DECOMPOSIÇÃO EM FORMA DE SMITH UTILIZANDO FORMA CANÔNICA DE JORDAN

Raphael de Oliveira Leite

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

____
Prof. João Carlos dos Santos Basílio, D.Phil.

____
Prof. Ramon Romankevicius Costa, D.Sc.

____
Prof. Paulo César Marques Vieira, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

JULHO DE 2006

LEITE, RAPHAEL DE OLIVEIRA

Decomposição em forma de Smith utilizando forma canônica de Jordan [Rio de Janeiro] 2006

X, 105 p. 29,7 cm (COPPPE/UFRJ, M.Sc., Engenharia Elétrica, 2006)

Dissertação – Universidade Federal do Rio de Janeiro, COPPE

1. Forma de Smith
2. Forma Canônica de Jordan
3. Base polinomial mínima
4. Forma de Smith McMillan
5. Sistemas multivariáveis
6. Matrizes polinomiais

I. COPPE/UFRJ II. Título (série)

Agradecimentos

"O que a gentileza livremente oferece, agradecimentos não podem pagá-lo."
John Masefield (1878-1967)

Enfim está concluído. Confesso que por muitas vezes pensei que este trabalho não chegaria ao fim e por muitas vezes pensei em desistir. Os obstáculos foram muitos, a peleja longa, mas enfim está concluído.

Muitos foram os que me auxiliaram nesta jornada. Para começar gostaria de agradecer aquele que mais me deu forças e incentivos para que continuasse e não desistisse, aquele que por muitas vezes partilhou de minhas dúvidas e incertezas, de meus resultados bons e ruins, aquele que mais que meu orientador foi meu amigo e conselheiro, agradeço ao Professor Basílio.

Agradeço também a minha família, minha mãe, minha irmã, pessoas que estiveram sempre a meu lado durante todo esse caminho que começou há anos, contribuindo com palavras de incentivo, partilhando compreensão. Pessoas que estiveram comigo não só durante este trabalho, mas também durante toda minha formação.

Agradeço também aos colegas e professores da COPPE/UFRJ, em especial do Programa de Engenharia Elétrica, que compartilharam seus conhecimentos e experiências comigo complementando o meu desenvolvimento e facilitando a concretização deste trabalho.

Por fim gostaria de agradecer a todos que, de alguma forma, foram responsáveis direta, ou indiretamente, pelo sucesso desta jornada.

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

DECOMPOSIÇÃO EM FORMA DE SMITH UTILIZANDO FORMA CANÔNICA DE JORDAN

Raphael de Oliveira Leite

Julho/2006

Orientador: João Carlos dos Santos Basílio

Programa: Engenharia Elétrica

Matrizes polinomiais têm sido objeto de interesse tanto na teoria matemática, quanto na engenharia de controle (análise e projeto de sistemas multivariáveis) e também em processamento de sinais.

No estudo de matrizes polinomiais, a forma de Smith desempenha um papel significativo. Utilizando a forma de Smith é possível, por exemplo, determinar o posto normal de uma matriz polinomial; além disso, através de sua variante, a Forma de Smith-McMillan, podem-se determinar os pólos e zeros de um sistema multivariável.

O objetivo principal desta dissertação é propor um algoritmo para a obtenção da forma de Smith de matrizes polinomiais quadradas (mxm) através da determinação da forma de Jordan da matriz de evolução de estados de uma realização de ordem mínima associada à matriz de transferência $G(s) = IA^{-1}(s)$, onde $A(s)$ denota a matriz polinomial cuja forma de Smith deve ser computada. São também determinadas as matrizes multiplicadoras $U(s)$ e $V(s)$ tais que $U(s)A(s)V(s) = S(s)$, onde $S(s)$ representa a forma de Smith de $A(s)$, através da obtenção de uma base polinomial mínima para o espaço nulo de uma determinada matriz polinomial e, em seguida, calculando os polinômios coeficientes da combinação linear dos vetores polinomiais da base que levem às matrizes unimodulares $U(s)$ e $V(s)$.

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

SMITH FORM DECOMPOSITION USING JORDAN CANONICAL FORM

Raphael de Oliveira Leite

July/2006

Advisor: João Carlos Basílio

Department: Electrical Engineering

Polynomial matrices have received considerable attention in mathematics, in control engineering (mainly in the analysis and design of multivariable systems) and in signal processing.

In the study of polynomial matrices, the Smith form plays an important role. Using the Smith form, it is possible, for instance, to determine the normal rank of a polynomial matrix and the poles and zeros of a multivariable system, using the so-called Smith-McMillan form.

In this dissertation, it is proposed a new method for the computation of Smith form of polynomial matrices of dimension $m \times m$. The key step of the algorithm is the computation of the Jordan form of the state transition matrix of the minimal realization associated with $G(s)=IA^{-1}(s)$, where $A(s)$ is the polynomial matrix whose Smith form must be computed. The matrices $U(s)$ and $V(s)$ such that $U(s)A(s)V(s)=S(s)$, where $S(s)$ is the Smith form of $A(s)$, are obtained by computing a minimal polynomial bases for the null space of a certain polynomial matrix are in the sequence by finding polynomials of the linear combination of the elements of the basis which make the matrices $U(s)$ and $V(s)$ unimodular.

Sumário

Lista de Tabelas

Lista de Abreviaturas e Símbolos

$\mathbb{C}$	Conjunto dos números complexos
$\mathscr{C}(.)$	Matriz de controlabilidade
Diag{.}	Matriz diagonal
Dim[.]	Dimensão de uma matriz
DVS	Decomposição por Valores Singulares
DFM	Descrição por Frações de Matrizes
Gr[.]	Grau de um polinômio ou matriz polinomial
I_n	Matriz identidade de dimensão n
$\mathscr{L}\{.\}$	Transformada de Laplace
m_i	Multiplicidade
MDC	Máximo Divisor Comum
MIMO	*Multi-input multi-output* (multivariável)
MMC	Mínimo Múltiplo Comum
min(a,b)	Menor entre *a* e *b*
$\mathscr{O}(.)$	Matriz de Observabilidade
$\mathscr{P}$	Conjunto de pólos de um sistema
$\mathbb{R}$	Conjunto dos números Reais
$\mathbb{R}^{p\times m}(s)$	Conjunto das matrizes racionais *p x m*
$\mathbb{R}^{p\times m}[s]$	Conjunto das matrizes polinomiais *p x m*
$\underset{\sim}{s}$	Smith equivalente, ou de mesma forma de Smith
SISO	*Single-input single-output* (monovariável)
W_c	Gramiano de Controlabilidade

W_o	Gramiano de Observabilidade
$\mathscr{Z}$	Conjunto de zeros de um sistema
$\mathscr{Z}_\infty$	Conjunto de zeros no infinito
$\mathscr{Z}_f$	Conjunto de zeros finitos
λ	Autovalor
$\rho(.)$	Posto de uma matriz
$\otimes$	Produto de Kronecker
>>	Sinal de solicitação do Matlab
$\leftarrow$	Símbolo de atribuição

1. Introdução

"No meio da dificuldade está
a oportunidade"
Einstein

O estudo e projeto de processos ou sistemas físicos pode ser realizado usando-se métodos empíricos. Para tanto, aplicam-se diversos sinais de entrada ao sistema físico e medem-se suas respostas. A partir destas, estima-se se seu desempenho está satisfatório e, em caso contrário, são feitos alguns ajustes, prosseguindo neste processo de tentativa-e-erro até que se encontre um resultado que atenda aos índices de desempenho pré-estabelecidos. Porém, estes métodos podem se mostrar impraticáveis se os sistemas físicos em estudo forem complexos demais, ou muito caros, ou ainda perigosos. Nestes casos, entram os processos analíticos. Os processos analíticos consistem basicamente da seguinte seqüência de procedimentos: modelagem, desenvolvimento de equações matemáticas, análise e projeto.

Tomando como base o número de entradas e saídas, os sistemas podem ser divididos em dois tipos básicos: os de uma entrada e uma saída (escalares, ou monovariáveis) e os sistemas de múltiplas-entradas e múltiplas-saídas (multivariáveis). A compreensão destes sistemas é muito importante em áreas como controle de processos, engenharia aeroespacial, meteorologia e economia, só para citar alguns exemplos.

A modelagem de sistemas multivariáveis objetiva a obtenção de matrizes de transferências, que são matrizes racionais. Tendo como motivação o caso escalar, essas matrizes racionais podem ser escritas como o produto de uma matriz polinomial pela inversa de outra matriz polinomial. Assim, no estudo de sistemas multivariáveis, as matrizes polinomiais têm destacada relevância. Matrizes polinomiais têm sido objeto de interesse tanto na teoria matemática [1], [2], quanto na engenharia de controle (análise e projeto de sistemas multivariáveis) [3], [4], [5], [6], [7], [8] e também em processamento de sinais [9], [10], [11]. Esta importância se deve ao fato que muitos conceitos relativos a sistemas monovariáveis só podem ser generalizados para o caso multivariável utilizando-se matrizes polinomiais.

A forma de Smith [12] é um conceito relevante dentro do estudo de matrizes polinomiais e de reconhecida importância como elemento chave na análise e projeto de sistemas de controle multivariáveis. Utilizando a forma de Smith é possível, por exemplo, determinar o posto normal de uma matriz polinomial; além disso, através de sua variante, a Forma de Smith-McMillan [13], podem-se determinar os pólos e zeros de um sistema multivariável.

A determinação da forma de Smith de uma matriz $A(s) \in \mathbb{R}^{p\times m}(s)$ consiste basicamente de sua diagonalização. O método clássico de obtenção da forma de Smith envolve a manipulação de matrizes unimodulares através de operações elementares de linhas e colunas. Por essa razão não é apropriado para cálculos numéricos, porque resulta em um número extraordinariamente grande de manipulações polinomiais, além de reconhecida sensibilidade numérica dos métodos de redução baseados em operações elementares de linhas e colunas.

Gantmacher [1] propôs um método direto para a redução de uma matriz polinomial à sua forma de Smith, o qual serviu de motivação para outros métodos [14], [15], que podem ser vistos como variações do método proposto por Gantmacher. Entretanto, todos estes métodos sofrem de instabilidade numérica e crescimento dos coeficientes porque o pivoteamento não é baseado nos coeficientes de s, mas sim em suas potências. Ainda nos anos 70 do século passado, Ramachandran [16] propôs um método para obtenção da redução exata de uma matriz polinomial à sua Forma de Smith, porém este método tem aplicação limitada a matrizes polinomiais monovariáveis.

Provavelmente um dos primeiros métodos polinomiais diretos apresentados, seja aquele proposto por Kaltofen et al. [17]; contudo ele não calcula os multiplicadores (ou transformações equivalentes) $U(s)$ e $V(s)$ tais que se $S(s)$ representa a forma de Smith de $A(s)$, $U(s)A(s)V(s) = S(s)$. Em outro algoritmo, Kaltofen et al. [18] propõem um algoritmo probabilístico para obter a forma de Smith, onde é mostrado que além da forma de Smith, também as matrizes multiplicadoras podem ser obtidas de forma estocástica. Esse algoritmo tem como base a multiplicação da matriz de entrada por uma outra matriz constante escolhida aleatoriamente, chamada “condicionante”, que tem por objetivo pré-condicionar a matriz de entrada para facilitar a obtenção da forma de Smith através de sua forma de Hermite [4]. A chave para seu sucesso é a boa escolha desta matriz constante, o que não é fácil, uma vez que não há uma maneira determinística de se encontrar essa matriz “condicionante”. Outros algoritmos baseados em métodos probabilísticos também foram apresentados recentemente [19], [20], [21], [22]. Além deles, pode ser encontrado na literatura também aplicações específicas recentes para a forma de Smith, principalmente na área de processamento de sinais [23].

Um grande número de artigos tem sido devotado ao problema de se estabelecer conexões entre matrizes polinomiais e realizações em espaço de estado, com particular atenção à matriz de evolução de estados estar na forma canônica de Jordan [24], [25], [26], [27]. Seguindo essa linha, Van Dooren, et al. [28] propõem um algoritmo para obtenção da Forma de Smith-Macmillan de matrizes racionais a partir de sua expansão em série de Laurent.

O objetivo principal desta dissertação é propor um algoritmo para a obtenção da forma de Smith de matrizes polinomiais quadradas (mxm) através da determinação da forma de Jordan da matriz de evolução de estados de uma realização de ordem mínima associada à matriz de transferência $G(s) = IA^{-1}(s)$. São também determinadas as matrizes multiplicadoras $U(s)$ e $V(s)$ através da obtenção de uma base polinomial mínima para o espaço nulo de uma determinada matriz polinomial [29] e, em seguida, calculando os polinômios coeficientes da combinação linear dos vetores polinomiais da base que levem às matrizes unimodulares $U(s)$ e $V(s)$.

Como ferramenta de apoio aos cálculos foi utilizado o *software* Matlab da Mathworks [30], tendo sido desenvolvidas funções que implementaram os algoritmos descritos neste trabalho e possibilitaram os cálculos dos exemplos apresentados. As funções aqui desenvolvidas visam complementar aquelas introduzidas por Barcelos [31] e compor o conjunto de funções (*toolbox*) denominado *Polymat*. Outros conjuntos de funções análogos já existem a disposição, por exemplo, a *Polynomial Toobox*, comercialmente chamada de *Polyx* (www.polyx.cz). A vantagem do método apresentado neste trabalho em relação à *Polyx*, por exemplo, vem de sua concepção. Todos os cálculos das funções que compõem *Polymat* são baseados em matrizes de coeficientes, ou seja, todas as manipulações envolvem apenas matrizes numéricas. Já as funções de *Polyx* são baseadas em matrizes simbólicas, o que faz necessária maior capacidade de processamento computacional dos resultados. Além disso, a *Polymat* visa ser disponibilizada para a área acadêmica de forma livre para que a mesma seja utilizada e melhorada.

Este trabalho está estruturado da seguinte forma. No capítulo 2 serão apresentados os fundamentos da teoria de sistemas multivariáveis. Ainda no capítulo 2, apresenta-se uma discussão dos assuntos relacionados à obtenção da forma de Smith de uma matriz polinomial, culminando com um algoritmo para o cálculo da mesma. Exemplos ilustrativos são também apresentados. No capítulo 3, um novo algoritmo para a obtenção da forma de Smith de matrizes polinomiais é proposto. Com vistas a mostrar o desempenho computacional deste algoritmo, e também para clarificar a sua utilização, são apresentados alguns exemplos. Por fim, no capítulo 4, é apresentada uma breve análise sobre os resultados obtidos e sugestões de trabalhos futuros.

2. Fundamentos Teóricos

"A aquisição da sabedoria é melhor que a das pérolas."
Jô, 28, 18

Este capítulo destina-se a apresentar parte da teoria envolvida no cálculo da forma de Smith de matrizes polinomiais. Na seção 2.1, apresenta-se uma breve comparação entre sistemas escalares (monovariáveis) e sistemas multivariáveis (múltiplas entradas e múltiplas saídas). Na seção 2.2, abordam-se os conceitos de controlabilidade e observabilidade e, em seguida, na seção 2.3, são revisadas as realizações de ordem mínima para sistemas multivariáveis. Na seção 2.4 discute-se a descrição por frações de matrizes (DFM). A seção 2.5 considera o problema de obter uma realização de ordem mínima a partir de uma DFM irredutível. Esse problema terá um importante papel no trabalho aqui desenvolvido. A seção 2.6 trata de pólos e zeros de sistemas multivariáveis e, para tanto, torna-se necessário introduzir a Forma de Smith de matrizes polinomiais. Na seção 2.7 é descrito um algoritmo para o cálculo da forma de Smith, deixando alguns exemplos para serem desenvolvidos na seção 2.8. Finalmente, na seção 2.9, é feita uma análise sobre alguns métodos presentes na literatura e são tecidos alguns comentários. Este capítulo foi escrito tomando-se por base [4], [6], [31], [32], [33].

2.1. Comparação entre Sistemas Escalares e Multivariáveis

Esta seção apresenta brevemente os conceitos de sistemas escalares buscando estendê-los aos sistemas multivariáveis.

2.1.1 Sistemas Escalares

Um sistema escalar, ou sistema monovariável (SISO), é aquele que apresenta apenas uma entrada e uma saída. Seja, portanto, um sistema linear e invariante no tempo (SLIT) com entrada $u(t)$ e saída $y(t)$. Então a função de transferência desse sistema será dada por:

$$G(s) = \frac{Y(s)}{U(s)} \tag{2.1}$$

sendo

$$G(s) = \frac{b(s)}{a(s)} = \frac{b_1 s^{n-1} + b_2 s^{n-2} + \ldots + b_{n-1}s + b_n}{s^n + a_1 s^{n-1} + a_2 s^{n-2} + \ldots + a_{n-1}s + a_n} \tag{2.2}$$

com $b(s)$ e $a(s)$ coprimos[1]. É sabido [4], [6] que, para o sistema descrito pela equação (2.2), é possível obter uma realização em espaço de estado de ordem n e que esta realização tem ordem mínima, isto é, possuir o menor número possível de estados. Uma dessas realizações é dada por:

$$\begin{cases} \dot{\underline{x}}(t) = A\underline{x}(t) + b\underline{u}(t) \\ y(t) = c\underline{x}(t) \end{cases}, \tag{2.3}$$

sendo

$$A = \begin{bmatrix} -a_1 & 1 & 0 & \cdots & 0 \\ -a_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_{n-1} & 0 & 0 & \cdots & 1 \\ -a_n & 0 & 0 & \cdots & 0 \end{bmatrix}, \; b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{bmatrix} \text{ e } c = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix}. \tag{2.4}$$

Considere agora o cálculo dos pólos/zeros de $G(s)$. Suponha que $\mathcal{P}$, $\mathcal{Z}$, $\mathcal{Z}_f$, $\mathcal{Z}_\infty$ denotem, respectivamente, os conjuntos dos pólos, zeros, zeros finitos e zeros no infinito de $G(s)$. Então:

[1] Dois polinômios são ditos coprimos se todos os seus máximos divisores comuns têm grau igual a zero, isto é, são escalares reais.

(i) $\mathscr{P} = \left\{ p \in \mathbb{C} : \lim_{s \to p} G(s) = \infty \right\} = \left\{ p \in \mathbb{C} : a(p) = 0 \right\}$

(ii) $\mathscr{Z} = \left\{ z \in \mathbb{C} : \lim_{s \to z} G(s) = 0 \right\} = \mathscr{Z}_f \cup \mathscr{Z}_\infty$, onde $\mathscr{Z}_f = \{ z \in \mathbb{C} : b(z) = 0 \}$ e $\mathscr{Z}_\infty$ tem cardinalidade igual a $grau[b(s)] - grau[a(s)]$.

2.1.2 Sistemas Multivariáveis

Considere agora um sistema multivariável (MIMO), com m entradas e p saídas. Assim, o modelo da função de transferência será dado por $y(s) = G(s)u(s)$, onde $u(s) \in \mathbb{R}^{m \times 1}(s)$ e $y(s) \in \mathbb{R}^{p \times 1}(s)$ denotam, respectivamente, as transformadas de Laplace dos vetores de entrada e saída e $G(s) \in \mathbb{R}^{p \times m}(s)$ denota a matriz de transferência, sendo $\mathbb{R}^{p \times m}(s)$ o conjunto das matrizes racionais p x m. A esta matriz de transferência podem-se associar diversas realizações de diferentes ordens, conforme será ilustrado no exemplo a seguir.

Exemplo 2.1: Considere a matriz de transferência $G(s)$ dada por:

$$G(s) = \begin{bmatrix} \dfrac{1}{(s-1)^2} & \dfrac{1}{(s-1)(s+3)} \\ \dfrac{-6}{(s-1)(s+3)^2} & \dfrac{s-2}{(s+3)^2} \end{bmatrix}. \tag{2.5}$$

(1) Como primeiro exemplo, considere a seguinte realização para $G(s)$.

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \Rightarrow \begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}$$

$$\Rightarrow y_i(s) = \sum_{j=1}^{2} g_{ij}(s) u_j(s) \Rightarrow \begin{cases} y_1(s) = y_{11}(s) + y_{12}(s) \\ y_2(s) = y_{21}(s) + y_{22}(s) \end{cases}.$$

A forma recursiva para a expressão apresentada acima é:

$$y_{ij}(s) = g_{ij}(s) u_j(s) \Rightarrow \begin{cases} \dot{\underline{x}}_{ij}(t) = A_{ij} \underline{x}_{ij}(t) + b_{ij} u(t) \\ y_{ij}(t) = c_{ij} x_{ij}(t) \end{cases},$$

onde

$$A_{11} = \begin{bmatrix} 2 & 1 \\ -1 & 0 \end{bmatrix}, \quad b_{11} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad c_{11} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$A_{12} = \begin{bmatrix} -2 & 1 \\ 3 & 0 \end{bmatrix}, \quad b_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad c_{12} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$A_{21} = \begin{bmatrix} -5 & 1 & 0 \\ -3 & 0 & 1 \\ 9 & 0 & 0 \end{bmatrix}, \quad b_{21} = \begin{bmatrix} 0 \\ 0 \\ -6 \end{bmatrix}, \quad c_{21} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \, .$$

$$A_{22} = \begin{bmatrix} -6 & 1 \\ -9 & 0 \end{bmatrix}, \quad b_{22} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \quad c_{22} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Definindo-se $\underline{x} = \begin{bmatrix} x_{11}^T(t) & x_{12}^T(t) & x_{21}^T(t) & x_{22}^T(t) \end{bmatrix}$, tem-se que uma realização para $G(s)$ será dada por:

$$\begin{cases} \dot{\underline{x}}(t) = \begin{bmatrix} 2 & 1 & & & & & & & \\ -1 & 0 & & & & & & & \\ & & -2 & 1 & & & & & \\ & & 3 & 0 & & & & & \\ & & & & -5 & 1 & 0 & & \\ & & & & -3 & 0 & 1 & & \\ & & & & 9 & 0 & 0 & & \\ & & & & & & & -6 & 1 \\ & & & & & & & -9 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ -6 & 0 \\ 0 & 1 \\ 0 & -2 \end{bmatrix} . \\ y(t) = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} x(t) \end{cases} \qquad (2.6)$$

Note que a ordem dessa realização é n_1=9.

(2) Escreva agora $G(s)= N(s)/d(s)$ onde

$$d(s) = (s-1)^2(s+3)^2 = s^4 + 4s^3 - 2s^2 - 12s + 9$$

e

$$N(s) = \begin{bmatrix} (s+3)^2 & (s-1)(s+3) \\ -6(s-1) & (s-2)(s-1)^2 \end{bmatrix} = \begin{bmatrix} s^2 + 6s + 9 & s^2 + 2s - 3 \\ -6s + 6 & s^3 - 4s^2 + 5s - 2 \end{bmatrix}.$$

Escrevendo $N(s) = N_1 s^3 + N_2 s^2 + N_3 \, s + N_4$, onde

$$N_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad N_2 = \begin{bmatrix} 1 & 1 \\ 0 & -4 \end{bmatrix}, \quad N_3 = \begin{bmatrix} 6 & 2 \\ -6 & 5 \end{bmatrix}, \quad N_4 = \begin{bmatrix} 9 & -3 \\ 6 & -2 \end{bmatrix},$$

e definindo $d(s) = s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4$, tem-se que uma outra realização para $G(s)$ será:

$$\begin{cases} \dot{\underline{x}}(t) = \begin{bmatrix} -a_1 I_2 & I_2 & 0 & 0 \\ -a_2 I_2 & 0 & I_2 & 0 \\ -a_3 I_2 & 0 & 0 & I_2 \\ -a_4 I_2 & 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} I_2 & 0 & 0 & 0 \end{bmatrix} x(t) \end{cases}, \tag{2.7}$$

cuja ordem é $n_2 = 8$ ($n_2 < n_1$).

Surge, então, uma pergunta natural. Qual a menor ordem possível para as realizações associadas a uma dada matriz de transferência? Para responder essa questão, são necessários os conceitos de controlabilidade e observabilidade, a serem apresentados a seguir.

2.2. Controlabilidade e Observabilidade

Esta seção introduz os conceitos de controlabilidade e observabilidade. A controlabilidade de um sistema está associada à capacidade de se controlar os estados, isto é, de se levar os estados de um sistema de uma posição inicial arbitrária para uma posição final também arbitrária em um intervalo de tempo finito. A observabilidade de um sistema está associada à capacidade de se determinar o estado inicial de um sistema a partir do conhecimento da saída. A seguir, são apresentados em maiores detalhes cada um desses conceitos.

2.2.1 Controlabilidade

Considere uma descrição em espaço de estado de um SLIT com n estados e m entradas

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t), \tag{2.8}$$

onde $A \in \mathbb{R}^{n \times n}$ e $B \in \mathbb{R}^{n \times m}$. Pode-se então apresentar o conceito de controlabilidade.

Definição 2.1 O par (A,B) da equação de estados (2.8) é controlável se e somente se para todo estado inicial $x(t_o) = x_o$ e todo estado final $x(t_1) = x_1$, para ($t_1 > t_o$), existir uma entrada $u(t)$, $t \in [t_0, t_1]$, tal que $x(t_1) = x_1$. □

A análise matemática da controlabilidade de um sistema é feita utilizando-se o seguinte teorema:

Teorema 2.1 É equivalente dizer que:

1. O par (A,B) é controlável.

2. A matriz n x n

$$W_c(t) = \int_0^t e^{A\tau} BB^T e^{A'\tau} d\tau = \int_0^t e^{A(t-\tau)} BB^T e^{A'(t-\tau)} d\tau \tag{2.9}$$

é não-singular para qualquer $t>0$.

3. A matriz de controlabilidade n x m

$$\mathscr{C} = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} \tag{2.10}$$

tem posto cheio, ou seja, tem posto igual a n.

4. A matriz $[A\text{–}\lambda I \;\; B]$ de dimensão n x (n+m) tem posto completo para todo autovalor, λ, de A.

Prova: Ver [6], páginas 145 a 147. □

Uma vez apresentada uma maneira de se verificar se um par (A,B) é controlável, o passo seguinte é apresentar uma maneira sistemática de, a partir de uma realização não controlável, obter uma outra realização equivalente à realização dada (no sentido de que ambas levem à mesma função de transferência) e que esta seja controlável. Isto será feito a seguir.

Lema 2.1 Transformações de similaridade não alteram o posto da matriz de controlabilidade, isto é, se

$$\begin{cases} \dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \\ \underline{y}(t) = C\underline{x}(t) \end{cases} \quad \text{e} \quad \begin{cases} \dot{\underline{x}}(t) = \hat{A}\hat{\underline{x}}(t) + \hat{B}\underline{u}(t) \\ \underline{y}(t) = \hat{C}\hat{\underline{x}}(t) \end{cases}$$

onde $\underline{x}(t) = T\hat{\underline{x}}(t)$ e, portanto, $\hat{A} = T^{-1}AT$, $\hat{B} = T^{-1}B$ e $\hat{C} = CT$, então

$$\rho[\mathscr{C}(A,B)] = \rho[\mathscr{C}(\hat{A},\hat{B})].$$

Prova: Ver [4] ou [6]. □

Teorema 2.2 Seja a realização em espaço de estado

$$\begin{cases} \dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \\ \underline{y}(t) = C\underline{x}(t) + D\underline{u}(t) \end{cases} \tag{2.11}$$

e suponha que $\rho[\mathscr{C}(A,B)]=r<n$. Defina:

(i) $Q=[q_1\ q_2\ \ldots\ q_r\ q_{r+1}\ \ldots\ q_n]$, onde q_i, $i = 1,\ldots,r$ são as r primeiras colunas linearmente independentes da matriz $\mathscr{C}(A,B)$ e q_i, $i = r+1, \ldots, n$ são quaisquer vetores escolhidos de modo que Q seja uma matriz não singular.

(ii) $\underline{x}(t) = Q\hat{\underline{x}}(t)$

Então, pode-se re-escrever o sistema apresentado em (2.11) como

$$\begin{cases} \dot{\hat{\underline{x}}}(t) = \hat{A}\hat{\underline{x}}(t) + \hat{B}\underline{u}(t) \\ \underline{y}(t) = \hat{C}\underline{x}(t) + D\underline{u}(t) \end{cases} \tag{2.12}$$

onde

$$\hat{A} = \begin{bmatrix} A_c & A_{12} \\ 0 & A_{\bar{c}} \end{bmatrix},\ \hat{B} = \begin{bmatrix} B_c \\ 0 \end{bmatrix} \text{ e } \hat{C} = \begin{bmatrix} C_c & C_{\bar{c}} \end{bmatrix}, \tag{2.13}$$

com as matrizes $A_c \in \mathbb{R}^{r\times r}$, $B_c \in \mathbb{R}^{r\times m}$ e $C_c \in \mathbb{R}^{p\times r}$. Pode-se dizer que:

(a) $\rho[\mathscr{C}(A_c,B_c)]=r$;

(b) $T(s) = C(sI-A)^{-1}B+D = C_c(sI-A_c)^{-1}B_c+D$.

Prova: Ver [6] □

Assim, de acordo com o Teorema 2.2, a realização em espaço de estado

$$\begin{cases} \underline{\dot{x}}_c(t) = A_c \underline{x}_c(t) + B_c \underline{u}(t) \\ \underline{y}(t) = C_c \underline{x}_c(t) + D\underline{u}(t) \end{cases} \tag{2.14}$$

é equivalente, tendo a realização (2.14) menor ordem.

2.2.2 Observabilidade

O conceito de observabilidade é dual do conceito de controlabilidade. De forma sucinta, controlabilidade estuda a possibilidade de se guiar o estado a partir da entrada, enquanto observabilidade estuda a possibilidade de se estimar o estado inicial a partir da saída de forma única.

Considere uma realização em espaço de estado

$$\begin{cases} \underline{\dot{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \\ \underline{y}(t) = C\underline{x}(t) + D\underline{u}(t) \end{cases}, \tag{2.15}$$

onde A, B, C e D são, respectivamente, matrizes constantes de ordem nxn, nxm, pxn e pxm,.

Definição 2.2 A realização (2.15) é observável se e somente se para todo estado inicial $x(0) = x_0$, existir um tempo finito $t_1>0$ tal que, conhecendo-se a entrada e a saída $u(t)$ e $y(t)$, respectivamente, definidas no intervalo $[0, t_1]$, o estado inicial x_0 é determinado unicamente. De outra forma, a realização é não-observável. □

Note que, como

$$\underline{x}(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau\,, \tag{2.16}$$

então

$$\begin{aligned} y(t) &= Ce^{At}x_0 + C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau + Du(t) \\ y(t) - C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau - Du(t) &= Ce^{At}x_0 \end{aligned}.$$

Definindo

$$\tilde{y}(t) = y(t) - C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau - Du(t) \tag{2.17}$$

e notando que $u(t)$ e $y(t)$ são conhecidos no intervalo $[0,t_1]$, então se pode escrever:

$$\tilde{y}(t) = Ce^{At}x_0 \,. \tag{2.18}$$

Observe que $\tilde{y}(t)$ pode ser vista como a resposta livre do sistema à condição inicial x_0. Assim, é possível enunciar uma definição equivalente à Definição 2.2.

Definição 2.3 A realização (A, B, C, D) da equação (2.15) é observável se e somente se o estado inicial puder ser unicamente determinado a partir do conhecimento da resposta livre em um intervalo de tempo finito. □

A verificação da controlabilidade de um sistema é feita utilizando-se os seguintes teoremas.

Teorema 2.3 **(Teorema da Dualidade)** O par (A,B) é controlável se e somente se o par (B^T,A^T) é observável.

Prova: Veja [6], página 156. □

Utilizando-se o Teorema 2.3, podem-se obter resultados análogos aos do Teorema 2.1.

Teorema 2.4 É equivalente dizer que:

1. O par (A,C) de dimensão n é observável.

2. A matriz $n \times n$

$$W_o(t) = \int_0^t e^{A^T t}C^T Ce^{At}dt \tag{2.19}$$

é não singular para qualquer $t > 0$.

3. A matriz de observabilidade $nq \times n$

$$\mathscr{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{2.20}$$

tem posto n (posto completo).

4. A matriz $(n+q)$ x n

$$\begin{bmatrix} A - \lambda I \\ C \end{bmatrix}$$

tem posto completo para todo e qualquer autovalor λ de A.

Prova: Veja [6], página 156. □

2.3. *Realização de ordem mínima*

Nesta seção será abordado o problema de se obter uma realização de ordem mínima. Será apresentado inicialmente o conceito de realizações equivalentes e a seguir será feita a caracterização das realizações de ordem mínima.

2.3.1 Realizações Equivalentes

Realizações equivalentes podem ser definidas como segue.

Definição 2.4 Duas realizações (A_1,B_1,C_1,D) e (A_2,B_2,C_2,D) de ordens diferentes são equivalentes quando elas levam à mesma matriz de transferência, isto é,

$$C_1(sI - A_1)^{-1}B_1 + D = C_2(sI - A_2)^{-1}B_2 + D.$$ □

Seja $G(s) = C(sI - A)^{-1}B + D$ a função de transferência de um sistema e considere a matriz $e^{At} = \mathscr{L}^{-1}\{(sI - A)^{-1}\}$. Sabe-se que

$$e^{At} = I + At + A^2\frac{t^2}{2!} + \ldots + A^n\frac{t^n}{n!} + \ldots. \tag{2.21}$$

Aplicando a transformada de Laplace em (2.21), chega-se a:

$$\mathscr{L}\left\{e^{At}\right\} = (sI - A)^{-1} = \mathscr{L}\left\{I + At + A^2\frac{t^2}{2!} + \ldots + A^n\frac{t^n}{n!} + \ldots\right\}. \tag{2.22}$$

Pode-se, portanto, escrever:

$$(sI - A)^{-1} = s^{-1} + As^{-2} + A^2 s^{-3} + \ldots + A^n s^{-(n+1)} + \ldots = \sum_{k=0}^{\infty} A^k s^{-(k+1)} = \sum_{k=1}^{\infty} A^{k-1} s^{-k} \; .$$

Portanto

$$G(s) = C\left(\sum_{k=1}^{\infty} A^{k-1} s^{-k}\right)B + D = \sum_{k=1}^{\infty} CA^{k-1} Bs^{-k} + D \; . \tag{2.23}$$

Definido-se $h_k = CA^{k-1}B$, k = 1, 2, ..., n, resulta:

$$G(s) = \sum_{k=1}^{\infty} h_k s^{-k} + D \; . \tag{2.24}$$

A partir de (2.23) e (2.24), pode-se introduzir os chamados Parâmetros de Markov.

Definição 2.5 Aos parâmetros $h_k = CA^{k-1}B$, k = 1, 2, ..., n, dá-se o nome de Parâmetros de Markov de *G(s)*. □

Note que, na Definição 2.5 é dito que h_k, para $k=1,2,...,n$, são os parâmetros de *G(s)* e não da realização (*A*,*B*,*C*,*D*), conforme enunciado a seguir.

Lema 2.2 Duas realizações (*A*,*B*,*C*,*D*) e $(\widetilde{A}, \widetilde{B}, \widetilde{C}, \widetilde{D})$, não necessariamente de mesma ordem são equivalentes se e somente se $D = \widetilde{D}$ e $CA^{k-1}B = \widetilde{C}\widetilde{A}^{k-1}\widetilde{B}$, para $k = 1, 2, 3, \ldots$ □

2.3.2 Realizações de Ordem Mínima

Definição 2.6 Uma realização (*A*,*B*,*C*,*D*) de *G*(*s*) é chamada de realização de ordem mínima de *G*(*s*) se *A* tem a menor dimensão possível, ou equivalentemente, o menor número de estados possível. □

Uma maneira de se verificar na prática se uma realização é de ordem mínima é dada pelo teorema a seguir.

Teorema 2.5 Uma realização (*A*,*B*,*C*,*D*) de uma matriz de transferência *G*(*s*) é dita de ordem mínima se e somente se (*A*,*B*) é controlável e (*C*,*A*) é observável.

Prova: Ver [33]. □

Para finalizar o estudo de realizações de ordem mínima, é importante ressaltar que existem infinitas realizações de ordem mínima. Porém, conforme será mostrado a seguir, dadas duas realizações de ordem mínima, existe uma e somente uma transformação de similaridade que as relaciona.

Teorema 2.6 Sejam (A,B,C,D) e $(\widetilde{A},\widetilde{B},\widetilde{C},\widetilde{D})$ realizações de ordem mínima de um sistema com função de transferência *G(s)*. Então, existe uma, e somente uma, transformação de similaridade *T*, tal que $\widetilde{A}=T^{-1}AT$, $\widetilde{B}=T^{-1}B$ e $\widetilde{C}=CT$. Além disso, define-se *T* e T^{-1}, respectivamente, como $T=(\mathscr{C}\widetilde{\mathscr{C}}^T)(\widetilde{\mathscr{C}}\widetilde{\mathscr{C}}^T)^{-1}$ e $T^{-1}=(\widetilde{\mathscr{O}}^T\widetilde{\mathscr{O}})^{-1}(\widetilde{\mathscr{O}}^T\mathscr{O})$, onde $\widetilde{\mathscr{O}}=\mathscr{O}(\widetilde{C},\widetilde{A})$, $\widetilde{\mathscr{C}}=\mathscr{C}(\widetilde{A},\widetilde{B})$, $\mathscr{O}=\mathscr{O}(C,A)$ e $\mathscr{C}=\mathscr{C}(A,B)$.

Prova: Ver [4], página 364. □

2.4. Descrição por Frações de Matrizes

O objetivo principal desta seção é o estudo das Descrições por Frações de Matrizes (DFM). Além disso, conceitos importantes como, por exemplo, matrizes unimodulares e máximo divisor comum de matrizes polinomiais, Identidade de Bezout, redução por coluna e realização na forma controlador serão abordados.

No decorrer do capítulo e dos subseqüentes considere que $\mathbb{R}^{p\times q}[s]$ denota o conjunto das matrizes polinomiais *pxq* e $\mathbb{R}^{p\times q}(s)$ denota o conjunto das matrizes racionais *pxq*.

2.4.1 Matrizes Unimodulares

Quando é necessário realizar-se operações elementares com linhas e colunas de matrizes polinomiais com o objetivo de, por exemplo, calcular o posto normal de uma matriz (Definição 2.7), ou uma redução por coluna, ou de quaisquer operações que envolvem pré ou pós-multiplicação por matrizes que não alteram o posto normal da mesma faz-se uso de matrizes unimodulares.

Definição 2.7 O posto normal de $G(s)$ é definido como o maior posto de $G(s)$ para todos os valores de $s \in \mathbb{C}$. □

Definição 2.8 Uma matriz polinomial quadrada $U(s)$ é denominada unimodular se e somente se ela é não singular para todo $s \in \mathbb{C}$, ou equivalentemente se e somente se o seu determinante é não-nulo e independente de s. □

Fato 2.1 Uma matriz polinomial $U(s)$ é unimodular se e somente se sua inversa $U^{-1}(s)$ é também unimodular.

Prova: A prova é imediata e será omitida. □

Ainda sobre matrizes unimodulares, é necessário citar uma importante propriedade: a multiplicação de uma matriz qualquer por uma matriz unimodular não altera o grau do determinante dessa matriz, isto é, se $V(s)$ for uma matriz polinomial qualquer e $U(s)$ uma matriz unimodular, então $X(s) = U(s)V(s)$ é tal que $gr(|X(s)|) = gr(|V(s)|)$.

2.4.2 Máximo Divisor Comum de Matrizes Polinomiais

Outro conceito importante para auxiliar o assunto central desta seção é o de Máximo Divisor Comum (MDC).

Definição 2.9 Sejam $N(s) \in \mathbb{R}^{p \times m}[s]$ e $D(s) \in \mathbb{R}^{q \times m}[s]$, isto é, matrizes polinomiais com o mesmo número de colunas. Então, a matriz $R(s) \in \mathbb{R}^{m \times m}[s]$ será um MDC de $N(s)$ e $D(s)$ à direita se ela satisfizer as seguintes condições:

(i) $R(s)$ é um divisor comum à direita de $N(s)$ e $D(s)$, isto é, existem matrizes polinomiais $N_R(s)$ e $D_R(s)$ tais que $N(s) = N_R(s)R(s)$ e $D(s) = D_R(s)R(s)$.

(ii) Se $R_1(s)$ é qualquer outro divisor comum à direita de $N(s)$ e $D(s)$, então $R_1(s)$ é um divisor à direita de $R(s)$, ou seja, existe uma matriz polinomial $W(s)$ tal que $R(s)=W(s)R_1(s)$. □

Da mesma forma, pode-se definir MDC à esquerda com as mudanças óbvias.

Definição 2.10 Sejam $N(s) \in \mathbb{R}^{p\times m}[s]$ e $D(s) \in \mathbb{R}^{p\times q}[s]$, isto é, matrizes polinomiais com o mesmo número de linhas (p). Então, a matriz $L(s) \in \mathbb{R}^{p\times p}[s]$ será um MDC à esquerda de $N(s)$ e $D(s)$ se ela satisfizer as seguintes condições:

(i) $L(s)$ é um divisor comum à esquerda de $N(s)$ e $D(s)$, ou seja, existem matrizes polinomiais $N_L(s)$ e $D_L(s)$ tais que $N(s) = L(s)N_L(s)$ e $D(s) = L(s)D_L(s)$.

(ii) Se $L_1(s)$ é qualquer outro divisor comum à esquerda de $N(s)$ e $D(s)$, então $L_1(s)$ é um divisor à esquerda de $L(s)$, ou seja, existe uma matriz polinomial $W(s)$ tal que $L(s) = L_1(s)W(s)$. □

Nos casos em que $N(s)$ e $D(s)$ estão associadas a um problema de controle, a matriz $R(s)$ será quadrada, isto é, $q=m$ ou $q=p$, conforme o caso.

O conceito de MDC de matrizes leva diretamente ao conceito de matrizes coprimas, quais sejam.

Definição 2.11 Seja $R(s)$ um MDC à direita de $N(s)$ e $D(s)$. Então, as matrizes $D(s)$ e $N(s)$ são coprimas à direita se e somente se $R(s)$ é unimodular. Analogamente, seja $L(s)$ um MDC à esquerda de $D(s)$ e $N(s)$. Logo, as matrizes $D(s)$ e $N(s)$ são coprimas à esquerda se e somente se $L(s)$ é unimodular. □

Por simplicidade, de agora em diante, a matriz $D(s)$ será suposta quadrada ($q=m$). Um importante resultado envolvendo MDC à direita de matrizes polinomiais é apresentado a seguir.

Teorema 2.7 Sejam $N(s) \in \mathbb{R}^{p\times m}[s]$ e $D(s) \in \mathbb{R}^{m\times m}[s]$ e suponha que $|D(s)| \neq 0$. Seja $U(s) \in \mathbb{R}^{(p+m)\times(p+m)}[s]$ uma matriz unimodular formada pelo produto de matrizes elementares tais que

$$U(s)\begin{bmatrix} D(s) \\ N(s) \end{bmatrix} = \begin{bmatrix} R(s) \\ 0 \end{bmatrix}. \tag{2.25}$$

Então $R(s)$ é um MDC à direita de $N(s)$ e $D(s)$.

Prova: Ver [4], página 377. □

Um fato importante sobre máximos divisores comuns de matrizes polinomiais é que eles não são únicos, isto é, se $R_1(s)$ e $R_2(s)$ são MDCs à direita de $N(s)$ e $D(s)$ ($|D(s)| \neq 0$), então existe uma matriz unimodular $U(s)$ tal que $R_1(s) = U(s)R_2(s)$. Também é importante que se diga que se um MDC é unimodular, então todos os MDCs serão unimodulares.

Para finalizar este assunto, uma importante conseqüência do Teorema 2.7 é a chamada Identidade de Bezout [2], [4], [29] apresentada a seguir.

Teorema 2.8 Duas matrizes $N(s) \in \mathbb{R}^{p\times m}[s]$ e $D(s) \in \mathbb{R}^{m\times m}[s]$ são coprimas se e somente se existirem matrizes $\tilde{X}(s) \in \mathbb{R}^{m\times m}[s]$ e $\tilde{Y}(s) \in \mathbb{R}^{m\times p}[s]$ tais que

$$\tilde{X}(s)D(s) + \tilde{Y}(s)N(s) = I_m. \tag{2.26}$$

À equação (2.26) acima se dá o nome de identidade de Bezout.

Prova: Ver [4]. □

De posse dos conceitos apresentados nessa subseção, pode-se abordar o assunto principal que é a Descrição por Frações de Matrizes (DFM) para matrizes racionais.

2.4.3 DFM à Direita e DFM à Esquerda

Em sistemas escalares, uma função racional $g(s)$ pode ser escrita como a razão entre dois polinômios $b(s)$ e $a(s)$, da seguinte forma:

$$g(s) = \frac{b(s)}{a(s)} = b(s)a^{-1}(s) = a^{-1}(s)b(s). \tag{2.27}$$

No caso matricial, uma matriz $G(s) \in \mathbb{R}^{pxm}(s)$ pode inicialmente ser escrita como

$$G(s) = \frac{1}{d(s)} N(s). \tag{2.28}$$

onde $N(s) \in \mathbb{R}^{p \times m}[s]$ e $d(s)$ é um polinômio correspondente ao MMC dos polinômios do denominador de $G(s)$. Note que, assim como no caso escalar, também a matriz $G(s)$ poderá ser escrita pelo produto de uma matriz polinomial pela inversa de outra polinomial. Para tanto, defina:

$$\begin{cases} B(s) = N(s) \\ \widetilde{A}(s) = d(s)I_p \end{cases} \quad e \quad \begin{matrix} A(s) = d(s)I_{\mathrm{m}} \\ \widetilde{B}(s) = \widetilde{N}(s) \end{matrix}.$$

Nessas condições, $G(s)$ poderá ser escrita como:

$$G(s) = B(s)A^{-1}(s) = \widetilde{A}^{-1}(s)\widetilde{B}(s). \tag{2.29}$$

Definição 2.12 Ao produto $B(s)A^{-1}(s)$ $[\widetilde{A}^{-1}(s)\widetilde{B}(s)]$ que satisfaz à equação (2.29), dá-se o nome de Descrição por Frações de Matrizes à direita (à esquerda) de $G(s)$. □

Note que existem diversas maneiras de se obter descrições por frações de matrizes.

2.4.4 DFMs Irredutíveis

Seja $G(s) = N(s)\,D^{-1}(s)$ uma DFM à direita de $G(s)$ e seja $R(s)$ um MDC à direita de $N(s)$ e $D(s)$, isto é, $N(s) = N_R(s)R(s)$ e $D(s) = D_R(s)R(s)$. Assim, pode-se escrever

$$G(s) = N(s)\,D^{-1}(s) = N_R(s)R(s)\,[D_R(s)R(s)]^{-1} = N_R(s)D_R^{-1}(s),$$

que mostra que $N_R(s)D_R^{-1}(s)$ é também uma DFM à direita de $G(s)$.

De acordo com a equação (2.25), particionando-se $U(s)$ apropriadamente, pode-se escrever:

$$U_{11}(s)D(s) + U_{12}(s)N(s) = R(s)$$

$$U_{11}(s)D_R(s)R(s) + U_{12}(s)N_R(s)R(s) = R(s)$$

$$\tilde{X}(s)D_R(s) + \tilde{Y}(s)N_R(s) = I\ ,$$

onde $\tilde{X}(s) = U_{11}(s)$ e $\tilde{Y}(s) = U_{12}(s)$. Assim, de acordo com a Teorema 2.8, $D_R(s)$ e $N_R(s)$ são coprimas. Isso leva à seguinte definição.

Definição 2.13 Uma DFM $G(s) = N(s)D^{-1}(s)$ é irredutível se, e somente se, $N(s)$ e $D(s)$ são coprimas à direita. □

É importante observar que DFMs irredutíveis não são únicas, basta pré-multiplicá-las por matrizes unimodulares.

2.4.5 DFMs Duplamente Coprimas

Uma importante aplicação da identidade de Bezout (2.26) às DFMs de uma matriz racional $G(s) \in \mathbb{R}^{pxm}(s)$ é a obtenção de uma fatoração duplamente coprima de uma dada matriz de transferência. Este problema pode ser formulado como segue.

Teorema 2.9 Sejam $G(s) = N(s)D^{-1}(s) = \tilde{D}^{-1}(s)\tilde{N}(s)$ DFMs irredutíveis à direita e à esquerda, respectivamente. Então existem matrizes polinomiais $\tilde{X}(s)$, $\tilde{Y}(s)$, $X(s)$ e $Y(s)$ tais que

$$\begin{bmatrix} \tilde{X}(s) & -\tilde{Y}(s) \\ -\tilde{N}(s) & \tilde{D}(s) \end{bmatrix}\begin{bmatrix} D(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{2.30a}$$

e

$$\begin{bmatrix} D(s) & Y(s) \\ N(s) & X(s) \end{bmatrix}\begin{bmatrix} \tilde{X}(s) & -\tilde{Y}(s) \\ -\tilde{N}(s) & \tilde{D}(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{2.30b}$$

Às equações (2.30a) e (2.30b) dá-se o nome de Identidade de Bezout Generalizada.

Prova: Ver [4], página 382. □

2.5. Obtenção de uma Realização de Ordem Mínima a Partir de uma DFM Irredutível

Nessa seção será feita uma conexão entre os resultados das seções 2.3 e 2.4. Para tanto, é fundamental o conceito de matrizes reduzidas por coluna.

2.5.1 Matrizes Reduzidas por Coluna

Por uma questão de simplicidade, serão consideradas apenas matrizes reduzidas por coluna, embora, usando a dualidade, seja possível reproduzir este estudo para matrizes reduzidas por linha. Antes de definir o conceito de matriz reduzida por coluna é importante citar o conceito de grau de um vetor polinomial.

Definição 2.14 Seja $d(s) \in \mathbb{R}^{m\times 1}[s]$ um vetor polinomial, o grau de $d(s)$ é dado pelo grau do polinômio de maior grau entre aqueles que pertencem a $d(s)$. □

Definição 2.15 Seja $D(s) \in \mathbb{R}^{m\times m}[s]$ e escreva $D(s)$ como $D(s) = [\underline{d}_1(s)\ \underline{d}_2(s)\ \ldots\ \underline{d}_m(s)]$, onde $\underline{d}_i(s)$, $i=1,2, \ldots,m$, são vetores polinomiais de grau k_i, $i=1,2, \ldots,m$. Então, $D(s)$ é reduzida por coluna se e somente se

$$gr\left\{\det\left[D(s)\right]\right\} = \sum_{i=1}^{m} k_i \,. \tag{2.31}$$ □

Uma outra forma de se caracterizar matrizes reduzidas por coluna é notando que toda matriz $D(s) \in \mathbb{R}^{m\times m}[s]$ pode ser escrita na seguinte forma:

$$D(s) = D_{hc}S(s) + D_{lc}\Psi(s), \tag{2.32}$$

onde $D_{hc} \in \mathbb{R}^{m\times m}$ é uma matriz formada pelos coeficientes de s^{k_i} de cada uma das colunas i de $D(s)$ de grau k_i,

$$S(s) \triangleq diag\left\{s^{k_1}, s^{k_2}, \ldots, s^{k_m}\right\} \tag{2.33}$$

e

$$\Psi(s)=\begin{bmatrix} s^{k_1-1} & & & \\ s^{k_1-2} & & & \\ \vdots & & & \\ s & & & \\ 1 & & & \\ & s^{k_2-1} & & \\ & s^{k_2-2} & & \\ & \vdots & & \\ & s & & \\ & 1 & & \\ & & \ddots & \\ & & & s^{k_m-1} \\ & & & s^{k_m-2} \\ & & & \vdots \\ & & & s \\ & & & 1 \end{bmatrix} \tag{2.34}$$

Pode-se então enunciar o seguinte resultado.

Fato 2.2 Seja $D(s)\in\mathbb{R}^{m\times m}[s]$ e escreva $D(s) = D_{hc}S(s) + D_{lc}\Psi(s)$. Então $D(s)$ é reduzida por coluna se e somente se $\det[D_{hc}]\neq 0$, ou equivalentemente D_{hc} é não-singular. □

A importância do conceito de matrizes reduzidas por coluna será mostrada na próxima seção. Nesse ponto, é importante ressaltar que dada uma matriz $D(s)$ não unimodular, existe uma matriz unimodular $U(s)$ tal que $\overline{D}(s)=D(s)U(s)$ seja reduzida por coluna [27]. Assim, dada uma DFM à direita irredutível $G(s)=N(s)D^{-1}(s)$ em que $D(s)$ não é reduzida por coluna, é sempre possível encontrar uma outra DFM irredutível $G(s)=\overline{N}(s)\overline{D}^{-1}(s)$ em que $\overline{D}(s)$ é reduzida por coluna. Para obter essa DFM basta multiplicar $D(s)$ por $U(s)$, onde $U(s)$ é a matriz unimodular que torna $\overline{D}(s)$ reduzida por coluna [27].

2.5.2 Realização na Forma Controlador

Seja $G(s) \in \mathbb{R}^{p\times m}[s]$, $G(s) = N(s)D^{-1}(s)$ onde $N(s)$ e $D(s)$ são coprimos ($G(s)$ irredutível) e $D(s)$ é reduzida por coluna. Suponha, sem perda de generalidade, que $G(s)$ é estritamente própria. Note que, se $G(s)$ for própria, então $G(s)$ pode ser escrita como $G(s) = G_{ep}(s) + D$, sendo $D = [d_{ij}] \in \mathbb{R}^{p\times m}$, com d_{ij} o quociente da divisão do polinômio do numerador pelo polinômio do denominador de $g_{ij}(s)$ e $G_{ep}(s)$ é estritamente própria. O objetivo dessa seção é obter uma realização em espaço de estado de ordem $n = gr[|D(s)|]$. Tal realização pode ser obtida da seguinte forma.

Como $y(s) = G(s)u(s)$, então $y(s) = N(s)D^{-1}(s)u(s)$. Definindo $\varepsilon(s) = D^{-1}(s)u(s)$, têm-se:

$$\begin{cases} u(s) = D(s)\varepsilon(s) \\ y(s) = N(s)\varepsilon(s) \end{cases}$$

Escrevendo $D(s) = D_{hc}S(s) + D_{lc}\Psi(s)$, onde D_{hc} é a matriz composta pelos coeficientes dos termos de maior grau das colunas de $D(s)$ e D_{lc}, os coeficientes dos demais termos, então como $D(s)$ é, por hipótese, reduzida por coluna, tem-se que D_{hc}^{-1} existe. Portanto $u(s)$ pode ser escrito como:

$$u(s) = D(s)\varepsilon(s) = D_{hc}S(s)\varepsilon(s) + D_{lc}\Psi(s)\varepsilon(s)$$

$$D_{hc}^{-1}u(s) = S(s)\varepsilon(s) + D_{hc}^{-1}D_{lc}\Psi(s)\varepsilon(s)$$

e, conseqüentemente:

$$\begin{cases} S(s)\varepsilon(s) = -D_{hc}^{-1}D_{lc}\Psi(s)\varepsilon(s) + D_{hc}^{-1}u(s) & (2.35a) \\ y(s) = N_{lc}\Psi(s)\varepsilon(s) & (2.35b) \end{cases},$$

uma vez que $G(s)$ é estritamente própria.

Seja $d_i(s)$ a i-ésima coluna de $D(s)$ e suponha que $gr[d_i(s)]=k_i$. Como

$$\varepsilon(s) = \begin{bmatrix} \varepsilon_1(s) & \varepsilon_2(s) & \dots & \varepsilon_m(s) \end{bmatrix}^T \text{ e } S(s) = \begin{bmatrix} s^{k_1} & & & \\ & s^{k_2} & & \\ & & \ddots & \\ & & & s^{k_m} \end{bmatrix}$$

então

$$\Psi(s)\varepsilon(s) = \begin{bmatrix} s^{k_1-1}\varepsilon_1(s) \\ s^{k_1-2}\varepsilon_1(s) \\ \vdots \\ \varepsilon_1(s) \\ \vdots \\ s^{k_m-1}\varepsilon_m(s) \\ s^{k_m-2}\varepsilon_m(s) \\ \vdots \\ \varepsilon_m(s) \end{bmatrix}_{(k_1+k_2+\ldots+k_m)\times 1 = n\times 1} \tag{2.36}$$

e

$$S(s)\varepsilon(s) = \begin{bmatrix} s^{k_1}\varepsilon_1(s) \\ s^{k_2}\varepsilon_2(s) \\ \vdots \\ s^{k_m}\varepsilon_m(s) \end{bmatrix} \tag{2.37}$$

As equações (2.35a), (2.36) e (2.37) sugerem a seguinte atribuição de estados:

$$\begin{cases} x_{11}(t) = \varepsilon_1^{(k_1-1)}(t), & x_{12}(t) = \varepsilon_1^{(k_1-2)}(t), & \cdots, & x_{1k_1}(t) = \varepsilon_1(t), \\ x_{21}(t) = \varepsilon_2^{(k_2-1)}(t), & x_{22}(t) = \varepsilon_2^{(k_2-2)}(t), & \cdots, & x_{2k_2}(t) = \varepsilon_2(t), \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1}(t) = \varepsilon_m^{(k_m-1)}(t), & x_{m2}(t) = \varepsilon_m^{(k_m-2)}(t), & \cdots, & x_{mk_m}(t) = \varepsilon_m(t) \end{cases}.$$

Note que:

$$\begin{cases} \dot{x}_{11}(t) = \varepsilon_1^{(k_1)}(t), & \dot{x}_{12}(t) = \varepsilon_1^{(k_1-1)}(t) = x_{11}(t), & \cdots, & \dot{x}_{1k_1}(t) = x_{1k_1-1}(t), \\ \dot{x}_{21}(t) = \varepsilon_2^{(k_2)}(t), & \dot{x}_{22}(t) = \varepsilon_2^{(k_2-1)}(t) = x_{21}(t), & \cdots, & \dot{x}_{2k_2}(t) = x_{2k_2-1}(t), \\ \vdots & \vdots & & \vdots \\ \dot{x}_{m1}(t) = \varepsilon_m^{(k_m)}(t), & \dot{x}_{m2}(t) = \varepsilon_m^{(k_m-1)}(t) = x_{m1}(t), & \cdots, & \dot{x}_{mk_m}(t) = x_{mk_m-1}(t) \end{cases}.$$

e, assim, calculando-se a transformada inversa de Laplace da equação (2.35a) resulta:

$$\begin{bmatrix} \dot{x}_{11}(t) \\ \dot{x}_{21}(t) \\ \vdots \\ \dot{x}_{m1}(t) \end{bmatrix} = -D_{hc}^{-1} D_{lc} x(t) + D_{hc}^{-1} u(t) .$$

Definindo-se $x_i(t) = [x_{i1}(t) \quad x_{i2}(t) \quad ... \quad x_{iki}(t)]^T$, $i = 1, 2, ..., m$ e $x(t) = \begin{bmatrix} x_1^T(t) & x_2^T(t) & \cdots & x_m^T(t) \end{bmatrix}^T$ e tomando a transformada de Laplace inversa da equação (2.35b), resulta:

$$\begin{cases} \dot{x}(t) = \left(\begin{bmatrix} A_{CO}^{(1)} & & & \\ & A_{CO}^{(2)} & & \\ & & \ddots & \\ & & & A_{CO}^{(m)} \end{bmatrix} - \begin{bmatrix} B_{CO}^{(1)} \\ B_{CO}^{(2)} \\ \vdots \\ B_{CO}^{(m)} \end{bmatrix} D_{hc}^{-1} D_{lc} \right) x(t) + \begin{bmatrix} B_{CO}^{(1)} \\ B_{CO}^{(2)} \\ \vdots \\ B_{CO}^{(m)} \end{bmatrix} D_{hc}^{-1} u(t) \\ y(t) = N_{lc} x(t) \end{cases} \quad (2.38)$$

onde:

$$A_{CO}^{(i)} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}_{k_i \times k_i} , \quad B_{CO}^{(i)} = \begin{bmatrix} e_i^t \\ 0^t \\ 0^t \\ \vdots \\ 0^t \end{bmatrix}_{k_i \times m}$$

com e_i^t, $i = 1, 2, ..., m$, denotando a i-ésima linha da matriz identidade de ordem m. Finalmente, definindo-se

$$A_{CO} = \begin{bmatrix} A_{CO}^{(1)} & & & \\ & A_{CO}^{(2)} & & \\ & & \ddots & \\ & & & A_{CO}^{(m)} \end{bmatrix} \text{ e } B_{CO} = \begin{bmatrix} B_{CO}^{(1)} \\ B_{CO}^{(2)} \\ \vdots \\ B_{CO}^{(m)} \end{bmatrix}$$

chega-se a:

$$\begin{cases} \dot{x}(t) = A_C x(t) + B_C u(t) \\ y(t) = C_C x(t) \end{cases} \quad (2.39)$$

onde

$$A_C = A_{CO} - B_{CO} D_{hc}^{-1} D_{lc} , \; B_C = B_{CO} D_{hc}^{-1} , \; C_C = N_{lc} .$$

A realização (2.39) é denominada forma controlador.

2.6. Pólos e Zeros de um Sistema Multivariável

Nessa seção será considerada a extensão do conceito de pólos e zeros para de sistemas multivariáveis. A maneira de se determinar esses pólos e zeros representa o assunto central desta dissertação, uma vez que isso é feito através da Forma de Smith de matrizes polinomiais. A forma de Smith é um conceito relevante para o estudo de matrizes polinomiais. Com ela é possível achar o posto normal de uma matriz polinomial e, além disso, através da forma de Smith-McMillan, é possível achar os pólos e zeros de um sistema multivariável.

2.6.1 Pólos e Zeros de Sistemas Escalares

Seja $g(s) = b(s)/a(s)$, a função de transferência de um sistema escalar, onde $a(s)$ e $b(s)$ são polinômios de graus n e m, respectivamente ($n \geq m$). É sabido que os conjuntos de pólos ($\mathscr{P}$) e zeros ($\mathscr{Z}$) de $g(s)$ são definidos da seguinte forma:

$$\mathscr{P} = \left\{ p \in \mathbb{C} : \lim_{s \to p} g(s) = \infty \right\} = \{ p \in \mathbb{C} : a(p) = 0 \} \tag{2.40}$$

e

$$\mathscr{Z} = \mathscr{Z}_f \cup \mathscr{Z}_\infty , \tag{2.41}$$

onde

$$\mathscr{Z}_f = \{ z \in \mathbb{C} : g(z) = 0 \} = \{ z \in \mathbb{C} : b(z) = 0 \} \tag{2.42}$$

e

$$\mathscr{Z}_\infty = \{ \infty \text{, com multiplicidade } n\text{–}m \}. \tag{2.43}$$

2.6.2 Forma de Smith de Matrizes Polinomiais

Teorema 2.10 Para qualquer matriz polinomial $N(s) \in \mathbb{R}^{pxm}[s]$, existem matrizes unimodulares $\{U(s), V(s)\}$, tais que

$$U(s)N(s)V(s) = \Sigma_N(s), \tag{2.44}$$

onde

$$\Sigma_N(s) = \left[\begin{array}{cccc|c} \sigma_1(s) & & & & \\ & \sigma_2(s) & & & 0_{r\times(m-r)} \\ & & \ddots & & \\ & & & \sigma_r(s) & \\ \hline & & 0_{(p-r)\times r} & & 0_{(p-r)\times(m-r)} \end{array}\right], \tag{2.45}$$

sendo que $\sigma_i(s) \mid \sigma_{i+1}(s)$, $i = 1, ..., r-1$ ($\sigma_i(s)$ divide $\sigma_{i+1}(s)$) e r é o posto (normal) de $N(s)$. Os polinômios $\sigma_i(s)$, $i = 1, ..., r$, são mônicos, sendo denominados polinômios invariantes de $N(s)$.

Além disso, seja $\Delta_i(s)$ o MDC mônico de todos os menores de ordem i x i de $N(s)$, então se pode identificar

$$\sigma_i(s) = \frac{\Delta_i(s)}{\Delta_{i-1}(s)},\ i = 1, ..., min(p,m) \tag{2.46}$$

onde, por definição, $\Delta_0(s) \triangleq 1$.

Prova: Ver [4], páginas 390-392. □

Definição 2.16 À matriz $\Sigma_N(s)$ definida acima, dá-se o nome de forma de Smith da matriz $N(s) \in \mathbb{R}^{pxm}[s]$. □

É importante ressaltar que, embora $\Sigma_N(s)$ seja única, as matrizes unimodulares $U(s)$ e $V(s)$ não o são. Além disso, como $U(s)$ e $V(s)$ são matrizes unimodulares, então existem as matrizes polinomiais $X(s) = U^{-1}(s)$ e $Y(s) = V^{-1}(s)$ tais que

$$N(s) = X(s)\Sigma_N(s)Y(s). \tag{2.47}$$

O segundo membro da equação acima é denominado decomposição em forma de Smith da matriz polinomial $N(s)$.

A forma de Smith de uma matriz permite generalizar o conceito de zeros para um sistema multivariável. Note que, no caso escalar,

$$g(s)=\frac{n(s)}{d(s)}=\frac{1}{d(s)}[n(s)], \tag{2.48}$$

e, portanto,

$$\Sigma_n(s)=\bar{n}(s), \tag{2.49}$$

com

$$\bar{n}(s)=\frac{n(s)}{a_0}, \tag{2.50}$$

sendo a_0 denotando o coeficiente da potência de s de maior grau de $n(s)$. Desta forma

$$\sigma_1(s)=\bar{n}(s)=(s-z_1)(s-z_2)\cdots(s-z_n), \tag{2.51}$$

o que mostra que os zeros finitos de $g(s)$ são raízes de $\sigma_1(s) = 0$.

Para o caso multivariável, a matriz $\Sigma_N(s)$ não mais ficará identicamente nula para os valores de $z \in \mathbb{C}$, tais que $\sigma_i(z) = 0$, $i = 1,..., r$, e sim perderá posto. Portanto, a forma mais geral de se definir os zeros finitos de uma matriz de transferência é a seguinte.

Definição 2.17 Os zeros finitos de uma matriz de transferência $G(s)\in\mathbb{R}^{pxm}(s)$ são os valores $z\in\mathbb{C}$ tais que $G(z)$ perde posto. □

2.6.3 Forma de Smith-McMillan

Uma vez introduzido o conceito da forma de Smith, relevante para o estudo de matrizes polinomiais, será considerada agora a chamada forma de Smith-McMillan, através da qual é possível determinar os pólos e zeros de um sistema multivariável. Para tanto, seja $G(s)\in\mathbb{R}^{pxm}(s)$, $G(s) = N(s)/d(s)$, sendo $N(s)\in\mathbb{R}^{pxm}[s]$ e $d(s)$ o menor múltiplo comum mônico dos denominadores de $G(s)$. Escreva $N(s)=X(s)\Sigma_N(s)Y(s)$, sendo $\Sigma_N(s)$ a forma de Smith de $N(s)$, $X(s) \in \mathbb{R}^{pxp}[s]$ e $Y(s) \in \mathbb{R}^{mxm}[s]$ matrizes unimodulares. Escreva $G(s)$ como

$$G(s) = \frac{1}{d(s)} X(s)\Sigma_N(s)Y(s) = X(s)\frac{\Sigma_N(s)}{d(s)}Y(s) = X(s)\Gamma_G(s)Y(s)\,, \tag{2.52}$$

onde

$$\Gamma_G(s) = \left[\begin{array}{cccc|c} \frac{\varepsilon_1(s)}{\psi_1(s)} & & & & \\ & \frac{\varepsilon_2(s)}{\psi_2(s)} & & & 0_{r\times(m-r)} \\ & & \ddots & & \\ & & & \frac{\varepsilon_r(s)}{\psi_r(s)} & \\ \hline & & 0_{(p-r)\times r} & & 0_{(p-r)\times(m-r)} \end{array}\right] \tag{2.53}$$

e

$$\frac{\varepsilon_i(s)}{\psi_i(s)} = \frac{\sigma_i(s)}{d(s)}\,,\ i = 1, 2, ..., r. \tag{2.54}$$

sendo $\{\varepsilon_i(s), \psi_i(s)\}$ polinômios coprimos e r o posto (normal) de $G(s)$.

Definição 2.18 À matriz $\Gamma_G(s)$ definida acima se dá o nome de forma de Smith-McMillan de $G(s)$. □

A partir da definição de $\varepsilon_i(s)$ e $\psi_i(s)$ pode-se verificar que:

$$\begin{cases} \psi_i(s) \mid \psi_{i+1}(s), & i = 1,\ldots,r-1, & (2.55) \\ \varepsilon_i(s) \mid \varepsilon_{i+1}(s), & i = 1,\ldots,r-1, & (2.56) \\ d(s) = \psi_1(s) & & (2.57) \end{cases}$$

As propriedades (2.55) e (2.56) são razoavelmente óbvias de se deduzir. Já para se obter a expressão (2.57) note que, se $\psi_1(s) \neq d(s)$ e como $\varepsilon_1(s)/\psi_1(s) = \sigma_1(s)/d(s)$, então $d(s)$ e $\sigma_1(s)$ deveriam ter um fator comum. Conseqüentemente, todos os elementos de $N(s)$ e $d(s)$ deveriam ter esse fator comum, o que contradiz a definição de $d(s)$.

Outro conceito relacionado é o de *grau de McMillan*.

Definição 2.19 O grau de McMillan de uma matriz de transferência é igual à soma dos graus dos polinômios $\psi_i(s)$, $i = 1, 2, ..., r$, da forma de Smith-McMillan dessa matriz, isto é:

$$n = \sum_{i=1}^{r} gr[\psi_i(s)] . \qquad (2.58) \quad \square$$

Pode-se mostrar que o grau de McMillan é igual à ordem da realização mínima associada à função de transferência $\varepsilon(s)$.

Por fim, apresenta-se a definição de pólos e zeros de sistemas multivariáveis.

Definição 2.20 Pólos e zeros finitos de um sistema multivariável são definidos como:

$$\mathscr{P} = \left\{ p \in \mathbb{C} : \lim_{s \to p} G(s) = \infty \right\} = \left\{ p \in \mathbb{C} : \psi_i(p) = 0, \quad i = 1, \ldots, r \right\} \qquad (2.59)$$

e

$$\mathscr{Z}_f = \left\{ z \in \mathbb{C} : \rho[G(z)] < r \right\} = \left\{ z \in \mathbb{C} : \varepsilon_i(z) = 0, \quad i = 1, \ldots, r \right\} \qquad (2.60)$$

onde r é o posto normal de $G(s)$.

2.7. Um Algoritmo para Obtenção da Forma de Smith Usando Operações Elementares de Linhas e Colunas

Uma maneira tradicionalmente usada para se obter a forma de Smith de uma matriz polinomial $A(s)$ é através de operações elementares de linhas e colunas. Um algoritmo foi desenvolvido por Pace e Barnet [14]. O algoritmo produz a forma de Smith

$$S(s) = U(s)A(s)V(s), \qquad (2.61)$$

para qualquer matriz polinomial $A(s) \in \mathbb{R}^{p \times m}[s]$, matrizes unimodulares $U(s) \in \mathbb{R}^{p \times p}[s]$ e $V(s) \in \mathbb{R}^{m \times m}[s]$ e

$$S(s) = \begin{bmatrix} \sigma_{11} & & & & \\ & \ddots & & & \\ & & \sigma_{rr} & & \\ & & & 0 & \\ & & & & \ddots \end{bmatrix}, \qquad (2.62)$$

onde r é o posto de $A(s)$ e $\sigma_{kk}|\sigma_{k+1,k+1}$, para $k = 1, 2, ..., r-1$.

O algoritmo consiste em executar operações elementares nas linhas e colunas de $A(s)$. Primeiro escolhe-se o polinômio diferente de zero de menor grau de $A(s)$ e leva-o para a posição (1,1) por trocas de linhas e colunas. Então, subtraem-se múltiplos deste polinômio dos outros polinômios na primeira linha e coluna de forma a reduzir os graus dos polinômios da primeira linha e coluna. Após isso, determina-se um novo polinômio em $S(s)$ ($S(s) \leftarrow A(s)$)[1] de menor grau, caso exista, e leva-o para a posição (1,1) e repete-se o processo para todos os polinômios na primeira linha e coluna até todos serem reduzidos a zero. A repetição dos passos acima nas linhas e colunas restantes produzirá a uma matriz constituída por dois blocos, no superior uma matriz diagonal e no inferior uma matriz polinomial qualquer.

Seja

$$S(s) \leftarrow \left[\begin{array}{ccc|ccc} \sigma_{11} & & & & & \\ & \ddots & & & & \\ & & \sigma_{k-1,k-1} & & & \\ \hline & & & \sigma_{kk} & \cdots & \sigma_{km} \\ & & & \vdots & & \vdots \\ & & & \sigma_{pk} & \cdots & \sigma_{pm} \end{array}\right] \qquad (2.63)$$

a matriz $S(s)$ cujo bloco superior tem dimensão $(k-1)$, com $\sigma_{11} \mid \sigma_{22} \mid ... \mid \sigma_{k-1,k-1}$.

Realizando o procedimento descrito acima na primeira linha e coluna do bloco inferior obtém-se:

[1] O símbolo $\leftarrow$ será usado para indicar o valor assumido pela matriz ao fim da operação descrita a seguir.

$$S(s) \leftarrow \begin{bmatrix} \sigma_{11} & & & & & \\ & \ddots & & & & \\ & & \sigma_{k-1,k-1} & & & \\ & & & \bar{\sigma}_{kk} & & \\ & & & & \bar{\sigma}_{k+1,k+1} & \cdots & \bar{\sigma}_{k+1,m} \\ & & & & \vdots & & \vdots \\ & & & & \bar{\sigma}_{p,k+1} & \cdots & \bar{\sigma}_{p,m} \end{bmatrix}.$$

Resta agora checar se $\sigma_{k-1,k-1}$ divide $\bar{\sigma}_{k,k}$, já que isto é uma precondição da forma de Smith. Confirmado isto, poder-se-á continuar com o algoritmo. Se não, alguns passos devem ser seguidos. O primeiro é calcular um máximo divisor comum $g(s)$ entre $\sigma_{k-1,k-1}$ e $\bar{\sigma}_{k,k}$ e expressá-lo na forma:

$$\sigma_{k-1,k-1}x(s) + \bar{\sigma}_{kk}\,y(s) = g(s)\,. \tag{2.64}$$

Em seguida, adiciona-se a linha $(k-1)$ multiplicada por $x(s)$ à linha k e adiciona-se também a coluna k multiplicada por $y(s)$ à coluna $(k-1)$ resultando

$$S(s) \leftarrow \begin{bmatrix} \sigma_{11} & & & & & \\ & \ddots & & & & \\ & & g & \bar{\sigma}_{kk} & & \\ & & \sigma_{k-1,k-1} & 0 & & \\ & & & & \bar{\sigma}_{k+1,k+1} & \cdots & \bar{\sigma}_{k+1,m} \\ & & & & \vdots & & \vdots \\ & & & & \bar{\sigma}_{p,k+1} & \cdots & \bar{\sigma}_{p,m} \end{bmatrix}. \tag{2.65}$$

Os passos principais do algoritmo são, então, recomeçados para um novo bloco diagonal inferior formado pelas $p-(k+2)$ linhas e $m-(k+2)$ colunas restantes de $S(s)$. O procedimento termina quando o bloco no canto inferior direito consistir inteiramente de zeros, ou $S(s)$ for uma matriz diagonal.

Já para as matrizes unimodulares $U(s)$ e $V(s)$, deve-se iniciá-las como $U(s) = I_p$ e $V(s) = I_m$ e executando as mesmas operações com linhas sobre $S(s)$ em $U(s)$ e as mesmas operações com colunas sobre $S(s)$ em $V(s)$, consegue-se a forma de Smith de $A(s)$, assim como as matrizes de transformação unimodulares $U(s)$ e $V(s)$.

O desenvolvimento acima pode ser resumido no seguinte algoritmo.

Algoritmo 2.1

Passo 1 Faça $U(s) = I_p$ e $V(s) = I_m$.

Passo 2 $k = 0$.

Passo 3 $k = k + 1$.

Passo 4 Determine a posição do polinômio diferente de zero de menor grau, chame-o σ_{ij}, o bloco de linhas $k, k+1, ..., p$ e colunas $k, k+1, ..., m$ de $S(s)$. Se for todo zero, faça $r = k - 1$ e pare.

Passo 5 Se $i \neq k$ troque as linhas i e k de ambas $S(s)$ e $U(s)$.

Passo 6 Se $j \neq k$ troque as colunas j e k de ambas $S(s)$ e $V(s)$.

Passo 7 Se há somente um polinômio diferente de zero na coluna k de $S(s)$ vá para o Passo 11.

Passo 8 Divida o coeficiente do termo de maior grau de σ_{kk} do termo de maior grau de σ_{nk}, $n = k+1, ..., p$, chamando o resultado de λ_n. Subtraia o grau de σ_{kk} do grau de σ_{nk}, $n = k+1, ..., p$, chamando o resultado de u_n.

Passo 9 Subtraia $\lambda_n s^{u_n}$ vezes a linha k de $S(s)$ da linha n, $n = k+1, ..., p$. Faça as mesmas operações em $U(s)$.

Passo 10 Vá para o Passo 4.

Passo 11 Se há somente um polinômio diferente de zero na linha k de $S(s)$, vá para o Passo 15.

Passo 12 Divida o coeficiente do termo de maior grau de σ_{kk} pelo coeficiente de maior grau de σ_{kn}, $n = k+1, ..., m$, chamando o resultado de μ_n. Subtraia o grau de σ_{kk} do grau de σ_{kn}, $n = k+1, ..., m$, chamando o resultado de v_n.

Passo 13 Subtraia $\mu_n s^{v_n}$ vezes a coluna k de $S(s)$ da coluna n, $n = k+1, \ldots, m$.

Realize as mesmas operações em $V(s)$.

Passo 14 Vá para o Passo 4.

Passo 15 Se $k = 1$ vá para o Passo 21.

Passo 16 Calcule $g(s)$, o MDC entre $\sigma_{k-1,k-1}$ e σ_{kk}, com os polinômios $x(s)$ e $y(s)$ tal que

$$\sigma_{k-1,k-1}x + \sigma_{k,k} = g.$$

Passo 17 Se $\sigma_{k-1,k-1} = g$ vá para o Passo 21.

Passo 18 Adicione x vezes a linha $(k-1)$ à linha k de ambas $S(s)$ e $U(s)$.

Passo 19 Adicione y vezes a coluna k à coluna $(k-1)$ de ambas $S(s)$ e $V(s)$.

Passo 20 $k = k - 2$.

Passo 21 Se $k \neq l$ e $k \neq m$, vá para o Passo 3, se não faça $r = k$ e pare. □

Completando o procedimento, a forma de Smith $S(s)$ substitui a matriz $A(s)$ dada. Isto é, $\sigma_{11}, \sigma_{22}, \ldots, \sigma_{rr}$ são os polinômios invariantes.

2.8. Exemplos Numéricos do Algoritmo

Para ilustrar o algoritmo proposto na seção anterior, serão apresentados, agora, alguns exemplos.

Exemplo 2.2: Considere a matriz $A(s) \in \mathbb{R}^{2\times3}[s]$, dada por:

$$A(s) = \begin{bmatrix} -s^2+1 & 0 & -s+1 \\ 0 & s+1 & 0 \end{bmatrix}.$$

Todos os passos envolvidos na redução serão descritos, sendo as matrizes do produto $S(s)=U(s)A(s)V(s)$, $U(s) \in \mathbb{R}^{2\times2}[s]$, $A(s) \in \mathbb{R}^{2\times3}[s]$ e $V(s) \in \mathbb{R}^{3\times3}[s]$, atualizadas após cada passo.

(1) Defina $U(s)=I_2$ e $V(s)=I_3$, sendo I_2 e I_3 matrizes identidade de ordens 2 e 3, respectivamente ($S(s) \leftarrow A(s)$). Portanto:

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s^2+1 & 0 & -s+1 \\ 0 & s+1 & 0 \end{bmatrix}.$$

(2) De acordo com o Passo 4 do Algoritmo 2.1, primeiro deve-se identificar a posição do polinômio de menor grau de $S(s)$: posição (1,3). Deve-se então, como no Passo 5 e no Passo 6, trocar as colunas 1 e 3 de $S(s)$ e $V(s)$:

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s+1 & 0 & -s^2+1 \\ 0 & s+1 & 0 \end{bmatrix}.$$

(3) O Passo seguinte a ser considerado (Passo 13) é reduzir o polinômio da linha 1, coluna 3 buscando sempre tornar $S(s)$ uma matriz diagonal. Para tanto, deve-se multiplicar a coluna 1 por s e diminuir o produto da coluna 3 de $S(s)$ e $V(s)$, obtendo-se:

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & -s \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s+1 & 0 & -s+1 \\ 0 & s+1 & 0 \end{bmatrix}.$$

(4) O elemento (1,3) pode ser ainda reduzido. Para tanto, deve-se subtrair a coluna 1 da coluna 3 de $S(s)$ e $V(s)$, resultando

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s+1 & 0 & 0 \\ 0 & s+1 & 0 \end{bmatrix}.$$

(5) Note que, o polinômio $(-s+1)$ não divide $(s+1)$, desta forma, como estabelece o Passo 16, calcule x e y da expressão (2.64). Assim, devem-se os calcular polinômios $x(s)$, $y(s)$ e $g(s)$ tais que:

$$(-s+1)x + (s+1)y = g,$$

onde $g(s)$ é o MDC entre $(-s+1)$ e $(s+1)$, que, neste caso, é uma constante qualquer. Para facilitar os cálculos, escolhe-se 2 e, portanto:

$$(-s+1)(1) + (s+1)(1) = 2.$$

(6) De acordo com o Passo 18, deve-se adicionar o produto da linha 1 por x=1 à linha 2 de ambas $S(s)$ e $U(s)$:

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s+1 & 0 & 0 \\ -s+1 & s+1 & 0 \end{bmatrix}.$$

(7) Em seguida (Passo 19), deve-se adicionar o produto da coluna 2 por y=1 à coluna 1 de $S(s)$ e $V(s)$, obtendo-se:

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} -s+1 & 0 & 0 \\ 2 & s+1 & 0 \end{bmatrix}.$$

(8) Identificado que o polinômio de menor grau de $S(s)$ está na posição (2,1), deve-se, de acordo com o Passo 5, trocar as linhas 1 e 2 de $S(s)$ e $U(s)$

$$U(s) \leftarrow \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} 2 & s+1 & 0 \\ -s+1 & 0 & 0 \end{bmatrix}.$$

(9) Mais uma vez, deve-se reduzir os polinômios da coluna 1, linha 2. Para tanto, deve-se subtrair o produto da linha 1 por $-0,5s$ da linha 2 de $S(s)$ e $U(s)$:

$$U(s) \leftarrow \begin{bmatrix} 1 & 1 \\ 0,5s+1 & 0,5s \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} 2 & s+1 & 0 \\ 1 & 0,5s^2+0,5s & 0 \end{bmatrix}.$$

(10) Em seguida, deve-se subtrair o produto da linha 1 por $0,5$ da linha 2 das matrizes $S(s)$ e $U(s)$, para obter:

$$U(s) \leftarrow \begin{bmatrix} 1 & 1 \\ 0,5s+0,5 & 0,5s-0,5 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} 2 & s+1 & 0 \\ 0 & 0,5s^2-0,5 & 0 \end{bmatrix}.$$

(11) Para reduzir o elemento (1,2), deve-se multiplicar a coluna 1 por (*0,5s+0,5*) e diminuir este produto da coluna 2 em ambas as matrizes *S*(*s*) e *V*(*s*):

$$U(s) \leftarrow \begin{bmatrix} 1 & 1 \\ 0,5s+0,5 & 0,5s-0,5 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 0 & 0 & 1 \\ 1 & -0,5s+0,5 & 0 \\ 1 & -0,5s-0,5 & -s-1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0,5s^2-0,5 & 0 \end{bmatrix}.$$

Assim, a forma de Smith *S*(*s*) = *U*(*s*)*A*(*s*)*V*(*s*) é:

$$S(s) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0,5s^2-0,5 & 0 \end{bmatrix}$$

e as matrizes unimodulares *U*(*s*) e *V*(*s*) são

$$U(s) = \begin{bmatrix} 1 & 1 \\ 0,5s+0,5 & 0,5s-0,5 \end{bmatrix} \text{ e } V(s) = \begin{bmatrix} 0 & 0 & 1 \\ 1 & -0,5s+0,5 & 0 \\ 1 & -0,5s-0,5 & -s-1 \end{bmatrix}. \qquad \square$$

Exemplo 2.3: Considere, agora, a seguinte matriz polinomial [24] $A(s) \in \mathbb{R}^{3\times 3}[s]$

$$A(s) = \frac{1}{s^4}\begin{bmatrix} s^3-s^2+1 & 1 & -s^3+s^2-2 \\ 1,5s+1 & s+1 & -1,5s-2 \\ s^3-9s^2-s+1 & -s^2+1 & s^3-s-2 \end{bmatrix}.$$

Para obter a forma de Smith de *A*(*s*), de acordo com o Algoritmo 2.1, é necessário, inicialmente, definir *S*(*s*) e as matrizes unimodulares correspondentes. Assim:

$$U(s) \leftarrow I_3,\; S(s) \leftarrow A(s) \text{ e } V(s) \leftarrow I_3,$$

ou seja,

$$U(s) \leftarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S(s) \leftarrow \begin{bmatrix} s^3-s^2+1 & 1 & -s^3+s^2-2 \\ 1,5s+1 & s+1 & -1,5s-2 \\ s^3-9s^2-s+1 & -s^2+1 & s^3-s-2 \end{bmatrix}, \quad V(s) \leftarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Feito isso, determina-se a posição do polinômio não identicamente nulo de menor grau na matriz *S*(*s*) acima, ou seja, o polinômio da posição (1,2). Desta forma, o passo seguinte é trocar as colunas 1 e 2 de *S*(*s*) e *V*(*s*) e, conseqüentemente,

$$U(s)\leftarrow\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},\quad S(s)\leftarrow\begin{bmatrix}1&s^3-s^2+1&-s^3+s^2-2\\s+1&1{,}5s+1&-1{,}5s-2\\-s^2+1&s^3-9s^2-s+1&s^3-s-2\end{bmatrix},\quad V(s)\leftarrow\begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}.$$

De acordo com os passos do algoritmo, devem-se reduzir os polinômios da primeira coluna de $S(s)$ a zero, a menos daquele na posição (1,1). Para tanto, deve-se fazer:

$$U(s)\leftarrow\begin{bmatrix}1&0&0\\-s-1&1&0\\s^2-1&0&1\end{bmatrix},\quad S(s)\leftarrow\begin{bmatrix}1&s^3-s^2+1&-s^3+s^2-2\\0&-s^4+s^2+0{,}5s&s^4-s^2+0{,}5s\\0&s^5-s^4-7s^2-s&-s^5+s^4+2s^3-3s^2-s\end{bmatrix},\quad V(s)\leftarrow\begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}.$$

O procedimento acima anulou os polinômios da primeira coluna de $S(s)$ abaixo do elemento (1,1). A seguir, deve-se, por procedimento análogo (Passos 11 a 14), reduzir os polinômios da primeira linha, exceto o do elemento (1,1). Assim, as matrizes $U(s)$, $S(s)$ e $V(s)$ resultantes serão:

$$U(s)\leftarrow\begin{bmatrix}1&0&0\\-s-1&1&0\\s^2-1&0&1\end{bmatrix},\quad S(s)\leftarrow\begin{bmatrix}1&0&0\\0&-s^4+s^2+0{,}5s&s^4-s^2+0{,}5s\\0&s^5-s^4-7s^2-s&-s^5+s^4+2s^3-3s^2-s\end{bmatrix},\quad V(s)\leftarrow\begin{bmatrix}0&1&0\\1&-s^3+s^2-1&s^3-s^2+2\\0&0&1\end{bmatrix}.$$

O passo seguinte é à diagonalização do bloco composto pelas linhas e colunas 2 e 3 de modo a ter as posições (2,3) e (3,2) anuladas. Após algumas manipulações que incluem trocas de linhas objetivando sempre ter o termo de menor grau na posição (2,2), chega-se à expressão:

$$U(s)\leftarrow\begin{bmatrix}1&0&0\\-0{,}02s^2+0{,}12s+0{,}14&0{,}01s^3-0{,}02s^2-0{,}11&0{,}01s^2-0{,}01\\23s^3-150s^2-210s-35&-23s^4+23s^3+160s+23&-23s^3+23s+12\end{bmatrix},$$

$$S(s)\leftarrow\begin{bmatrix}1&0&0\\0&-0{,}04s&0\\0&0&-46s^6+210s^5+120s^4-210s^3\end{bmatrix},$$

$$V(s) \leftarrow \begin{bmatrix} 0 & 1 & 0,81s^4 - 3,8s^3 - 1,3s^2 + 4s - 1 \\ 1 & -s^3 + s^2 - 1 & -0,81s^7 + 4,6s^6 + 2,5s^5 - 6,1s^4 + 9,8s^3 - 0,72s^2 - 4s + 3 \\ 0 & 0 & 1 \end{bmatrix}.$$

Por fim, para obterem-se polinômios mônicos, realizam-se algumas multiplicações por escalares nas matrizes $S(s)$ e $U(s)$. Com isso, a forma de Smith de *A(s)* é

$$S(s) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s^6 - 4{,}5s^5 - 2{,}5s^4 + 4{,}5s^3 \end{bmatrix}$$

e as matrizes unimodulares são

$$U(s) = \begin{bmatrix} 1 & 0 & 0 \\ 0{,}4s^2 + 2{,}7s - 3{,}1 & -0{,}4s^3 + 0{,}48s^2 - 0{,}1s + 2{,}7 & -0{,}4s^2 + 0{,}07s - 0{,}37 \\ 0{,}49s^3 + 3{,}2s^2 + 4{,}5s + 0{,}74 & 0{,}49s^4 - 0{,}49s^3 + 3{,}5s - 0{,}49 & 0{,}49s^3 - 0{,}49s - 0{,}24 \end{bmatrix}$$

e

$$V(s) = \begin{bmatrix} 0 & 1 & 0,81s^4 - 3,8s^3 - 1,3s^2 + 4s - 1 \\ 1 & -s^3 + s^2 - 1 & -0,81s^7 + 4,6s^6 + 2,5s^5 - 6,1s^4 + 9,8s^3 - 0,72s^2 - 4s + 3 \\ 0 & 0 & 1 \end{bmatrix}. \qquad \square$$

Para o próximo exemplo, será feito uso do *Matlab*. Uma função para cálculo da forma de Smith, assim como das matrizes unimodulares, foi desenvolvida a partir do Algoritmo 2.1. Esta função, chamada *smithkuc.m*, está listada na seção A.1 do apêndice.

Exemplo 2.4: Considere agora a matriz $A(s) \in \mathbb{R}^{3\times 2}[s]$,

$$A(s) = \begin{bmatrix} s + 2 & s + 1 \\ s^2 + 3s + 3 & s^2 - 1 \\ s + 1 & s^2 + 3s + 2 \end{bmatrix}.$$

Para dar entrada na função *smithkuc*, a matriz polinomial $A(s)$ deve ser escrita na forma de matriz de coeficientes:

$$A(s)=\begin{bmatrix}0 & 0\\ 1 & 1\\ 0 & 1\end{bmatrix}s^2+\begin{bmatrix}1 & 1\\ 3 & 0\\ 1 & 3\end{bmatrix}s+\begin{bmatrix}2 & 1\\ 3 & -1\\ 1 & 2\end{bmatrix},$$

para o Matlab,

$$A(s)=\begin{bmatrix}0 & 0 & 1 & 1 & 2 & 1\\ 1 & 1 & 3 & 0 & 3 & -1\\ 0 & 1 & 1 & 3 & 1 & 2\end{bmatrix}.$$

Assim, as linhas de comando são:

```
>> A=[0 0 1 1 2 1;1 1  3 0 3 -1; 0 1 1 3 1 2];

>> [S,U,V,r] = smithkuc(A,2);
```

A função apresenta os seguintes resultados:

(1) Forma de Smith $S(s)$ da matriz de entrada $A(s)$ é:

$$S(s)=\begin{bmatrix}1 & 0\\ 0 & s+1\\ 0 & 0\end{bmatrix};$$

(2) As matrizes unimodulares $U(s)$ e $V(s)$ são:

$$U(s)=\begin{bmatrix}-s-1 & 1 & 0\\ 0{,}14s^2+0{,}71s+1 & -0{,}14s-0{,}57 & -0{,}28\\ 0 & 0 & 0\end{bmatrix} \text{ e } V(s)=\begin{bmatrix}1 & 2s+2\\ 0 & 1\end{bmatrix}$$

e ainda,

(3) Posto normal da matriz $A(s)$:

$$r = 2. \qquad \square$$

Exemplo 2.5: Desta vez considere uma matriz com a presença de uma linha nula. Seja a matriz $A(s)\in\mathbb{R}^{3\times 3}[s]$,

$$A(s)=\begin{bmatrix}s+1 & 0 & s^2+s\\ 0 & s+2 & s^2+2s\\ 0 & 0 & 0\end{bmatrix}.$$

Assim, escreve-se $A(s)$ na forma de matriz de coeficientes

$$A(s)=\begin{bmatrix}0&0&1\\0&0&1\\0&0&0\end{bmatrix}s^2+\begin{bmatrix}1&0&1\\0&1&2\\0&0&0\end{bmatrix}s+\begin{bmatrix}1&0&0\\0&2&0\\0&0&0\end{bmatrix}$$

e, para o *Matlab*, as linhas de comando são:

```
>> A=[0 0 1 1 0 1 1 0 0;0 0 1 0 1 2 0 2 0;0 0 0 0 0 0 0 0 0];
```

```
>> [S,U,V,r] = smithkuc(A);
```

Após o cálculo chega-se ao seguinte resultado, primeiro a forma de Smith $S(s)$ da matriz de entrada $A(s)$:

$$S(s)=\begin{bmatrix}1&0&0\\0&s^2+3s+2&0\\0&0&0\end{bmatrix}.$$

A seguir, as matrizes unimodulares $U(s)$ e $V(s)$:

$$U(s)=\begin{bmatrix}-1&0{,}37&0\\-2{,}6s-5{,}3&s+1&0\\0&0&0\end{bmatrix}\text{ e }V(s)=\begin{bmatrix}1&-0{,}37s-0{,}75&-s\\2{,}6&-s-1&-s\\0&0&1\end{bmatrix}.$$

Por fim, o posto normal de $A(s)$ é $r = 2$. □

Exemplo 2.6: Como último exemplo desta seção, considere a matriz $A(s)\in\mathbb{R}^{3\times3}[s]$,

$$A(s)=\begin{bmatrix}9{,}5\times10^{-7}s^2+0{,}44s+0{,}41 & 0{,}49s^2+0{,}92s+0{,}35 & 0{,}46s^2+8{,}1\times10^{-9}s+0{,}14\\ 2{,}3\times10^{-7}s^2+0{,}62s+0{,}89 & 0{,}89s^2+0{,}74s+0{,}81 & 0{,}019s^2+1{,}9\times10^{-8}s+0{,}2\\ 6{,}1\times10^{-7}s^2+0{,}79s+0{,}058 & 0{,}76s^2+0{,}18s+0{,}0099 & 0{,}82s^2+1{,}8\times10^{-8}s+0{,}2\end{bmatrix}$$

Escrevendo $A(s)$ na forma de matriz de coeficientes e aplicando a função *smithkuc*, observa-se que os resultados são sensíveis à precisão adotada. Ou seja, caso a precisão adotada seja menor que seis casas decimais (1×10^{-6}), pode-se arredondar $A(s)$ para

$$A(s)=\begin{bmatrix}0{,}44s+0{,}41 & 0{,}49s^2+0{,}92s+0{,}35 & 0{,}46s^2+0{,}14\\ 0{,}62s+0{,}89 & 0{,}89s^2+0{,}74s+0{,}81 & 0{,}019s^2+0{,}2\\ 0{,}79s+0{,}058 & 0{,}76s^2+0{,}18s+0{,}0099 & 0{,}82s^2+0{,}2\end{bmatrix}$$

e sua forma de Smith é

$$S(s) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s^5 + 6,4s^4 + 20s^3 - 1,2s^2 + 3,5s - 0,06 \end{bmatrix}.$$

Contudo, caso a precisão adotada seja igual ou superior a sete casas decimais (1×10^{-7}), observa-se que o Algoritmo 2.1 não consegue determinar a forma de Smith de $A(s)$. Desta forma demonstra-se, aqui, uma limitação do método clássico de obtenção da forma de Smith de uma matriz polinomial por operações elementares de linhas e colunas.

2.9. Análise e Conclusões

Este capítulo trouxe uma breve introdução ao conceito de matrizes polinomiais. Foram abordados vários assuntos relacionados à sua análise, pontos que serviram de base para a introdução do tema central desta dissertação que é a determinação da forma de Smith de matrizes polinomiais. Sobre ela, foram descritos seu conceito e sua formulação e apresentada sua forma relacionada, a forma de Smith-McMillan. Foi listado, ainda, um algoritmo para obtenção da forma de Smith usando operações elementares de linhas e colunas e, por fim, foram desenvolvidos alguns exemplos de aplicação deste algoritmo.

São vários os métodos e algoritmos presentes na literatura para a obtenção da forma de Smith de matrizes polinomiais tal qual aquele apresentado neste capítulo. Entre eles, destacam-se os métodos baseados em soluções determinísticas usando trocas de linhas e colunas [1], [14], [15] e métodos probabilísticas [19], [20], [21], [22] que são baseados principalmente na determinação de multiplicadores aleatórios para facilitar o processo de cálculo. Outros métodos determinísticos conhecidos [16], [17], [23] também podem ser citados, porém há de se observar que estes apresentam aplicação limitada a certos tipos de matrizes, ou a dimensões reduzidas.

O método clássico de obtenção da forma de Smith de matrizes polinomiais, ou seja, o método de manipulações elementares de linhas e colunas, apesar de ser o mais comumente usado também apresenta limitações. Para sistemas de ordem superior o número de cálculos envolvendo polinômios é muito grande, o que compromete o desempenho dos algoritmos baseados neste método, portanto o método sofre de instabilidade numérica.

Neste capítulo, através do Exemplo 2.6, pode-se observar também que o método clássico é sensível à precisão adotada. Para polinômios com coeficientes muito próximos de zero não é possível atingir o resultado desejado através de operações elementares de linhas e colunas. Isto se deve ao seguinte fato: a diagonalização da matriz de entrada se dá por meio do pivoteamento de suas colunas e esse pivoteamento é função desses coeficientes muito próximos de zero. Assim, não se encontram fatores que consigam anular os polinômios indesejados nas posições que não pertençam à diagonal e a matriz na forma de Smith não é obtida.

3. Obtenção da forma de Smith de matrizes polinomiais via descrição em espaço de estado e base polinomial mínima

"Não basta saber, é preciso também aplicar; não basta querer, é preciso também agir."
Goethe (1749-1832)

Neste capítulo será proposto um método para obtenção da forma de Smith de matrizes polinomiais. O método proposto nesse trabalho consiste basicamente em se obter a forma de Smith de uma dada matriz polinomial $A(s)$ através da determinação dos autovalores da matriz de estados de uma realização controlável (forma controlador) de uma matriz de transferência associada à matriz polinomial em questão e do comprimento das cadeias de Jordan [6], [25], [26] associadas a esses autovalores. Serão também obtidas as matrizes unimodulares que levam à forma de Smith.

Este capítulo está dividido da seguinte forma: na seção 3.1 são apresentados alguns fundamentos matemáticos que são ferramentas para os cálculos subseqüentes, entre eles é introduzido o Teorema de Binet-Cauchy [1], [4] que será crucial no desenvolvimento do método proposto. Na seção 3.2 são apresentados alguns resultados preliminares para o que é apresentando na seção 3.3, a formulação do problema de se obter a decomposição na forma de Smith a partir de uma determinada realização em espaço de estado associada a uma matriz de transferência e na seção 3.4, apresenta-se um algoritmo para obtenção da forma de Smith de matrizes polinomiais. Por fim, na seção 3.5 são apresentados alguns exemplos de aplicação deste algoritmo.

3.1. Fundamentos Matemáticos

3.1.1 Fórmula de Binet-Cauchy

Suponha que uma matriz quadrada $A \in \mathbb{R}^{m \times m}$ seja o produto de duas matrizes retangulares $B \in \mathbb{R}^{m \times p}$ e $C \in \mathbb{R}^{p \times m}$, isto é,

$$\begin{bmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mm} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mp} \end{bmatrix} \begin{bmatrix} c_{11} & \cdots & c_{1m} \\ c_{21} & \cdots & c_{2m} \\ \vdots & \ddots & \vdots \\ c_{p1} & \cdots & c_{pm} \end{bmatrix}, \tag{3.1}$$

sendo que,

$$a_{ij} = \sum_{\alpha=1}^{p} b_{i\alpha} c_{\alpha j} \quad (i, j = 1, 2, \dots, m). \tag{3.2}$$

Suponha ainda que $A^{i_1, i_2, \dots, i_p}_{k_1, k_2, \dots, k_p}$, denote o menor de ordem *p* da matriz *A* de dimensão *m x m*, formado pelas linhas i_1, i_2, ..., i_p, e colunas k_1, k_2, ..., k_p, tal que $1 \le i_1 < i_2 < \dots < i_p$ e $1 \le k_1 < k_2 < \dots < k_p$ (p<m). Portanto,

$$A^{i_1, i_2, \dots, i_p}_{k_1, k_2, \dots, k_p} = \begin{vmatrix} a_{i_1 k_1} & a_{i_1 k_2} & \cdots & a_{i_1 k_p} \\ a_{i_2 k_1} & a_{i_2 k_2} & \cdots & a_{i_2 k_p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i_p k_1} & a_{i_p k_2} & \cdots & a_{i_p k_p} \end{vmatrix}. \tag{3.3}$$

Estabelece-se agora a fórmula de Binet-Cauchy que expressa o determinante de *A* em termos dos menores de *B* e *C*.

Lema 3.1 (Fórmula de Binet-Cauchy) Sejam as matrizes $A \in \mathbb{R}^{m \times m}, B \in \mathbb{R}^{m \times p}$ e $C \in \mathbb{R}^{p \times m}$, tais que *A*=*BC*. Então o determinante de *A* pode ser escrito como:

$$\begin{vmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mm} \end{vmatrix} = \sum_{k_1,k_2,\ldots,k_m \in \kappa_m} \left\{ \begin{vmatrix} b_{1k_1} & \cdots & b_{1k_m} \\ \vdots & \ddots & \vdots \\ b_{mk_1} & \cdots & b_{mk_m} \end{vmatrix} \begin{vmatrix} c_{k_1 1} & \cdots & c_{k_1 m} \\ \vdots & \ddots & \vdots \\ c_{k_m 1} & \cdots & c_{k_m m} \end{vmatrix} \right\} \tag{3.4}$$

ou, utilizando a notação da equação (3.3),

$$A_{1,2,\ldots,m}^{1,2,\ldots,m} = \sum_{k_1,k_2,\ldots,k_m \in \kappa_m} \left\{ B_{k_1,k_2,\ldots,k_m}^{1,2,\ldots,m} C_{1,2,\ldots,m}^{k_1,k_2,\ldots,k_m} \right\}, \tag{3.5}$$

onde κ_m denota o conjunto formado por todas as

$$\binom{p}{m} = \frac{p!}{(p-m)!m!}, \tag{3.6}$$

possíveis combinações das colunas (linhas) de B (C).

Prova: Ver [1]. □

De acordo com a fórmula (3.5), o determinante de A é a soma dos produtos de todos os menores de ordem m de B pelos menores correspondentes de mesma ordem de C.

A fórmula de Binet-Cauchy permite, de forma geral, expressar os menores de $A=BC$ em termos dos menores de seus fatores. Para tanto,escreve-se B e C como

$$B = \begin{bmatrix} \underline{b}_1^T \\ \underline{b}_2^T \\ \vdots \\ \underline{b}_m^T \end{bmatrix}, \; C = \begin{bmatrix} \underline{c}_1 & \underline{c}_2 & \cdots & \underline{c}_m \end{bmatrix}, \tag{3.7}$$

onde $\underline{b}_i$, $\underline{c}_i \in \mathbb{R}^p$. Portanto,

$$A_{j_1,j_2,\ldots,j_p}^{i_1,i_2,\ldots,i_p} = \left| \begin{bmatrix} \underline{b}_{i_1}^T \\ \underline{b}_{i_2}^T \\ \vdots \\ \underline{b}_{i_p}^T \end{bmatrix} \begin{bmatrix} \underline{c}_{j_1} & \underline{c}_{j_2} & \cdots & \underline{c}_{jp} \end{bmatrix} \right|. \tag{3.8}$$

3.1.2 Autovalores e Forma Canônica de Jordan

Uma das maneiras de se obter a forma canônica de Jordan de uma dada matriz, é através do método desenvolvido por Rubin [34]. O método foi derivado do seguinte teorema.

Teorema 3.1 Seja $A \in \mathbb{R}^{m\times m}$ uma matriz m x m e suponha que λ seja um autovalor de A. Para cada $\mu=1, 2, 3, ...$, seja s_μ o número de blocos menores de ordem μ, em um bloco superior M associado ao autovalor λ, da forma canônica de Jordan de A. Então s_μ é dado por:

$$s_\mu = r_{\mu+1} - 2r_\mu + r_{\mu-1}, \quad \mu = 1,2,3,... \tag{3.9}$$

onde[1],

$$r_\mu = posto(B^\mu)=\rho(B^\mu) \tag{3.10}$$

e

$$B^\mu = A - \lambda I_m. \tag{3.11}$$

A dimensão do maior dos blocos de menores é o menor inteiro não negativo p tal que:

$$r_p = r_{p+1}. \tag{3.12}$$

Prova: Ver [34]. □

Para a determinação da forma canônica de Jordan de uma dada matriz A deve, inicialmente determinar os autovalores λ desta matriz (não é necessário encontrar as suas multiplicidades, caso existam). Então, para cada autovalor λ distinto, proceda de acordo com o seguinte algoritmo.

Algoritmo 3.1 (Método de Segundas Diferenças)

Passo 1 Determine $B = A - \lambda I_m$, onde I_m é uma matriz identidade de ordem m.

Passo 2 Monte uma tabela para cada autovalor com os campos μ (índice), r_μ (posto de B^μ), σ_1 (primeira diferença) e σ_2 (segunda diferença), conforme Tabela

[1] Por convenção, $B^0 = I_m$ para qualquer matriz B de ordem m x m. Assim, $r_0 = posto(B^0) = posto(I_m) = m$.

3.1, a seguir. A coluna r_μ representa o posto da matriz B elevada ao índice μ, para o autovalor λ em questão.

Tabela 3.1: Tabela para determinação do algoritmo de Rubin.

μ	B^μ	r_μ	σ_1	σ_2

Passo 3 Insira o valor $\mu = 0$ na coluna adequada da tabela.

Passo 4 Calcule B^0 e o resultado da expressão $r_0 = \rho(B^0)$ e preencha a tabela nos campos especificados.

Passo 5 Insira o valor $\mu = 1$ na linha seguinte.

Passo 6 Calcule B^1 e $r_1 = \rho(B^1)$ e insira nas respectivas linhas da tabela.

Passo 7 Subtraia os dois últimos valores da coluna do $r_\mu = \rho(B^\mu)$ e insira na coluna σ_1. Observe que este termo deve ficar entre duas linhas das colunas anteriores.

Passo 8 Insira o valor $\mu = 2$ na linha seguinte.

Passo 9 Calcule o resultado da expressão $r_\mu = \rho(B^\mu)$ e insira na respectiva linha da tabela.

Passo 10 Como no Passo 7, subtraia os dois últimos valores da coluna do $r_\mu = \rho(B^\mu)$ e insira na coluna σ_1.

Passo 11 Subtraia os dois últimos campos da coluna σ_1 e insira na coluna σ_2.

Passo 12 Se os dois últimos resultados da coluna r_μ forem iguais, como estabelecido em (3.12), pare, caso contrário incremente μ em 1 (um) e volte para o Passo 9. □

Em resumo, preencha os campos da Tabela 3.1 conforme segue:

Campo	Descrição
μ	representando uma numeração seqüencial (índice) começando em zero;
B^{μ}	matriz resultante da relação $(A - \lambda I_m)^{\mu}$, onde λ é autovalor em questão e I_m é uma matriz identidade de ordem m;
r_{μ}	posto de B^{μ};
σ_1	resultado da subtração dos últimos dois resultados da coluna do posto;
σ_2	resultado da subtração dos últimos dois resultados da coluna σ_1.

Tabela 3.2: Tabela do algoritmo de Rubin preenchida.

μ	B^{μ}	r_{μ}	σ_1	σ_2
0	B^0	r_0	$\sigma_{11} = r_0 - r_1$	$\sigma_{11} - \sigma_{12}$
1	B^1	r_1	$\sigma_{12} = r_1 - r_2$	
2	B^2	r_2	$\sigma_{13} = r_2 - r_3$	$\sigma_{12} - \sigma_{13}$
3	B^3	r_3	...	...
...	...	...		...

Depois de realizado o Algoritmo 3.1, o resultado da segunda diferença σ_2 encontrado na primeira linha representa o número de blocos de Jordan de primeira ordem, o número da segunda diferença presente na segunda linha da mesma coluna representa o número de blocos de Jordan de segunda ordem, o número da terceira linha é o número de blocos de terceira ordem e assim sucessivamente. As estruturas para os blocos dos outros autovalores de $A(s)$ podem ser obtidas de forma similar. □

3.1.3 Base polinomial mínima

Sejam $T(s)$ e $F(s) = [f_1(s)\ f_2(s)\ ...\ f_n(s)]$, matrizes polinomiais tal que $T(s)F(s)=0$, com graus coluna $\mu_1 \leq \mu_2 \leq \ldots \leq \mu_n$. Então, as seguintes afirmações são equivalentes:

1. As colunas de $F(s)$ formam uma base polinomial mínima para o espaço nulo à direita de $T(s)$.
2. $F(s)$ é reduzida por coluna e irredutível [seu posto é cheio para todos os valores (finitos) de s].
3. $F(s)$ tem ordem mínima, isto é, $\sum_{i=1}^{n} \mu_i$ é mínimo.

Prova: Ver [4], páginas 458-459. □

Um algoritmo robusto para obtenção de uma base polinomial mínima para o espaço nulo de uma matriz foi proposto por Basílio e Moreira [29].

3.2. Resultados Preliminares

Seja $A(s) \in \mathbb{R}^{m \times m}[s]$ uma matriz polinomial quadrada, cuja decomposição na forma de Smith se deseja obter, isto é,

$$A(s) = L(s)S(s)R(s), \tag{3.13}$$

onde $L(s), R(s) \in \mathbb{R}^{m \times m}[s]$ são matrizes unimodulares e $S(s)$ está na forma de Smith, equação (2.45).

Conforme será visto, a obtenção de $S(s)$ será baseada na determinação da forma canônica de Jordan associada a A_C, uma matriz de estado de uma realização de ordem mínima de $G(s)=I\,A^{-1}(s)$. Como I e $A(s)$ são coprimas [27], então se $A(s)$ for reduzida por coluna e tiver todas as colunas com graus maiores que 0 (zero), então, uma realização que pode ser obtida de forma imediata é a chamada forma controlador, equação (2.39).

Assim, quando $A(s)$ não for reduzida por coluna é necessário encontrar uma matriz $A_R(s)$, reduzida por coluna, que seja equivalente a $A(s)$. Isso pode ser feito utilizando-se o algoritmo proposto em [27], pode-se obter uma matriz unimodular $V_1(s)$ tal que

$$A_R(s) = A(s)V_1(s) \tag{3.14}$$

seja reduzida por coluna. Note que, como $V_1(s)$ é unimodular, então $A_R(s)$ e $A(s)$ têm a mesma forma de Smith. Assim, sem perda de generalidade, a partir desse ponto será suposto que $A(s)$ seja uma matriz reduzida por coluna.

Será agora considerada a situação em que $A(s)$ tem colunas com graus nulos. Suponha, então,

$$J = \{j_1, j_2, ..., j_k\}, \tag{3.15}$$

$k<m$, represente o conjunto formado pelos índices das colunas de $A(s)$ que têm grau 0 (zero) e forme a seguinte matriz diagonal

$$D(s) = diag\left\{d_1(s), \quad d_2(s), \quad \cdots, \quad d_m(s)\right\}, \tag{3.16}$$

onde

$$d_i(s) = \begin{cases} 1 & , \quad i \notin J \\ (s+s_0) & , \quad i \in J \end{cases}$$

para, $i=1, 2, ..., m$, sendo $s_0 \in \mathbb{N}$ escolhido de tal forma que $A(s_0)$ seja não-singular[1], isto é, s_0 não seja um zero de $A(s)$. Essa restrição se deve ao fato de que, conforme será visto a seguir, como a determinação da forma de Smith será baseada no cálculo dos autovalores de A_C, deve-se então, evitar a introdução de autovalores com multiplicidade maior que 1 (um). Caso a matriz de entrada $A(s)$ não tenha colunas com grau zero, $D(s)$ será igual à matriz identidade de ordem m. Portanto,

---

[1] Foi escolhido o conjunto dos números naturais $\mathbb{N}$ por simplicidade, contudo o mesmo vale para o conjunto dos números reais $\mathbb{R}$.

$$\overline{A}(s) = A(s)D(s) \tag{3.17}$$

tem todas as colunas com graus maiores ou iguais a 1 (um).

A multiplicação de $A(s)$ pela matriz $D(s)$ não é empecilho para o cálculo da forma de Smith de $A(s)$, pois embora todos os cálculos sejam feitos usando-se $\overline{A}(s)$ e, portanto, a forma de Smith encontrada seja a de $\overline{A}(s)$, tem-se que $S(s)$, a forma de Smith de $A(s)$, pode ser obtida a partir de $\overline{S}(s)$, a forma de Smith de $\overline{A}(s)$, de maneira imediata. Este é um resultado inédito, conforme será mostrado na proposição a seguir em sua prova, desenvolvidas especialmente para este trabalho.

Proposição 3.1 Seja $A(s) \in \mathbb{R}^{m \times m}[s]$ e suponha que a forma de Smith de *A(s)* seja *S(s)*, isto é,

$$A(s) \underset{\sim}{s} S(s), \tag{3.18}$$

onde $\underset{\sim}{s}$ denota equivalência de Smith. Suponha ainda que $s_0 \in \mathbb{N}$ não seja um zero de $A(s)$, ou seja, $A(s_0)$ é não-singular. Defina

$$\overline{A}(s) = A(s)D(s), \tag{3.19}$$

sendo que $D(s)$ é definida de acordo com a equação (3.16). Se $\overline{S}(s)$ denota a forma de Smith da matriz $\overline{A}(s)$, então $\overline{S}(s)$ e $S(s)$ estão relacionados da seguinte forma:

$$S(s) = \overline{S}(s)\left[diag\{\underbrace{1,1,...,1}_{(m-k)vezes}, \underbrace{s+s_0, s+s_0, ..., s+s_0}_{k \;\; vezes}\} \right]^{-1}. \tag{3.20}$$

Prova: Suponha que $A(s)$, tenha a seguinte forma de Smith:

$$S(s) = \left[\begin{array}{cccc|ccc} \sigma_1(s) & & & & & & \\ & \sigma_2(s) & & & & & \\ & & \ddots & & & & \\ & & & \sigma_r(s) & & & \\ \hline & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \\ \multicolumn{4}{c|}{\underbrace{\hphantom{\sigma_1(s)\ \sigma_2(s)\ \ddots\ \sigma_r(s)}}_{r}} & \multicolumn{3}{c}{\underbrace{\hphantom{0\ \ddots\ 0}}_{m-r}} \end{array}\right], \tag{3.21}$$

onde $\sigma_i(s)$, $i = 1, 2,..., r$, são polinômios mônicos, únicos, obedecendo a propriedade de divisão

$$\sigma_i(s) \mid \sigma_{i+1}(s), \quad i = 1, \ldots, r-1. \tag{3.22}$$

Note que, definindo $\Delta_i(s)$ como o máximo divisor comum (MDC) de todos os menores ixi de $A(s)$, tem-se que

$$\sigma_i(s) = \frac{\Delta_i(s)}{\Delta_{i-1}(s)}, \qquad \Delta_0(s) = 1. \tag{3.23}$$

Suponha, agora, sem perda de generalidade, que as colunas de $A(s)$ tenham sido rearranjadas de forma que as k primeira colunas tenham grau igual a zero. Isto não altera a forma de Smith tendo em vista que isto é feito com operações elementares. Portanto, $D(s)$ terá a seguinte forma

$$D(s) = diag\left[\underbrace{s - s_0 \quad s - s_0 \quad \cdots \quad s - s_0}_{k \ \ vezes} \quad 1 \quad \cdots \quad 1\right]. \tag{3.24}$$

Seja $\overline{A}(s) = A(s)D(s)$ e seja $\overline{S}(s)$ a forma de Smith de $\overline{A}(s)$. Escreva $\overline{A}(s)$, $A(s)$ e $D(s)$ da seguinte forma:

$$\overline{A}(s) = \begin{bmatrix} \overline{a}_{11} & \overline{a}_{12} & \cdots & \overline{a}_{1m} \\ \overline{a}_{21} & \overline{a}_{22} & \cdots & \overline{a}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{a}_{m1} & \overline{a}_{m2} & \cdots & \overline{a}_{mm} \end{bmatrix}, \tag{3.25}$$

$$A(s)=\begin{bmatrix}\underline{a}_1^T\\ \underline{a}_2^T\\ \vdots\\ \underline{a}_m^T\end{bmatrix} \tag{3.26}$$

e

$$D(s)=\begin{bmatrix}(s-s_0)\underline{e}_1 & (s-s_0)\underline{e}_2 & \cdots & (s-s_0)\underline{e}_k & \underline{e}_{k+1} & \cdots & \underline{e}_m\end{bmatrix}. \tag{3.27}$$

Utilizando a equação (3.8) tem-se que os máximos divisores comuns de ordem p de $\overline{A}(s)$ ($\overline{\Delta}_p(s)$) e os MDC dos menores de ordem p de $A(s)$ ($\Delta_p(s)$) possuem a seguinte relação

(i) $\left|\overline{A}\right|=\left|A\right|\left|D\right|\Rightarrow\left|\overline{A}_k\right|=\left|A\right|(s-s_0)\Rightarrow\overline{\Delta}_m(s)=\Delta_m(s)(s-s_0)^k$.

(ii) $$\overline{A}_{1,2,\ldots,m-1}^{1,2,\ldots,m-1}=\left|\begin{bmatrix}\underline{a}_1^T\\ \underline{a}_2^T\\ \vdots\\ \underline{a}_{m-1}^T\end{bmatrix}\begin{bmatrix}(s-s_0)\underline{e}_1 & (s-s_0)\underline{e}_2 & \cdots & (s-s_0)\underline{e}_k & \underline{e}_{k+1} & \cdots & \underline{e}_{m-1}\end{bmatrix}\right|$$

$$=\left|\begin{bmatrix}\underline{a}_1^T\\ \underline{a}_2^T\\ \vdots\\ \underline{a}_{m-1}^T\end{bmatrix}\begin{bmatrix}\underline{e}_1 & \underline{e}_2 & \cdots & \underline{e}_k & \underline{e}_{k+1} & \cdots & \underline{e}_{m-1}\end{bmatrix}\right|(s-s_0)^k=A_{1,2,\ldots,m-1}^{1,2,\ldots,m-1}\cdot(s-s_0)^k, \tag{3.28}$$

$$\overline{A}_{1,2,\ldots,m-1}^{2,3,\ldots,m}=\left|\begin{bmatrix}\underline{a}_1^T\\ \underline{a}_2^T\\ \vdots\\ \underline{a}_{m-1}^T\end{bmatrix}\begin{bmatrix}(s-s_0)\underline{e}_2 & (s-s_0)\underline{e}_3 & \cdots & (s-s_0)\underline{e}_k & \underline{e}_{k+1} & \cdots & \underline{e}_m\end{bmatrix}\right|=A_{1,2,\ldots,m-1}^{2,3,\ldots,m-1}\cdot(s-s_0)^{k-1}.$$

(3.29)

Assim, é possível verificar que:

$$\overline{\Delta}_m(s)=\Delta_m(s)(s-s_0)^k. \tag{3.30}$$

(iii) Note ainda que, para $i<m$-k,

$$\overline{A}^{k+1,\ldots,m}_{1,2,\ldots,m-k} = A^{k+1,\ldots,m}_{1,2,\ldots,m-k} \tag{3.31}$$

e, portanto, $\overline{\Delta}_i = \Delta_i$. Conseqüentemente

$$\overline{\Delta}_{m-l}(s) = \begin{cases} \Delta_{m-l}(s)\cdot(s-s_0)^{k-l}, & l = 0,\ldots,k-1 \\ \Delta_{m-l}(s), & l = k,\ldots,m \end{cases}, \tag{3.32}$$

o que permite concluir:

$$\overline{\sigma}_m(s) = \frac{\overline{\Delta}_m(s)}{\overline{\Delta}_{m-1}(s)} = \frac{\Delta_m(s)\cdot(s+s_0)^k}{\Delta_{m-1}(s)\cdot(s+s_0)^{k-1}} = \sigma_m(s)\cdot(s+s_0)$$

$$\overline{\sigma}_{m-k+1} = \frac{\overline{\Delta}_{m-k+1}(s)}{\overline{\Delta}_{m-k}(s)} = \frac{\Delta_{m-k+1}(s)\cdot(s+s_0)}{\Delta_{m-k}(s)} = \sigma_{m-k+1}(s)\cdot(s+s_0)$$

Com isso,

$$\overline{S}(s) = \left[\begin{array}{cccc|ccc} \overline{\sigma}_1 & & & & & & \\ & \overline{\sigma}_2 & & & & & \\ & & \ddots & & & & \\ & & & \overline{\sigma}_r & & & \\ \hline & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \end{array}\right]$$

$$= \left[\begin{array}{cccc|ccc} \sigma_1 & & & & & & \\ & \sigma_2 & & & & & \\ & & \ddots & & & & \\ & & & \sigma_r & & & \\ \hline & & & & 0 & & \\ & & & & & \ddots & \\ & & & & & & 0 \end{array}\right] \cdot \left[\begin{array}{ccc|cccc} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ \hline & & & s+s_0 & & & \\ & & & & s+s_0 & & \\ & & & & & \ddots & \\ & & & & & & s+s_0 \end{array}\right]$$

$$= S(s)\left[\begin{array}{ccc|cccc} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ \hline & & & s+s_0 & & & \\ & & & & s+s_0 & & \\ & & & & & \ddots & \\ & & & & & & s+s_0 \end{array}\right]. \tag{3.33}$$

O que conclui a prova. □

3.3. Obtenção da Decomposição em Forma de Smith de uma Matriz Polinomial

À luz de todas as informações apresentadas nesta seção, a forma de Smith de matrizes polinomiais pode ser encontrada através da determinação de seus autovalores e respectivos comprimentos de cadeias de Jordan.

Teorema 3.2 Seja $A(s) \in \mathbb{R}^{m \times m}[s]$ e suponha que o posto normal de $A(s)$ seja igual a m e sejam ainda $\{\lambda_1, \lambda_2, ..., \lambda_n\}$ ($n \leq n_C$) os autovalores distintos de $A_C \in \mathbb{R}^{n_C \times n_C}$, com suas respectivas multiplicidades m_1, m_2, ..., m_n (sendo $m_1 + m_2 + ... + m_n = n_C$), onde A_C é resultante da determinação de uma realização na forma controlador da matriz de transferência $G(s)=IA^{-1}(s)$. Suponha que n_{Ci} denote o número de cadeias de Jordan associadas ao autovalor λ_i, $i=1, 2, ..., n$, e seja v_{ij}, com $j=1, 2, ..., n_{Ci}$, o comprimento das cadeias de Jordan tal que v_{ij} atenda às seguintes condições:

$$v_{ij} \leq v_{i(j+1)} \tag{3.34}$$

e

$$\sum_{j=1}^{n_{Ci}} v_{ij} = m_i . \tag{3.35}$$

Então a forma de Smith de $A(s)$ de ordem m como

$$S(s) = diag\{\sigma_1(s), \sigma_2(s), \ldots, \sigma_m(s)\} \tag{3.36}$$

onde,

$$\sigma_{m-k}(s) = \prod_{i=1}^{n} (s - \lambda_i)^{v_{i(n_{Ci}-k)}} \tag{3.37}$$

com $k=1, 2, ..., m–1$. Considere sempre que se $(n_{Ci} - k) \leq 0$ então $v_{i(n_{Ci}-k)} = 0$.

Prova: Ver [24].

Determinada a matriz $S(s)$ que representa a forma de Smith, o passo seguinte é determinar o par de matrizes unimodulares $\{U(s), V(s)\}$ que completam a equação

$$U(s)A(s)V(s) = S(s). \tag{3.38}$$

A partir de (3.38) pode-se escrever $S(s)V^{-1}(s) - U(s)A(s) = 0$. Chamando $R(s)=V^{-1}(s)$, tem-se que $S(s)R(s) - U(s)A(s) = 0$, ou melhor ainda,

$$S(s)R(s)I - IU(s)A(s) = 0. \tag{3.39}$$

Antes de dar prosseguimento, o seguinte resultado é necessário:

Lema 3.2 Seja $A(s)$, $B(s)$ e $T(s) \in \mathbb{R}^{m\times m}(s)$ e defina que $\underline{\Theta}(s) = A(s)T(s)B(s)$. Se $\underline{\theta}_i(s)$ e $\underline{t}_i(s)$, $i = 1, 2, ..., m$ denotam as i-ésimas colunas de $\underline{\Theta}(s)$ e $T(s)$, respectivamente, então

$$\begin{bmatrix} \underline{\theta}_1(s) \\ \underline{\theta}_2(s) \\ \vdots \\ \underline{\theta}_m(s) \end{bmatrix} = B^T(s) \otimes A(s) \begin{bmatrix} \underline{t}_1(s) \\ \underline{t}_2(s) \\ \vdots \\ \underline{t}_m(s) \end{bmatrix}, \tag{3.40}$$

onde $\otimes$ denota o produto de Kronecker.

Prova: Ver [35], página 123. □

A partir do enunciado no Lema 3.2, pode-se re-escrever a equação (3.39) da seguinte forma:

$$I \otimes S(s)\underline{r}(s) - A^T(s) \otimes I\underline{u}(s) = \underline{0}. \tag{3.41}$$

Escrevendo (3.41) na forma matricial chega-se a

$$\begin{bmatrix} \underbrace{I \otimes S(s)}_{M_1(s)} & \underbrace{-A^T(s) \otimes I}_{M_2(s)} \end{bmatrix} \begin{bmatrix} \underline{r}(s) \\ \underline{u}(s) \end{bmatrix} = \underline{0}, \tag{3.42}$$

onde

$$\underline{r} = \begin{bmatrix} \underline{r}_1 \\ \underline{r}_2 \\ \vdots \\ \underline{r}_m \end{bmatrix} \text{ e } \underline{u} = \begin{bmatrix} \underline{u}_1 \\ \underline{u}_2 \\ \vdots \\ \underline{u}_m \end{bmatrix} \tag{3.43}$$

com $\underline{r}_i$ ($\underline{u}_i$) denotando a i-ésima coluna de $R(s)$ [$U(s)$].

Definindo

$$M(s) = \begin{bmatrix} I \otimes S(s) & -A^T(s) \otimes I \end{bmatrix}. \tag{3.44}$$

então, o problema de se encontrar matrizes unimodulares $R(s)$ e $U(s)$ se reduz a encontrar uma base polinomial mínima para o espaço nulo à direita de $M(s)$.

Note que, como $M(s) \in \mathbb{R}^{m^2 \times 2m^2}[s]$, então o espaço nulo à direita de $M(s)$ tem dimensão pelo menos igual a m^2. Assim, existe uma matriz $X(s) \in \mathbb{R}^{2m^2 \times p}[s]$, onde $p=m^2-\rho$, ρ denota o posto normal de $M(s)$ tal que $M(s)X(s) = 0$. Seja então:

$$X(s) = \begin{bmatrix} \underline{r}_1(s) & \underline{r}_2(s) & \underline{r}_3(s) & \cdots & \underline{r}_p(s) \\ \underline{u}_1(s) & \underline{u}_2(s) & \underline{u}_3(s) & \cdots & \underline{u}_p(s) \end{bmatrix}, \tag{3.45}$$

onde os vetores $\underline{r}_i(s), \underline{u}_i(s) \in \mathbb{R}^{m^2 \times 1}[s]$, $i=1, ..., m$, formam as matrizes $R_i(s)$ e $U_i(s)$. Assim, é possível obter:

$$\begin{bmatrix} R_L(s) \\ U_L(s) \end{bmatrix} = \begin{bmatrix} R_1(s) & R_2(s) & R_3(s) & \cdots & R_p(s) \\ U_1(s) & U_2(s) & U_3(s) & \cdots & U_p(s) \end{bmatrix}. \tag{3.46}$$

Finalmente, a determinação de $R(s)$ e $U(s)$ será finita, definindo-se o vetor polinomial

$$\underline{\theta}(s) = \begin{bmatrix} \theta_1(s) & \theta_2(s) & \theta_3(s) & \ldots & \theta_p(s) \end{bmatrix}^T \tag{3.47}$$

tal que

$$\begin{bmatrix} R_L(s) \\ U_L(s) \end{bmatrix} \Theta(s) = \begin{bmatrix} R(s) \\ U(s) \end{bmatrix}, \tag{3.48}$$

onde

$$\Theta(s)=\begin{bmatrix}\theta_1(s)I_m & \theta_2(s)I_m & \theta_3(s)I_m & \dots & \theta_p(s)I_m\end{bmatrix}^T \tag{3.49}$$

e as matrizes $R(s)$ e $U(s)$ são unimodulares. As matrizes $R_L(s)$ e $U_L(s)$ têm dimensão m x mp e a matriz $\Theta(s)$ tem dimensão mp x m, onde mp é o produto de m por p. Assim, precisa-se determinar

$$\begin{cases} R(s) = R_L(s)\Theta(s) \\ U(s) = U_L(s)\Theta(s) \end{cases}. \tag{3.50}$$

Uma maneira de conseguir o resultado desejado é igualando os determinantes das matrizes dos dois lados de cada igualdade, para cada uma delas. Para encontrar o vetor polinomial (3.47), solução para o problema, através do determinante de $R(s)$ e $U(s)$, faz-se uso da fórmula de Binet-Cauchy, equação (3.4), que diz que o determinante destas matrizes é igual à soma dos menores de ordem m de $R_L(s)$, ou $U_L(s)$, e $\Theta(s)$. Assim, escreve-se

$$\begin{vmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mm} \end{vmatrix} = \sum_{k_1,k_2,\dots,k_m \in \kappa_m} \left\{ \begin{vmatrix} r_{L_{1k_1}} & \cdots & r_{L_{1k_1m}} \\ \vdots & \ddots & \vdots \\ r_{L_{mk_1}} & \cdots & r_{L_{k_mm}} \end{vmatrix} \begin{vmatrix} \theta_{k_11} & \cdots & \theta_{k_1m} \\ \vdots & \ddots & \vdots \\ \theta_{k_m1} & \cdots & \theta_{k_mm} \end{vmatrix} \right\} \tag{3.51}$$

e

$$\begin{vmatrix} u_{11} & \cdots & u_{1m} \\ \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{mm} \end{vmatrix} = \sum_{k_1,k_2,\dots,k_m \in \kappa_m} \left\{ \begin{vmatrix} u_{L_{1k_1}} & \cdots & u_{L_{1k_1m}} \\ \vdots & \ddots & \vdots \\ u_{L_{mk_1}} & \cdots & u_{L_{k_mm}} \end{vmatrix} \begin{vmatrix} \theta_{k_11} & \cdots & \theta_{k_1m} \\ \vdots & \ddots & \vdots \\ \theta_{k_m1} & \cdots & \theta_{k_mm} \end{vmatrix} \right\}. \tag{3.52}$$

onde κ_m denota o conjunto dos menores formado por todas as

$$c = \begin{pmatrix} mp \\ m \end{pmatrix} = \frac{(mp)!}{(mp-m)!m!} \tag{3.53}$$

combinações possíveis das colunas de $R_L(s)$, ou $U_L(s)$, e linhas de $\Theta(s)$. Portanto, sejam os conjuntos dos menores de ordem m das colunas de $R_L(s)$ e $U_L(s)$ representados por $\underline{m}_R(s)$ e $\underline{m}_U(s) \in \mathbb{R}^{1\times c}[s]$ e os menores de ordem m das linhas de $\Theta(s)$ representados por

$\underline{m}_{\Theta}(s) \in \mathbb{R}^{c\times 1}[s]$, assim, os determinantes de $R(s)$ e de $U(s)$ devem ser iguais aos produtos de $\underline{m}_R(s)$ por $\underline{m}_{\Theta}(s)$ e de $\underline{m}_U(s)$ pelo mesmo $\underline{m}_{\Theta}(s)$, respectivamente. Ou seja, sejam

$$\begin{aligned} \underline{m}_R(s) &= \begin{bmatrix} m_{R1}(s) & m_{R2}(s) & \cdots & m_{Rc}(s) \end{bmatrix} \\ \underline{m}_U(s) &= \begin{bmatrix} m_{U1}(s) & m_{U2}(s) & \cdots & m_{Uc}(s) \end{bmatrix}, \\ \underline{m}_{\theta}(s) &= \begin{bmatrix} m_{\theta 1}(s) & m_{\theta 2}(s) & \cdots & m_{\theta c}(s) \end{bmatrix}^T \end{aligned} \tag{3.54}$$

então,

$$\left|U(s)\right| = \sum_{i=1}^{c} m_{Ui}(s) m_{\theta i}(s) \tag{3.55}$$

e

$$\left|R(s)\right| = \sum_{i=1}^{c} m_{Ri}(s) m_{\theta i}(s). \tag{3.56}$$

Feito isso, encontram-se então duas expressões em função de $\underline{\theta}(s)$, uma para cada matriz unimodular. Basta agora lembrar que para que $R(s)$ e $U(s)$ sejam matrizes unimodulares e tais que $U(s)A(s)V(s)=S(s)$, os seus determinantes devem ser não-nulos e independentes de s. Além disso, como

$$\left|A(s)\right| = \left|A_{hc}(s)\right| \phi(s), \tag{3.57}$$

onde $\phi(s)$ é um polinômio em s, então, deve-se ter:

$$\left|U(s)\right| = 1 \text{ e } \left|R(s)\right| = \left|A_{hc}\right|. \tag{3.58}$$

Assim, na equação (3.55) o determinante de $U(s)$ deve ser feito igual a 1 (um) e na equação (3.56), o determinante de $R(s)$ deve ser feito igual ao determinante da matriz A_{hc}, onde

$$A(s) = A_{hc}\, diag\left\{ s^{\delta_1}, s^{\delta_2}, \ldots, s^{\delta_m} \right\} + A_{lc}\Psi(s), \tag{3.59}$$

como na equação (2.32). Portanto, realizando este passo, chega-se a um sistema cujo número de incógnitas é determinado pelo maior grau entre $\underline{m}_R(s)$ e $\underline{m}_U(s)$, escolhendo-se o grau de $\underline{m}_{\Theta}(s)$ de forma que o número de equações seja igual ao número de incógnitas. Encontrado o valor de $\underline{\theta}(s)$ deve-se substituí-lo na equação (3.49) encontrando-se $\Theta(s)$. Finalmente, as matrizes unimodulares $R(s)$ e $U(s)$ são encontradas através da equação (3.48).

Analisando estes últimos resultados, observa-se que, conforme se aumenta a dimensão das matrizes, o número de combinações e, conseqüentemente, de equações do sistema a serem consideradas aumenta rapidamente. Contudo, por ser a matriz $\Theta(s)$ formada por matrizes diagonais, muitos dos menores considerados em (3.51) e (3.52) são nulos, reduzindo bastante o número de combinações úteis para os cálculos. Por meio de análises experimentais, ou seja, gerando diversos exemplos com matrizes de diferentes ordens, pode-se notar que o número de produtos de menores neste problema é da ordem de m^{2m} para solução de cada uma das matrizes $R(s)$ e $U(s)$, o que ainda é muito grande. Considerando, por exemplo, uma matriz de dimensão 5 x 5, o número de menores de $\Theta(s)$ não nulos possíveis é 9.765.625 (nove milhões, setecentos e sessenta e cinco mil, seiscentos e vinte e cinco).

Encontradas as matrizes $R(s)$ e $U(s)$, basta realizar a operação $V(s)=V_1(s)R^{-1}(s)$, onde $V_1(s)$ é a matriz unimodular que multiplicada por $A(s)$ criou uma matriz reduzida por coluna, e estão determinadas as matrizes unimodulares $U(s)$ e $V(s)$. Assim, chega-se a todos os componentes da igualdade

$$S(s) = U(s)A(s)V(s). \tag{3.60}$$

3.4. Algoritmo para Determinação da Forma de Smith de Matrizes Polinomiais via Descrição em Espaço de estado

Esta seção apresenta um algoritmo que resume e esquematiza o procedimento apresentado na seção anterior. Ele está dividido em duas partes: a primeira é a determinação da forma de Smith $S(s)$ da matriz de entrada $A(s)$; na segunda são apresentados os procedimentos para determinação das matrizes unimodulares, ou de transformação, $U(s)$ e $V(s)$.

Algoritmo 3.2 Algoritmo para determinação da Forma de Smith de Matrizes Polinomiais.

Parte 1: Determinação da forma de Smith $S(s)$ de uma dada matriz de entrada $A(s) \in \mathbb{R}^{m \times m}(s)$.

Passo 1 Se $A(s)$ não for uma matriz reduzida por coluna, obtenha uma matriz unimodular $V_1(s)$ (Algoritmo 2 de [27]) tal que

$$A_R(s) = A(s)V_1(s)$$

seja reduzida por coluna. Caso contrário, $V_1(s)=I_m$.

Passo 2 Determine os graus das colunas de $A_R(s)$.

Passo 3 Identifique as colunas com grau *0* (zero), se existirem, e seja

$$J = \{j_1, j_2, ..., j_k\},$$

$(k < m)$ o conjunto que representa essas colunas.

Passo 4 Se não existirem colunas com grau zero, defina $D(s) = I_m$. Caso contrário, forme a matriz diagonal

$$D(s) = diag\left\{d_1(s), \quad d_2(s), \quad \cdots, \quad d_m(s)\right\},$$

onde, para $i=1, 2, ..., m$

$$d_i(s) = \begin{cases} 1 & , \quad i \notin J \\ (s+s_0) & , \quad i \in J \end{cases}$$

sendo $s_0 \in \mathbb{N}$ escolhido de tal forma que $A(s_0)$ seja não-singular, isto é, s_0 não seja zero de $A(s)$.

Passo 5 Obtenha,

$$\overline{A}_R(s) = A_R(s)D(s).$$

Passo 6 Obtenha uma realização na forma controlador, equação (2.39), para a matriz de transferência

$$G(s) = I\overline{A}_R^{-1}(s) = \left[\begin{array}{c|c} A_C & B_C \\ \hline C_C & 0 \end{array}\right],$$

onde $A_C \in \mathbb{R}^{n_C \times n_C}$.

Passo 7 Encontre os autovalores de A_C. Sejam $\{\lambda_1, \lambda_2, ..., \lambda_n\}$ $(n \leq n_C)$ os autovalores distintos de A_C com suas respectivas multiplicidades m_1, m_2, ..., m_n (sendo $m_1+m_2+...+m_n=n_C$).

Passo 8 Utilize o Algoritmo 3.1 para encontrar a quantidade e o comprimento dos blocos de Jordan dos autovalores de A_C.

Passo 9 Suponha que n_{Ci} denote o número de cadeias de Jordan associadas ao autovalor λ_i, $i=1, 2, ..., n$, e seja v_{ij} com $j=1, 2, ..., n_{Ci}$ o comprimento das cadeias de Jordan tal que v_{ij} atenda as seguintes condições:

$$v_{ij} \leq v_{i(j+1)} \text{ e } \sum_{j=1}^{n_{Ci}} v_{ij} = m_i .$$

Defina a forma de Smith de $\overline{A}_R(s)$ como

$$\overline{S}(s) = diag\{\sigma_1(s), \sigma_2(s), \ldots, \sigma_m(s)\}$$

onde,

$$\sigma_{m-k}(s) = \prod_{i=1}^{n} (s-\lambda_i)^{v_{i(n_{Ci}-k)}} ,$$

com $k=1, 2, ..., m-1$. Considere sempre que se $(n_{Ci} - k) \leq 0$, $v_{i(n_{Ci}-k)} = 0$.

Passo 10 Determine então a forma de Smith $S(s)$ de $A(s)$ a partir de

$$S(s)=\overline{S}(s)\left[\begin{array}{ccc|cccc} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ \hline & & & s+s_0 & & & \\ & & & & s+s_0 & & \\ & & & & & \ddots & \\ & & & & & & s+s_0 \end{array}\right]^{-1} .$$

(with the block of $s+s_0$ entries spanning k rows, marked $\}k$)

Parte 2: Determinação das matrizes unimodulares *U*(*s*) e *V*(*s*) tal que *S*(*s*) = *U*(*s*)*A*(*s*)*V*(*s*).

Passo 11 Faça o produto de Kronecker das matrizes *I*, matriz identidade de ordem *m*, e *S*(*s*), forma de Smith de *A*(*s*), encontrada na Parte 1.

Passo 12 Faça o produto de Kronecker entre o negativo da transposta da matriz *A*(*s*) e a matriz identidade *I* de ordem *m*.

Passo 13 Forme a matriz $M(s)=\begin{bmatrix} I\otimes S(s) & -A^T(s)\otimes I\end{bmatrix}$ com os produtos dos dois passos anteriores e encontre uma base polinomial mínima *X*(*s*) para o espaço nulo à direita de *M*(*s*), isto é, *M*(*s*)*X*(*s*)=*0*. A matriz *X*(*s*) tem a seguinte forma:

$$X(s)=\begin{bmatrix} \underline{r}_1(s) & \underline{r}_2(s) & \underline{r}_3(s) & \cdots & \underline{r}_p(s) \\ \underline{u}_1(s) & \underline{u}_2(s) & \underline{u}_3(s) & \cdots & \underline{u}_p(s) \end{bmatrix}$$

onde, $\underline{r}_i(s),\underline{u}_i(s)\in\mathbb{R}^{m^2\times 1}[s]$ e $p=m^2-\rho$, ρ denota o posto normal de *M*(*s*).

Passo 14 Escreva a partir de *X*(*s*), determine a seguinte igualdade

$$\begin{bmatrix} R_L(s) \\ U_L(s) \end{bmatrix}=\begin{bmatrix} R_1(s) & R_2(s) & R_3(s) & \cdots & R_p(s) \\ U_1(s) & U_2(s) & U_3(s) & \cdots & U_p(s) \end{bmatrix},$$

onde $\underline{r}_i(s)$ e $\underline{u}_i(s)$ do passo anterior correspondem, respectivamente, a $R_i(s)$ e $U_i(s)$ com suas colunas empilhadas.

Passo 15 Forme os vetores polinomiais $\underline{m}_R(s)\in\mathbb{R}^{1\times c}[s]$, $\underline{m}_U(s)\in\mathbb{R}^{1\times c}[s]$ e $\underline{m}_\Theta(s)\in\mathbb{R}^{c\times 1}[s]$, onde

$$c = \binom{mp}{m} = \frac{(mp)!}{(mp-m)!m!}$$

tais que,

$$\underline{m}_R(s) = \begin{bmatrix} m_{R1}(s) & m_{R2}(s) & \cdots & m_{Rc}(s) \end{bmatrix}$$
$$\underline{m}_U(s) = \begin{bmatrix} m_{U1}(s) & m_{U2}(s) & \cdots & m_{Uc}(s) \end{bmatrix}$$
$$\underline{m}_\theta(s) = \begin{bmatrix} m_{\theta 1}(s) & m_{\theta 2}(s) & \cdots & m_{\theta c}(s) \end{bmatrix}^T$$

sejam, respectivamente, o conjuntos de todos os menores possíveis de ordem m das colunas de $R_L(s)$ e $U_L(s)$ e das linhas de $\Theta(s)$, onde

$$\Theta(s) = \begin{bmatrix} \theta_1(s)I_m & \theta_2(s)I_m & \theta_3(s)I_m & \dots & \theta_m(s)I_m \end{bmatrix}^T.$$

Passo 16 Multiplique $\underline{m}_U(s)$ por $\underline{m}_\Theta(s)$ e $\underline{m}_R(s)$ também por $\underline{m}_\Theta(s)$. Assim, os determinantes são dados pelos somatórios

$$|U(s)| = \sum_{i=1}^{c} m_{Ui}(s) m_{\theta i}(s)$$

e

$$|R(s)| = \sum_{i=1}^{c} m_{Ri}(s) m_{\theta i}(s).$$

Onde, $|U(s)| = 1$ e $|R(s)| = \left|A_{R_{hc}}\right|$, sendo $A_{R_{hc}}$ a matriz formada pelos coeficientes de $\overline{A}_R(s)$ correspondentes aos maiores graus de suas colunas. Determine, a partir daí, o vetor $\underline{\theta}(s) = \begin{bmatrix} \theta_1(s) & \theta_2(s) & \theta_3(s) & \dots & \theta_p(s) \end{bmatrix}^T$.

Passo 17 Encontre as matrizes unimodulares $R(s)$ e $U(s)$ tal que

$$\begin{bmatrix} R_L(s) \\ U_L(s) \end{bmatrix} \Theta(s) = \begin{bmatrix} R(s) \\ U(s) \end{bmatrix}.$$

onde $R(s)$ e $U(s)$ são resultantes da combinação linear dos $R_i(s)$ e $U_i(s)$, $i=1,2,\dots,p$, que formam $R_L(s)$ e $U_L(s)$.

Passo 18 Faça $V(s) = V_1(s)R^{-1}(s)$. □

3.5. Exemplos de Aplicação do Algoritmo

Nesta seção serão apresentados alguns exemplos de aplicação do Algoritmo 3.2 da seção anterior.

Exemplo 3.1: Considere a matriz polinomial $A(s) \in \mathbb{R}^{2\times 2}(s)$

$$A(s) = \begin{bmatrix} s+1 & 3s \\ 2 & 2s+1 \end{bmatrix}.$$

Seguindo o que está estabelecido no algoritmo, vê-se que as colunas de $A(s)$ têm graus-colunas iguais a 1 (um), não sendo, portanto, necessário multiplicá-la por $D(s)$, isto é, $D(s)=I_2$, desta forma,

$$\overline{A}(s) = A(s).$$

A seguir é fácil verificar que a matriz $A(s)$ é reduzida por coluna. Assim,

$$A_R(s) = A(s) \text{ e } V_1(s) = I_2.$$

Conseqüentemente, usando o Algoritmo 1 de [27], é possível obter uma realização em espaço de estado na forma controlador para $G(s) = I\overline{A}_R^{-1}(s)$, dada por:

$$A_C = \begin{bmatrix} 2 & 1,5 \\ -1 & -0,5 \end{bmatrix},\ B_C = \begin{bmatrix} 1 & -1,5 \\ 0 & 0,5 \end{bmatrix} \text{ e } C_C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Feito isso, o passo seguinte é a determinação dos autovalores de A_C, que são dados por: $\lambda_1 = 1$ e $\lambda_2 = 0,5$. Como pode-se notar são dois autovalores distintos, assim, de acordo com o Teorema 3.2, a forma de Smith de $A(s)$ pode ser escrita como:

$$S(s) = \begin{bmatrix} 1 & 0 \\ 0 & (s-1)(s-0,5) \end{bmatrix}.$$

Passa-se agora para o cálculo das matrizes unimodulares $U(s)$ e $V(s)$, tais que a igualdade $S(s)=U(s)A(s)V(s)$ seja estabelecida. Primeiro, calcula-se os produtos de Kronecker

$$I \otimes S(s) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & (s-1)(s-0,5) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (s-1)(s-0,5) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & (s-1)(s-0,5) \end{bmatrix}$$

e

$$-A^T(s) \otimes I = -\begin{bmatrix} s+1 & 3s \\ 2 & 2s+1 \end{bmatrix}^T \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -s-1 & 0 & -2 & 0 \\ 0 & -s-1 & 0 & -2 \\ -3s & 0 & -2s-1 & 0 \\ 0 & -3s & 0 & -2s-1 \end{bmatrix}.$$

Assim, pode-se escrever

$$M(s) = \begin{bmatrix} I \otimes S(s) & -A^T(s) \otimes I \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & -s-1 & 0 & -2 & 0 \\ 0 & (s-1)(s-0,5) & 0 & 0 & 0 & -s-1 & 0 & -2 \\ 0 & 0 & 1 & 0 & -3s & 0 & -2s-1 & 0 \\ 0 & 0 & 0 & (s-1)(s-0,5) & 0 & -3s & 0 & -2s-1 \end{bmatrix}.$$

Feito isso, deve-se encontrar, agora, uma base polinomial mínima $X(s)$ para o espaço nulo à direita de $M(s)$, isto é, $M(s)X(s)=0$. Desta forma, a partir de [29] chega-se a:

$$X(s) = \begin{bmatrix} -0,33s-0,07 & -0,095s+0,57 & 0,043-0,15s & -0,19s+0,37 \\ -0,17 & -0,28 & 0,072 & 0,38 \\ -0,74s+0,13 & 0,38s+0,33 & -0,26s+0,096 & -0,0084s+0,28 \\ -0,27 & -0,099 & 0,62 & 0,029 \\ -0,33 & -0,095 & -0,15 & -0,19 \\ -0,17s+0,18 & -0,28s-0,04 & 0,072s-0,58 & 0,38s+0,16 \\ 0,13 & 0,33 & 0,096 & 0,28 \\ 0,12s-0,13 & 0,37s-0,049 & 0,2s+0,31 & -0,56s+0,015 \end{bmatrix}.$$

A matriz $X(s)$ é composta pelos seguintes vetores:

$$X(s) = \begin{bmatrix} \underline{r}_1(s) & \underline{r}_2(s) & \underline{r}_3(s) & \underline{r}_4(s) \\ \underline{u}_1(s) & \underline{u}_2(s) & \underline{u}_3(s) & \underline{u}_4(s) \end{bmatrix}$$

onde as matrizes $\underline{r}_i(s)$ e $\underline{u}_i(s)$, $i=1,...,4$, são as matrizes $R_i(s)$ e $U_i(s)$, respectivamente, com suas colunas empilhadas. Assim, escreve-se

$$R_L(s) = \begin{bmatrix} R_1(s) & R_2(s) & R_3(s) & R_4(s) \end{bmatrix}$$

e

$$U_L(s) = \begin{bmatrix} U_1(s) & U_2(s) & U_3(s) & U_4(s) \end{bmatrix}$$

onde,

$$R_1(s) = \begin{bmatrix} -0,33s - 0,07 & -0,74s + 0,13 \\ -0,17 & -0,27 \end{bmatrix},$$

$$R_2(s) = \begin{bmatrix} -0,095s + 0,57 & 0,38s + 0,33 \\ -0,28 & -0,099 \end{bmatrix},$$

$$R_3(s) = \begin{bmatrix} -0,15s + 0,043 & -0,26s + 0,096 \\ 0,072 & 0,62 \end{bmatrix},$$

$$R_4(s) = \begin{bmatrix} -0,19s + 0,37 & -0,0084s + 0,28 \\ 0,38 & 0,029 \end{bmatrix}$$

e

$$U_1(s) = \begin{bmatrix} -0,33 & 0,13 \\ -0,17s + 0,18 & 0,12s - 0,13 \end{bmatrix},$$

$$U_2(s) = \begin{bmatrix} -0,095 & 0,33 \\ -0,28s - 0,04 & 0,37s - 0,049 \end{bmatrix},$$

$$U_3(s) = \begin{bmatrix} -0,15 & 0,096 \\ 0,072s - 0,58 & 0,2s + 0,31 \end{bmatrix},$$

$$U_4(s) = \begin{bmatrix} -0,19 & 0,28 \\ 0,38s + 0,16 & -0,56s + 0,015 \end{bmatrix}.$$

Encontradas as matrizes $R_L(s)$ e $U_L(s)$, o Passo 15 do algoritmo estabelece que se deva agora encontrar todos os menores de ordem m=2 de $R_L(s)$, $U_L(s)$ e também de $\Theta(s) = \begin{bmatrix} \theta_1(s)I_2 & \theta_2(s)I_2 & \theta_3(s)I_2 & \theta_4(s)I_2 \end{bmatrix}^T$, agrupando-os nos vetores $\underline{m}_R(s)$, $\underline{m}_U(s)$ e

$\underline{m}_{\Theta}(s)$, respectivamente. Obtidos os vetores $\underline{m}_R(s)$ e $\underline{m}_U(s)$, observa-se que o grau de $\underline{m}_R(s)$ é igual a 1 (um) e o grau de $\underline{m}_U(s)$ é também igual a 1 (um), portanto, como $\underline{\theta}(s) \in \mathbb{R}^{4\times 1}[s]$ então, é necessário que o grau de $\underline{m}_{\Theta}(s)$ seja igual a 1 (um) para que se tenha um sistema de 4 (quatro) equações e 4 (quatro) incógnitas. Portanto,

$$\underline{\theta}(s) = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}^T$$

Os vetores $\underline{m}_R(s)$, $\underline{m}_U(s)$ e $\underline{m}_{\Theta}(s)$ estão mostrados abaixo.

$$\underline{m}_R(s) = \begin{bmatrix} -0{,}037s+0{,}041 \\ 0{,}076s+0{,}12 \\ 0{,}097s+0{,}063 \\ -0{,}05s+0{,}0023 \\ -0{,}25s-0{,}027 \\ -0{,}16s+0{,}037 \\ -0{,}011s+0{,}046 \\ 0{,}18s+0{,}11 \\ 0{,}17s+0{,}075 \\ -0{,}093s+0{,}021 \\ -0{,}52s+0{,}11 \\ -0{,}33s+0{,}15 \\ -0{,}024s+0{,}079 \\ 0{,}11s+0{,}036 \\ -0{,}048+0{,}053 \\ -0{,}013s+0{,}38 \\ -0{,}089s+0{,}32 \\ -0{,}0051s+0{,}095 \\ 0{,}012s+0{,}028 \\ 0{,}21s+0{,}21 \\ 0{,}13s+0{,}16 \\ 0{,}01s+0{,}037 \\ -0{,}074s+0{,}019 \\ -0{,}044s-0{,}01 \\ -0{,}038s-0{,}019 \\ 0{,}019s-0{,}19 \\ -0{,}0024s-0{,}17 \\ -0{,}0024s-0{,}096 \end{bmatrix}^T \quad \underline{m}_U(s) = \begin{bmatrix} -0{,}019s+0{,}021 \\ 0{,}076s+0{,}03 \\ -0{,}066s+0{,}043 \\ -0{,}05s+0{,}22 \\ -0{,}051s-0{,}12 \\ -0{,}16s-0{,}019 \\ 0{,}23s-0{,}056 \\ -0{,}025s-0{,}018 \\ 0{,}0076s+0{,}038 \\ 0{,}028s-0{,}097 \\ 0{,}015s+0{,}054 \\ 0{,}074s-0{,}004 \\ -0{,}011s+0{,}039 \\ 0{,}057s+0{,}018 \\ -0{,}048s+0{,}049 \\ 0{,}0077s-0{,}025 \\ -0{,}089s-0{,}023 \\ 0{,}13s+0{,}0098 \\ 0{,}079s-0{,}2 \\ 0{,}031s+0{,}11 \\ 0{,}2s+0{,}044 \\ -0{,}29s+0{,}019 \\ -0{,}037s+0{,}0097 \\ -0{,}044s-0{,}13 \\ 0{,}064s+0{,}16 \\ 0{,}075s+0{,}074 \\ -0{,}11s-0{,}085 \\ -0{,}012s-0{,}048 \end{bmatrix}^T \quad \underline{m}_{\theta}(s) = \begin{bmatrix} \theta_1^2 \\ 0 \\ \theta_1\theta_2 \\ 0 \\ \theta_1\theta_3 \\ 0 \\ \theta_1\theta_4 \\ -\theta_1\theta_2 \\ 0 \\ -\theta_1\theta_3 \\ 0 \\ -\theta_1\theta_4 \\ 0 \\ \theta_2^2 \\ 0 \\ \theta_2\theta_3 \\ 0 \\ \theta_2\theta_4 \\ -\theta_2\theta_3 \\ 0 \\ -\theta_2\theta_4 \\ 0 \\ \theta_3^2 \\ 0 \\ \theta_3\theta_4 \\ -\theta_3\theta_4 \\ 0 \\ \theta_4^2 \end{bmatrix}$$

A seguir, deve-se determinar $\underline{\theta}(s) = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}^T$ tal que

$$\begin{bmatrix} R_L(s) \\ U_L(s) \end{bmatrix} \Theta(s) = \begin{bmatrix} R(s) \\ U(s) \end{bmatrix},$$

para isso, encontram-se os somatórios do Passo 16. Considerando que para $R(s)$ e $U(s)$ unimodulares e tais que $U(s)A(s)V(s)=S(s)$ deve-se ter $|U(s)| = 1$ e $|R(s)| = |A_{R_{hc}}| = 2$. Assim, $\underline{\theta}(s)$ deve ser a solução do seguinte sistema:

$$\begin{cases} -0,019\theta_1^2 - 0,041\theta_1\theta_2 - 0,079\theta_1\theta_3 + 0,156\theta_1\theta_4 + 0,057\theta_2^2 - 0,0713\theta_2\theta_3 - 0,07\theta_2\theta_4 - 0,037\theta_3^2 - 0,011\theta_3\theta_4 - 0,012\theta_4^2 = 0 \\ 0,021\theta_1^2 + 0,061\theta_1\theta_2 - 0,023\theta_1\theta_3 - 0,052\theta_1\theta_4 + 0,018\theta_2^2 + 0,175\theta_2\theta_3 - 0,0342\theta_2\theta_4 + 0,0097\theta_3^2 + 0,086\theta_3\theta_4 - 0,048\theta_4^2 = 2 \\ -0,037\theta_1^2 - 0,083\theta_1\theta_2 - 0,157\theta_1\theta_3 + 0,319\theta_1\theta_4 + 0,11\theta_2^2 - 0,025\theta_2\theta_3 - 0,1351\theta_2\theta_4 - 0,074\theta_3^2 - 0,057\theta_3\theta_4 - 0,0024\theta_4^2 = 0 \\ 0,041\theta_1^2 - 0,047\theta_1\theta_2 - 0,048\theta_1\theta_3 - 0,104\theta_1\theta_4 + 0,036\theta_2^2 + 0,352\theta_2\theta_3 - 0,065\theta_2\theta_4 + 0,019\theta_3^2 - 0,209\theta_3\theta_4 - 0,096\theta_4^2 = 1 \end{cases}.$$

Resolvendo o sistema anterior com uso da função *fsolve* do Matlab, obtém-se:

$$\underline{\theta}(s) = [-0,1401 \;\; 1,1765 \;\; 1,0440 \;\; 0,3087]^T.$$

Com isso, as matrizes $U(s)$ e $R(s)$ são (Passo 17):

$$U(s) = \begin{bmatrix} -0,46 & 0,77 \\ -0,68s - 0,6 & 1,2s + 0,13 \end{bmatrix} \text{ e } R(s) = \begin{bmatrix} -0,046s + 1,1 & 0,17s + 0,77 \\ -0,68 & 0,25 \end{bmatrix}.$$

Basta agora determinar, como no Passo 18, $V(s) = V_1(s)R^{-1}(s) = I_2R^{-1}(s)$, assim

$$V(s) = \begin{bmatrix} 0,32 & -0,21s - 0,96 \\ 0,85 & -0,57s + 1,4 \end{bmatrix}.$$

Como pode ser demonstrado, as matrizes $U(s)$ e $V(s)$ são de fato unimodulares, pois seus determinantes são, respectivamente, $|U(s)| = 0,4$ e $|V(s)| = 1,25$, ainda, a igualdade $S(s) = U(s)A(s)V(s)$ pode ser comprovada. □

Nos próximos exemplos serão citadas funções computacionais para auxiliar os cálculos das diversas etapas do procedimento. Algumas das funções citadas foram retiradas de [31], ou aperfeiçoadas a partir daquelas lá encontradas. Outras foram desenvolvidas especificamente para este trabalho. Para todas as funções citadas a matriz polinomial $A(s)$, ou outra que dela derive, deverá ser escrita na forma de matriz de coeficientes.

Exemplo 3.2: Considere, agora, a matriz polinomial $A(s) \in \mathbb{R}^{2\times 2}(s)$

$$A(s) = \begin{bmatrix} 2 & s+1 \\ 0 & -2s+1 \end{bmatrix}.$$

Primeiro, Passo 1, deve-se encontrar uma matriz reduzida por coluna equivalente a $A(s)$. Usa-se a função *reducol.m* (ver anexo A.16 de [31]) para encontrar esta matriz reduzida por coluna. Contudo, como $A(s)$ já é reduzida por coluna, tem-se que $A_R(s)=A(s)$ e $V_1(s)=I_2$.

O Passo 2 é a determinação do grau das colunas de $A(s)$. Para isso, usa-se a função *coldeg.m* (ver anexo A.9 de [31]). Com ela verifica-se que os graus das colunas de $A(s)$ são iguais a 0 (zero) e 1 (um). Portanto, o grau da primeira coluna da matriz é igual a zero e $J=\{1\}$. Com isso, como estabelece o Passo 4 do algoritmo, é necessário multiplicar $A(s)$ por

$$D(s) = \begin{bmatrix} s+s_0 & 0 \\ 0 & 1 \end{bmatrix}.$$

Para o exemplo, escolhe-se $s_0=1$. Pode-se comprovar que $|A(1)| = -2$, portanto $A(s)$ não é singular para $s_0=1$, este não é zero de $A(s)$. Assim,

$$\overline{A}(s) = A(s)\begin{bmatrix} s+1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2s+2 & s+1 \\ 0 & -2s+1 \end{bmatrix}.$$

Sendo $\overline{A}(s)$ reduzida por coluna, deve-se encontrar, agora (Passo 6), uma realização na forma controlador para a matriz de transferência $G(s) = I\overline{A}^{-1}(s)$ associada à matriz polinomial $\overline{A}(s)$. Para tanto, usa-se a função *nmcop2ss.m* (ver anexo A.2 deste trabalho), que por sua vez faz uso da função *dhcdlcsplit.m* (ver anexo A.3). As matrizes de estados que compõem a forma controlador relacionadas à realização são:

$$A_C = \begin{bmatrix} -1 & -0,75 \\ 0 & 0,5 \end{bmatrix},\ B_C = \begin{bmatrix} 0,5 & 0,25 \\ 0 & -0,5 \end{bmatrix} \text{ e } C_C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

O passo seguinte é encontrar os autovalores de A_C que são λ={-1, 0,5}, dois autovalores distintos com multiplicidade 1 (um), cada. Assim, a forma de Smith de $\overline{A}(s)$ é dada por:

$$\overline{S}(s) = \begin{bmatrix} 1 & 0 \\ 0 & (s-0,5)(s+1) \end{bmatrix} = S(s)\begin{bmatrix} 1 & 0 \\ 0 & (s+1) \end{bmatrix}.$$

Por conseqüência, a matriz que representa a forma de Smith da matriz de entrada $A(s)$ é

$$S(s) = \begin{bmatrix} 1 & 0 \\ 0 & s-0,5 \end{bmatrix}.$$

No cálculo das matrizes unimodulares $U(s)$ e $V(s)$, tais que $S(s)=U(s)A(s)V(s)$, encontra-se $M(s) = \left[I \otimes S(s) \quad -A^T(s) \otimes I \right]$. Os produtos de Kronecker $I \otimes S(s)$ (Passo 11) e $-A^T(s) \otimes I$ (Passo 12), são determinados com a função *matkron.m* (ver seção A.4 dos anexos). Assim,

$$M(s) = \begin{bmatrix} 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & s-0,5 & 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 1 & 0 & -s-1 & 0 & 2s-1 & 0 \\ 0 & 0 & 0 & s-0,5 & 0 & -s-1 & 0 & 2s-1 \end{bmatrix}.$$

O próximo passo é achar uma matriz polinomial $X(s)$ tal que $M(s)X(s)=0$ onde $X(s)$ seja uma base polinomial mínima para o espaço nulo à direita de $M(s)$. Neste passo (Passo 13), a função utilizada é a *polybases.m* (ver anexo A.18 de [31]). Assim,

$$X(s) = \begin{bmatrix} -0,66 & 0,31 & 0,26 & 0,19 \\ 0 & 0 & 0,075 & -0,41 \\ -0,50 & 0,23 & 0,37s+0,011 & -0,58s+0,44 \\ -0,38 & -0,81 & -0,62s-0,45 & -0,26s-0,15 \\ -0,33 & 0,15 & 0,13 & 0,097 \\ 0 & 0 & 0,038s-0,19 & -0,2s+0,1 \\ -0,17 & 0,077 & 0,12 & 0,34 \\ 0,19 & 0,40 & 0,33s+0,24 & 0,027s-0,027 \end{bmatrix}.$$

Como pede o Passo 14, escreve-se

$$R_L(s) = \begin{bmatrix} R_1(s) & R_2(s) & R_3(s) & R_4(s) \end{bmatrix}$$

e

$$U_L(s) = \begin{bmatrix} U_1(s) & U_2(s) & U_3(s) & U_4(s) \end{bmatrix}$$

que, por simplicidade, serão omitidas, porém são de fácil determinação a partir de $X(s)$, como já foi demonstrado. Com isso, determinam-se as matrizes unimodulares $U(s)$ e $R(s)$ diretamente a partir da primeira coluna de $X(s)$ que forma uma base polinomial mínima, suprimindo o Passo 15. Assim,

$$U(s) = \begin{bmatrix} -0,33 & -0,17 \\ 0 & 0,19 \end{bmatrix} \text{ e } R(s) = \begin{bmatrix} -0,66 & -0,50 \\ 0 & -0,38 \end{bmatrix}.$$

Finalmente, fazendo

$$V(s) = V_1(s)R^{-1}(s) = I_2R^{-1}(s) = \begin{bmatrix} -1,51 & 1,98 \\ 0 & -2,63 \end{bmatrix} \qquad \square$$

está completa a igualdade $S(s)=U(s)A(s)V(s)$.

O próximo exemplo apresentará um caso onde a matriz $A(s)$ de dimensão 3x3 possui autovalores repetidos.

Exemplo 3.3: Forma de Smith da matriz polinomial $A(s) \in \mathbb{R}^{3\times3}(s)$

$$A(s) = \begin{bmatrix} 1 & -2s-2 & 0 \\ -s & 2s^2+s-1 & s^2+2s+1 \\ 0,25s & s+1 & -0,5s^3-s^2-0,5s \end{bmatrix}.$$

O Passo 1 é determinar a forma reduzida por coluna da matriz $A(s)$. Assim,

$$A_R(s) = \begin{bmatrix} -0,49s & 1 & 0,76 \\ -0,73s-0,24 & -s & 0,76 \\ 0,12 & 0,25s & -0,19s-0,38 \end{bmatrix}$$

e a matriz unimodular $V_1(s)$ é

$$V_1(s) = \begin{bmatrix} 0,49s^3 - 0,49s^2 - 0,49s & 1 & 0,76s^2 \\ -0,24s^2 & 0 & 0,38s - 0,38 \\ -0,24s - 0,24 & 0 & 0,38 \end{bmatrix}.$$

O próximo passo é encontrar os graus das colunas de $A(s)$, são eles 1, 1 e 1, respectivamente. Observa-se que todos são maiores que zero. Neste caso, de acordo com o Passo 4, a matriz $D(s)=I_3$, conseqüentemente,

$$\overline{A}_R(s) = A_R(s)I_3 = A_R(s).$$

A partir desta, determina-se as matrizes A_C, B_C e C_C (Passo 6) da realização na forma controlador associada à matriz de transferência $G(s) = I\overline{A}_R^{-1}(s)$, sendo dado por:

$$A_C = \begin{bmatrix} 0 & 2,0616 & 1,5584 \\ -0,2425 & -1,5000 & -0,3780 \\ -0,3208 & -0,6614 & -1,5000 \end{bmatrix},\ B_C = \begin{bmatrix} -2,0616 & 0 & 0 \\ 1,5000 & -1,0000 & 0 \\ 0,6614 & -1,3229 & -5,2915 \end{bmatrix} \text{ e}$$

$$C_C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Para este caso, encontram-se autovalores repetidos para A_C, $\lambda=\{-1\}$ com multiplicidade 3 (três). A determinação da forma de Smith passa pela determinação do comprimento das cadeias de Jordan (Passo 8) que é encontrado através do Algoritmo 3.1, resumido na função *rubin.m* (ver anexo A.5). Vê-se, então, que o autovalor apresenta um bloco de ordem 1 (um) e um bloco de ordem 2 (dois). Assim, a forma de Smith de $A(s)$ é:

$$S(s) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & s+1 & 0 \\ 0 & 0 & (s+1)^2 \end{bmatrix}.$$

Para o cálculo das matrizes unimodulares, segue-se o mesmo procedimento apresentado nos exemplos anteriores, suprimido aqui por conveniência. Após estes passos, as matrizes unimodulares $U(s)$ e $V(s)$ tais que $S(s) = U(s)A(s)V(s)$ são:

$$U(s)=\begin{bmatrix} -0,09 & 0 & 0 \\ -0,02s-0,099 & 0,079 & 0 \\ -0,2s-0,24 & 0,24s+0,32 & 0,16 \end{bmatrix}$$

e

$$V(s)=\begin{bmatrix} -5,3s^2+17s+11 & -26s^2-78s-52 & 13s^2+25s+13 \\ -2,7s+11 & -13s-26 & 6,3s+6,3 \\ -2,7 & -13 & 6,3 \end{bmatrix}. \qquad \square$$

Todos os procedimentos do Algoritmo 3.2 até agora apresentados nos exemplos foram reunidos na forma de uma função, chamada *smithbl.m* . Ela está listada no anexo A.7 e será usada no cálculo dos exemplos que seguem. Portanto, serão apresentados apenas os resultados finais provenientes da função, ou seja, as matrizes unimodulares e a matriz polinomial diagonalisada representado a forma de Smith da matriz de entrada. Quando necessário serão feitos ponderações sobre o exemplo.

Exemplo 3.4: Para este exemplo considere a matriz polinomial $A(s)\in\mathbb{R}^{3\times 3}(s)$

$$A(s)=\begin{bmatrix} 2 & -4s & 0 \\ -2s & 4s^2-2s & 2s^2+4s \\ 0,5s & 2s & -s^3-2s^2 \end{bmatrix}.$$

Para a determinação da forma de Smith de $A(s)$ aplica-se a função *smithbl*. Antes, porém, $A(s)$ deve ser escrita na forma de matriz de coeficientes:

$$A(s)=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}s^3+\begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 2 \\ 0 & 0 & -2 \end{bmatrix}s^2+\begin{bmatrix} 0 & -4 & 0 \\ -2 & -2 & 4 \\ 0,5 & 2 & 0 \end{bmatrix}s+\begin{bmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Abaixo são apresentados os comandos e os resultados.

```
>> A=[0 0 0 0 0 0 0 -4 0 2 0 0;0 0 0 0 4 2 -2 -2 4 0 0 0;0 0 -1 0 0 -2
0,5 2 0 0 0 0;];

>> [S,U,V] = smithbl(A);
```

A função apresenta como resultado a matriz polinomial $S(s)$, forma de Smith de $A(s)$:

$$S(s)=\begin{bmatrix}1 & 0 & 0\\ 0 & s & 0\\ 0 & 0 & s(s+2)\end{bmatrix}$$

e as matrizes unimodulares:

$$U(s)=\begin{bmatrix}0,032 & 0 & 0\\ 0,12s & 0,15 & 0\\ 0,093s & -0,04-0,08 & -0,37\end{bmatrix}$$

e

$$V(s)=\begin{bmatrix}5,6s^2+5,6s+16 & 2,6s^2-1,4s & -2,7s^2-5,4s\\ 2,8s+2,8 & 1,3s-0,72 & -1,3s-2,7\\ 2,8 & 1,3 & -1,3\end{bmatrix}.$$

A igualdade $S(s) = U(s)A(s)V(s)$ pode ser verificada. □

Exemplo 3.5: Neste exemplo será considerada uma matriz polinomial $A(s)\in\mathbb{R}^{3\times 3}(s)$

$$A(s)=\begin{bmatrix}2 & -4s-4 & 0\\ -s & 2s^2+s-1 & s^2-2s-3\\ 0,25s & s+1 & -0,5s^3+s^2+1,5s\end{bmatrix}.$$

Na sua forma de matriz de coeficientes, $A(s)$ é escrita como:

$$A(s)=\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & -0,5\end{bmatrix}s^3+\begin{bmatrix}0 & 0 & 0\\ 0 & 2 & 1\\ 0 & 0 & 1\end{bmatrix}s^2+\begin{bmatrix}0 & -4 & 0\\ -1 & 1 & -2\\ 0,25 & 1 & 1,5\end{bmatrix}s+\begin{bmatrix}2 & -4 & 0\\ 0 & -1 & -3\\ 0 & 1 & 0\end{bmatrix}.$$

Aplicando a função *smithbl*

```
>> A=[0 0 0 0 0 0 0 -4 0 2 -4 0; 0 0 0 0 2 1 -1 1 -2 0 -1 -3; 0 0 -0.5 0 0 1 0.25 1 1 0 1 0];

>> [S,U,V] = smithbl(A);
```

encontra-se a forma de Smith da matriz de entrada $A(s)$

$$S(s) = \begin{bmatrix} 1 & & \\ & s+1 & \\ & & (s+1)(s-3) \end{bmatrix}$$

e as matrizes unimodulares

$$U(s) = \begin{bmatrix} -0,13 & 0 & 0 \\ 0,024s - 0,037 & 0,12 & 0 \\ 0,14s + 0,15 & 0,097s + 0,12 & 0,17 \end{bmatrix}$$

e

$$V(s) = \begin{bmatrix} -4,2s^2 + 13s + 13 & -1,4s - 13s - 12 & 12s^2 - 24s - 36 \\ -2,1s + 8,6 & -0,69s - 6 & 6s - 18 \\ -2,1 & -0,69 & 6 \end{bmatrix}$$

que verificam a igualdade $S(s) = U(s)A(s)V(s)$. □

O próximo exemplo é o mesmo do capítulo anterior onde, através do método clássico não foi possível obter a forma de Smith da matriz de entrada.

Exemplo 3.6: Considere uma matriz polinomial $A(s) \in \mathbb{R}^{3\times 3}[s]$, onde

$$A(s) = \begin{bmatrix} 9,5\cdot 10^{-7}s^2 + 0,44s + 0,41 & 0,49s^2 + 0,92s + 0,35 & 0,46s^2 + 8,1\cdot 10^{-9}s + 0,14 \\ 2,3\cdot 10^{-7}s^2 + 0,62s + 0,89 & 0,89s^2 + 0,74s + 0,81 & 0,019s^2 + 1,9\cdot 10^{-8}s + 0,2 \\ 6,1\cdot 10^{-7}s^2 + 0,79s + 0,058 & 0,76s^2 + 0,18s + 0,0099 & 0,82s^2 + 1,8\cdot 10^{-8}s + 0,2 \end{bmatrix}.$$

Escrevendo $A(s)$ na forma de matriz de coeficientes e aplicando a função *smithbl* obtém-se de forma simples e rápida a matriz diagonal

$$S(s) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s^6 - 6,4\times 10^4 s^5 - 4,1\times 10^5 s^4 - 1,3\times 10^6 s^3 + 7,6\times 10^4 s^2 - 2,2\times 10^5 s + 3,8\times 10^3 \end{bmatrix}$$

que é a forma de Smith da matriz de entrada. As matrizes unimodulares que completam a igualdade $S(s)=U(s)A(s)V(s)$ são:

$$U(s) = \begin{bmatrix} 0 & -1 & 0 \\ 2\times 10^{-6}s^2 - 8,7s + 9,9 & -4,5\times 10^{-6}s^4 + 0,28s^3 + 1,1s^2 + 2,2s - 4,2 & 4,7s - 5,9 \\ -3,1\times 10^{-7}s^3 + 1,3s^2 - 0,13s + 0,29 & 6,7\times 10^{-7}s^5 - 0,042s^2 - 0,21s^3 - 0,56s^2 - 0,041s - 0,13 & -0,71s^2 + 0,15s - 0,074 \end{bmatrix}$$

e

$$V(s)=\begin{bmatrix} 0{,}89s-0{,}56 & -0{,}28s+0{,}14 & 2\times10^{-6}s+1{,}2\times10^{6} \\ -2{,}3\times10^{7}s-0{,}62 & 6{,}3\times10^{-8}s+0{,}23 & 1{,}5\times10^{6}s-1{,}5\times10^{5} \\ 0 & 5{,}2\times10^{-7}s-1{,}5 & -7{,}3\times10^{6}s+9{,}4\times10^{5} \end{bmatrix}.$$

Assim, procurou-se mostrar através deste exemplo que o método proposto funciona onde o método clássico falhou.

4. Conclusões

"Para atingirmos a verdade, é necessário aplicarmos a razão, pensar, refletir. E para isso, devemos começar a pensar, não a partir do que já sabemos, mas do que não sabemos."
Sócrates

O desenvolvimento de equações matemáticas é um importante passo na obtenção de modelos analíticos no estudo e projeto de processos ou sistemas físicos. É dentro desse contexto, mais especificamente, na modelagem matemática de sistemas multivariáveis que se insere o trabalho aqui desenvolvido tendo, para tanto, sido necessário abordar vários conceitos relacionados ao estudo de sistemas multivariáveis, destacando-se: realizações de ordem mínima, máximo divisor comum de matrizes, descrição por frações de matrizes, matrizes reduzidas por coluna e realização na forma controlador. Também, como caráter ilustrativo e didático, foi apresentado um algoritmo para a obtenção da forma de Smith de matrizes polinomiais baseado em operações elementares de linhas e colunas.

Contudo, o ponto principal deste trabalho foi a proposição de um novo algoritmo para a obtenção da forma de Smith de matrizes polinomiais que evita a necessidade de se realizar operações elementares de linhas e colunas, evitando assim as dificuldades numéricas desse tipo de algoritmo. O algoritmo aqui proposto tem como base a determinação dos autovalores da matriz de estados de uma realização controlável (na forma controlador) associada à matriz polinomial em análise e ao comprimento das cadeias de Jordan associadas a esses autovalores. Para ilustrar sua aplicação foram apresentados alguns exemplos. Mais uma vez, este método mostrou-se de fácil aplicação e apresenta-se como mais uma alternativa na determinação da forma de Smith de matrizes polinomiais.

Embora a forma de Smith seja de fácil determinação através do método proposto, o cálculo das matrizes unimodulares $U(s)$ e $V(s)$ para sistemas de ordem igual ou superior a 5 (cinco) se torna impraticável, pois requer o cálculo de m^{2m} (onde m representa a ordem da matriz de entrada) determinantes para cada uma das duas matrizes unimodulares durante a determinação dos menores, necessários na solução do Passo 15 do Algoritmo 3.2. Esse ponto requer uma investigação mais aprofundada no futuro.

Um outro ponto a ser considerado é a extensão do presente algoritmo para matrizes pxm. Esse ponto é mais simples, tendo em vista que se utilizando o algoritmo para redução por coluna a uma matriz retangular em que $m>p$, pode-se reduzir a matriz $A(s) \in \mathbb{R}^{p\times m}[s]$ a uma matriz $\overline{A}(s) \in \mathbb{R}^{p\times m}[s]$, onde

$$\overline{A}(s) = \begin{bmatrix} A_0(s) & 0 \end{bmatrix}$$

sendo $A_0(s) \in \mathbb{R}^{p\times p}[s]$. Assim,

$$\overline{A}(s) = A(s)R(s)$$

onde $R(s)$ é unimodular.

Por fim, uma outra sugestão de trabalho futuro é um estudo comparativo dos métodos para obtenção da forma de Smith de matrizes polinomiais disponíveis na literatura a fim de avaliar seu desempenho. Por hora apresenta-se um quadro onde se mostra uma comparação quanto ao tempo de resposta dos algoritmos para obtenção da forma de Smith citados nesta dissertação. Os métodos usados são o método clássico considerando operações elementares de linhas e colunas, representado pelo algoritmo descrito no Capítulo 2, um algoritmo para cálculo da forma de Smith através de matrizes simbólicas da *toolbox* Polyx, citada na Introdução. Além deles, apresenta-se também o desempenho do método apresentado neste trabalho. Sobre este, fazem-se duas análises, na primeira somente é considerada a forma de Smith $S(s)$ e na segunda, além de $S(s)$, também são calculadas as matrizes unimodulares. Foram analisadas as respostas aos exemplos do Capítulo 3. Para cálculo do tempo de resposta foi utilizada a função *etime.m* do Matlab, que calcula a diferença de tempo entre um instante inicial e outro final.

Portanto, considere a seguinte tabela onde a primeira coluna (Exemplo) traz o número do exemplo e a segunda (Dimensão) traz a dimensão da matriz de entrada. A terceira coluna (Polyx) representa o tempo de resposta usando a função *smith.m* da *toolbox* Polyx, a quarta (Clássico) mostra o desempenho usando o método clássico (função *smithkuc.m*), a quinta (Smithbl-1) traz os valores dos tempos de resposta para o cálculo da forma de Smith da matriz polinomial de entrada usando o algoritmo apresentado neste trabalho, porém é calculada apenas a matriz $S(s)$ e na sexta e última coluna (Smithbl-2) é considerado todo o Algoritmo 3.2, onde também são calculadas as matrizes polinomiais. Os resultados apresentados a seguir foram extraídos diretamente do Matblab e os tempos estão medidos em milisegundos (ms).

Tabela 4.1: Quadro Comparativo.

Exemplo	**Dimensão**	**Polyx (ms)**	**Clássico (ms)**	**Smithbl-1 (ms)**	**Smithbl-2 (ms)**
3.1	2x2	31,0	16,0	0	328,0
3.2	2x2	31,0	16,0	0	0
3.3	3x3	47,0	16,0	0	47,0
3.4	3x3	47,0	15,0	16,0	47,0
3.5	3x3	47,0	31,0	0	47,0

Observado a Tabela 4.1 nota-se que o cálculo da forma de Smith apenas através do método proposto é extremamente rápido para os casos apresentados. Porém, só através de uma análise mais aprofundada, pode-se tirar outras conclusões a respeito do desempenho de tais métodos.

Apêndice – Funções Matlab

Este apêndice traz as funções desenvolvidas no *Matlab* para execução dos algoritmos apresentados nos capítulos anteriores. Estas funções serão incorporadas a, já referida, *toolbox* desenvolvida chamada *Polymat*.

A.1. Smithkuc

A primeira função é *smithkuc.m*, desenvolvida com base no Algoritmo 2.1 (seção 2.7) extraído de [14].

```
%SMITHKUC Smith form of a polynomial matrix
%   S=SMITHKUC(A,m) it supplies the Smith Form of a coefficient polynomial matrix.
%
%   To use it, the user enters with the coefficient matrix and the number of columns (in case
%   the matrix is not square) and the function returns one coefficient matrix of the polynomials
%   with the Smith form S, where S = U*A*V, for any given polynomial matrix A.
%
%   [S,U,V,r]=SMITHKUC(A,m) it finds, besides the matrix S, the unimodular matrices and
%   the rank r of the matrix S.

%     Author(s):  R.O.Leite
%     $Revision: 0.0 $  $Date: 12-Nov-2005 $

function [S,U,V,r]=smithkuc(A,m);
tol=1e-5;
[p la]=size(A);
if nargin==1;m=p;end
U=eye(p);V=eye(m);
flag=1;p15=1;k=1;S=A;
while flag
        aij=[];
        for i=0:(la/m-1);aij=[aij S(k:p,(k:m)+i*m)];end
        [imin,jmin]=posmin(aij,m-k+1);
        imin=imin+k-1;jmin=jmin+k-1;
```

```
if aij==0
        r=k-1;flag=0;
else
        eyel=eye(p);eyev=eye(m);
        if imin ~= k;
                eyel([imin k],:)=eyel([k imin],:);
                S=convmat(eyel,S,m);
                U=convmat(eyel,U);
        end
        if jmin ~= k;
                eyev(:,[k jmin])=eyev(:,[jmin k]);
                S=convmat(S,eyev);
                V=convmat(V,eyev);
        end
        [contlin,contcol]=contpolzero(S,k,m);
        if contcol~=1
                [S,U]=premulti(S,U,k,m);
                la=size(S,2);
        elseif contlin~=1
                S=transpol(S,m);
                V=transpol(V,m);
                [S,V]=premulti(S,V,k,p);
                S=transpol(S,p);la=size(S,2);
                V=transpol(V,m);lv=size(V,2);
        elseif k~=1 & p15
                s_ant=S(k-1,k-1:m:la);
                s_atu=S(k,k:m:la);
                g=mdcpoly(s_ant,s_atu);
                for j=1:(size(s_ant,2)-size(g,2));g=[0 g];end;
                p17=deconv(cutzeros(s_ant),cutzeros(g));
                degp17=coldeg(p17);
                if sc(g-s_ant)>tol & degp17~=0;
                        [p1,q1]=reseqpol(s_ant,s_atu,g);
                        S(k,k-1:m:la)=g;
                        U(k,:)=U(k,:)+conv(p1,U(k-1,:));
                        Bv=[];for j=1:m;Bv=[Bv;conv(q1,V(j,k:m:end))];end
                        V(:,k-1:m:end)=V(:,k-1:m:end)+Bv;
                        k=k-1;
                else
                        p15=0;
```

```
                end
            else
                if k~=p & k~=m
                    k=k+1;p15=1;
                else
                    r=k;flag=0;
                end
            end
    end
end
sdiag=diagpol(S,m);
veci=[];lsdiag=size(sdiag,1);
for i=1:lsdiag
    vi=cutzeros(sdiag(i,:));
    if vi(1)~=0;veci=[veci 1/vi(1)];end
end
veci=diag(veci);
[row,col]=size(veci);
if row<p
    veci=[veci; zeros(p-row,col)];
    [row,col]=size(veci);
end
if col<p;veci=[veci zeros(row,p-col)];end
S=convmat(veci,S,m);S(abs(S)<tol)=0;
U=convmat(veci,U,p);U(abs(U)<tol)=0;
%******************** Auxiliar Function ********************************
function [S,U]=premulti(S,U,k,m)
[p,la]=size(S);
u=eye(p);
uvec=matsplit(u);
muvec=size(uvec,1);
aij=cutzeros(S(k,k:m:la));
for i=k+1:p;
    kp=i+p*(k-1);
    bij=S(i,k:m:la);
    [qij,rij]=deconv(bij,aij);
    luvec=size(uvec,2);
    lqij=length(qij);
    uvec=[zeros(muvec,lqij-luvec)  uvec];
    uvec(kp,:)=-qij;
```

```
end
lambda_n=vecmat(uvec,p);
S=cutzeros(convmat(lambda_n,S,m),m);
U=cutzeros(convmat(lambda_n,U,p),p);
```

A.2. Nmcop2ss

A função *nmcop2ss.m* visa encontrar as matrizes que compõem a forma controlador a partir de uma DFM coprima. Esta função foi escrita com base no Algoritmo 1 de [27].

```
% NMCOP2SS Realization in the controlator form given a MFD coprime.
%   [Ac,Bc,Cc,Dc]=NMCOP2SS(N,M) calculate the realization (Ac,Bc,Cc,Dc)
%   in the controlator form from a right coprime Matrix Fraction
%   Description (MFD) G(s)=N(s)M^(-1)(s). The degree of the columns of
%   the matrix M must be bigger than that of N.
%
%   See also DHCDLCSPLIT, RCMFD, LCMFD.

%       Original by J.C.Basilio
%       $Revision: 0.0 $  $Date: 07-Nov-2005 $

function [ac,bc,cc,dc]=nmcop2ss(n,m)
tol=1e-6;
mlin=size(m,1);graucol=coldeg(m);[mhc,mlc]=dhcdlcsplit(m);
[nhc,nlc]=dhcdlcsplit([eye(size(n,1),mlin) n],mlin);nhc=[];
if abs(det(mhc))<tol
        disp('Leading coefficient matrix of M(s) approximately singular');
        ac=[];bc=[];cc=[];dc=[];return;
end
ac=[];bc=[];
for i=1:mlin;
        ac=blkdiag(ac,[zeros(1,graucol(i));eye(graucol(i)-1) zeros(graucol(i)-1,1)]);
        bc=blkdiag(bc,eye(graucol(i),1));
end
mhcinv=inv(mhc);prod=mhcinv*mlc;
for i=1:length(graucol)
        if i==1;ni=1;else ni=norm(graucol(1:i-1),1)+1;end
        ac(ni,:)=-prod(i,:);bc(ni,:)=mhcinv(i,:);
end
```

```
cc=nlc;dc=zeros(size(n,1),mlin);
```

A.3. Dhcdlcsplit

A função *dhcdlcsplit.m* é aquela que decompõe uma matriz polinomial *D(s)* nas matrizes D_{hc} e D_{lc}.

```
% DHCDLCSPLIT Decomposes a polinomial matrix in Dhc and Dlc.
%   [DHC,DLC]=DHCDLCSPLIT(D,ncol) is the decomposition of the polynomial
%   coefficient matrix D with ncol columns in the matrices Dhc and Dlc.
%
%   See also POLYBASES.

%      Original by J.C.Basilio
%      $Revision: 0.0 $  $Date: 12-Apr-2006 $

function [dhc,dlc]=dhcdlcsplit(d,m)
p=size(d,1);if nargin==1;m=p;end
graucol=coldeg(d,m);dvec=matsplit(d,m);
dhc=[];dlc=[];ncdvec=size(dvec,2);
for i=1:m
	dhc=[dhc dvec((i-1)*p+1:i*p,ncdvec-graucol(i))];
	if graucol(i)==0
		dlc=[dlc zeros(p,1)];
	else
		dlc=[dlc dvec((i-1)*p+1:i*p,ncdvec-graucol(i)+1:ncdvec)];
	end
end
```

A.4. Matkron

A função *matkron.m* realiza o produto de Kronecker entre duas matrizes polinomiais.

```
% MATKRON Kronecker tensor product of polynomial matrices.
%   MATKRON(A,B,ma,mb) is the Kronecker tensor product of two polynomial
%   coefficient matrices A with ma columns and B with mb columns.
%   The result is a large matrix formed by taking all possible
%   products between the polynomials of A and those of B.
%   MATKRON(A,B) works when the matrices A and B are squared.
%
```

```
%   See also KRON.

%       Author(s):  R.O.Leite
%       $Revision: 0.0 $  $Date: 24-Nov-2005 $

function K=matkron(A,B,ma,mb)
[pa,cola]=size(A);[pb,colb]=size(B);
if nargin==2;
        ma=pa;mb=pb;
elseif nargin==3
        error('It is necessary to inform both the number of columns of A and B.')
end
K=[];
for d=1:ma
        for c=1:mb
                for i=1:pa
                        for j=1:pb;
                                K=[K;conv(A(i,d:ma:cola),B(j,c:mb:colb))];
                        end
                end
        end
end
K=vecmat(K,pa*pb,ma*mb);
```

A.5. Rubin

Outra função importante é *rubin.m*, que determina o número e comprimento das cadeias de Jordan. Esta função foi escrita de acordo com o Algoritmo 3.1 extraído de [34].

```
%RUBIN Length of Jordan blocks.
%   RUBIN(A), by the Rubin's Algorithm, we calculate the length of the blocks
%   of repeated eigenvalues in the Jordan form of a square matrix A.
%
%   This function writes the length of Jordan blocks (JB) in the form:
%        [av(1)   av(2)    ...   av(n)   ] Eigenvalues (av)
%        [JB(1,1)JB(2,1) ...   JB(n,1)] No. of blocks of length(JB) 1 of the av 1
%   JB=[JB(1,2)JB(2,2) ...   JB(n,2)] No. of blocks of length(JB) 2 of the av 2
%        [...        ...         ...   ...        ] ...
%       [JB(1,n) JB(2,n) ...   JB(n,n)] No. of blocks of length(JB) n of the av n
```

```
%
%   See also EIG.

%   Author(s):  Raphael de Oliveira Leite
%   $Revision: 1.0 $  $Date: 07-Aug-2006 $

function cc=rubin(A)
tol=1e-5;
row=size(A,1);
av=eig(A);
av=sort(arred(av,tol));
av_d=av(1);lambda=[];
for i=2:length(av)
	if ~ismember(av(i),av_d);
		av_d=[av_d; av(i)];		% distinct eigenvalues
	elseif ~ismember(av(i),lambda);
		lambda=[lambda; av(i)];		% repeted eigenvalues
	end
end
cont=length(av_d);
if cont==length(av)	% if all eigenvalues are distinct
	cc=[];
	for i=1:length(av)
		cc=[cc; 1];
	end
	ii=cc;lambda=av.';
else			% if there are repeated eigenvalues
	cc=[];ii=[];
	for j=1:length(lambda)
		flag=1;mi=0;i=0;
		p=[];sigma1=[];sigma2=[];	% p: rank
		while flag
			i=i+1;
			Ami=arred((A-lambda(j)*eye(row))^mi,tol);
			p=[p; rank(Ami)];
			if i>1
				sigma1=[sigma1; p(i-1)-p(i)];
				if length(sigma1)>1
					sigma2=[sigma2; sigma1(end-1)-sigma1(end)];
				end
```

```
                    if p(i-1)==p(i);flag=0;end
                end
                mi=mi+1;
            end
            cc=[cc; sigma2];
            ii=[ii; length(sigma2)];
        end
end
% Write cc in the form described above
m=ii(1);for i=2:length(ii);if m<ii(i);m=ii(i);end;end
cc_aux=cc;
cc=zeros(m+1,length(ii));
cc(1,:)=[lambda];
cc_ind=1;
for i=1:length(ii)
        if i>length(ii)
            cc(2,i)=1;
        else
            for j=1:ii(i)
                cc(j+1,i)=cc_aux(cc_ind);
                cc_ind=cc_ind+1;
            end
        end
end
for i=1:length(av_d)
        if ~ismember(av_d(i),cc(1,:))
            cci=[av_d(i);1;zeros(size(cc,1)-2,1)];
            cc=[cc cci];
        end
end
```

A.6. Polyinv

A função *polyinv.m* encontra a inversa de matrizes polinomiais, ela foi escrita com base nos algoritmos apresentados em [27].

```
%POLYINV Inverse of column or non-column reduced matrices.
%   [N,d]=POLYINV(D) is the inverse of column reduced matrices. Where G is a
%   numerator matrix and d is a denominator polynomium.
%
%   [N,d,U]=POLYINV(D) provides the means to obtain column reduced and
```

```
%   unimodular matrices (G and U, respectively) such that for a given non-column
%   reduced square matrix D.
%
%   See also INV.

%       Original by R.O.Leite
%       $Revision: 1.0 $  $Date: 24-June-2006 $
%       Last Revision by R.O.Leite

%   This is a function based on Algorithm 3 of J.C.Basilio, “Inversion of polynomial
%   matrices via state-space”. Linear Algebra and its Aplications 357 (2002) 259-271.

function [N,d,U]=polyinv(D)
 [p c]=size(D);m=c/p;
if fix(m)~=m;error('Matrix must be square.');end
if c==p;
        N=inv(D);d=1;U=eye(p);
        return;
else
        if polydet(D)==0;
                warning('Matrix is singular to working precision.')
                return
        end
end
[Dbarra,U]=reducol(D,p);
grcdbr=coldeg(Dbarra);
indgrcbr=find(grcdbr==0);lindgrcbr=length(indgrcbr);
if lindgrcbr==p
        Ndbinv=inv(Dbarra);d=1;
else
        diagdelta=[zeros(p,1) ones(p,1)];
        if ~isempty(indgrcbr);diagdelta(indgrcbr,:)=[1 0];end
        delta=diaglamb(diagdelta);
        Dtil=cutzeros(convmat(Dbarra,delta));
        [Ac,Bc,Cc]=nmcop2ss(eye(p),Dtil);
        for i=1:p;
                [Ndtilinvi,d]=ss2tf(Ac,Bc,Cc,zeros(p),i);
                if i==1;Ndtilinv=Ndtilinvi;else Ndtilinv=Matform(Ndtilinv,Ndtilinvi,i-1,1);end
        end
        Ndbinv=cutzeros(convmat(delta,Ndtilinv));
```

```
end
N=cutzeros(convmat(U,Ndbinv));
flag=1;
while flag
        if d(end)==0 & N(:,end-p+1:end)==zeros(p)
                N=N(:,1:end-p);
                d=d(1:end-1);
        else
                flag=0;
        end
end
```

A.7. Smithbl

A função *smithbl* reúne todas as rotinas necessárias para determinação da forma de Smith de matrizes polinomiais. O algoritmo que deu origem à função é o Algoritmo 3.2

```
%SMITHBL Smith form of a polynomial matrix.
%   SMITHBL(A) is the Smith form of a square coeficient polynomial matrix A.
%
%   [S,U,V]=SMITHBL(A) also calculate the unimodular matrices U and V that
%   form the equation S = U*A*V.
%
%   The matrix A must be written as A=[An An-1 ... A2 A1 A0], where
%   A(s)=An*s^n+An-1*s^(n-1)+...+A2*s^2+A1*s+A0.
%
%   See also SMITHKUC, RUBIN.

%       Original by R.O.Leite
%       $Revision: 2.0 $  $Date: 07-Aug-2006 $
%       Last Update by R.O.Leite

function [S,U,V]=smithbl(a)
global tol
tol=1e-5;
[AR,V1]=reducol(a);
[p,la]=size(AR);m=p;
Im=eye(m);
grcar=coldeg(AR);
J=find(grcar==0);
if isempty(J)
```

```
        D=Im;
else
        flag=1;
        while flag
                s0=round(rand(1)*10);
                rdetA=roots(polydet(AR));
                if ~ismember(s0,rdetA)
                        flag=0;
                end
        end
        D=[];
        for i=1:m
                if ismember(i,J)
                        di=[1 s0];
                else
                        di=1;
                end
                D=concatena(D,di);
        end
        D=diagmat(D,m);
end
ARbarra=cutzeros(convmat(AR,D));
[ac,bc,cc,dc]=nmcop2ss(Im,ARbarra);
ccjac=rubin(ac);
avacd=ccjac(1,:);
n=size(ccjac,2);
nci=[];sigma=[];
for i=1:n;
        nci=[nci sum(ccjac(2:end,i))];
end
for i=1:n
        j=1;
        for p=1:size(ccjac,1)-1
                for c=1:ccjac(p+1,i)
                        ni_ij(i,j)=p;
                        j=j+1;
                end
        end
end
for k=0:m-1
```

```
        sigmai=[];
        for i=1:n
                nci_k=nci(i)-k;
                if nci_k<=0
                        ni_incik=0;
                        pol=1;
                else
                        ni_incik=ni_ij(i,nci_k);
                        pol=[1 -avacd(i)];
                        for j=2:ni_incik
                                pol=conv(pol,[1 -avacd(i)]);
                        end
                end
                if isempty(sigmai)
                        sigmai=pol;
                else
                        sigmai=conv(sigmai,pol);
                end
        end
        sigma=concatena(sigma,sigmai);
end
Sbarra=sigma(end:-1:1,:);
Dbarra=[];
for i=1:m
        if i<=m-length(J)
                Dbarra=concatena(Dbarra,1);
        else
                Dbarra=concatena(Dbarra,[1 s0]);
        end
end
S=[];
for i=1:m
        if isempty(S)
                S=deconv(cutzeros(Sbarra(i,:)),cutzeros(Dbarra(i,:)));
        else
                S=concatena(S,deconv(cutzeros(Sbarra(i,:)),cutzeros(Dbarra(i,:))));
        end
end
S=diagmat(S);
% ************ UNIMODULAR MATRICES CALCULATION ************
```

```
I=eye(m);
M1=matkron(I,S);M2=matkron(-transpol(a,m),I);
M=matform(M1,M2,m^2,m^2);
X=arred(polybases(M,m^2,2*m^2),tol);
ri=X(1:m^2,:);ui=X(m^2+1:end,:);
RL=[];UL=[];
for i=1:m^2;
        Ri=cutzeros(vecmat(ri(:,i:m^2:end),m));
        Ui=cutzeros(vecmat(ui(:,i:m^2:end),m));
        tr=isunimodular(Ri);tu=isunimodular(Ui);
        if tr & tu
                U=Ui;[V,dv]=polyinv(Ri);
                if S-convmat(U,convmat(a,V))<tol;return;end
        end
        RL=matform(RL,Ri,m*(i-1),m);
        UL=matform(UL,Ui,m*(i-1),m);
end
ind=combntns(1:m^3,m);
lind=size(ind,1);
Mr=[];Mu=[];
i=1;
for i=1:lind
        mr=[];mu=[];
        for j=1:m
                mr=matform(mr,RL(:,ind(i,j):m^3:end),j-1,1);
                mu=matform(mu,UL(:,ind(i,j):m^3:end),j-1,1);
        end
        mr=cutzeros(mr);mu=cutzeros(mu);
        Mr=concatena(Mr,polydet(mr));
        Mu=concatena(Mu,polydet(mu));
end
M=[Mr Mu];
alpha0=rand([m^2,1]);
options=optimset('Display','off');
[ahc,alc]=dhcdlcsplit(a);detA=det(ahc);
alpha=fsolve(@alphastar,alpha0,options,M,m,detA,ind);
R=vecmat(convmat(ri,alpha,1),m);
U=vecmat(convmat(ui,alpha,1),m);
[invR,dr]=polyinv(R);
V=convmat(V1,invR);
```

```
tv=isunimodular(V);tu=isunimodular(U);
if ~(tv & tu)
        U=zeros(m);V=U;
        disp('Error: it is not possible to calcule the unimodular matrices U and V')
end
%******************** Auxiliar Function #1 ********************************
function F=alphastar(x,M,m,dv,ind)
% ALPHASTAR to use with fsolve.
%   F is the solution to fsolve above, x is a inicial value,
%   M,m and dv are others input parameters.

X=[];
for i=1:length(x);X=[X;x(i)*eye(m)];end
lind=size(ind,1);
xvec=[];
for i=1:lind
        mx=[];
        for j=1:m
                mx=[mx;X(ind(i,j),:)];
        end
        xvec=[xvec;polydet(mx)];
end
mM=size(M,2);fvec=[];
for i=1:mM
        fvec=[fvec M(:,i).*xvec];
end
fveci=zeros(1,m^2);fveci(m)=1/dv;fveci(m^2)=1;
fvec=concatena(fvec,-fveci);
F=sum(fvec)';
%******************** Auxiliar Function #2 ********************************
function t=isunimodular(W)
% ISUNIMODULAR True for unimodular matrix.
%   ISUNIMODULAR(W) returns true if a given square matrix W
%   is a unimodular matrix and false if not.

%       Original by R.O.Leite
%       $Revision: 0.0 $  $Date: 26-Jun-2006 $

global tol
detw=cutzeros(polydet(W));
```

```
c1=abs(detw)>tol;
c2=length(detw)==1;
if c1 & c2
	t=1;
else
	t=0;
end
```

A.8. Diagmat

A função *diagmat.m* cria uma matriz polinomial diagonal a partir de um vetor de polinômios.

```
%DIAGMAT Diagonal polynomial matrix.
%   DIAGMAT(V) returns the coefficient diagonal polynomial matrix from a
%   column vector V.
%
%   DIAGMAT(V,cols) is used when V is a row vector, then it's necessary
%   to inform the number of columns of V (cols).
%
%   See also TRANSPOL, DIAG.

%   Original by R.O. Leite
%   $Revision: 0.0 $  $Date: 02-Jan-2006 $

function A=diagmat(V,cols)
if nargin==1;cols=1;end
[p,m]=size(V);
if p==1;
	V=transpol(V,cols);
	[p,m]=size(V);
end
A=[];I=eye(p);
for i=1:m
	for j=1:p
		As(j,:)=V(j,i).*I(j,:);
	end
	A=[A As];
End
```

A.9. Concatena

A função *concatena.m* realiza a concantenação de polinômios em um único vetor.

```
%CONCATENA Polynomium concatenation in a vector.
%   S=CONCATENA(S,sa) is the vertical concatenation of the polynomium
%    sa in the vector S.
%
%   See also MATFORM.

%   Author(s):  Raphael de Oliveira Leite
%   $Revision: 0.0 $  $Date: 08-Mar-2006 $

function S=concatena(S,sa)
if length(sa)>size(S,2)
        saux=zeros(size(S,1),size(sa,2));
        df=size(saux,2)-size(S,2);
        for j=size(S,2):-1:1
                saux(1:size(S,1),j+df)=S(:,j);
        end
        S=saux;
elseif size(S,2)>length(sa)
        df=size(S,2)-length(sa);
        saux=[];
        for j=1:df
                saux=[saux 0];
        end
        sa=[saux sa];
end
S=[S;sa];
```

A.10. Ordenapoly

A função *ordenapoly.m* ordena a seqüência de polinômios fornecida através de um vetor de forma crescente considerando o grau dos polinômios.

```
%ORDENAPOLY Sort a polynomial vector.
%   VEC=ORDENAPOLY(VEC) sort the polynomial column vector vec increasing by
%   the polynomial grade.
```

```
function vec=ordenapoly(vec)
 [row,col]=size(vec);
if row>1
	flag=1;
	while flag
		for i=1:row-1
			if vec(i,1)>vec(i+1,1);vec([i i+1],:)=vec([i+1 i],:);end
		end
		v=[];
		for i=1:row-1;v=[v;vec(i,1)>vec(i+1,1)];end
		if v==0;flag=0;end
	end
	a=rand(1);
	for i=1:row-1
		if vec(i,1)==vec(i+1,1) & vec(i,1)~=a
			a=vec(i,1);ind=[];
			ind=find(vec(:,1)==a);
			vaux=vec(ind,2:end);
			if ~isempty(vaux);vaux=ordenapoly(vaux);end
			vec(ind,2:end)=vaux;
		end
	end
end
```

Referências Bibliográficas

"Não se pode ensinar tudo a alguém, pode-se apenas ajudá-lo a encontrar por si mesmo."
Galileu Galilei

[1] GANTMACHER, F.R., *The Theory of Matrices, vols. I e II.* New York, Chelse Publishing Company, 1959.

[2] GOHBERGB, I., LANCASTER, P., RODMAN, L., *Matrix Polynomials*. New York, Academic Press, 1982.

[3] WOLOVICH, W.A., *Linear Multivariable Systems*. New York, Spriger, 1974.

[4] KAILATH, T., *Linear Systems.* Englewood Cliffs, N.J., Prentice Hall, 1980.

[5] STEFANIDIS,P., PAPLINSKI,A.P., GIBBARD, M.J., “Numerical Operations with Polynomial Matrices”, *Lecture Notes in Control and Information Sciences*, vol. 171, Berlin, Springer, 1992.

[6] CHEN, C.T., *Linear System Theory and Design*. 3rd Edition. New York, Oxford University Press, 1999.

[7] MORF, M., LÉVY, B.C., KUNG, S.Y., New Results in 2-D Systems Theory, Part I: 2-D Polynomial Matrices, Factorization, and Coprimeness”, *Proceedings of the IEEE*, vol. 65, no. 6, June 1977.

[8] CASANOVA, A., FRAGOPOULOS, D., GRIMBLE, M.J., “A Polynomial Solution to the Scalar 4-Block H_∞ General Distance Problem”, *International Journal of Control*, vol. 65, no. 1, pp. 33-52, 1996.

[9] FORNASINI, E., VALCHER, M.E., “nD Polynomial Matrices with Aplications to Multidimensional Signal Analysis”, *Multidimensional Systems and Signal Processing*, vol. 8, pp. 387-408, 1997.

[10] KARLSSON, G., VETTERLI, M., “Theory of Two-Dimensional Multirate Filter Banks”, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 6, pp. 925-937, June 1990.

[11] CVETKOVIC, Z., VETTERLI, M., “Oversampled Filter Banks”, *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1245-1255, May 1998.

[12] SMITH, H.J.S., “On Systems of Linear Indeterminate Equations and Congruencies”, *Philosophical Transactions Royal Society London,* vol. 151, pp. 293-326, 1861.

[13] MACMILLAN, B., “Introduction to Formal Realizability Theory”, *Bell. Systems Technology Journal*, vol. 31, pp. 217-239, 541-600, 1952.

[14] PACE,I.S., BARNETT, S.,“Efficient algorithms for linear system calculations. Part 1, Smith form and common divisor of polynomial matrices”, *International Journal of Systems Science*, vol. 5, pp. 403-411, 1974.

[15] BRADLEY, G.H., “Algorithms for Hermite and Smith Normal Matrices and Linear Diophantine Equations”, *Math. Computation*, vol.25, pp.897-907, 1971.

[16] RAMACHANDRAN, V., “Exact Reduction of a Polynomial Matrix to the Smith Normal Form”, *IEEE Transactions on Automatic Control*, vol. AC-24, pp. 638-641, August 1979.

[17] KALTOFEN, E., KRISHNAMOORTHY, M.S., SAUNDERS, B.D., "Fast Parallel Computation of Hermite and Smith Forms of Polynomial Matrices", *SIAM J. Algebraic and Discrete Methods*, vol. 8, pp. 683-690, 1987.

[18] KALTOFEN, E., KRISHNAMOORTHY, M.S., SAUNDERS, B.D., "Mr. Smith goes to Las Vegas: Randomized Parallel Computation of the Smith Normal Form of Polynomial Matrices". In: *EUROCAL'87*, LNCS 378, pp. 317-322, Springer Verlag, 1989.

[19] STORJOHANN, A., LABAHN, G., "A Fast Las Vegas Algorithm for Computing the Smith Normal Form of a Polynomial Matrix", *Linear Algebra and It's Applications*, vol. 253, pp. 155-173, 1997.

[20] VILLARD G., "Computation of the Smith Normal Form of Polynomial Matrices". In: *Proceedings of the 1993 International Symposium on Symbolic and Algebraic Computation, ISSAC'93*, pp. 209-217, ACM Press, New York, 1993.

[21] VILLARD G., "Fast Parallel Computation of the Smith Normal Form of Polynomial Matrices". In: *Proceedings of the 1994 International Symposium on Symbolic and Algebraic Computation, ISSAC'94*, pp. 312-317, ACM Press, New York, 1994.

[22] JÄGER, G., "A New Algorithm for Computing the Smith Normal Form and its Implementation on Parallel Machines". In: *Proceedings of the 18th International Parallel and Distributed Processing Symposium, IPDPS'2004*,pp. 175-182, ACM Press, New York, 2004.

[23] LIN, Y.P., “Smith Form of FIR Pseudocirculants”, *IEEE Transactions Signal Processing Letters*, vol. 9, pp. 256-258, August 2002.

[24] KALMAN, R.E., “Irreducible realizations and the degree of a rational matrix”, *SIAM J. Control,* vol. 13, pp. 520-544, June 1965.

[25] PANDA, S. P., CHEN, C. T., “Irreducible Jordan form realization of a rational matrix”, *IEEE Transactions on Automatic Control*, vol. AC-14, pp. 66-69, February 1969.

[26] KUO, Y. L., “On the irreducible Jordan form realization and the degree of a rational matrix”, *IEEE Transactions on Circuit Theory*, vol. CT-17, pp. 322-332, August 1970.

[27] BASILIO, J. C., “Inversion of polynomial matrices via state-space”, *Linear Algebra and its Applications*, vol. 357, pp. 259-271, 2002.

[28] VAN DOOREN, P. M., DEWILDE, P. VANDEWALLE, J., “On the determination of the Smith-MacMillan form of a rational matrix from its Laurent Expansion”, *IEEE Transactions on Circuits and Systems,* vol. CAS-26, March 1979.

[29] BASILIO, J.C., MOREIRA, M.V., “A Robust Solution of the Generalized Polynomial Bezout Identity”, *Linear Algebra and its Applications*, vol. 385, pp. 287-303, 2004.

[30] MATH WORKS, *Matlab Programming Tips*. Natick, M.A., The Math Works Inc., 2005

[31] BARCELOS, A.F., *Algoritmos para Matrizes Polinomiais*. Dissertação de M.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2004.

[32] SKOGESTAD, S., POSTLETHWAITE, I., *Multivariable Feedback Control Analysis and Design*. 2nd Edition. New York, John Wiley & Sons, 2001.

[33] BASILIO, J.C., *Notas de Aula de Sistemas Lineares II*, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2003.

[34] RUBIN, W.B., "A Simple Method for Finding the Jordan Form of a Matrix", *IEEE Transactions on Automatic Control*, pp. 145-146, February 1972.

[35] BASILIO, J. C., *Multivariable Generalized Nyquist Design.* Tese de D.Sc., University of Oxford, Oxford, UK,1995.

“Não há homem, por sábio que seja, que em alguma época da sua mocidade não tenha levado uma vida ou não tenha pronunciado umas palavras que não lhe agrade recordar e que quisesse ver anuladas. Mas na verdade não deve senti-lo inteiramente, pois não pode estar certo de ter alcançado a sabedoria, na medida do possível, sem passar por todas as encarnações ridículas ou odiosas que a precedem... A sabedoria não se transmite, é preciso que a gente mesma a descubra depois de uma caminhada que ninguém pode fazer em nosso lugar, e que ninguém nos pode evitar, porque a sabedoria é uma maneira de ver as coisas.”

Proust (1871-1922)