. Esta dificuldade não é decisiva na aplicação do Algoritmo 6.1, pois é preciso calcular zeros de polinômios escalares cujo grau é no máximo aquele do polinômio matricial, independente da dimensão dos coeficientes.

A principal vantagem do Algoritmo 6.1 é a de possibilitar o cálculo de qualquer solvente triangular superior estável. Por outro lado, o método permite conhecer, antes do cálculo efetivo do solvente, se o conjunto de raízes fornecidas corresponde ou não a um solvente simples. Mais ainda, é possível calcular uma estimativa, a priori, do erro com o qual será calculado o solvente.

A triangularização simultânea dos coeficientes é, de fato, uma restrição importante, porém, ela não deve ser supervalorizada, primeiro porque existem teoremas que caracterizam os conjuntos de matrizes simultaneamente triangularizáveis, e segundo porque os polinômios com coeficientes triangulares podem ser utilizados como aproximantes, e as suas soluções podem servir como uma aproximação inicial para o Algoritmo de Newton. Além disso, constatada a possibilidade de resolver este tipo de equações, será possível encontrar aplicações, que fogem ao objetivo deste trabalho, onde os polinômios matriciais com coeficientes simultaneamente triangularizáveis podem ser empregados, veja-se por exemplo o trabalho de Balendra, Tat e Lee [1], onde os coeficientes do problema são substituídos por matrizes

que comutam.

Pode-se observar facilmente que o Algoritmo 6.1 apresenta vantagens significativas sobre o método de Newton. Se neste último método for empregada como aproximação inicial uma matriz diagonal cujos elementos são os empregados para aplicar o algoritmo da Seção 6, O Algoritmo 2.1 requer no mínimo duas avaliações do polinômio matricial e da resolução de dois sistemas lineares. Em número de operações, isto representa mais do dobro do custo computacional.

A comparação com o algoritmo da Seção 5 é um pouco mais simples. O Algoritmo 5.1 somente calcula um solvente dominante e, além disso não aproveita o conhecimento das particularidades dos coeficientes.

Resumindo, o Algoritmo 6.1 apresenta condições ótimas de aplicabilidade e, portanto, deve ser utilizado sempre que seja constatada a possibilidade da sua utilização.

Capítulo IV

Transformação simultânea de matrizes

Levando-se em conta as qualidades do algoritmo apresentado na Seção III-6, este capítulo é dedicado ao estudo da existência de transformações que triangularizam simultaneamente um conjunto de matrizes. Das duas seções nas quais este capítulo é dividido, a primeira é dedicada ao estudo de conjuntos de matrizes simultaneamente triangularizáveis, e na segunda, apresentam-se alguns resultados sobre diagonalização simultânea, com o intuito de mostrar quais são as restrições envolvidas na simplificação do cálculo dos solventes de um polinômio matricial.

1. Triangularização simultânea

O principal resultado desta seção, o Teorema 1.1, caracteriza os conjuntos de matrizes que podem ser triangularizados simultaneamente por transformações unitárias. O maior esforço é realizado no sentido de provar o Lema 1.1, o qual fornece uma condição necessária para a existência de um autovetor comum a um conjunto de matrizes.

Lema 1.1 ([13]). *Se $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$ forem matrizes tais que, para cada polinômio escalar $p(z_1, \dots, z_m)$ nas variáveis z_1, \dots, z_m e para cada ij , a matriz*

$$p(A_1, \dots, A_m)(A_i A_j - A_j A_i) \quad (1.1)$$

for nilpotente, então, as matrizes A_1, \dots, A_m têm um autovetor em comum. Além disso, dado qualquer vetor η não nulo, é possível construir um polinômio escalar $h(z_1, \dots, z_m)$ tal que $h(A_1, \dots, A_m)\eta$ é um autovetor comum de A_1, \dots, A_m .

PROVA. Pode-se supor, sem perda da generalidade, que as matrizes A_1, \dots, A_m são todas não nulas.

Seja \mathcal{H}_s a seguinte hipótese: Dado um vetor não nulo η , existe um polinômio escalar $h(z_1, \dots, z_s)$ tal que o vetor $h(A_1, \dots, A_s)\eta$ é um autovetor comum das matrizes A_1, \dots, A_s .

\mathcal{H}_1 é verdadeira, pois os vetores $\{A_1^k \eta ; k = 0, \dots, n\}$ são linearmente dependentes e, portanto, existe um polinômio escalar $q(z)$ para o qual $q(A_1)\eta$ é o vetor nulo. Se λ_0 for uma raiz de $q(z)$ e $q_0(z)$ for o cociente da divisão de $q(z)$ por $z - \lambda_0$, então o vetor $q_0(A_1)\eta$ é um autovetor de A_1 .

Se a afirmação \mathcal{H}_{s-1} for verdadeira, seja $\xi_1 = h(A_1, \dots, A_{s-1})\eta$ um autovetor comum de A_1, \dots, A_{s-1} . A hipótese \mathcal{H}_1 aplicada à matriz A_s assegura que existe um polinômio escalar $g(z)$ tal que $g(A_s)\xi_1$ é um autovetor de A_s .

Para $i < s$ seja λ_i o autovalor de A_i associado a ξ_1 . Se, para todo ij , $C_{ij} = A_i A_j - A_j A_i$ for o comutador das matrizes A_i e A_j , existem duas possibilidades:

- (1) $C_{i,s} A_s^k \xi_1 = 0$ para $i < s$ e para todo k ,
- (2) existem inteiros i_1 e k_1 para os quais $C_{i_1,s} A_s^{k_1} \xi_1 \neq 0$.

No primeiro caso tem-se que:

$$A_i g(A_s) \xi_1 = g(A_s) A_i \xi_1 = \lambda_i g(A_s) \xi_1, \quad (i = 1, \dots, s-1)$$

e segue-se, da hipótese de indução, que o vetor

$$g(A_s) \xi_1 = g(A_s) h(A_1, \dots, A_{s-1}) \eta$$

é um autovetor comum das matrizes A_1, \dots, A_s .

Se existirem inteiros i_1 e k_1 satisfazendo $C_{i_1,s} A_s^{k_1} \xi_1 \neq 0$, a hipótese \mathcal{H}_{s-1} mostra que existe um autovetor comum de A_1, \dots, A_{s-1} da forma:

$$\xi_2 = q_1(A_1, \dots, A_{s-1}) C_{i_1,s} A_s^{k_1} \xi_1.$$

Observe-se que, se o vetor $C_{i,r,s} A_s^{k_r} \xi_r$ for não nulo para alguns inteiros i_r e k_r , então o vetor ξ_{r+1} definido por:

$$\xi_{r+1} = q_r(A_1, \dots, A_{s-1}) A_s^{k_r} C_{i_r,s} \xi_r$$

é não nulo.

Se a seqüência $\{\xi_1, \xi_2, \dots\}$ for finita, isto é, se existir um inteiro p tal que $\xi_p \neq 0$ e $\xi_{p+1} = 0$, aplicando a hipótese \mathcal{H}_1 ao vetor ξ_p , obtém-se um autovetor $g(A_s)\xi_p$ de A_s para o qual

$$A_i g(A_s)\xi_p = g(A_s)A_i \xi_p = \lambda_i g(A_s)\xi_p, \quad (i = 1, \dots, s-1)$$

e portanto, o vetor $g(A_s)\xi_p$ é um autovetor comum de A_1, \dots, A_s .

Finalmente, observe-se que deve existir um inteiro $p \leq n+1$ tal que $\xi_p \neq 0$ e $\xi_{p+1} = 0$, pois se isto não acontecer, existirão escalares $\lambda_1, \dots, \lambda_{n+1}$ não todos nulos tais que:

$$\sum_{k=1}^{n+1} \lambda_k \xi_k = 0.$$

Logo, se j for o menor inteiro tal que $\lambda_j \neq 0$,

$$\sum_{k=j+1}^{n+1} \lambda_k \xi_k = -\lambda_j \xi_j,$$

e pela definição dos vetores ξ_r , tem-se:

$$u(A_1, \dots, A_s) C_{i,j,s} A_s^{k_j} \xi_j = -\lambda_j \xi_j,$$

assim sendo, $-\lambda_j$ é um autovalor não nulo de $A_s^{k_j} u(A_1, \dots, A_s) C_{i,j,s}$, contrariando a hipótese de que a matriz $A_s^{k_j} u(A_1, \dots, A_s) C_{i,j,s}$ é nilpotente. Assim sendo, o lema fica demonstrado. \square

Observe-se que a prova do lema acima fornece um método para construir um autovetor comum a um conjunto de matrizes; este algoritmo foi empregado no programa TRIANGUL, e os detalhes da sua implementação encontram-se na Seção V-1.

Como mostra o Teorema 1.1, a condição (1.1) é muito forte. Os exemplos a seguir fornecem uma idéia aproximada do que um par de matrizes precisa para ter um autovetor em comum.

Exemplo 1.1. Considere-se o movimento das três massas representadas na Figura 1.1. Se $\xi = (x_1, x_2, x_3)^t$, a equação de movimento das três massas é determinada por:¹

$$M\ddot{\xi} + C\dot{\xi} + K\xi = 0, \quad (1.2)$$

onde

$$M = \begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{pmatrix}, \quad C = \begin{pmatrix} c_1 & -c_1 & 0 \\ -c_1 & c_1 + c_2 & -c_2 \\ 0 & -c_2 & c_2 \end{pmatrix}$$

$$K = \begin{pmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{pmatrix}.$$

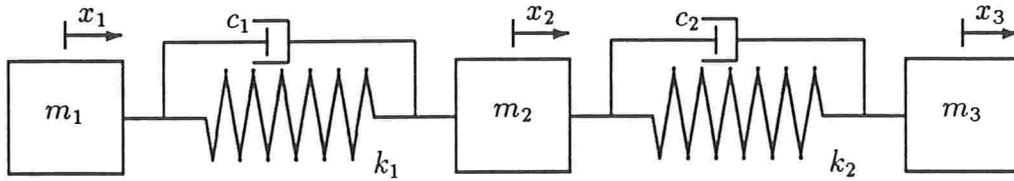


FIGURA 1.1

Sejam $m_1 = m_2 = m_3 = 1$, neste caso as matrizes I , C e K sempre têm um autovetor comum, o vetor

$$\xi = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

porém, a matriz $CK - KC$ é nilpotente se, e somente se,

$$c_2k_1 = c_1k_2,$$

isto é, se K for um múltiplo de C . Observe-se que a matriz $CK - KC$ é singular e que o vetor ξ é um autovetor de $CK - KC$ associado ao autovalor 0, pois

$$CK - KC = \begin{pmatrix} 0 & c_2k_1 - c_1k_2 & c_1k_2 - c_2k_1 \\ c_1k_2 - c_2k_1 & 0 & c_2k_1 - c_1k_2 \\ c_2k_1 - c_1k_2 & c_1k_2 - c_2k_1 & 0 \end{pmatrix}.$$

¹Os Exemplos 1.1 e 1.4 foram extraídos de Pestel e Lechie [36], e a notação neles empregada corresponde àquela da referência citada.

De fato, se existir um autovetor comum, o determinante de todos os comutadores é zero, porém, a recíproca não vale, como fica demonstrado pelo seguinte exemplo.

Exemplo 1.2. Considere as matrizes

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 2 & 2 \\ 0 & 2 & 2 \end{pmatrix} \quad \text{e} \quad C = \begin{pmatrix} 1 & -1 & 2 \\ 0 & 2 & -2 \\ 0 & 0 & 0 \end{pmatrix},$$

pode-se mostrar que as matrizes $AB - BA$, $BC - CB$ e $AC - CA$ têm determinante zero, porém, A , B e C não têm nenhum autovetor em comum, mesmo que qualquer subconjunto de duas matrizes tenha.

Até o final do capítulo, o conjunto das matrizes unitárias de dimensão n é denotado por \mathcal{U}_n , e o conjunto das matrizes triangulares superiores é representado por \mathcal{T}_n .

Teorema 1.1 ([13]). *Para quaisquer matrizes $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$, as seguintes afirmações são equivalentes:*

- (1) *Para cada polinômio escalar $p(z_1, \dots, z_m)$ nas variáveis z_1, \dots, z_m e para cada ij , a matriz*

$$p(A_1, \dots, A_m)(A_i A_j - A_j A_i)$$

é nilpotente.

- (2) *Existe uma matriz unitária U tal que as matrizes $U^* A_1 U, \dots, U^* A_m U$ são triangulares superiores.*
- (3) *Para cada $i = 1, \dots, n$, existe uma ordem dos autovalores $\lambda_i^{(1)}, \dots, \lambda_i^{(n)}$ da matriz A_i , tal que para qualquer função racional $r(z_1, \dots, z_m)$, os autovalores $\lambda_1, \dots, \lambda_n$ da matriz $r(A_1, \dots, A_m)$ são definidos por:*

$$\lambda_k = r(\lambda_1^{(k)}, \dots, \lambda_m^{(k)}) \quad (k = 1, \dots, n).$$

PROVA.

(1) \Rightarrow (2). Se a afirmação (1) for verdadeira, segue-se, do Lema 1.1, que existe um autovetor comum ζ das matrizes A_1, \dots, A_m . Pode-se supor, sem perda da generalidade, que $\|\zeta\|_2 = 1$, assim sendo, se $U \in \mathcal{U}_n$ for uma matriz cuja primeira coluna é ζ , tem-se que:

$$U^* A_i U = \begin{pmatrix} \lambda_i & \xi_i \\ 0 & A_i' \end{pmatrix} \quad (i = 1, \dots, m).$$

Sendo todas as matrizes de dimensão 1 triangulares, e como a hipótese (1) também é satisfeita pelas matrizes $A'_1, \dots, A'_m \in \mathbb{C}^{(n-1) \times (n-1)}$, o resultado obtém-se por indução sobre a dimensão das matrizes. \square

(2) \Rightarrow (3). Como $r(z_1, \dots, z_m)$ é uma função racional, existem dois polinômios $p_1(z_1, \dots, z_m)$ e $p_2(z_1, \dots, z_m)$ tais que:

$$r(z_1, \dots, z_m) = \frac{p_1(z_1, \dots, z_m)}{p_2(z_1, \dots, z_m)}.$$

matriz $U \in \mathcal{U}_n$ satisfizer

$$U^* A_i U = T_i \in \mathcal{T}_n \quad (i = 1, \dots, m),$$

então

$$\begin{aligned} U^* p_1(A_1, \dots, A_m) U &= p_1(T_1, \dots, T_m) \\ U^* p_2(A_1, \dots, A_m) U &= p_2(T_1, \dots, T_m). \end{aligned}$$

Considerando que $p_1(T_1, \dots, T_m)$ e $p_2(T_1, \dots, T_m)$ são matrizes triangulares, um dos autovalores de $p_l(T_1, \dots, T_m)$ é da forma

$$p_l(\lambda_1, \dots, \lambda_m), \quad (l = 1, 2)$$

onde λ_i é algum autovalor de A_i . Assim sendo, os autovalores de $r(A_1, \dots, A_m)$ são:

$$\frac{p_1(\lambda_1, \dots, \lambda_m)}{p_2(\lambda_1, \dots, \lambda_m)}. \quad \square$$

(3) \Rightarrow (1). Como todo polinômio é uma função racional, cada autovalor da matriz

$$B = p(A_1, \dots, A_m) (A_i A_j - A_j A_i)$$

é da forma

$$\lambda_B = p(\lambda_1, \dots, \lambda_m) (\lambda_i \lambda_j - \lambda_j \lambda_i),$$

onde λ_k é algum autovalor da matriz A_k . Uma vez que $\lambda_B = 0$, B é nilpotente. \square

O fato de U ser uma matriz unitária (hipótese (2) do teorema) de maneira alguma representa uma limitação para o conjunto de matrizes unitarizáveis, pois

(2') Se existir uma matriz inversível P para a qual $P^{-1} A_1 P, \dots, P^{-1} A_m P$ são matrizes triangulares superiores,

a prova de (2') \Rightarrow (3) decorre como a de (2) \Rightarrow (3).

A vantagem de usar matrizes unitárias é percebida na hora de implementar o algoritmo de triangularização, pois os erros de arredondamento introduzidos pelas transformações unitárias são menores.

É bem conhecido o fato de que um conjunto de matrizes simétricas reais diagonalizam simultaneamente desde que as matrizes comutem duas a duas; o seguinte corolário mostra o que acontece na ausência da hipótese de simetria.

Corolário 1.1. *Se A_1, \dots, A_m for um conjunto de matrizes que comutam duas a duas, então existe uma matriz unitária U tal que*

$$U^* A_i U \in \mathcal{T}_n \quad (i = 1, \dots, m).$$

Do Teorema 1.1, obtém-se como corolário um conhecido resultado sobre a diagonalização de matrizes normais.

Corolário 1.2. *Se $A \in \mathbb{C}^{n \times n}$ for uma matriz normal, então A é similar a uma matriz diagonal. Além disso, a matriz A^* pode ser expressa como um polinômio escalar avaliado em A .*

PROVA. Sendo A uma matriz normal e definindo $A_1 = A$ e $A_2 = A^*$, segue-se, do Corolário 1.1, que existe uma matriz $U \in \mathcal{U}_n$ tal que

$$T_1 = U^* A U \in \mathcal{T}_n \quad \text{e} \quad T_2 = U^* A^* U \in \mathcal{T}_n.$$

Como $T_1 = T_2^*$, as duas matrizes são diagonais.

A segunda parte do corolário decorre de observar que, para quaisquer números complexos $\lambda_1, \dots, \lambda_n$, existe um polinômio escalar $q(z)$ tal que $q(\lambda_i) = \bar{\lambda}_i$, para todo $i = 1, \dots, n$. \square

O seguinte corolário fornece uma outra condição para a triangularização simultânea no caso em que todas as matrizes forem normais.

Corolário 1.3. *Se cada uma das matrizes A_1, \dots, A_m for normal, então as três afirmações estabelecidas no Teorema 1.1 são equivalentes à condição*

$$A_i A_j = A_j A_i \quad (i, j = 1, \dots, m). \quad (1.3)$$

IV. TRANSFORMAÇÃO SIMULTÂNEA DE MATRIZES

Uma vez que A_1, \dots, A_m são matrizes normais, segue-se, do Corolário 1.2, que as matrizes U^*A_1U, \dots, U^*A_mU são diagonais. Assim sendo, as matrizes A_1, \dots, A_m comutam.

A recíproca é óbvia, pois (1.3) implica que a condição (1) do Teorema 1.1 é verdadeira. □

Exemplo 1.3. Sejam A_1 e A_2 as matrizes definidas por:

$$A_1 = \begin{pmatrix} -1 & -6 \\ 2 & -9 \end{pmatrix} \quad \text{e} \quad A_2 = \begin{pmatrix} 0 & 12 \\ -2 & 14 \end{pmatrix}.$$

O comutador de A_1 e A_2 é a matriz $\begin{pmatrix} -12 & 12 \\ -12 & 12 \end{pmatrix}$, a qual tem $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ como “único” autovetor.

No Exemplo acima, as matrizes A_1 e A_2 têm um único autovetor comum e, portanto, o resultado do Teorema 1.1 não pode ser melhorado no sentido de fornecer mais autovetores linearmente independentes, sem impor mais condições às matrizes A_1, \dots, A_m . Estas novas condições são analisadas na Seção 1.1 e na Seção 2.

1.1. Existência de autovetores comuns linearmente independentes

O Lema 1.1 mostra que se as matrizes A_1, \dots, A_m satisfizerem a condição (1) do Teorema 1.1, existe sempre pelo menos um autovalor em comum; levando-se em conta a equivalência de (1) com (2) e (3), este resultado não pode ser melhorado sem impor novas restrições às matrizes A_1, \dots, A_m (ver Exemplo 1.3). Nesta seção, são analisadas condições para a existência de um número maior de autovetores linearmente independentes.

Definição 1.1. As matrizes A_1, \dots, A_m são chamadas **quase-comutativas**, se cada matriz A_k comuta com as matrizes

$$A_i A_j - A_j A_i \quad (i, j = 1, \dots, m).$$

O Teorema 1.2 mostra que se as matrizes forem quase-comutativas (ou comutativas), o número de autovetores linearmente independentes é maior.

Teorema 1.2 ([13]). *Se as matrizes $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$ forem quase-comutativas, então a hipótese (1) do Teorema 1.1 é verdadeira. Além disso, para qualquer*

autovalor λ_1 de A_1 , existe um autovetor comum η das matrizes A_1, \dots, A_m satisfazendo $A_1\eta = \lambda_1\eta$.

PROVA. Levando-se em conta a comutatividade de A_k com cada um dos comutadores C_{ij} , para demonstrar a primeira parte do teorema é suficiente mostrar que C_{ij} é nilpotente. Como o comutador de duas matrizes tem traço nulo, se r for um inteiro positivo, a matriz C_{ij}^r tem traço zero, pois:

$$C_{ij}^r = C_{ij}^{r-1}(A_i A_j - A_j A_i) = A_i C_{ij}^{r-1} A_j - C_{ij}^{r-1} A_j A_i.$$

Se $\lambda_1, \dots, \lambda_p$ forem os autovalores distintos de C_{ij} com multiplicidades m_1, \dots, m_p respectivamente, tem-se:

$$m_1 \lambda_1^r + \dots + m_p \lambda_p^r = 0 \quad (r = 1, 2, \dots)$$

e, em particular,

$$V(\lambda_1, \dots, \lambda_p) \begin{pmatrix} m_1 r_1 \\ m_2 r_2 \\ \vdots \\ m_p r_p \end{pmatrix} = 0.$$

Como a matriz $V(\lambda_1, \dots, \lambda_p)$ é inversível, o sistema acima tem somente a solução trivial, assim, o único autovalor de C_{ij} é o zero e, portanto, o comutador é nilpotente.

Para provar a segunda parte do teorema, seja \mathcal{H}_s a seguinte afirmação: dado qualquer autovalor λ_1 de A_1 , existe um autovetor comum η das matrizes A_1, \dots, A_s e a C_{ij} , $i, j = 1, \dots, m$ para o qual $A_1\eta = \lambda_1\eta$.

Considere primeiro o caso $s = 1$. Seja B uma matriz cujas colunas formam uma base do subespaço dos autovetores de A_1 associados ao autovalor λ_1 . Em geral, B é uma matriz retangular com k colunas que satisfaz

$$A_1 C_{ij} B = C_{ij} A_1 B = \lambda^{(1)} C_{ij} B \quad (i, j = 1, \dots, m),$$

e portanto, cada coluna de $C_{ij} B$ é o vetor nulo ou é um autovetor de A_1 associado a λ_1 e, portanto, cada coluna de C_{ij} é uma combinação linear das colunas de B . Assim sendo, para cada ij existe uma matriz quadrada M_{ij} de dimensão k tal que $C_{ij} B = B M_{ij}$. Levando-se em conta que as matrizes C_{ij} comutam e que as colunas de B são vetores linearmente independentes, as matrizes M_{ij} comutam. O Teorema 1.1 mostra que existe um vetor ζ , autovetor de todas as matrizes M_{ij} , assim sendo, $B\zeta$ é um autovetor de A_1 e de cada uma das matrizes C_{ij} . Observe-se que por definição, $\lambda^{(1)}$ é o autovalor de A_1 associado ao vetor $B\zeta$.

Se, por hipótese de indução, \mathcal{H}_{s-1} for verdadeira, seja ζ um autovetor comum de A_1, \dots, A_{s-1} e de cada matriz C_{ij} , $i, j = 1, \dots, m$, e sejam $\lambda_2, \dots, \lambda_{s-1}$ os autovalores de A_2, \dots, A_{s-1} associados a ζ . Uma vez que C_{ij} é nilpotente, todos os seus autovalores são nulos e, portanto,

$$C_{ij}A_s^k\zeta = A_s^kC_{ij}\zeta = 0 \quad (i, j = 1, \dots, m; k = 0, 1, \dots) \quad (1.4)$$

e

$$A_iA_s^k\zeta = \lambda_iA_s^k\zeta \quad (i = 1, \dots, s-1; k = 1, 2, \dots). \quad (1.5)$$

Sendo que sempre existe um polinômio escalar $p(z)$, tal que $p(A_s)\zeta$ é um autovetor de A_s , segue-se, das equações (1.4) e (1.5), que $p(A_s)\zeta$ também é um autovetor de A_1, \dots, A_{s-1} e de cada C_{ij} , $i, j = 1, \dots, m$, com autovalor λ_1 em relação a A_1 . Logo, a afirmação \mathcal{H}_s é verdadeira para todo s . \square

Observe-se que o teorema acima mostra que as matrizes A_1, \dots, A_m têm no mínimo tantos autovetores comuns linearmente independentes quantos forem os autovalores distintos de cada matriz.

Corolário 1.4. *Se $A, B \in \mathbb{C}^{n \times n}$ forem matrizes quase-comutativas e se A tiver k autovalores distintos, então existe um polinômio escalar $p(z)$ tal que a matriz $B - p(A)$ tem no mínimo k autovetores linearmente independentes associados ao autovalor zero.*

PROVA. Sejam $\lambda_1, \dots, \lambda_k$ os autovalores distintos de A ; o Teorema 1.2 mostra que para cada λ_i existe um autovetor comum ξ_i de A e B (associado ao autovalor β_i de B). Além disso, para qualquer polinômio escalar $p(z)$ tem-se que:

$$(B - p(A))\xi_i = (\beta_i - p(\lambda_i))\xi_i \quad (i = 1, \dots, k).$$

Escolhendo o polinômio escalar $p(z)$ de forma tal que $p(\lambda_i) = \beta_i$, para todo $i = 1, \dots, k$, obtém-se o resultado. \square

1.2. Triangularização por transformações bi-unitárias

Nesta seção, estuda-se o problema de transformar simultaneamente um conjunto de matrizes à forma triangular superior por transformações bi-unitárias. Embora não haja uma caracterização dos conjuntos de matrizes que satisfazem esta propriedade, algumas condições suficientes são aqui enunciadas.

Definição 1.2. Duas matrizes $A, B \in \mathbb{C}^{n \times n}$ são chamadas **equivalentes**, se existirem matrizes inversíveis P e Q tais que

$$B = P^{-1}AQ.$$

Se P e Q forem matrizes unitárias, A e B são chamadas **unitariamente equivalentes**, e a transformação que associa a cada matriz A a matriz P^*AQ , é chamada de **transformação bi-unitária**.

O seguinte lema é trivial, porém, é muito útil na prova do Teorema 1.3, e só foi incluído nesta seção por motivos didáticos.

Lema 1.2. Para cada matriz $A \in \mathbb{C}^{n \times n}$, existe uma matriz unitária U tal que U^*A é uma matriz triangular superior.

PROVA. A demonstração decorre por indução sobre a dimensão da matriz. Como toda matriz de dimensão 1 é triangular, o resultado é verdadeiro para $n = 1$. Seja e_1 o primeiro vetor da base canônica e seja U_1 uma matriz unitária tal que:

$$U_1^*Ae_1 = \|Ae_1\|_2 e_1.$$

Assim sendo, existe uma matriz \tilde{A} de dimensão $n - 1$, satisfazendo

$$U_1^*A = \begin{pmatrix} \|Ae_1\|_2 & a \\ 0 & \tilde{A} \end{pmatrix}.$$

Se, por hipótese de indução, o lema vale para as matrizes de dimensão $n - 1$, então existe $U_2 \in \mathbb{C}^{(n-1) \times (n-1)}$ para a qual

$$U_2^*\tilde{A} = T \in \mathcal{T}_{n-1},$$

definindo a matriz U por:

$$U = U_1 \begin{pmatrix} 1 & 0 \\ 0 & U_2 \end{pmatrix},$$

tem-se que U^*A é uma matriz triangular superior. \square

O Teorema 1.3 estabelece uma condição sob a qual um conjunto de matrizes é unitariamente equivalente a um conjunto de matrizes triangulares.

Teorema 1.3. Dadas as matrizes $A_0, \dots, A_m \in \mathbb{C}^{n \times n}$, se A_0 for uma matriz inversível, então A_0, \dots, A_m são unitariamente equivalentes a matrizes triangulares superiores se, e somente se, as matrizes $A_0^{-1}A_1, \dots, A_0^{-1}A_m$ satisfizerem alguma das hipóteses do Teorema 1.1.

PROVA. Sejam U e V matrizes unitárias satisfazendo

$$U^* A_i V = T_i \in \mathcal{T}_n \quad (i = 0, \dots, m).$$

Se A_0 for uma matriz inversível, a matriz inversa de T_0 satisfaz

$$T_0^{-1} = V^* A_0^{-1} U \in \mathcal{T}_n,$$

logo,

$$T_0^{-1} T_i = V^* A_0^{-1} A_i V \in \mathcal{T}_n \quad (i = 1, \dots, m),$$

e portanto, as matrizes $A_0^{-1} A_1, \dots, A_0^{-1} A_m$ satisfazem a hipótese (2) do Teorema 1.1.

Reciprocamente, se a matriz A_0 for inversível e as matrizes $A_0^{-1} A_1, \dots, A_0^{-1} A_m$ triangularizarem simultaneamente, isto é, se existir $V \in \mathcal{U}_n$ tal que

$$V^* A_j^{-1} A_i V = \tilde{T}_i,$$

o Lema 1.2 mostra que existe uma matriz unitária U , que satisfaz

$$U^* A_0 V = T_0 \in \mathcal{T}_n, \quad (1.6)$$

logo,

$$\begin{aligned} \tilde{T}_i &= V^* A_0^{-1} A_i V \\ &= V^* A_0^{-1} U U^* A_i V \\ &= T_0^{-1} U^* A_i V, \end{aligned}$$

e portanto,

$$U^* A_i V = T_0 \tilde{T}_i = T_i \quad (i = 1, \dots, m).$$

Uma vez que A_0 satisfaz (1.6), o teorema fica demonstrado. \square

O seguinte exemplo mostra que nem sempre um conjunto de matrizes triangularizáveis por transformações bi-unitárias satisfaz alguma das hipóteses do Teorema 1.1.

Exemplo 1.4. Considere-se o sistema descrito pela Figura 1.2. O movimento deste sistema é controlado pela equação:

$$M\ddot{\xi} + C\dot{\xi} + K\xi = 0, \quad (1.7)$$

onde

$$M = \begin{pmatrix} m_1 & 0 \\ 0 & m_2 \end{pmatrix}, \quad C = \begin{pmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 \end{pmatrix}$$

e

$$K = \begin{pmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 + k_3 \end{pmatrix}.$$

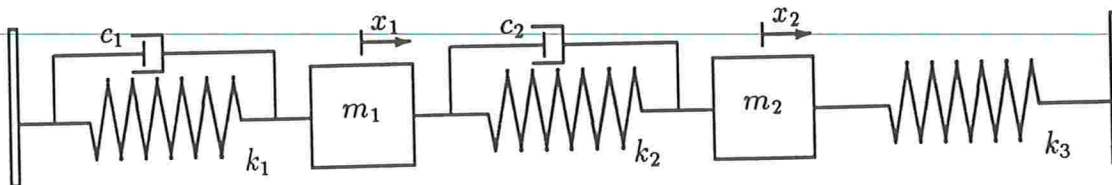


FIGURA 1.2

Sejam $m_1 = 2$, $m_2 = 1$, $c_1 = c_2 = 1$, $k_1 = 3$ e $k_2 = k_3 = 1$. Neste caso, tem-se

$$M = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix} \quad \text{e} \quad K = \begin{pmatrix} 4 & -1 \\ -1 & 2 \end{pmatrix}.$$

Pode-se mostrar facilmente que o comutador de C e K não é nilpotente e, portanto, as matrizes não podem triangularizar simultaneamente. Porém, como as matrizes M , C e K são uma combinação das matrizes

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{e} \quad \begin{pmatrix} 4 & -1 \\ -1 & 2 \end{pmatrix},$$

pelo teorema acima existem matrizes unitárias U e V que triangularizam simultaneamente as matrizes A_i . Pode-se mostrar que, uma escolha possível para U e V é

$$U = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & -\sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix} \quad \text{e} \quad V = \frac{1}{\sqrt{3}} \begin{pmatrix} \sqrt{2} & -1 \\ 1 & \sqrt{2} \end{pmatrix}.$$

O Teorema 1.4 fornece uma outra condição suficiente para que um conjunto de matrizes seja unitariamente equivalente a um conjunto de matrizes triangulares superiores.

Teorema 1.4. *Sejam $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$, matrizes tais que o conjunto S_1 definido por:*

$$S_1 = \{A_i^* A_j\}_{i,j=1}^m$$

é Abeliano, então existem matrizes unitárias U e V tais que:

$$U^* A_i V \in \mathcal{T}_n \quad (i = 1, \dots, m).$$

PROVA. Se S_1 for um conjunto comutativo, as matrizes $A_i^* A_j$, $i, j = 1, \dots, m$ são normais e, portanto, o Corolário 1.2 mostra que existe uma base ortonormal de autovetores ξ_1, \dots, ξ_n tal que:

$$A_i^* A_j \xi_k = \lambda_{ij}^{(k)} \xi_k \quad (i, j = 1, \dots, m; k = 1, \dots, n).$$

Pode-se supor, sem perda da generalidade, que para algum s , os vetores ξ_1, \dots, ξ_s pertencem ao núcleo de todas as A_i e que para $k > s$, existe um índice i_k tal que $A_{i_k} \xi_k \neq 0$, desta maneira podem-se definir os vetores η_k por

$$\eta_k = \frac{1}{\sqrt{\lambda_{i_k i_k}^{(k)}}} A_{i_k} \xi_k \quad (k = s + 1, \dots, n), \quad (1.8)$$

Observe-se que $\eta_{s+1}, \dots, \eta_n$ formam um conjunto ortonormal de vetores, pois para $s + 1 \leq k, j \leq n$ tem-se:

$$\langle A_{i_k} \xi_k, A_{i_j} \xi_j \rangle = \lambda_{i_k i_j}^{(j)} \langle \xi_k, \xi_j \rangle = \lambda_{i_k i_j}^{(j)} \delta_{kj},$$

e portanto,

$$\lambda_{i_k i_k}^{(k)} = \|A_{i_k} \xi_k\|_2^2.$$

Estendendo $\eta_{s+1}, \dots, \eta_n$ a uma base ortonormal η_1, \dots, η_n e definindo

$$U = [\eta_1, \dots, \eta_n] \quad \text{e} \quad V = [\xi_1, \dots, \xi_n],$$

tem-se que as matrizes $U^* A_i V$ são triangulares superiores e, portanto, o teorema fica demonstrado. \square

Note-se que as condições do Teorema 1.4 não são necessárias, pois as matrizes do Exemplo 1.3 triangularizam simultaneamente, porém não satisfazem a hipótese do teorema.

Corolário 1.5. *Dadas as matrizes $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$, se o conjunto $\{A_i^* A_j\}_{i,j=1}^m$ satisfizer alguma das hipóteses do Teorema 1.1, então existem matrizes unitárias U e V , tais que $U^* A_i V \in \mathcal{T}_n$ para $i = 1, \dots, m$.*

PROVA. Se as matrizes $\{A_i^* A_j\}_{i,j=1}^m$ satisfizerem alguma das hipóteses do Teorema 1.1, então existe uma matriz unitária V tal que:

$$V^* A_i^* A_j V = T_{ij} \in \mathcal{T}_n \quad (i, j = 1, \dots, m).$$

Sendo $T_{ij}^* = T_{ij}$, a matriz T_{ij} é diagonal para todo $i, j = 1, \dots, m$. Logo, as matrizes $\{A_i^* A_j\}_{i,j=1}^m$ formam um conjunto Abelian, assim o resultado decorre do Teorema 1.4. \square

2. Diagonalização simultânea

O problema de diagonalização simultânea de matrizes quadradas surge na discussão do desacoplamento das equações na teoria linear das vibrações [1].

O Teorema a seguir caracteriza os conjuntos de matrizes similares a conjuntos de matrizes diagonais.

Teorema 2.1 ([13]). *Para quaisquer matrizes $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$, as seguintes afirmações são equivalentes:*

(1) *Cada uma das matrizes A_1, \dots, A_m é diagonalizável e para $i, j = 1, \dots, m$ tem-se:*

$$A_i A_j = A_j A_i.$$

(2) *Existe uma matriz P inversível, tal que as matrizes $P^{-1} A_1 P, \dots, P^{-1} A_m P$ são diagonais.*

(3) *Existe uma matriz diagonalizável A tal que, cada uma das matrizes A_i pode ser expressa como um polinômio escalar em A .*

PROVA.

(1) \Rightarrow (2). Seja \mathcal{H}_s a seguinte afirmação: existe uma matriz inversível P_s , tal que as matrizes

$$P_s^{-1} A_1 P_s, \dots, P_s^{-1} A_s P_s$$

são diagonais.

Uma vez que A_1 é diagonalizável, existe uma matriz inversível P_1 tal que $P_1^{-1} A_1 P_1$ é diagonal, logo \mathcal{H}_1 é verdadeira.

Se a afirmação \mathcal{H}_{s-1} for verdadeira, existe uma matriz P_{s-1} tal que as matrizes $P_{s-1}^{-1} A_1 P_{s-1}, \dots, P_{s-1}^{-1} A_{s-1} P_{s-1}$ são diagonais. Considerando que A_s comuta com cada uma das matrizes A_1, \dots, A_{s-1} , a matriz $P_{s-1}^{-1} A_s P_{s-1}$ comuta com

$$P_{s-1}^{-1} A_1 P_{s-1}, \dots, P_{s-1}^{-1} A_{s-1} P_{s-1}. \quad (2.1)$$

Assim sendo, o elemento ij de $P_{s-1}^{-1} A_s P_{s-1}$ é não nulo, se para cada uma das matrizes em (2.1), o i -ésimo e o j -ésimo elemento da diagonal coincidirem.

Empregando uma matriz de permutação P' adequada, podem-se reordenar as linhas e colunas de $P_{s-1}^{-1} A_s P_{s-1}$ para transformá-la em uma matriz diagonal por blocos A'_s .

Observe-se que a matriz $(P_{s-1} P')^{-1} A_s P_{s-1} P'$ pode ser reduzida à sua forma diagonal, usando uma transformação de similaridade P com a mesma estrutura de

blocos que A'_s . Logo, para cada $i = 1, \dots, s-1$, a matriz $A'_i = (P_{s-1}P')^{-1} A_s P_{s-1}P'$ tem a mesma estrutura de blocos que A'_s e, portanto, a matriz

$$(P_{s-1}P'P)^{-1} A_s P_{s-1}P'P$$

é diagonal, pois os blocos diagonais de A'_i são múltiplos da matriz identidade. Assim sendo, definindo a matriz P_s por:

$$P_s = P_{s-1}P'P$$

a afirmação \mathcal{H}_s resulta verdadeira para todo s . \square

(2) \Rightarrow (3). Seja $D = \text{diag}(d_1, \dots, d_n)$ uma matriz diagonal tal que os elementos d_1, \dots, d_n são distintos. Como $P^{-1}A_iP = \text{diag}(\lambda_i^{(1)}, \dots, \lambda_i^{(n)})$, para cada $i = 1, \dots, m$ pode-se escolher um polinômio escalar $p_i(z)$ tal que:

$$p_i(d_k) = \lambda_i^{(k)} \quad (k = 1, \dots, n; i = 1, \dots, m).$$

Assim sendo, $P^{-1}A_iP = p_i(D)$ e, portanto, se a matriz A for definida por $A = PDP^{-1}$, tem-se que $A_i = p_i(A)$. \square

(3) \Rightarrow (1). É óbvia, pois as matrizes A_1, \dots, A_m são polinômios escalares avaliados em uma mesma matriz diagonalizável e, portanto, são diagonalizáveis e comutam duas a duas. \square

Observação 2.1. Se a hipótese (1) do Teorema 2.1 for substituída pela existência de uma base ortonormal de autovetores, então a matriz que faz a transformação na hipótese (2) pode ser escolhida unitária.

O Teorema 2.2 caracteriza os conjuntos de matrizes simultaneamente diagonalizáveis por transformações bi-unitárias.

Teorema 2.2 ([20]). *As matrizes $A_1, \dots, A_m \in \mathbb{C}^{n \times n}$ são unitariamente equivalentes a matrizes diagonais se, e somente se, os conjuntos*

$$S_1 = \{A_i^* A_j\}_{i,j=1}^m \quad e \quad S_2 = \{A_i A_j^*\}_{i,j=1}^m$$

forem Abelianos.

PROVA. É óbvio que a comutatividade das matrizes nos conjuntos S_1 e S_2 é uma condição necessária para a existência de matrizes unitárias U e V , tais que as matrizes $U^* A_1 V, \dots, U^* A_m V$ sejam diagonais.

Se o conjunto S_1 for Abelian, todas as matrizes $A_i^* A_j$ são normais. Assim sendo, o Corolário 1.2 mostra que as matrizes $A_i^* A_j$ tem uma base comum de vetores ortonormais ξ_1, \dots, ξ_n tal que:

$$A_i^* A_j \xi_k = \lambda_{ij}^{(k)} \xi_k. \quad (2.2)$$

Como na prova do Teorema 1.4, pode-se supor, sem perda da generalidade, que para algum s , os vetores ξ_1, \dots, ξ_s pertencem ao núcleo de A_i para todo i , e que para $k > s$ existe um índice i_k tal que $A_{i_k} \xi_k \neq 0$. Logo, podem-se definir os vetores η_k por:

$$\eta_k = \frac{1}{\sqrt{\lambda_{i_k i_k}^{(k)}}} A_{i_k} \xi_k \quad (k = s + 1, \dots, n), \quad (2.3)$$

os quais, como no Teorema 1.4, resultam ser vetores ortonormais. Além disso,

$$\langle \eta_k, A_i \xi_l \rangle = \lambda_{i_k i}^{(l)} \delta_{kl} (\lambda_{i_k i_k}^{(k)})^{-1/2}, \quad (2.4)$$

e portanto, no subespaço gerado pelos vetores $\eta_{s+1}, \dots, \eta_n$, o vetor $A_i \xi_l$ só tem componente na direção de η_l . Levando-se em conta que:

$$A_i A_i^* A_i A_j^* A_j \xi_l = A_i A_j^* A_i A_i^* A_j \xi_l$$

e usando (2.2), obtém-se a relação

$$\lambda_{ii}^{(l)} \lambda_{jj}^{(l)} = \lambda_{ji}^{(l)} \lambda_{ij}^{(l)} = |\lambda_{ij}^{(l)}|^2. \quad (2.5)$$

Com o auxílio de (2.5) e (2.4), tem-se que:

$$|\langle \eta_l, A_i \xi_l \rangle| = |\lambda_{i_i i}^{(l)}| (\lambda_{i_i i_i}^{(l)})^{-1/2} = (\lambda_{i_i i}^{(l)})^{1/2} = \|A_i \xi_l\|_2,$$

e portanto, $A_i \xi_l$ e η_l são vetores paralelos. Observe-se que este é o único ponto onde foi usada a comutatividade de S_2 . Estendendo $\eta_{k+1}, \dots, \eta_n$ a uma base ortonormal $\eta_1, \dots, \eta_k, \eta_{k+1}, \dots, \eta_n$ pode-se escrever que:

$$A_i \xi_k = \rho_i^{(k)} \eta_k \quad (i = 1, \dots, m; k = 1, \dots, n),$$

e portanto,

$$A_i = (\eta_1, \dots, \eta_n) \begin{pmatrix} \rho_i^{(1)} & & 0 \\ & \ddots & \\ 0 & & \rho_i^{(n)} \end{pmatrix} \begin{pmatrix} \xi_1^* \\ \vdots \\ \xi_n^* \end{pmatrix}. \quad \square$$

A comutatividade das matrizes nos conjuntos S_1 e S_2 , foi crucial para a prova do Teorema 2.2; como mostra o seguinte teorema, se uma das matrizes A_i for não singular, é suficiente, para a diagonalização simultânea de A_1, \dots, A_m , que S_1 ou S_2 sejam Abelianos.

Teorema 2.3 ([20]). *Sejam A_1, \dots, A_m matrizes tais que A_1 é não singular e o conjunto $S_1 = \{A_i^* A_j\}_{i,j=1}^m$ é Abeliano, então existem matrizes unitárias U e V , tais que as matrizes $U^* A_1 V, \dots, U^* A_m V$ são diagonais. Neste caso, $S_2 = \{A_i A_j^*\}_{i,j=1}^m$ é um conjunto Abeliano.*

PROVA. Sejam ξ_1, \dots, ξ_n , os vetores escolhidos na prova do Teorema 2.2, e sejam η_k os vetores definidos por:

$$\eta_k = \frac{A_1 \xi_k}{\|A_1 \xi_k\|_2} \quad (k = 1, \dots, n).$$

Os vetores η_1, \dots, η_n estão definidos, pois A_1 é não singular. Como no Teorema 2.2, tem-se que:

$$\langle \eta_k, A_i \xi_l \rangle = \lambda_{1i}^{(l)} \delta_{kl} (\lambda_{11}^{(k)})^{-1/2}, \quad (2.6)$$

e portanto, os vetores η_1, \dots, η_n formam uma base ortonormal de \mathbb{C}^n , assim sendo, segue-se, da equação (2.6), que $A_i \xi_k$ é proporcional a η_k para todo k e, portanto, a demonstração decorre como no Teorema 2.2. \square

Note-se que o teorema poderia ter sido provado mesmo definindo o conjunto S_1 como $S_1 = \{A_1^* A_j\}_{j=1}^m$, de fato, esta foi a hipótese empregada na demonstração para definir os vetores η_1, \dots, η_n .

Capítulo V

Programas

Neste capítulo, apresentam-se alguns comentários sobre a implementação e uso dos programas que empregam o Algoritmo III-6.1 para resolver equações polinomiais matriciais com coeficientes triangulares (programa POLTRIN), e sobre o método descrito no Lema IV-1.1 que permite calcular um autovetor comum a um conjunto de matrizes (programa TRIANGUL).

1. Descrição

A linguagem Fortran 77 foi escolhida para a elaboração dos programas, os quais podem rodar em qualquer ambiente com poucas modificações no que diz respeito aos arquivos de entrada e saída.

Para minimizar, dentro do possível, o espaço de memória reservado, os vetores e matrizes de dimensão maior foram arranjados em blocos COMMON, segundo mostram as Figuras 1.1, 1.2 e 1.3.

As constantes mais importantes estão contidas no arquivo "parametros.f" (ver Figura 1.4), podendo ser modificadas, se for preciso, alterando somente este arquivo.

Nos programas, preferiu-se privilegiar o espaço de memória sobre a legibilidade. Assim sendo, os elementos acima da diagonal principal de uma matriz triangular superior foram armazenados por linhas num vetor coluna. Com isto, obteve-se uma redução de quase 50% no espaço reservado pelo programa POLTRIN.

```

INTEGER*4 IAS(MAXM)
COMPLEX*16 AS(MAXN,MAXN,MAXM)
COMMON /MATR1 / IAS
COMMON /MATR2 / AS

```

FIGURA 1.1. Arquivo common0.f

```

COMPLEX*16 COEF(MAXN2,MAXM)
COMMON /POLINO/ COEF

```

FIGURA 1.2. Arquivo common1.f

```

LOGICAL*1 DIVO(MAXN2)
COMPLEX*16 BS(MAXN2,0:MAXM),PEN(MAXM),ORD(MAXM),T(MAXN2)
EQUIVALENCE (T(1),BS(1,0))
COMMON /AVAL1 / PEN,ORD,BS
COMMON /AVAL2 / DIVO

```

FIGURA 1.3. Arquivo common2.f

O programa POLTRIN. O ponto mais complicado na aplicação do Algoritmo III-6.1 encontra-se no cálculo das raízes de polinômios escalares. No programa, escolheu-se o algoritmo de Laguerre devido a sua simplicidade. Nos exemplos testados o algoritmo não apresentou nenhum problema; se preciso for, este algoritmo pode ser trocado sem maiores dificuldades. As subrotinas ZROOTS e LAGUERRE, que calculam as raízes de polinômios escalares, foram obtidas em [37].

Os dados devem ser fornecidos no arquivo "poltrian.dados", da seguinte forma:

- (1) Na primeira linha, os valores de M , N e $RESPAR$, onde M é o grau do polinômio matricial, N é a dimensão das matrizes, e $RESPAR$ é uma variável lógica que controla a saída de resultados parciais.
- (2) Nas linhas seguintes, devem ser fornecidos os coeficientes do polinômio matricial, começando pelo coeficiente do termo independente (A_m). Cada matriz deve ser incluída em N linhas, cada uma das quais deve conter somente os elementos à direita da diagonal principal.

A saída do programa foi dividida em três arquivos. Os solventes do polinômio matricial estão no arquivo "poltrian.saída", onde estão armazenadas cada uma das soluções obtidas seguida pelo valor do polinômio matricial avaliado na solução. Todas as matrizes são armazenadas por linhas, começando pelo elemento diagonal.

```

PARAMETER (MAXN = 11)      ! DIM. MAX. PERMITIDA AS MATRIZES E 10
PARAMETER (MAXRN = 2)      ! MAXRN = SQRT(MAXN/2)
PARAMETER (MAXN2 = 55)     ! MAXN2 = MAXN*(MAXN-1)/2
PARAMETER (MAXM = 5)      ! GRAU MAX, PERMITIDO AOS POLINOMIOS E 4
PARAMETER (MAXM1 = 4)     ! MAXM - 1
REAL*8    EPSS
PARAMETER (EPSS = 0.5E-6)
PARAMETER (MAXIT = 200)   ! NUMERO MAXIMO DE ITERACOES EM LAGUER
COMPLEX*8  IMAG,UM,ZERO
PARAMETER (IMAG = (0.,1.), UM = (1.,0.), ZERO = (0.,0.))

```

FIGURA 1.4. Arquivo parametros.f

O programa **TRIANGUL**. Como já mencionado, o programa aqui apresentado corresponde ao algoritmo empregado na prova do Lema IV-1.1. A implementação tem dois pontos críticos: o cálculo de um autovetor comum e o cálculo de uma base ortonormal a partir de um vetor dado. Para desenvolver o segundo ponto, o algoritmo mais conhecido é o de Gram-Schmidt, o qual, embora muito empregado teoricamente, numericamente poderia introduzir erros consideráveis. Este método foi utilizado, exceto para vetores de dimensão dois, obtendo-se resultados surpreendentemente bons quando aplicado a vetores de dimensão pequena.

No algoritmo para calcular um autovetor comum, mereceram um cuidado especial os seguintes pontos:

- (1) Para cada vetor ξ , calcular um polinômio escalar $p(z)$ tal que

$$p(A)\xi = 0. \quad (1.1)$$

- (2) Determinar os zeros de um polinômio escalar.
 (3) Para um polinômio escalar $q(z)$, avaliar $q(A)$.

Para o primeiro ponto poderiam ser utilizados os polinômios característico ou minimal da matriz A , porém, sabe-se que calcular raízes de polinômios de grau elevado é um problema complicado, e de custo computacional considerável. Sendo que seria preciso calcular as raízes de $p(z)$, preferiu-se calcular, para cada vetor ξ , o polinômio mônico de grau mínimo que satisfaz a equação (1.1).

Para obter este polinômio mônico de grau mínimo, foi empregado um procedimento análogo ao usado na redução de uma matriz por operações elementares. O procedimento consiste em reduzir a matriz cujas linhas são as potências da matriz A multiplicadas pelo vetor ξ . Observe-se que se $p(z)$ for o polinômio

$$p(z) = \sum_{l=0}^k a_{k-l} z^l$$

e a matriz B e o vetor η forem definidos por:

$$B = \begin{pmatrix} \xi^t \\ (A\xi)^t \\ \vdots \\ (A^k\xi)^t \end{pmatrix} \quad \text{e} \quad \eta = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix},$$

segue-se que

$$p(A)\xi = 0 \quad \Longleftrightarrow \quad B\eta = 0.$$

O procedimento de redução das linhas de B é desenvolvido até achar o primeiro pivô nulo; se este pivô estiver na linha i , o coeficiente a_i é igualado a 1 e para $j = i+1, \dots, n$, os coeficientes a_j são igualados a zero. Os coeficientes a_1, \dots, a_{i-1} são obtidos por retro-substituição.

Certamente, neste procedimento os vetores $A^k\xi$ não precisam ser calculados desde o início, pelas características do algoritmo, eles são calculados somente quando forem necessários, isto é, só quando a linha acima tiver pivô não nulo. Em geral, este procedimento fornece polinômios de grau menor que o polinômio minimal, como mostra o seguinte exemplo.

Exemplo 1.1. Considere

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{e} \quad \xi = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

O polinômio minimal de A é z^3 enquanto que o polinômio fornecido pelo algoritmo acima é z .

Para calcular as raízes dos polinômios escalares, como no programa `PC`, empregou-se o método de Laguerre.

Para avaliar $q(A)$, empregou-se um algoritmo de Van Loan [41] que reduz o número de multiplicações.

1. DESCRIÇÃO

Os dados para este programa devem estar contidos no arquivo "matrivel.saida" da seguinte forma

- (1) Na primeira linha M , N , *RESPAR*.
- (2) Nas linhas seguintes, as matrizes a serem transformadas, armazenadas por linhas.

A complexidade do algoritmo implementado no programa TRIANGUL é da ordem de $2mn^4$ operações para o cálculo de um autovetor comum a um conjunto de m matrizes de dimensão n . Em geral este número é bem maior que o verificado na prática, pois o grau dos polinômios que satisfazem a equação (1.1) decresce com o número de iterações. Pode-se dizer que o número medio de operações desenvolvidas neste processo é da ordem mn^3 .

A saída é o arquivo "trans_U.saida", que contém a matriz unitária empregada na transformação, armazenada por linhas.

Para uma melhor compreensão dos algoritmos apresentados, a seguir são incluídos os Diagramas de Fluxo das rotinas mais importantes.

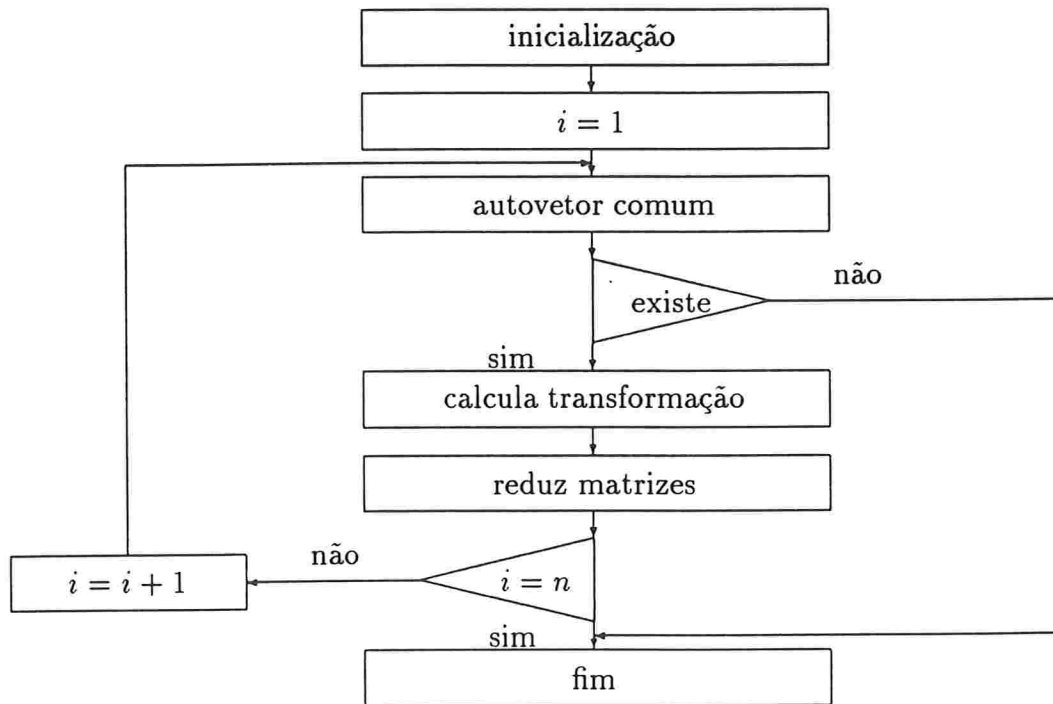


FIGURA 1.5. Diagrama de fluxo do programa de triangulação simultânea

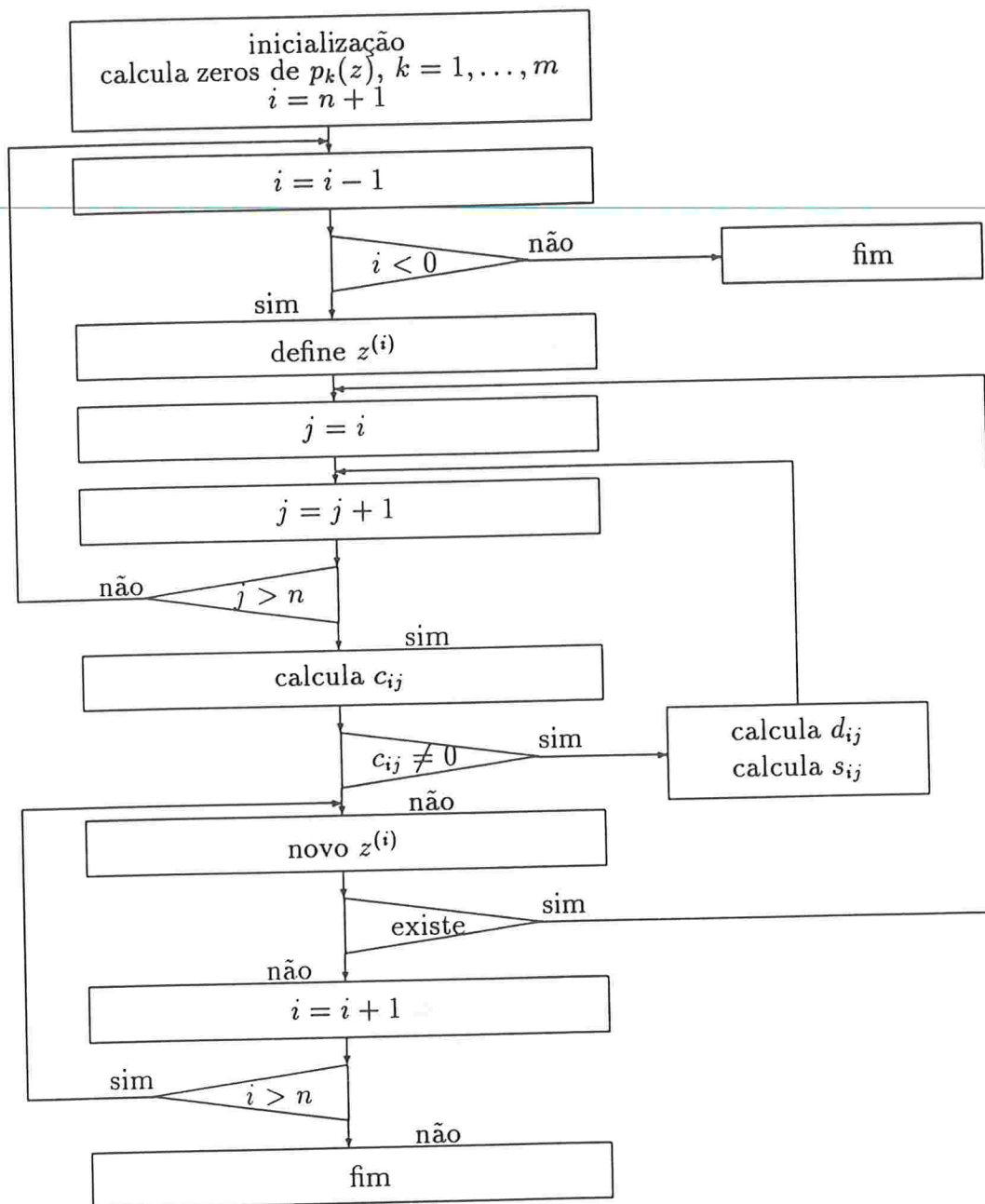


FIGURA 1.6. Diagrama de fluxo do programa POLTRIN

2. LISTAGENS

2. Listagens

```

C
PROGRAM TRIANGUL! TRIANGULACAO SIMULTANEA
C*-----
C*
C* DADO UM CONJUNTO DE MATRIZES "AS", O PROGRAMA DETERMINA SE ELE
C* PODE SER SIMULTANEAMENTE TRIANGULARIZADO.
C* EM CASO AFIRMATIVO CALCULA UMA TRANSFORMACAO UNITARIA QUE
C* CONDUZ A TRIANGULARIZACAO SIMULTANEA DAS MATRIZES.
C*-----
C* DESCRICAO DAS VARIAVEIS
C*-----
C* AS      EM CADA ETAPA CONTEM UM CONJUNTO DE MATRIZES PARA O QUAL O
C*          PROGRAMA PROCURARA UM VETOR PROPRIO COMUM.
C* AT      CONTEM A TRANSFORMADA DE UMA MATRIZ.
C* M      NUMERO DE MATRIZES.
C* N      DIMENSAO DAS MATRIZES.
C* U      EM CADA ETAPA CONTEM UMA MATRIZ UNITARIA DE DIMENSAO
C*          APROPRIADA QUE CONTEM NA PRIMEIRA COLUNA O VETOR PROPRIO
C*          COMUM A TODAS AS MATRIZES "AS".
C* UTUT    MATRIZ UNITARIA QUE AGLIHA AS TRANSFORMACOES "U".
C* RESPAR  = .TRUE. IMPRIME RESULTADOS PARCIAIS NA UNIDADE 6.
C*-----
C*          ARQUIVOS DE ENTRADA
C*-----
C*          UNIDADE  ARQUIVO
C*-----
C*          1        MATRIZEL SAIDA
C*-----
C*          ARQUIVOS DE SAIDA
C*-----
C*          UNIDADE  ARQUIVO
C*-----
C*          2        TRANS_U SAIDA
C*          6        TELA SAIDA
C*          22       TRIAMEA SAIDA
C*-----
C
INCLUDE 'parametros.f'
INCLUDE 'common0.f'
C
LOGICAL*1 ACHOU,RESPAR,ERRO
INTEGER*4 I, IDIM, J, M, N, NDIM
COMPLEX*16 U(MAXN,MAXN),UTUT(MAXN,MAXN),X(MAXN)
C
INICIALIZA A TRANSFORMACAO.
C
DO I=1,MAXN
  DO J=1,MAXN
    UTUT(J,I) = ZERO
  ENDDO
  UTUT(I,I) = UM
ENDDO
C
OPEN(UNIT = 6,
$  STATUS = 'NEW',
$  NAME = 'tela.saída')
C
OPEN(UNIT = 1,
$  ERR = 9999,
$  STATUS = 'OLD',
$  NAME = 'matrizel.saída')
C
OPEN(UNIT = 2,
$  STATUS = 'NEW',
$  NAME = 'trans_u.saída')
C
OPEN(UNIT = 22,
$  STATUS = 'NEW',
$  NAME = 'trineas.saída')
C
ACHOU = .FALSE.
CALL LEITURA_MATRIZES (M,N,RESPAR,ERRO)
IF (ERRO) THEN
  WRITE(22,12000) 0

```

```

GO TO 10000
ENDDIF
C
DO I=1,M
  IAS(I) = I
ENDDO
C
DO IDIM=1,N-1
  ! AS SUB-ROTINAS FORAM IMPLEMENTADAS PARA N>1
  NDIM = N - IDIM + 1
  CALL AUTOVETOR_COMUM (NDIM,M,X,RESPAR,ACHOU,ERRO)
  IF (ERRO) THEN
    WRITE(6,*) 'IDIM = ',IDIM
    GO TO 10000
  ENDDIF
  IF (.NOT. ACHOU) THEN
    WRITE (6,11000)
    WRITE (22,12000) 0
    GO TO 9998
  ENDDIF
  CALL SALVA (MAXN,1,NDIM,1,X,U(1,1))
  CALL GERA_MATRIZ_ORCUNORMAL (NDIM,1,U)
  CALL PRODUTO_DE_ORCUNORMALS (N,NDIM,UTUT,U)
C
  REDUCAO DAS MATRIZES
C
  DO I=1,N
    CALL REDUCE_MATRIZ (NDIM,AS(1,1,I,IAS(I)),U(1,1))
  ENDDO
  IF (RESPAR) CALL SAIDA_PARCIAL (NDIM-1,N,M,X,U)
  CALL JOGA_FORA_MATRIZ_NULA (M,NDIM-1)
  IF (M .EQ. 0) GO TO 9997
  ! ACHOU TRANSFORMACAO
ENDDO
  ! ACHOU TRANSFORMACAO
9997 WRITE (22,12000) 1
C
  WRITE (2,15000)
  CALL GRAVA_MATRIZ (2,N,UTUT)
C
9998 CLOSE (UNIT=2)
CLOSE (UNIT=6)
CLOSE (UNIT=22)
STOP 'SAIDA NORMAL --- TRIANGULACAO SIMULTANEA '
C
9999 WRITE (6,13000)
10000 CLOSE (UNIT=6)
STOP 'SAIDA COM ERRO --- TRIANGULACAO SIMULTANEA '
C
11000 FORMAT (' O SISTEMA NAO E TRIANGULARIZAVEL.')
12000 FORMAT (' ',15)
13000 FORMAT (' ERRO NA ABERTURA DA UNIDADE 1')
15000 FORMAT (' ----- MATRIZ U -----,')
END
C
SUBROUTINE AUTOVETOR_COMUM (N,M,ETA,RESPAR,ACHOU,ERRO)
C*-----
C*
C* GERENCIA O CALCULO DE UM AUTOVETOR COMUM A TODAS AS MATRIZES. SE
C* AS MATRIZES NAO TEM AUTOVETOR COMUM "ACHOU" = .FALSE.
C*-----
C
INCLUDE 'parametros.f'
INCLUDE 'common0.f'
C
LOGICAL*1 ACHOU,ERRO,RESPAR
INTEGER*4 N
COMPLEX*16 ETA(0)
C
LOGICAL*1 VOLTA
INTEGER*4 I,K
COMPLEX*16 LAMEDA
C
ETA(1) = UM
DO I=2,N
  ETA(I) = ZERO
ENDDO
C

```

```

DO K=1,M
IF (RESPAR) WRITE (6,*) 'AUTOVETOR COMUM, K = ',K
CALL AUTOVETOR (N,AS(1,1,IAS(1)),ETA,LAMEDA)
IF (ERRO) THEN
WRITE (6,*) 'AUTOVETOR COMUM 1, K = ',K
RETURN
ENDIF
IF (M.GT. 1) THEN
DO I=2,M
IF (RESPAR) WRITE (6,*) 'AUTOVETOR COMUM, I = ', I
CALL AUTOVETOR (N,AS(1,1,IAS(I)),ETA,LAMEDA,RESPAR,ERRO)
IF (ERRO) THEN
WRITE (6,*) 'AUTOVETOR COMUM 2, K = ',K,' I = ',I
RETURN
ENDIF
CALL AUTOVS (N,I,ETA,LAMEDA,VOLTA,RESPAR,ERRO)
IF (ERRO) THEN
WRITE (6,*) 'AUTOVETOR COMUM 3, K = ',K,' I = ',I
RETURN
ENDIF
ENDIF
IF (.NOT. VOLTA) GO TO 20
ENDDO
ENDIF
ACHOU = .TRUE.
RETURN
CONTINUE
20 ENDDO
ACHOU = .FALSE.
C
RETURN
END
C
C
C
SUBROUTINE AUTOVETOR (N,A,ETA,LAMEDA,RESPAR,ERRO)
C *****
C *
C * COORDENA O CALCULO DE UM AUTOVETOR DA MATRIZ "A" QUE SEJA UM
C * POLINOMIO ESCALAR NA MESMA MATRIZ AVALIADO NO VETOR "ETA".
C *
C * *****
C
INCLUDE 'parametros.f'
C
LOGICAL*1 ERRO,RESPAR
INTEGER*4 N
COMPLEX*8 A(MAXN,MAXN)
COMPLEX*16 ETA(0),LAMEDA
C
INTEGER*4 I,M,M2
COMPLEX*16 Q1,Q2,Q(MAXN),QINV(MAXN),XI(MAXN)
C
CALL POLINOMIO_MOMMO (N,A,ETA,M,Q)
C
C "QINV" E O POLINOMIO QUE TEM COMO RAIZES AS O INVERSO DA RAIZES DE "Q"
C
M2 = M + 2
DO I=1,M+1
QINV(I) = Q(M2-I)
ENDDO
C
IF (RESPAR) WRITE (6,*) (' Q(' ,I,') = ',Q(I),I=1,M+1)
C
C CALCULO DA MELHOR RAIZ DE "QINV".
C
LAMEDA = ZERO
IF (M.GT. 1 .AND. CDABS(Q(1)) .NE. 0.) THEN
CALL LAGUERRE (QINV,M,LAMEDA,EPSS,.FALSE.,ERRO)
IF (ERRO) THEN
WRITE (6,300)
CLOSE (UNIT=2)
RETURN
ENDIF
ELSE
LAMEDA = UM
ENDIF
C
C CALCULO DA MAIOR RAIZ DE "Q" USANDO COMO VALOR INICIAL O INVERSO DO
C VALOR OBTIDO NO PASO ANTERIOR

```

```

C
LAMEDA = 1./LAMEDA
CALL LAGUERRE (Q,M,LAMEDA,EPSS,.TRUE.,ERRO)
IF (ERRO) THEN
WRITE (6,300)
CLOSE (UNIT=2)
RETURN
ENDIF
C
C DEFLACAO
C
Q1 = Q(M+1)
DO I=M,1,-1
Q2 = Q(I)
Q(I) = Q1
Q1 = LAMEDA*Q1 + Q2
ENDDO
C
IF (RESPAR) WRITE (6,*) (' Q(' ,I,') = ',Q(I),I=1,M+1)
C
CALL VANLOAN (N,M-1,A,ETA,Q,XI) ! AVALIA XI = Q(A)ETA
CALL NORMALIZACAO (N,XI,ERRO)
IF (ERRO) THEN
WRITE (6,100)
CLOSE (UNIT=2)
RETURN
ENDIF
C
C MELHORA O AUTOVALOR E O AUTOVETOR.
C
CALL MELHORA_AUTOVETOR (N,A,XI,LAMEDA,RESPAR,ERRO)
IF (ERRO) THEN
WRITE (6,200)
CLOSE (UNIT=2)
RETURN
ENDIF
C
CALL SALVA (MAXN,1,N,1,XI,ETA)
C
RETURN
100 FORMAT (' VETOR MUITO PEQUO DO VETOR NULO --- ROTINA AUTOVETOR?')
200 FORMAT (' ERRO EM AUTOVETOR ')
END
C
C
C
SUBROUTINE AUTOVS (N,IS,ETA,LAMEDA,RESPAR,VOLTA,ERRO)
C *****
C *
C * VERIFICA SE ETA E AUTOVETOR DAS MATRIZES "AS" COM INDICE MENOR A
C * "IS". EM CASO AFIRMATIVO "VOLTA" = .TRUE.
C * CASO CONTRARIO "VOLTA" = .FALSE. E "ETA" E UM VETOR QUE NAO PERTENCE
C * AO NUCLEO DO COMPUTADOR "CIS". ONDE "I" E UM INDICE TAL QUE "ETA"
C * NAO E' AUTOVETOR DE "A".
C *
C * *****
C
INCLUDE 'parametros.f'
INCLUDE 'common0.f'
C
LOGICAL*1 ERRO,RESPAR,VOLTA
INTEGER*4 N
COMPLEX*16 ETA(0),LAMEDA
C
INTEGER*4 IS,I,J
COMPLEX*16 AX(MAXN),Y(MAXN)
C
DO I=1,IS-1
CALL VERIFICA_AUTOVETOR (N,AS(1,1,IAS(I)),ETA,AX,RESPAR,VOLTA)
IF (.NOT. VOLTA) THEN
CALL PRODUTO_MATRIZ_VETOR (N,AS(1,1,IAS(IS)),AX,Y)
DO J=1,N
Y(J) = LAMEDA*AX(J) - Y(J)
ENDDO
CALL NORMALIZACAO (N,Y,ERRO)
IF (ERRO) THEN
CALL NOVO_VETOR (N,AS(1,1,IAS(I)),AS(1,1,IAS(IS)),ETA)
ELSE
CALL SALVA (MAXN,1,N,1,Y,ETA)

```

2. LISTAGENS

```

      ENDDIF
      RETURN
    ENDDIF
  ENDDO
C
  RETURN
  END
C
C
C
C
SUBROUTINE NOVO_VETOR (N, AI, AS, ETA)
C * =====
C *
C * PROCURA UM VETOR QUE NAO PERTECA AO NUCLEO DO COMPUTADOR "CIS"
C * ENTRE OS QUE SEJAM DA FORMA UM POLINOMIO ESCALAR AVALIADO EM "AS"
C * VEZES "ETA".
C *
C * =====
C
  INCLUDE 'parametros.f'
C
  INTEGER*4 N
  COMPLEX*16 AI(MAXN, MAXN), AS(MAXN, MAXN), ETA(MAXN)
C
  LOGICAL*1 ERRO
  INTEGER*4 J, K
  COMPLEX*16 CIS(MAXN, MAXN), X(MAXN)
C
C CALCULO DO COMPUTADOR DE "AI" E "AS".
C
  DO J=1, N
    CALL PRODUTO_MATRIZ_VETOR (N, AI, AS(1, J), CIS(1, J))
    CALL PRODUTO_MATRIZ_VETOR (N, AS, AI(1, J), X)
    DO K=1, N
      CIS(K, J) = CIS(K, J) - X(K)
    ENDDO
  ENDDO
C
  ERRO = .FALSE.
  DO J=1, N
    CALL PRODUTO_MATRIZ_VETOR (N, CIS, ETA, X)
    CALL NORMALIZACAO (N, X, ERRO)
    IF (ERRO) THEN
      CALL PRODUTO_MATRIZ_VETOR (N, AS, X, ETA)
    ELSE
      CALL SALVA (MAXN, 1, N, 1, X, ETA)
    RETURN
  ENDDIF
  ENDDO
C
  STOP ' PROBLEMAS DE PROGRAMACAO EM NOVO_VETOR'
C
  END
C
C
C
SUBROUTINE POLINOMIO_MINIMO (N, A, ETA, M, Q)
C * =====
C *
C * CALCULA O POLINOMIO "Q" DE MENOR GRAU TAL QUE "Q(A)ETA = 0".
C * EMPREGA O METODO DE TRIANGULACAO E PROCURA PELO PRIMEIRO PIVOTE NULO,
C * POIS OS VETORES X, AX, AAX, ... , AA...AX SAO NECESSARIAMENTE
C * LINEARMENTE DEPENDENTES.
C *
C * =====
C
  INCLUDE 'parametros.f'
C
  INTEGER*4 N, M
  COMPLEX*16 A(MAXN, MAXN), ETA(M), Q(M+1)
C
  INTEGER*4 BP(MAXN), I, J, JEP, K, KEP
  REAL*4 NORMA_DEFINITO, NORMA
  REAL*8 XPI
  COMPLEX*16 X1, X(MAXN), B(MAXN, MAXN), AX(MAXN)
C
  DO I=1, N
    Q(I) = ZERO
    BP(I) = I
  ENDDO

```

```

    X(I) = ETA(I)
    AX(I) = X(I)
  ENDDO
C
  NORMA = NORMA_DEFINITO (N, A)
  IF (NORMA .LT. EPSS**2) NORMA = 1.
C
  DO I=1, N
    CALL ACHA_PIVO (N, I, AX, BP)
    X1 = AX(BP(I))
    XPI = CDABS(X1)
    IF (XPI .LE. NORMA*EPSS) THEN
      M = I-1
      GO TO 120
    ENDDIF
C
    DO J=1, I
      B(J, I) = AX(BP(J))
    ENDDO
    DO J=I+1, N
      B(J, I) = - AX(BP(J))/X1
    ENDDO
    ! CALCULO DOS MULTIPLICADORES
C
    CALL PRODUTO_MATRIZ_VETOR (N, A, X, AX)
    CALL SALVA (MAXN, 1, N, 1, AX, X)
    ! CALCULA OUTRA POTENCIA
C
    DO J=1, I
      ! APLICA OS MULTIPLICADORES A NOVA COLUNA
      JEP = BP(J)
      DO K=J+1, N
        KEP = BP(K)
        AX(KEP) = AX(KEP) + B(K, J)*AX(JEP)
      ENDDO
    ENDDO
  ENDDO
  M = N
C
  120 Q(M+1) = (1., 0.)
  DO I=1, M
    Q(I) = -AX(BP(I))
  ENDDO
C
  DO I=N, 1, -1
    ! CALCULO DOS COEFICIENTES DO POLINOMIO
    DO J=I+1, M
      Q(I) = Q(I) - Q(J)*B(I, J)
    ENDDO
    Q(I) = Q(I)/B(I, I)
  ENDDO
C
  RETURN
  END
C
C
C
SUBROUTINE VERIFICA_AUTOVETOR (N, A, X, AX, RESPAR, SIM)
C * =====
C *
C * VERIFICA SE "X" E AUTOVETOR DE "A". SE NAO FOR O CASO "AX" CONTEM
C * O PRODUTO "A"."X".
C *
C * =====
C
  INCLUDE 'parametros.f'
C
  LOGICAL*1 RESPAR, SIM
  INTEGER*4 N
  COMPLEX*16 A(MAXN, MAXN), X(M), AX(M)
C
  INTEGER*4 I
  REAL*4 CNDORM
  COMPLEX*16 PIVTOR, LAMBDA, Y(MAXN)
C
  CALL PRODUTO_MATRIZ_VETOR (N, A, X, AX)
C
  IF (RESPAR) THEN
    DO I = 1, N
      WRITE(6, *) 'X(' , I, ') = ', X(I), 'AX(' , I, ') = ', AX(I)
    ENDDO
  ENDDIF
C

```


2. LISTAGENS

```

2 STATUS = 'NEW',
* NAME = 'tela.saída')
C
OPEN(UNIT = 1,
1 ERR = 99999,
2 STATUS = 'OLD',
* NAME = 'poltrian.dados')
C
OPEN(UNIT = 2,
2 STATUS = 'NEW',
* NAME = 'poltrian.saída')
C
OPEN(UNIT = 24,
2 STATUS = 'NEW',
* NAME = 'poltrim.saída')
C
CALL LECTURA_POLINOMIO (M,N,RESPAR,ERR0)
IF (ERR0) GO TO 99998
C
CALL POLINOMIOS (M,N,RAIZES,IDIFER,MODIF,ERR0)
IF (ERR0) THEN
GO TO 99998
ELSE
IF (RESPAR) CALL SAIDA_PARCIAL2 (M,N,RAIZES)
ENDIF
C
DO J=1,N
RAIZUT(J) = 0
EMDDO
EQUAC = N + 1
ACHU = .FALSE.
C
IF (RESPAR) THEN
WRITE (2,*) N, '.true.'
ELSE
WRITE (2,*) N, '.false.'
ENDIF
100 EQUAC = EQUAC - 1
IF (EQUAC .EQ. 0) THEN ! ENCONTROU UMA SOLUCAO
WRITE (2,10000)
CALL GRAVA_MATRIZ_TRIANGULAR (2,N,T)
WRITE (2,12000)
CALL GRAVA_MATRIZ_TRIANGULAR (2,N,BS(1,M))
ACHU = .TRUE.
EQUAC = 1
ENDIF
110 RAIZUT(EQUAC) = RAIZUT(EQUAC) + 1
IF (RAIZUT(EQUAC) .GT. MODIF(EQUAC)) THEN
C
C NAO ACHU SOLUCAO PARA ESE CONJUNTO DE RAIZES. VAI MUDAR O CONJUNTO
C
EQUAC = EQUAC + 1
IF (EQUAC .EQ. N+1) THEN ! O PROBLEMA NAO TEM SOLUCAO
WRITE (6,13000)
CLOSE (UNIT=2)
IF (ACHU) THEN
WRITE (24,*) 1
CLOSE (UNIT=24)
CLOSE (UNIT=6)
STOP 'O PROBLEMA NAO TEM MAIS SOLUCOES -- POLTRIAN'
ELSE
WRITE (24,*) 0
CLOSE (UNIT=24)
CLOSE (UNIT=6)
STOP 'O PROBLEMA NAO TEM SOLUCOES -- POLTRIAN'
ENDIF
ELSE
DO J=1,EQUAC-1
RAIZUT(J) = 0
EMDDO
GO TO 110
ENDIF
ENDIF
C
C JA TEMS UM NOVO CONJUNTO. PODE SER UMA AMPLIACAO.
C
J = RAIZUT(EQUAC)
K = IDIFER(J,EQUAC)
RR = RAIZES(K,EQUAC)

```

```

EE = APOSTA (EQUAC,EQUAC,N)
BS(EE,0) = RR
COLUMA = EQUAC
CALL PRODUTO_ELEMENTO_DIAGONAL (M,N,EQUAC)

```

```

C
C TENTA RESOLVER TODAS AS EQUACOES LINEARES A DIREITA.
C
120 COLUMA = COLUMA + 1
IF (COLUMA .GT. N) GO TO 100 ! CONSIGUIU RESOLVER A LINHA
C
CALL CALCULA_ELEMENTO (M,N,EQUAC,COLUMA)
EC = APOSTA (EQUAC,COLUMA,N)
C
C RESOLVE A EQUACAO RESULTANTE DESDE QUE POSSIVEL.
C
C SE POR ACASO PENHO FOR "0. ASSUMIRA QUE A SOLUCAO NAO EXISTE SE SE XX
C NO FOR "0. OMBE "0 QUER DIZER QUE O MODULO E MENOR DO QUE "EPSS**4".
C

```

```

IF (CDABS(PENHO) .LT. EPSS**4) THEN ! COEF = ZERO
IF (CDABS(ORDO) .LT. EPSS**4) THEN ! ORD = ZERO

```

```

ASSUMIMOS QUE A SOLUCAO E ZERO.

```

```

T(EC) = ZERO
DIVO(EC) = .TRUE.
CALL PRODUTO_ELEMENTO_FURA (M,N,EQUAC,COLUMA)
ELSE

```

```

NAO TEM SOLUCAO

```

```

GO TO 110
ENDIF
ELSE

```

```

CALCULA A SOLUCAO.

```

```

T(EC) = - ORDO/PENHO
DIVO(EC) = .FALSE.
CALL PRODUTO_ELEMENTO_FURA (M,N,EQUAC,COLUMA)
IF (RESPAR) THEN
WRITE (6,*) 'PRODUTOS PARCIAIS NO HORROR'
DO J=1,N
WRITE (6,*) ' J = ', J
WRITE (6,*) ' PEN = ', PEN(J), ' ORD = ', ORD(J)
CALL GRAVA_MATRIZ_TRIANGULAR (6,N,BS(1,J))
EMDDO
ENDIF
ENDIF
GO TO 120

```

```

C
99999 WRITE(6,10000)
99998 CLOSE (UNIT=6)
STOP 'SAIDA COM ERRO -- POLTRIM'

```

```

C
10000 FORMAT (' ERRO NA ABERTURA DA UNIDADE 1')
11000 FORMAT (' --- SOLUCAO --- ')
12000 FORMAT (' --- AVALIACAO --- ')
13000 FORMAT (' O PROBLEMA NAO TEM MAIS SOLUCOES')
END

```

```

SUBROUTINE CALCULA_ELEMENTO (M,N,LINHA,COLUMA)

```

```

C *****
C *
C * ESTA SUBROTINA EFETIVIZA A AVALIACAO DE UM POLINOMIO COM COEFI-
C * CIENTES TRIANGULARES NUMA MATRIZ TRIANGULAR. SUPOE QUE NA POSICAO
C * (FILA,COLUMA) TEMS UMA INCOGNITA E QUE TEMS CALCULADO O POLINOMIO
C * QUANDO ESA INCOGNITA E ZERO.
C * ESTAMOS SUPONDO QUE SEMPRE "LINHA" E MENOR QUE "COLUMA"
C *
C *****

```

```

INCLUDE 'parametros.f'
INCLUDE 'common1.f'
INCLUDE 'common2.f'

```

```

INTEGER*4 COLUMA,LINHA,N,N

```

```

C INTEGER*4 J,K,MJ,MJ1,APONTA,LL,CC,LC,LK,EC

```

```

C MJ1 = M + 1
C LL = APONTA (LINHA,LINHA,M)
C LC = APONTA (LINHA,COLUNA,M)
C CC = APONTA (COLUNA,COLUNA,M)
C PEN(1) = COEF(LL,MJ1)
C ORD(1) = COEF(LC,M)

```

```

C DO K=LINHA+1,COLUNA
C LK = APONTA (LINHA,K,M)
C KC = APONTA (K,COLUNA,M)
C ORD(1) = ORD(1) + COEF(LK,MJ1)*T(KC)
C ENDDO

```

```

C DO J=2,M
C MJ1 = J - 1
C MJ = M - J + 1
C PEN(J) = T(CC)*PEN(MJ1) + BS(LL,MJ1)
C ORD(J) = T(CC)*ORD(MJ1) + COEF(LC,MJ)
C DO K=LINHA+1,COLUNA-1
C LK = APONTA (LINHA,K,M)
C KC = APONTA (K,COLUNA,M)
C ORD(J) = ORD(J) + BS(LK,MJ1)*T(KC)
C ENDDO
C ENDDO

```

```

C RETURN
C END

```

```

C SUBROUTINE PRODUTO_ELEMENTO_DIAGONAL (M,N,LINHA)

```

```

C *****
C * REALIZA O PRODUTO QUANDO ACRESCENTAMOS UM UNICO ELEMENTO NA DIA-
C * GONAL, NA LINHA "LINHA".
C * AO MESMO TEMPO REALIZA O REEMPLAZO DOS ELEMENTOS (LINHA,LINHA)
C * NAS MATRIZES BS
C *****

```

```

C INCLUDE 'parametros.f'
C INCLUDE 'common1.f'
C INCLUDE 'common2.f'

```

```

C INTEGER M,N,LINHA
C INTEGER J,APONTA,LL

```

```

C LL = APONTA (LINHA,LINHA,M)
C BS(LL,1) = T(LL)*COEF(LL,M+1) + COEF(LL,M)
C DO J=2,M
C BS(LL,J) = T(LL)*BS(LL,J-1) + COEF(LL,M-J+1)
C ENDDO

```

```

C RETURN
C END

```

```

C SUBROUTINE PRODUTO_ELEMENTO_FORA (M,N,LINHA,COLUNA)

```

```

C *****
C * REALIZA O PRODUTO QUANDO ACRESCENTAMOS UM UNICO ELEMENTO FORA DA
C * DIAGONAL, NA NA POSICAO (LINHA,COLUNA).
C *****

```

```

C INCLUDE 'parametros.f'
C INCLUDE 'common1.f'
C INCLUDE 'common2.f'

```

```

C INTEGER M,N,LINHA,COLUNA
C INTEGER J,APONTA,LC

```

```

C LC = APONTA (LINHA,COLUNA,M)
C DO J=1,M

```

```

BS(LC,J) = PEN(J)*T(LC) + ORD(J)

```

```

C ENDDO
C RETURN
C END

```

```

C SUBROUTINE LEITURA_POLINOMIO (M,N,RESPAR,ERRO)

```

```

C *****
C * ESTA SUBROUTINA LE A UNIDADE 1, ARQUIVO 'POLINOMIO DADOS A1'
C * QUE CONTEM OS DADOS DO PROBLEMA.
C *****

```

```

C INCLUDE 'parametros.f'
C INCLUDE 'common1.f'

```

```

C LOGICAL*1 RESPAR,ERRO
C INTEGER*4 M,N

```

```

C INTEGER J,K,L,T,APONTA

```

```

C ERRO = .FALSE.
C READ(1,*) M,N,RESPAR
C IF (M.GT.MAXM .OR. N.GT.MAXN) THEN
C WRITE(2,100)
C CLOSE(UNIT=1)
C CLOSE(UNIT=2)
C ERRO = .TRUE.
C RETURN
C ENDDIF

```

```

C DO 50 J=M+1,-1
C DO 40 K=1,M
C T = APONTA (K,K,M)
C READ(1,*) (COEF(T+L,J),L=0,M-K)

```

```

40 CONTINUE
50 CONTINUE

```

```

C CLOSE(UNIT=1)
C RETURN

```

```

C 100 FORMAT(' DADOS FORA DOS LIMITES PERMITIDOS')
C END

```

```

C SUBROUTINE POLINOMIOS (M,N,RAIZES,IDIIFER,IDIIFER,ERRO)

```

```

C *****
C * GERA OS POSSIVEIS AUTOVALORES DOS SOLVENTES.
C *****

```

```

C INCLUDE 'parametros.f'
C INCLUDE 'common1.f'

```

```

C LOGICAL*1 ERRO
C INTEGER*4 M,N,IDIIFER(MAXM1,MAXM),IDIIFER(MAXM)
C COMPLEX*16 RAIZES(MAXM1,MAXM)

```

```

C INTEGER*4 APONTA,J,K,K1,L
C COMPLEX*16 POL(MAXM),RAIZ(MAXM1)

```

```

C GERA E RESOLVE OS POLINOMIOS UNIDIMENSIONAIS DA DIAGONAL.

```

```

C ERRO = .FALSE.

```

```

C DO J=1,M
C K1 = APONTA (J,J,M)
C DO K=1,M+1
C POL(K) = COEF(K1,K)
C ENDDO
C CALL ZROOTS (POL,M,RAIZ,.TRUE.,ERRO)
C IF (ERRO) THEN
C WRITE (6,*) 'ERRO EM ZROOTS'
C RETURN

```

```

! SOBRE AS MATRIZES
! SOBRE AS LINHAS

```

```

! CALCULO RAIZES

```


2. LISTAGENS

```

      ENDIF
      DO K=1,M
        RAIZES(K,J) = RAIZ(O)
      ENDDO
    ENDDO
  C
  C PARA MINIMIZAR O TRABALHO DESCARTAMOS A MULTIPLICIDADE DAS RAIZES
  C DE CADA POLINOMIO
  C
  DO J=1,N
    IDIFER(1,J) = 1
    L = 1
    DO K=2,M
      Z = RAIZES(K,J)
      DO K1=1,L
        IF (Z .EQ. RAIZES(IDIFER(K1,J),J)) GO TO 10
      ENDDO
      L = L+1
      IDIFER(L,J) = K
    CONTINUE
  10  ENDDO
      IDIFER(J) = L
    ENDDO
  C
  C RETURN
  C
  C
  C
  C SUBROUTINE LETURA_MATRIZ (UNI,N,A)
  C *****
  C *
  C * ESTA SUBROUTINA LE A MATRIZ "A" (ARMAZENADA POR LINHAS) DA
  C * UNIDADE UNI
  C * *****
  C
  C INCLUDE 'parametros.f'
  C
  C INTEGER*4 N,UNI
  C COMPLEX*16 A(MAXN,MAXN)
  C
  C INTEGER*4 K,L
  C DO K=1,N
    READ(UNI,*) (A(K,L),L=1,N)
  ENDDO
  C
  C RETURN
  C
  C
  C
  C SUBROUTINE LETURA_MATRIZES (M,N,RESPAR)
  C *****
  C *
  C * ESTA SUBROUTINA LE A UNIDADE 1 COM OS DADOS DO PROBLEMA.
  C * O ARQUIVO TEM A SEGUINTE ESTRUTURA:
  C *
  C * NA PRIMEIRA LINHA DUAS CONSTANTES INTEIRAS E UMA LOGICA. (VER
  C * PROGRAMA PRINCIPAL)
  C *
  C * NA SEQUENTES LINHAS "M" MATRIZES DE ORDEN "M"*"M".
  C * *****
  C
  C INCLUDE 'parametros.f'
  C INCLUDE 'common0.f'
  C
  C LOGICAL*1 RESPAR
  C INTEGER*4 M,N
  C
  C INTEGER*4 J,K,L
  C
  C READ(1,*) M,N,RESPAR
  C IF (M .GT. MAXN .OR. N+1 .GT. MAXN) THEN
    WRITE(6,100)
    CLOSE(UNIT=1)
    CLOSE(UNIT=2)
    STOP ' SAIDA COM ERRO'
  ENDDIF

```

```

  C
  C DO J=N,1,-1
    DO K=1,N
      READ(1,*) (AS(K,L,J),L=1,N)
    ENDDO
  ENDDO
  C
  C RETURN
  C
  C FORMAT(' DADOS FORA DOS LIMITES PERMITIDOS')
  C
  C
  C
  C SUBROUTINE ACHA_PIVO (N,IS,X,INDICE)
  C *****
  C *
  C * SEJA O VETOR "X", ORDENADO SEGUNDO INDICE, E DADO "IS" <= "N". A
  C * SUB-ROUTINA ACHA A MAIOR COMPONENTE DE "X" COM INDICE MAIOR O IGUAL
  C * A "IS" Y REORDENA INDICE PARA QUE ESTE VALOR ESTE NA POSICAO "IS".
  C * *****
  C
  C
  C INTEGER*4 N,IS,INDICE(O)
  C COMPLEX*16 X(O)
  C
  C INTEGER*4 I,J
  C REAL*4 XH1,XH2
  C
  C XH1 = CDABS(X(INDICE(IS)))
  C J = IS
  C DO I=IS+1,N
    XH2 = CDABS(X(INDICE(I)))
    IF (XH2 .GT. XH1) THEN
      XH1 = XH2
      J = I
    ENDDIF
  ENDDO
  C
  C I = INDICE(IS)
  C INDICE(IS) = INDICE(J)
  C INDICE(J) = I
  C
  C RETURN
  C
  C
  C
  C SUBROUTINE VANLOAN (N,M,A,Z,P,Y)
  C *****
  C *
  C * CALCULA P(A)Z = Y DA FORMA MAIS EFICIENTE. VAN LOAN AND ....
  C * *****
  C
  C INCLUDE 'parametros.f'
  C
  C INTEGER*4 N,M
  C COMPLEX*16 A(MAXN,MAXN),P(O+1),Y(O),Z(O)
  C
  C INTEGER*4 I,J,K,MORF,MORPR,PM1,PM
  C COMPLEX*16 EJ(MAXN,MAXN),AS(MAXN,MAXN),X(MAXN),AX(MAXN)
  C
  C IF (M .EQ. 0) THEN
    DO I=1,N
      Y(I) = Z(I)*P(1)
    ENDDO
    RETURN
  ENDDIF
  C
  C RM = IFIX(SQRT(OH/2.))
  C DO I=1,N
    CALL SALVA (MAXN,1,M,1,1,A(1,D),AS(1,D))
    Y(I) = ZERO
  ENDDO
  C
  C
  C CALCULA A**PM.
  C
  C DO K=1,PM-1
    DO I=1,N

```


2. LISTAGENS

```

C      INCLUDE 'parametros.f'
C
C      INTEGER*4 N
C      COMPLEX*16 X(O), AX(O), A(MAXN,MAXN)
C
C      INTEGER*4 I, J, INDICE(MAXN)
C      REAL*4      APXY(MAXN)
C      COMPLEX*16 PXY(MAXN), SOMA
C
C      DO J=1, N
C        DO I=1, N
C          PXY(I) = A(J, I) * X(I)
C          APXY(I) = CDABS(PXY(I))
C        ENDDO
C
C      CALL ORDENA (N, APXY, INDICE)
C
C      SOMA = ZERO
C      DO I=N, 1, -1
C        SOMA = SOMA + PXY(INDICE(I))
C      ENDDO
C      AX(J) = SOMA
C      ENDDO
C
C      RETURN
C      END
C
C
C
C      SUBROUTINE ORDENA (NDIM, LISTA, ORDEM)
C *-----
C *
C *      ESTA SUBROTINA DA COMO RESULTADO O VETOR ORDEM TAL QUE ORDEM(K)
C *      O NUMERO DA COMPONENTE DE LISTA QUE TEM A K-ESIMA MAIOR COMPONENTE.
C *-----
C
C      INTEGER*4 NDIM, ORDEM(NDIM)
C      REAL*4    LISTA(NDIM)
C
C      ORDEM(1) = 1
C      IF (NDIM .EQ. 1) RETURN
C
C      DO J=2, NDIM
C        VALOR = LISTA(J)
C        K = J-1
C        IF (K .GT. 0) THEN
C          IF (VALOR .GT. LISTA(ORDEM(K))) THEN
C            ORDEM(K+1) = ORDEM(K)
C            ORDEM(K) = J
C            K = K-1
C          ELSE
C            ORDEM(K+1) = J
C            K = 0
C          ENDDIF
C        GO TO 10
C      ENDDO
C
C      RETURN
C      END
C
C
C      INTEGER FUNCTION APONTA*4 (FILA, COLUMA, N)
C *-----
C *
C *      DADA A "FILA" E A "COLUMA" DE UM ELEMENTO DA MATRIZ ACHA A
C *      POSICAO DO ELEMENTO NO VETOR NO QUAL A MATRIZ ESTA ARMAZENADA.
C *      A MATRIZ E SUPOSTA DE ORDEN "N".
C *-----
C
C      INTEGER*4 FILA, COLUMA
C
C      APONTA = (FILA - 1) * (2 * N - FILA) / 2 + COLUMA
C
C      RETURN
C      END

```

```

C
C
C      SUBROUTINE LEITURA_MATRIZ_TRIANGULAR (N, AT)
C *-----
C *
C *      LE A MATRIZ "AT" DA UNIDADE 1.
C *-----
C
C      INCLUDE 'parametros.f'
C
C      INTEGER*4 N
C      COMPLEX*16 AT(MAXN2)
C
C      INTEGER*4 I, J, K, NMI
C
C      K = 0
C      DO I=0, N-1
C        NMI = N - I
C        READ (1, *) (AT(K+J), J=1, NMI)
C        K = K + NMI
C      ENDDO
C
C      RETURN
C      END
C
C
C      SUBROUTINE GRAVA_MATRIZ_TRIANGULAR (UNIT, N, AT)
C *-----
C *
C *      GRAVA A MATRIZ "AT" NA UNIDADE "UNIT".
C *-----
C
C      INCLUDE 'parametros.f'
C
C      INTEGER*4 N, UNIT
C      COMPLEX*16 AT(MAXN2)
C
C      INTEGER*4 I, J, K, NMI
C
C      K = 0
C      DO I=0, N-1
C        NMI = N - I
C        WRITE (UNIT, 100) (DREAL(AT(K+J)), DIMAG(AT(K+J)), J=1, NMI)
C        K = K + NMI
C      ENDDO
C
C      RETURN
C 100  FORMAT(' ', 2(' ', E14.7, ', ', E14.7, '), ', ', ', ')
C      END
C
C
C      SUBROUTINE GRAVA_MATRIZ (UNIT, N, U)
C *-----
C *
C *      GRAVA A MATRIZ "U" NA UNIDADE "UNIT".
C *-----
C
C      INCLUDE 'parametros.f'
C
C      INTEGER*4 N, UNIT
C      COMPLEX*16 U(MAXN, MAXN)
C
C      INTEGER*4 I, J
C
C      DO I=1, N
C        WRITE (UNIT, 100) (REAL(U(I, J)), DIMAG(U(I, J)), J=1, N)
C      ENDDO
C
C      RETURN
C 100  FORMAT(' ', 2(' ', E14.7, ', ', E14.7, '), ', ', ', ')
C      END
C
C

```

```

C
C SUBROUTINE SALVA (NDIM,MDIM,N,M,X,XS)
C *****
C * SALVA O CONTEUDO DA MATRIZ 'X' NA MATRIZ 'XS'. 'NDIM' E 'MDIM' SAO
C * AS DIMENSOES COM QUE FORAM DEFINIDAS AS MATRIZES 'X' E 'XS'.
C *
C *****
C
C INTEGER*4 N,M
C COMPLEX*16 X(NDIM,MDIM),XS(NDIM,MDIM)
C
C INTEGER*4 I,J
C
C DO I=1,M
C DO J=1,MDIM
C XS(J,I) = X(J,I)
C ENDDO
C
C RETURN
C END
C
C
C
C REAL FUNCTION NORMA_INFINITO*4 (N,A)
C *****
C * CALCULA A NORMA INFINITO DA MATRIZ "A".
C *
C *****
C
C INCLUDE 'parametros.f'
C
C INTEGER*4 N
C COMPLEX*16 A(MAXN,MAXN)
C
C INTEGER*4 I,J
C REAL*4 NORMA,NUEVA
C
C NORMA = 0.
C DO I=1,N
C NUEVA = 0.
C DO J=1,N
C NUEVA = NUEVA + CDABS(A(I,J))
C ENDDO
C IF (NUEVA .GT. NORMA) NORMA = NUEVA
C ENDDO
C
C NORMA_INFINITO = NORMA
C
C RETURN
C END
C
C
C
C SUBROUTINE GERA_MATRIZ_ORTONORMAL (N,N1,U)
C *****
C * CONSTRUI UMA BASE ORTONORMAL A PARTIR DAS "N1" PRIMEIRAS COLUNAS
C * DE "U" APLICANDO O PROCESSO DE ORTOGONALIZACAO DE GRAM-SCHMIDT AO
C * CONJUNTO. SE FOR PRECISO ACRESCENTA A BASE CANONICA.
C *
C *****
C
C INCLUDE 'parametros.f'
C
C INTEGER*4 N,N1
C COMPLEX*16 U(MAXN,MAXN)
C
C LOGICAL*1 ERR
C INTEGER*4 I,J,K
C COMPLEX*16 U1(MAXN,MAXN)
C
C K=2
C DO I=2,N1
C CALL GRAM_SCHMIDT (N,K,U,U(1,I),ERR)
C IF (.NOT. ERR) K = K + 1
C ENDDO

```

```

C
C IF (K .GT. N) RETURN ! COMPLETOU A BASE
C
C DO I=1,N
C DO J=1,N
C U1(J,I) = ZERO
C ENDDO
C U1(I,I) = UM
C ENDDO
C
C DO I=1,N
C CALL GRAM_SCHMIDT (N,K,U,U(1,I),ERR)
C IF (.NOT. ERR) THEN
C K = K + 1
C IF (K .GT. N) RETURN ! COMPLETOU A BASE
C ENDDO
C ENDDO
C STOP 'PARADA COM ERRO NA SUB-ROTTINA GERA_MATRIZ_ORTONORMAL'
C
C END
C
C
C
C SUBROUTINE GRAM_SCHMIDT (N,N1,U,X,ERR)
C *****
C * PROCESSO DE ORTOGONALIZACAO DE GRAM_SCHMIDT.
C * SUPOE QUE AS COLUNAS 1 ... N1-1 SAO VETORES ORTONORMAIS E APLICA O
C * PROCESSO AO VETOR X PARA GERAR UMA BASE COM MAIS UM ELEMENTO. SE
C * 'X' FOR LINEARMENTE DEPENDENTE ERR = .TRUE.
C *
C *****
C
C INCLUDE 'parametros.f'
C
C LOGICAL*1 ERR
C INTEGER*4 N,N1
C COMPLEX*16 U(MAXN,MAXN),X(MAXN)
C
C INTEGER*4 I,J
C COMPLEX*16 PINTUR,PIY
C
C DO I=1,N1-1
C PIY = PINTUR (N,X,U(1,I))
C DO J=1,N1
C X(J) = X(J) - PIY*U(J,I)
C ENDDO
C ENDDO
C
C CALL NORMALIZACAO (N,X,ERR)
C IF (.NOT. ERR) CALL SALVA (MAXN,1,N,1,X,U(1,N1))
C
C RETURN
C END
C
C
C
C SUBROUTINE GERA_ESTRELA (N,V)
C *****
C * SUBSTITUI A MATRIZ "V" PELA SUA TRANSPONSTA CONJUGADA.
C *
C *****
C
C INCLUDE 'parametros.f'
C
C INTEGER*4 N
C COMPLEX*16 V(MAXN,MAXN)
C
C INTEGER*4 I,J
C COMPLEX*16 X
C
C DO I=1,N
C V(I,I) = DCNUG(V(I,I))
C DO J=I+1,N
C X = DCNUG(V(I,J))
C V(I,J) = DCNUG(V(J,I))
C V(J,I) = X
C ENDDO

```

2. LISTAGENS

143

```
      ENDDO
C      RETURN
      END
C
C-----
C      SUBROUTINE JOGA_FORA_MATRIZ_NULA (M,N)
C-----
C*      ESTA SUB-ROTINA DEIXA DE LADO AS MATRIZES NULAS.
C*-----
C
C      INCLUDE 'parametros.f'
C      INCLUDE 'common0.f'
C
C      INTEGER*4 M,N
C      INTEGER*4 I,J,K,MPROV
C
C      MPROV = M
C      DO I=MPROV,1,-1
C        DO J=1,N
C          DO K=1,N
C            IF (AS(K,J,IASC(I)) .NE. ZERO) GO TO 1
C          ENDDO
C          K = IASC(I)
C          IASC(I) = IASC(J)
C          IASC(J) = K
C          M = M-1
C        CONTINUE
C      ENDDO
C
C      RETURN
      END
C
```


Apêndice A

Álgebra Linear

Neste apêndice, são apresentados alguns conceitos e símbolos de uso freqüente, como assim também alguns resultados simples ou pouco conhecidos aos quais se fez referência nos capítulos precedentes.

1. Definições

Denota-se por \mathbb{C}^r , o espaço dos vetores coluna complexos de dimensão r , e por $\mathbb{C}^{r \times s}$, a classe das matrizes complexas com r linhas e s colunas; a notação $\xi \in \mathbb{C}^r$ representa o vetor $\xi \in \mathbb{C}^r$ com coordenadas ξ_1, \dots, ξ_r , e $A = (a_{ij})_1^r$ denota uma matriz quadrada de dimensão r com entradas a_{ij} .

Para o vetor $\xi = (\xi_i)_1^r \in \mathbb{C}^r$, empregam-se com freqüência as seguintes normas vetoriais:

- (1) Norma infinito $\|\xi\|_\infty = \max_{i=1, \dots, r} |\xi_i|$,
- (2) Norma um $\|\xi\|_1 = \sum_{i=1}^r |\xi_i|$,
- (3) Norma dois $\|\xi\|_2^2 = \sum_{i=1}^r |\xi_i|^2$.

Se $A = (a_{ij})_1^n \in \mathbb{C}^{n \times n}$ for uma matriz, a norma matricial $\|\cdot\|$ induzida pela norma vetorial $\|\cdot\|$ é definida por:

$$\|A\| = \max_{\|\xi\|=1} \|A\xi\|.$$

Uma outra norma matricial muito empregada, é a chamada **norma de Frobenius**, a qual para toda matriz A é definida por:

$$\|A\|_F^2 = \text{tr}(AA^*) = \sum_{ij=1}^n |a_{ij}|^2,$$

onde $\text{tr}(A)$ representa o traço da matriz A , isto é,

$$\text{tr}(A) = \sum_{i=1}^r a_{ii},$$

e A^* é a matriz transposta conjugada de A .

Teorema 1.1 (Decomposição de Schur). Se A for uma matriz quadrada, existem uma matriz unitária U e uma matriz triangular superior T tais que:

$$U^*AU = T.$$

PROVA. A demonstração é muito simples e decorre por indução na dimensão de A , empregando o fato de que uma matriz tem pelo menos um autovetor. Para mais detalhes da prova ver Golub e Van Loan [30]. \square

1.1. Produto de Kronecker

O produto de Kronecker de duas matrizes é de grande utilidade para o estudo das equações polinomiais matriciais. Nesta seção, enumeram-se algumas das propriedades mais importantes deste produto.

Definição 1.1. Dadas as matrizes $A \in \mathbb{C}^{r \times s}$ e B dimensão arbitrária, o **produto de Kronecker** entre A e B é definido por:

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1s}B \\ a_{21}B & a_{22}B & \cdots & a_{2s}B \\ \vdots & \vdots & & \vdots \\ a_{r1}B & a_{r2}B & \cdots & a_{rs}B \end{pmatrix}.$$

Algumas das propriedades mais importantes do produto de Kronecker estão resumidas no seguinte teorema.

Teorema 1.2. *Se $A, B, C,$ e D forem matrizes com dimensões apropriadas, então as seguintes afirmações são verdadeiras:*

- (1) $(A \otimes B)(C \otimes D) = (AC \otimes BD).$
- (2) $(A \otimes B)^t = A^t \otimes B^t.$
- (3) $\overline{A \otimes B} = \overline{A} \otimes \overline{B}.$
- (4) $(A \otimes B)^* = A^* \otimes B^*.$
- (5) *Se A e B forem duas matrizes quadradas, então todos os autovalores de $A \otimes B$ são da forma $\lambda_A \lambda_B$, onde λ_A e λ_B são quaisquer autovalores de A e B respectivamente.*

PROVA. Ver Marcus e Minc [14]. □

A partir da afirmação (5) do teorema acima, obtém-se em forma direta o seguinte corolário.

Corolário 1.1. *Se A e B forem duas matrizes quadradas não singulares, então:*

- (1) $A \otimes B$ é não singular,
- (2) $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}.$

O produto de Kronecker pode ser utilizado para reescrever numa forma mais compacta o sistema $AXB = C$, da seguinte maneira

$$(B^t \otimes A)v(X) = v(C),$$

onde $v(X)$ é o vetor obtido a partir da matriz X , dispondo as colunas de X uma embaixo a outra num vetor coluna de dimensão n^2 , isto é, se X_1, \dots, X_n forem as colunas de X ,

$$v(X) = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{pmatrix}.$$

Usando as definições acima, o seguinte teorema tem uma demonstração simples.

Teorema 1.3 ([15]). *Dadas $A, B \in \mathbb{C}^{n \times n}$, a equação*

$$AX - XB = 0 \tag{1.1}$$

tem uma única solução (a trivial) se, e somente se, A e B não tiverem autovalores em comum.

PROVA. Dadas duas matrizes A e B arbitrárias, o Teorema 1.1 mostra que existem matrizes unitárias U e V e duas matrizes triangulares superiores T_A, T_{B^t} satisfazendo

$$U^*AU = T_A \quad V^*B^tV = T_{B^t}. \quad (1.2)$$

Da definição do produto de Kronecker, segue-se que a equação (1.1) é equivalente ao sistema

$$[(I \otimes A) - (B^t \otimes I)]v(X) = 0.$$

Como $(V \otimes U)^{-1} = V^* \otimes U^*$, a equação (1.2) e o Teorema 1.2 mostram que:

$$\begin{aligned} (V^* \otimes U^*) [(I \otimes A) - (B^t \otimes I)] (V \otimes U) &= (I \otimes U^*AU) - (V^*B^tV \otimes I) \\ &= (I \otimes T_A) - (T_{B^t} \otimes I). \end{aligned} \quad (1.3)$$

Assim sendo, existe uma solução não trivial de (1.2) se, e somente se, existir uma solução não trivial de (1.3). Levando-se em conta que os autovalores da matriz

$$(I \otimes T_A) - (T_{B^t} \otimes I) \quad (1.4)$$

são da forma

$$\lambda_A - \lambda_{B^t},$$

onde λ_A, λ_{B^t} são autovalores de A e B^t respectivamente, a matriz (1.4) é $\neq 0$ se, e somente se, A e B não tiverem autovalores em comum. \square

2. Matrizes definidas por blocos

Nesta seção, apresentam-se alguns resultados envolvendo matrizes definidas por blocos que foram empregados nos primeiros capítulos.

Uma matriz A é definida por blocos ou em blocos quando

$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1s} \\ A_{21} & A_{22} & \cdots & A_{2s} \\ \vdots & \vdots & \cdots & \vdots \\ A_{r1} & A_{r2} & \cdots & A_{rs} \end{pmatrix},$$

onde cada A_{ij} é uma matriz, neste caso, os A_{ij} são chamados **blocos** da matriz A e em geral são denotados por letras maiúsculas. A notação $(A_{ij})_1^r$ representa uma matriz definida por blocos com r linhas e r colunas.

A matriz transposta por blocos de A é a matriz B com blocos B_{ij} onde $B_{ij} = A_{ji}$; em geral, a matriz transposta por blocos de A é denotada por A^T , para distinguir-la da matriz transposta, para a qual emprega-se a notação usual A^t .

Note-se que a definição acima coincide com aquela da matriz transposta quando a dimensão dos blocos for 1.

A seguir, são enunciados dois lemas empregados no Capítulo II para o estudo da matriz companheira por blocos.

Lema 2.1. *Seja $A \in \mathbb{C}^{n \times n}$ uma matriz não singular. Neste caso, para cada $k \leq n$ existe uma permutação das colunas de A , \tilde{A} , tal que*

$$\tilde{A} = \begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{pmatrix},$$

onde $\tilde{A}_{11} \in \mathbb{C}^{k \times k}$, e os blocos \tilde{A}_{11} e \tilde{A}_{22} são matrizes inversíveis.

PROVA. Ver [11]. □

Lema 2.2. *Se $A \in \mathbb{C}^{nm \times nm}$ for uma matriz inversível, então existe uma permutação das colunas de A , $\tilde{A} = (\tilde{A}_{ij})_1^m$, onde $\tilde{A}_{ij} \in \mathbb{C}^{n \times n}$, tal que todos os blocos diagonais \tilde{A}_{ii} , são matrizes inversíveis.*

PROVA. A demonstração consiste em aplicar varias vezes o Lema 2.1. Ver [11]. □

2.1. Funções de matrizes triangulares por blocos

Nesta seção, apresenta-se um teorema simples sobre as funções de matrizes triangulares por blocos, de fundamental importância para o estudo das equações polinomiais matriciais com coeficientes triangulares superiores. O objetivo é mostrar que, apesar de X^r não ser linear como função dos elementos de X , os blocos acima da diagonal principal podem ser determinados pela solução de sistemas lineares.

Teorema 2.1. *Se $X \in \mathbb{C}^{n \times n}$ for uma matriz triangular superior por blocos e $r \leq 2$ for um inteiro arbitrário, então $(X^r)_{ij}$ só depende de X_{ij} e de blocos X_{kl} que ficam mais próximos da diagonal principal do que X_{ij} , isto é, $(X^r)_{ij}$ só depende dos blocos X_{kl} para os quais $l - k < j - i$. Além disso, o bloco $(X^r)_{ij}$ é linear em X_{ij} .*

PROVA. A afirmação decorre por indução em r . Seja $Y = X^r$ e seja $Z = X^{r-1}$, então

$$Y_{ij} = \sum_{s=i}^j Z_{is} X_{sj} = Z_{ij} X_{jj} + Z_{ii} X_{ij} + \sum_{s=i+1}^{j-1} Z_{is} X_{sj}. \quad (2.1)$$

O resultado é verdadeiro para $r = 2$, pois neste caso $Z = X$. Se o teorema for válido para $r - 1$, observe-se que na equação (2.1) o termo $\sum_{s=i+1}^{j-1} Z_{is} X_{sj}$ não depende de X_{ij} e os termos $Z_{ij} X_{jj}$ e $Z_{ii} X_{ij}$ são lineares em X_{ij} . Logo, o teorema fica demonstrado. \square

Observação 2.1. A partir da demonstração do Teorema 2.1, pode-se verificar facilmente que para $k \geq 2$, o bloco $(X^k)_{ij}$ satisfaz:

$$(X^k)_{ij} = \sum_{l=0}^{k-1} X_{ii}^l X_{ij} X_{jj}^{k-1-l} + W_{ij},$$

onde a matriz W_{ij} não depende de X_{ij} .

Do Teorema 2.1 obtém-se como corolário o seguinte resultado sobre matrizes triangulares.

Corolário 2.1 ([30]). *Se $X \in \mathbb{C}^{n \times n}$ for uma matriz triangular superior e $r \leq 2$ for um inteiro arbitrário, então X^r é uma matriz triangular superior, e o coeficiente $(X^r)_{ij}$ só depende de x_{ij} e das entradas x_{kl} que ficam mais próximas da diagonal principal do que x_{ij} . Além disso, $(X^r)_{ij}$ é linear em x_{ij} .*

Uma descrição bem detalhada dos elementos de X^r , quando X for uma matriz triangular superior, é fornecida por Golub e Van Loan [30].

Corolário 2.2. *Se X for uma matriz triangular superior por blocos e $p(z)$ for um polinômio escalar, então a matriz $Y = p(X)$ é uma matriz triangular superior por blocos. Além disso, o ij -ésimo bloco de Y é linear em X_{ij} e só depende dos blocos X mais próximos da diagonal principal do que X_{ij} .*

PROVA. Sendo $p(X)$ um polinômio escalar, $p(X)_{ij}$ é uma combinação linear dos blocos ij das potências de X . Assim sendo, o corolário decorre trivialmente do Teorema 2.1. \square

O Teorema 2.2 generaliza o resultado acima, quando $p(z)$ for uma função analítica, e mostra como os blocos acima da diagonal principal podem ser calculados.

Teorema 2.2. *Se X for uma matriz triangular superior por blocos e $p(z)$ for uma função analítica, então $Y = p(X)$ tem a mesma estrutura de blocos que X , e para $i < j$*

$$X_{ii}Y_{ij} - Y_{ij}X_{jj} = \sum_{k=i}^{j-1} Y_{ik}X_{kj} - \sum_{k=i+1}^j X_{ik}Y_{kj} \quad (2.2)$$

PROVA. Ver Parlett [34]. □

Note-se que o ij -ésimo bloco de Y só depende dos blocos de Y e X mais próximos da diagonal principal do que X_{ij} . Assim, os blocos diagonais de Y podem ser obtidos da relação $Y_{ii} = p(X_{ii})$ e os restantes podem ser calculados a partir da fórmula (2.2), uma supra-diagonal de cada vez, começando pela primeira supra-diagonal. Quando a fórmula for utilizada desta maneira, o lado direito é conhecido e, portanto, os elementos de Y_{ij} são determinados por um sistema de equações lineares do tipo

$$X_{ii}Y_{ij} - Y_{ij}X_{jj} = R,$$

assim sendo, existirá um único Y_{ij} se os autovalores de X_{ii} e X_{jj} forem diferentes (ver Teorema 1.3).

Lamentavelmente, este teorema não pode ser generalizado para a avaliação de polinômios matriciais, pois falta a hipótese fundamental da comutatividade entre X e $P(X)$.

A. ÁLGEBRA LINEAR

Bibliografia

- [1] T. Balendra, C. W. Tat, and S. L. Lee. Modal damping for torsionally coupled buildings on elastic foundation. *Earthquake Engineering and Structural Dynamics*, 10:735–756, 1982.
- [2] S. R. M. Barros. Comunicação Particular.
- [3] A. N. Beavers and E. D. Denman. Matrix sign function and computations in systems. *Appl. Math. Comput.*, 2:63–94, 1976.
- [4] A. Björck and S. Hammarling. A Schur method for the square root of a matrix. *Lin. Alg. Appl.*, 52:127–140, 1983.
- [5] H. Campos Velho. *Matriz não modal na integração e identificação num modelo Barotrópico*. PhD thesis, Universidade Federal de Porto Alegre, Porto Alegre, 1992.
- [6] J. Claeysen, E. Gallicchio, and M. Vilena. Inversion of linear matrix difference and differential equations by using dynamical solutions. *J. of Math. Analysis and Applications*, 1990.
- [7] S. Cohn. *Methods of sequential estimation for determining initial data in numerical weather prediction*. PhD thesis, New York University, 1982.
- [8] G. Cross and P. Lancaster. Square roots of complex matrices. *Linear and Mult. Alg.*, 1:289–293, 1974.

- [9] G. Davis. Numerical solution of quadratic matrix equation. *SIAM J. Sci. Stat. Comput.*, 2(2):164–175, 1981.
- [10] P. Davis. *Circulant Matrices*. John Wiley & sons, 1979.
- [11] J. Dennis, J. Traub, and R. Weber. The algebraic theory of matrix polynomials. *SIAM J. Numer. Anal.*, 13(6):831–845, 1976.
- [12] J. Dennis, J. Traub, and R. Weber. Algorithms for solvents of matrix polynomials. *SIAM J. Numer. Anal.*, 15(3):523–533, 1978.
- [13] M. Drazin, J. Dungey, and K. Gruenberg. Some theorems on commutative matrices. *J. London Math. Soc.*, 26:221–228, 1951.
- [14] Mac Duffee. *The theory of matrices*. Chelsea, 1956.
- [15] F. Gantmacher. *The theory of matrices*. Chelsea Publishing Company, 1960.
- [16] B. Garbow. Algorithm 535 the QZ algorithm to solve the generalized eigenvalue problem to complex matrices. *ACM TOMS*, 4:404–410, 1978.
- [17] I. Gohberg, P. Lancaster, and L. Rodman. *Matrix polynomials*. Academic Press, Inc., 1982.
- [18] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins series in the mathematical sciences. The Johns Hopkins University Press, Baltimore, Maryland, 1983.
- [19] R. Gregory and D. Karney. *A collection of matrices for testing computational algorithms*. Wiley-Interscience, 1969.
- [20] W. Grimus and G. Ecker. On the simultaneous diagonalizability of matrices. *J. Phys. A*, 19:3917–3919, 1986.
- [21] V. Hernandez and F. Incertis. A block bidiagonal form for block companion matrices. *Linear Algebra and Its Applications*, 75:241–256, 1986.
- [22] W. Hoskins and D. Walton. A faster method for computing the square root of a matrix. *IEEE Trans. Autom. Control*, 23(3):494–495, 1978.
- [23] L. Jódar. Boundary value problems for second order operator differential equation. *Linear Alg. Appl.*, 83:29–38, 1986.
- [24] L. Jódar. Algebraic and differential operators equations. *Lin. Algebra Appl.*, 102:35–53, 1988.
- [25] L. Jódar. Explicit solutions for second order operator differential equations with two boundary value conditions. *Lin. Algebra Appl.*, 103:73–86, 1988.
- [26] L. Jódar and E. Navarro. On complete set of solutions for polynomial matrix equations. *Appl. Math. Lett.*, 3(1):15–18, 1990.
- [27] L. Jódar and E. Navarro. Rectangular co-solutions of polynomial matrix equations and applications. *Appl. Math. Lett.*, 4(2):13–16, 1991.
- [28] L. Kantorovich and P. Akilov. *Functional analysis in normed spaces*. Pe. mon Press, 1964.
- [29] W. Kratz and E. Stickel. Numerical solution of matrix polynomial equations by Newton's method. *IMA J. Numer. Anal.*, 7:355–369, 1987.

- [30] P. Lancaster. *Lambda matrices and vibrating systems*. Pergamon Press, 1966.
- [31] P. Lancaster and P. Weber. Jordan chains for lambda matrices. *Linear Algebra and Appl.*, 1:563–569, 1968.
- [32] P. Lynch. *DYNAMO*. Irish Meteorological Office, Dublin, 1985.
- [33] C. Moler and G. Stewart. An algorithm for generalized matrix eigenvalue problems. *SIAM J. Numer. Anal.*, 12:835–853, 1975.
- [34] B. Parlett. A recurrence among the elements of triangular matrices. *Linear Alg. Appl.*, 14:117–121, 1976.
- [35] M. S. Paterson and L. Stockmeyer. On the number of nonscalar multiplications necessary to evaluate polynomials. *SIAM J. Comput.*, 2:60–66, 1973.
- [36] E. Pestel and F. Lechie. *Matrix methods in elastomechanics*. McGraw-Hill Book Company, Inc., New York, 1963.
- [37] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes*. Cambridge University Press, Cambridge, 1987.
- [38] A. Shabana. *Theory of vibrations*. Springer Verlag, 1991.
- [39] G. Stewart. On the sensitivity of the eigenvalue problem $Ax = \lambda Bx$. *SIAM J. Numer. Anal.*, 9:669–686, 1972.
- [40] J. F. Traub. A class of globally convergent iteration functions for the solution of polynomial equations. *Math. Comput.*, 20:113–138, 1966.
- [41] C. Van Loan. A note on the evaluation of matrix polynomials. *IEEE Trans. Autom. Control*, 24(2):320–321, 1979.
- [42] J. M. Varah. Rigorous machine bounds for the eigensystem of a general complex matrix. *Math. Comp.*, 22:793–801, 1968.
- [43] L. Verde and Star. Operator identities and the solution of linear matrix difference and differential equations. *Studies in Applied Mathematics*, 1992. to appear.
- [44] R. Ward. The combination shift QZ algorithm. *SIAM J. Numer. Anal.*, 10:241–256, 1973.
- [45] J. Wilkinson. *Rounding errors in algebraic processes*. Prentice Hall, 1963.

Índice

- algoritmo 48
 - de Horner 48
 - QZ 65
- autovalor generalizado 65
- autovetor generalizado 65
- cadeia de solventes 33
- comutador 108
- conjunto completo 13
 - de co-solventes 41
 - de solventes 13
- co-solvente 14, 40
- co-solventes 41
 - conjunto completo de 41
- equação polinomial matricial 9
- fator espectral à direita 11
- flops 49
- lambda matriz 10
- matriz 11
 - blocos de uma 148
 - companheira por blocos 11, 34, 40
 - de Vandermonde 12
 - de Vandermonde por blocos 12
 - definida por blocos 148
 - dominadora 72
 - em blocos 148
 - traço 146
 - transposta 16
 - transposta por blocos 149
- matrizes 117
 - equivalentes 117
 - quase-comutativas 114
 - unitariamente equivalentes 117
- método 60
 - de Heron 68
 - de Jordan 60
 - de Newton 61
 - de Newton modificado 69
- norma 145
 - de Frobenius 146
 - euclídeana 145
 - infinito 145
 - matricial induzida 146
 - um 145
- número de condição 68
- polinômio matricial 9
 - cociente 27
 - derivada de Fréchet 15
 - regular 15
 - divisão à esquerda 27
 - grau 9
 - mônico 11
 - ordem 9
 - resto 27
 - solvente 11
- polinômios fundamentais 43
- polinômios matriciais 9
 - divisão entre 26
 - fundamentais 43
 - interpolação de 43
- produto de Kronecker 146
- projedor 14
- quase-comutatividade 114
- raiz latente 11
- raiz quadrada 67
- solvente 11
 - à direita 11
 - à esquerda 11
 - dominante 72
 - simples 15
- solventes 13

cadeia de 33
conjunto completo de 13, 77
Teorema 10
 de Bézout generalizado 10, 28
 de Schur 146
traço 146
transformação bi-unitária 117
unidade de arredondamento 49
vetor latente à direita 12
