Gildson Queiroz de Jesus

Algoritmos Array para Filtragem de Sistemas Lineares

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica

Área de Concentração: Sistemas Dinâmicos
Orientador: Prof. Dr. Marco Henrique Terra
Co-orientador: João Yoshiuki Ishihara

São Carlos
2007

Sumário

Resumo	vii
Abstract	ix
Lista de Figuras	xi
Lista de Tabelas	xiii
1 Introdução	1
1.1 Motivação	1
1.2 Estrutura do Texto	2
2 Revisão Bibliográfica	3
2.1 Algoritmos <i>Array</i>	3
2.1.1 Exemplo de <i>Array</i>	4
2.2 Algoritmos <i>Array</i> Rápidos	5
2.3 Sistemas Lineares Sujeitos a Saltos Markovianos	7
2.4 Sistemas Singulares	8
3 Filtro para SLSSM na Forma de Informação	11
3.1 Resultados Preliminares	11
3.2 Filtro para SLSSM na Forma de Informação	16
4 Algoritmos <i>Array</i> para Filtragem de SLSSM	19
4.1 Algoritmo <i>Array</i> do Filtro para SLSSM	19
4.2 Algoritmo <i>Array</i> do Filtro para SLSSM na Forma de Informação	23
4.3 Exemplos Numéricos	25
5 Algoritmos <i>Array</i> Rápidos para Filtragem de Sistemas Singulares	29
5.1 Estimativa Filtrada Nominal	29
5.2 Estimativa Preditora Nominal	31

5.3	Estimativa Filtrada Robusta	33
5.4	Estimativa Preditora Robusta	36
5.5	Exemplos Numéricos	38
6	Conclusão	41
	Referências Bibliográficas	43
A	Resultados Matriciais Importantes	45
A.1	Complemento de Schur	45
A.2	Lema da Inversão de Matrizes	46
A.3	Transformações Unitárias	46
A.3.1	Transformações de Householder	46
A.3.2	Transformações de Givens	47
A.3.3	Rotações de Givens Hiperbólicas	48
A.4	Teoremas Auxiliares	49
A.4.1	Rotações de Bases	49
A.4.2	Transformações J-unitárias	49

A Idelberto e Jocélia, meus pais, com amor, admiração e gratidão por sua compreensão, carinho, presença e incansável apoio ao longo do período de elaboração deste trabalho.

Agradecimentos

A Deus, que me concedeu graça, força e capacidade na execução deste trabalho. A Ele toda honra e glória eternamente.

A Idelberto, Jocélia, Ainá e Raquel, minha família, pelo carinho e constante apoio nesta jornada.

A Jaíra, Darci, Lúgia e Humberto, meus tios, pelo carinho durante este trabalho.

Ao Prof. Dr. Marco Henrique Terra, que, nos anos de convivência, muito me ensinou, contribuindo para meu crescimento científico e intelectual.

Ao Prof. Dr. João Y. Ishihara, pela atenção e apoio durante o processo deste trabalho .

Ao pessoal do Laboratório de Sistemas Inteligentes (LASI) pela amizade e companherismo cotidiano.

Ao pessoal da Comunidade Cristã de São Carlos pela amizade e suporte espiritual.

Aos amigos que alcancei durante este tempo que muito contribuíram para o meu desenvolvimento, Samuel, Aline, Roberto, Eduardo, Fabíolo, Renata, Fernanda, Nazira, Evandro, Bruno, entre outros.

À Escola de Engenharia de São Carlos, pela oportunidade de realização do curso de mestrado.

Resumo

Jesus, G. Q. Algoritmos Array para Filtragem de Sistemas Lineares. 2007. 64 f. Tese (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Paulo, 2007.

Esta dissertação desenvolve filtro de informação, algoritmos *array* para estimador do erro médio mínimo quadrático para sistemas lineares sujeitos a saltos Markovianos e algoritmos *array* rápidos para filtragem de sistemas singulares convencionais. Exemplos numéricos serão apresentados para mostrarem as vantagens dos algoritmos *array* deduzidos. Parte dos resultados obtidos nesta pesquisa serão publicados no seguinte artigo

Terra et al. (2007). Terra, M. H., Ishihara, J. Y. and Jesus, G. Q. (2007). *Information Filtering and Array Algorithms for Discrete-Time Markovian Jump Linear Systems. Proceedings of the American Control Conference ACC07.*

Palavras-Chave: Filtro informação, algoritmo *array*, tempo discreto, sistemas lineares, saltos Markovianos, sistemas singulares.

Abstract

Jesus, G.Q. Array Algorithms for Filtering of Linear Systems. 2007. 64 f. Thesis (Master) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Paulo, 2007.

This dissertation develops information filter and array algorithms for linear minimum mean square error estimator (LMMSE) of discrete-time Markovian jump linear systems (MJLSs) and fast array algorithms for filtering of standard singular systems. Numerical examples to show the advantage of the array algorithms are presented. Some results obtained in this research are published in the following paper

Terra et al. (2007). Terra, M. H., Ishihara, J. Y. and Jesus, G. Q. (2007). Information Filtering and Array Algorithms for Discrete-Time Markovian Jump Linear Systems. Proceedings of the American Control Conference ACC07.

Keywords: *Information filter, array algorithm, discrete-time, linear systems, Markovian jump, singular systems.*

Lista de Figuras

4.1	Valores Singulares de $\tilde{Z}_{i i-1}$	27
4.2	Valores Singulares de $\tilde{Z}_{i i-1}^{-1}$	27
5.1	Valores Singulares da Estimativa Filtrada Robusta $P_{i i}^{-1}$	40

Lista de Tabelas

4.1	Erro médio quadrático para o cálculo dos valores singulares de $\tilde{Z}_{i i-1}$ e $\tilde{Z}_{i i-1}^{-1}$. . .	26
5.1	Erro médio quadrático para o cálculo dos valores singulares de $P_{i i}^{-1}$	39

Capítulo 1

Introdução

1.1 Motivação

A abordagem para a filtragem de sistemas desenvolvida em (Kalman (1960)), sintetizada através dos filtros de Kalman, tem sido aplicada em vários problemas práticos em áreas como aeronáutica, processamento de sinais e controle de processos industriais. Neste trabalho serão abordados os problemas de filtragem de sistemas lineares sujeitos a saltos Markovianos (SLSSMs), mais especificamente, o filtro para o estimador linear do erro mínimo médio quadrático deduzido em (Costa e Guerra (2002)) e de sistemas singulares formulados em (Terra et al. (2007)).

Dentre as várias limitações que têm sido apresentadas pelos filtros de Kalman, tanto para sistemas convencionais quanto para SLSSMs, pode-se exemplificar problemas de convergência devido a falta de fidelidade dos algoritmos numéricos ou modelagem não apropriada dos sistemas a serem estimados (Jazwinski (1970)). Para contornar estes problemas, têm sido desenvolvidos novos algoritmos para diferentes implementações do filtro. Em particular destaca-se neste trabalho os algoritmos *array*, introduzidos por Potter (Potter e Stern (1963)), os algoritmos *array* rápidos (Morf e Kailath (1974)) e também os filtros de informação (Anderson e Moore (1979)).

O filtro de Kalman calculado via algoritmo *array* apresenta algumas vantagens sobre o filtro de Kalman calculado através da equação de Riccati pois aumenta a eficiência e a estabilidade numérica nas implementações dos filtros devido ao uso de transformações ortogonais nos cálculos. Algoritmos *array* reduzem a faixa dinâmica dos valores calculados em implementações por aritmética de ponto fixo. A consequência mais importante dessa estabilidade numérica é a garantia de soluções Hermitianas no cálculo da matriz de covariância do erro de estimativa. Tais

algoritmos têm sido pesquisados por vários autores nas últimas décadas, dentre eles podemos destacar (Dyer e McReynolds (1969)), (Kaminski e Schmidt (1971)), (Kailath e Hassibi (2000)), (Verhaegen e Van Dooren (1986)), (Morf e Kailath (1975)), (Morf e Kailath (1974)).

Para sistemas singulares convencionais, os algoritmos *array* rápidos são usados para a estimativa recursiva em modelos no espaço de estado invariantes no tempo ou em uma certa classe de modelos em que a variação no tempo é estruturada. Uma das principais vantagens desta recursão é a redução do esforço computacional, enquanto que no filtro convencional de Kalman e na recursão do algoritmo *array* ambos requerem $O(n^3)$ operações por iteração (onde n é o número de estados do modelo), a estimativa recursiva do algoritmo *array* rápido requer somente $O(n^2)$ operações por iteração (Hassibi et al. (1999)). A principal característica deste algoritmo recursivo é propagar ao invés da matriz de covariância P_i uma matriz auxiliar M_i tal que $M_i S M_i^T = P_{i+1} - P_i$, para todo i , sendo S uma matriz assinatura. Para mais detalhes sobre algoritmos *array* rápidos, veja as seguintes referências (Morf e Kailath (1974)), (Morf e Kailath (1975)), (Verhaegen e Van Dooren (1986)), (Sayed e Kailath (1994)), (Babak Hassibi e Sayed (2000)).

Neste trabalho o filtro para sistemas lineares sujeitos a saltos Markovianos desenvolvido em (Costa e Guerra (2002)) é redefinido para a forma dita de informação e também formulado em termos de algoritmos *array*. Em seguida, serão deduzidos algoritmos *array* rápidos para sistemas singulares convencionais considerando os filtros desenvolvidos em (Terra et al. (2007)).

1.2 Estrutura do Texto

Este texto está organizado da seguinte maneira: No Capítulo 2, é apresentada uma revisão bibliográfica dos principais assuntos abordados nesta pesquisa: algoritmos *array* para filtros de Kalman, sistemas lineares sujeitos a saltos Markovianos (SLSSM) e sistemas singulares. No Capítulo 3 é deduzida a versão informação do filtro para SLSSM. No Capítulo 4 são deduzidos algoritmos *array* do filtro para SLSSM e para sua forma de informação e são apresentados resultados de simulação. No Capítulo 5 são deduzidos os algoritmos *array* rápidos para filtragem de sistemas singulares. Por fim, são apresentadas as conclusões e um apêndice com alguns resultados auxiliares utilizados neste trabalho.

Capítulo 2

Revisão Bibliográfica

2.1 Algoritmos *Array*

Potter (Potter e Stern (1963)) foi o primeiro que introduziu o conceito de algoritmo *array* para o problema de atualização da medida do filtro de Kalman no espaço de estado, em um caso especial quando o sistema apresenta observações escalares. Em razão das características numéricas do algoritmo e sua relativa simplicidade, ele foi implementado nos filtros de navegação da nave espacial Apollo, em meados da década de 1960 (Kailath e Hassibi (2000)).

O significado físico da matriz P_i na recursão de Riccati é a variância da predição do erro de estado, conseqüentemente esta matriz é (semi)definida positiva. Os erros de arredondamento que podem ocorrer no cálculo da variância através da equação de Riccati, podem causar uma perda desta propriedade. Por este e outros motivos já citados, os algoritmos *array* para filtragem de sistemas têm recebido atenção na literatura. Com o algoritmo *array*, ao invés de P_i , é propagado o seu fator raiz quadrada $P_i^{1/2}$, ou seja, uma matriz A_i , tal que $P_i = A_i A_i^T$. Apesar da possibilidade de haver erros de arredondamento em A_i , o produto dos fatores multiplicados $P_i = A_i A_i^T$ é sempre uma matriz (semi)definida positiva, evitando assim erros de arredondamento que geram vários problemas computacionais. Essas raízes quadradas não são únicas. Sendo θ uma matriz unitária, ou seja, $\theta\theta^T = \theta^T\theta = I$, então pode-se deduzir que $A\theta$ também é um fator raiz quadrada de P_i . Este fator pode ser único se forem impostas restrições adicionais como P sendo triangular ou Hermitiana. Por conveniência, consideram-se as seguintes igualdades

$$P_i = P_i^{1/2}(P_i^{1/2})^T = P_i^{1/2}P_i^{T/2}. \quad (2.1)$$

Em geral, os fatores raiz quadrada de uma equação recursiva podem ser propagados por algoritmo *array* da seguinte maneira

1. Forma-se um determinado pré-*array* de números baseados nos dados do tempo i .
2. Este pré-*array* é reduzido para uma forma específica (geralmente triangular) por uma sequência de operações unitárias elementares (rotações e reflexões).
3. Os valores desejados do tempo $i + 1$ podem ser imediatamente lidos a partir dos chamados pós-*array*.

2.1.1 Exemplo de *Array*

Antes de apresentar algoritmos *array* para filtragem de SLSSM, nesta seção será apresentado um exemplo do uso de *array* para o filtro de Kalman no espaço de estado baseado em (Hassibi et al. (1999)). Seja a equação de Riccati para o filtro de Kalman no espaço de estado dada pela seguinte equação recursiva

$$\begin{aligned} P_{i+1} &= F_i P_i F_i^T + G_i Q_i G_i^T - K_{p,i} R_{e,i}^{-1} K_{p,i}^T, \\ P_0 &= \pi_0, \quad R_{e,i} = R_i + H_i P_i H_i^T, \quad K_{p,i} = F_i P_i H_i^T R_{e,i}^{-1} \end{aligned} \quad (2.2)$$

e F_i, G_i, H_i, Q_i, R_i são matrizes de parâmetros conhecidas. O complemento de Schur da equação (2.2) é dado por

$$\begin{bmatrix} R_i + H_i P_i H_i^T & H_i P_i F_i^T \\ F_i P_i H_i^T & F_i P_i F_i^T + G_i Q_i G_i^T \end{bmatrix} \quad (2.3)$$

fatorando (2.3), obtém-se

$$\begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} & 0 \\ 0 & F_i P_i^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \begin{bmatrix} R_i^{T/2} & 0 \\ P_i^{T/2} H_i^T & P_i^{T/2} F_i^T \\ 0 & Q_i^{T/2} G_i^T \end{bmatrix}. \quad (2.4)$$

A estimativa é realizada através de uma transformação unitária θ_i , que triangulariza o pré-*array* como mostrado a seguir

$$\begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} & 0 \\ 0 & F_i P_i^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \theta_i = \begin{bmatrix} X & 0 & 0 \\ Y & Z & 0 \end{bmatrix}. \quad (2.5)$$

Considerando o quadrado desta expressão e utilizando o fato de que θ_i é unitária, obtém-se

$$\begin{bmatrix} R_i + H_i P_i H_i^T & H_i P_i F_i^T \\ F_i P_i H_i^T & F_i P_i F_i^T + G_i Q_i G_i^T \end{bmatrix} = \begin{bmatrix} X X^T & X Y^T \\ Y X^T & Y Y^T + Z Z^T \end{bmatrix}. \quad (2.6)$$

Igualando os respectivos termos desta igualdade obtém-se

$$\begin{aligned} X X^T &= R_i + H_i P_i H_i^T = R_{e,i} \\ X &= R_{e,i}^{1/2} \\ Y X^T &= F_i P_i H_i^T \\ Y &= F_i P_i H_i^T R_{e,i}^{-T/2} \\ Y &= K_{p,i} R_{e,i}^{1/2} \\ Z Z^T &= F_i P_i F_i^T + G_i Q_i G_i^T - Y Y^T \\ Z Z^T &= F_i P_i F_i^T + G_i Q_i G_i^T - Y X^T (X X^T)^{-1} X Y^T \\ Z Z^T &= F_i P_i F_i^T + G_i Q_i G_i^T - F_i P_i H_i^T R_{e,i}^{-1} H_i P_i F_i^T \\ Z Z^T &= P_{i+1} \\ Z &= P_{i+1}^{1/2}. \end{aligned} \quad (2.7)$$

Tem-se então a seguinte estimativa calculada através do algoritmo *array*

$$\begin{bmatrix} R_i^{1/2} & H_i P_i^{1/2} & 0 \\ 0 & F_i P_i^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i}^{1/2} & 0 & 0 \\ K_{p,i} R_{e,i}^{1/2} & P_{i+1}^{1/2} & 0 \end{bmatrix}. \quad (2.8)$$

2.2 Algoritmos *Array* Rápidos

Nesta seção será apresentado o algoritmo *array* rápido para o filtro de Kalman no espaço de estado baseado em (Sayed e Kailath (1994)). Seja $\delta P_{i+1} = P_{i+2} - P_{i+1}$, para sistemas invariantes no tempo. δP_i é uma matriz Hermitiana. Pode-se fatorá-la da seguinte forma

$$\delta P_{i+1} = P_{i+2} - P_{i+1} = M_{i+1} S_{i+1} M_{i+1}^T \quad (2.9)$$

sendo S_{i+1} uma matriz assinatura (uma matriz diagonal com ± 1 sobre as diagonais onde δP_{i+1} tem autovalores positivos e negativos). Considera-se $S_{i+1} = S$. Pode-se reescrever a equação

(2.9) como

$$\begin{aligned}
P_{i+2} - P_{i+1} &= -P_{i+1} + FP_{i+1}F^T - \bar{K}_{p,i+1}\bar{K}_{p,i+1}^T + GQG^T \\
P_0 &= \Pi_0 = 0, \quad P_1 = GQG^T \\
\bar{K}_{p,i+1} &= FP_{i+1}H^T R_{e,i+1}^{-1/2}, \quad R_{e,i+1} = R + HP_{i+1}H^T
\end{aligned} \tag{2.10}$$

e F, G, H, Q e R são matrizes conhecidas. Considere agora o complemento de Schur da equação (2.10)

$$\begin{bmatrix} R + HP_{i+1}H^T & HP_{i+1}F^T \\ FP_{i+1}H^T & -P_{i+1} + FP_{i+1}F^T + GQG^T \end{bmatrix} \tag{2.11}$$

pode-se fatorar (2.11) da seguinte maneira

$$\begin{bmatrix} R_{e,i}^{1/2} & HM_i \\ \bar{K}_{p,i} & FM_i \end{bmatrix} \begin{bmatrix} R_{e,i}^{1/2} & \bar{K}_{p,i}^T \\ M_i^T H^T & M_i^T F^T \end{bmatrix}. \tag{2.12}$$

O pré-array portanto é dado por

$$A_i = \begin{bmatrix} R_{e,i}^{1/2} & HM_i \\ \bar{K}_{p,i} & FM_i \end{bmatrix}. \tag{2.13}$$

Seja θ_i matriz J-unitária que triangulariza A_i . Então

$$A_i \theta_i = \begin{bmatrix} X & 0 \\ Y & Z \end{bmatrix}. \tag{2.14}$$

Elevando ao quadrado ambos os lados da igualdade (2.14) e comparando os elementos obtém-se

$$\begin{aligned}
XX^T &= R_{e,i} + HM_i SM_i^T H^T \\
&= R_{e,i} + H(P_{i+1} - P_i)H^T \\
&= R + HP_i H^T + HP_{i+1} H^T - HP_i H^T \\
X &= R_{e,i+1}^{1/2}
\end{aligned}$$

$$YX^T = FP_iH^T + FM_iSM_i^TH^T \quad (2.15)$$

$$= FP_iH^T + F(P_{i+1} - P_i)H^T$$

$$Y = \bar{K}_{p,i+1}$$

$$YY^T + ZZ^T = FP_iH^TR_{e,i}^{-1}HP_iF^T + FM_iSM_i^TF^T$$

$$= FP_iH^TR_{e,i}^{-1}HP_iF^T + F(P_{i+1} - P_i)F^T$$

$$ZZ^T = P_{i+2} + P_{i+1}$$

$$Z = L_{i+1} \quad (2.16)$$

Que resulta na seguinte recursão dita de Chandrasekhar

$$\begin{bmatrix} R_{e,i}^{1/2} & HM_i \\ \bar{K}_{p,i} & FM_i \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix} \quad (2.17)$$

2.3 Sistemas Lineares Sujeitos a Saltos Markovianos

Sistemas sujeitos a saltos Markovianos caracterizam sistemas sujeitos a mudanças que podem ser devidas, por exemplo, a distúrbios abruptos no ambiente, reparos ou falhas nos componentes, mudanças em subsistemas interconectados, etc. Exemplos de tais comportamentos podem ser encontrados em sistemas econômicos, em sistemas de controle de avião, sistemas manipuladores robóticos entre outros. Em alguns casos estes sistemas podem ser modelados por um conjunto de sistemas lineares discretos no tempo cuja transição pode ser dada por uma cadeia de Markov, ou seja, um modelo probabilístico que fornece uma indicação quantitativa das probabilidades de ocorrência dos vários cenários. Uma categoria de modelos para descrever tais comportamentos dinâmicos é a descrita por sistemas lineares sujeitos a saltos Markovianos (SLSSM). O algoritmo *array* que será apresentado a seguir está baseado no seguinte SLSSM discreto no tempo

$$x_{i+1} = F_{\Theta(i)}x_i + G_{\Theta(i)}u_i, \quad (2.18)$$

$$y_i = H_{\Theta(i)}x_i + D_{\Theta(i)}v_i, \quad i = 0, 1, \dots$$

sendo x_i o estado que pertence ao $\mathcal{R}^n$, u_i o distúrbio aleatório do estado que pertence ao $\mathcal{R}^p$, y_i a sequência de saída que pertence ao $\mathcal{R}^m$ e v_i o distúrbio aleatório de saída que pertence ao $\mathcal{R}^p$; Θ_i a cadeia de Markov discreta no tempo com espaço de estado finito $\{i = 1, \dots, N\}$ e $P = [p_{ij}]$ a matriz de probabilidade de transição; F_k, G_k, H_k , e D_k , $k = 1, \dots, N$, são matrizes de dimensões

apropriadas. Considera-se que $D_k D_k^T > 0$ para todo $k = 1, \dots, N$; as seqüências dos distúrbios aleatórios $\{u_i\}$ e $\{v_i\}$ possuem média nula, são estacionárias e mutuamente independentes com matrizes de covariância iguais a identidade; $x_0 1_{\{\Theta_0=k\}}$, $k = 1, \dots, N$, são vetores aleatórios de segunda ordem com $\mathcal{E}\{x_0 1_{\{\Theta_0=k\}}\} = \mu_k$ e $\mathcal{E}\{x_0 x_0^T 1_{\{\Theta_0=k\}}\} = V_k$, $k = 1, \dots, N$; $x(0)$ e $\Theta(i)$ são independentes de $\{u_i\}$ e $\{v_i\}$. Aqui, $1_{\{\cdot\}}$ denota a medida de Dirac (considerando o espaço de probabilidade $(\Omega, \mathcal{F}, \mathcal{P})$, para qualquer $\mathcal{U} \in \mathcal{F}$ e $\omega \in \Omega$, $1_{\mathcal{U}}(\omega) = 1$ se $\omega \in \mathcal{U}$ e $1_{\mathcal{U}}(\omega) = 0$ em caso contrário)(Costa et al. (2005)).

2.4 Sistemas Singulares

Filtragem e controle de sistemas singulares têm recebido grande atenção na literatura. Este interesse é motivado pelo fato de que muitos sistemas podem ser modelados naturalmente como um sistema singular. Aplicações para este tipo de modelo podem ser encontradas, por exemplo, em sistemas econômicos, circuitos elétricos e robótica. Os sistemas singulares foram mencionados pela primeira vez na literatura em 1973 (Singh e Liu (1973)). Em muitas publicações os sistemas singulares são chamados também de semi-estado, sistemas degenerados e sistemas algébricos-diferenciais, por envolverem equações algébricas e equações diferenciais.

Os algoritmos *array* rápidos que serão apresentados neste trabalho estão baseados no seguinte sistema singular linear discreto no tempo

$$\begin{aligned} (E_{i+1} + \delta E_{i+1})x_i &= (F_i + \delta F_i)x_i + w_i, i = 1, \dots, N \\ y_i &= (H_i + \delta H_i)x_i + v_i \end{aligned} \quad (2.19)$$

sendo $x_i \in \mathcal{R}^n$ a variável singular, $y_i \in \mathcal{R}^p$ a medida de saída, $w_i \in \mathcal{R}^m$ e $v_i \in \mathcal{R}^p$ são os ruídos de processo e de medida, $E_{i+1} \in \mathcal{R}^{m \times n}$ matriz singular, $F_i \in \mathcal{R}^{m \times n}$ e $H_i^{p \times n}$, matrizes do sistemas nominal conhecidas. Quando E_i é não singular, o sistema recai no espaço de estado. δE_{i+1} , δF_i e δH_i são perturbações variantes no tempo para as matrizes do sistema nominal definidas como

$$\begin{aligned} \delta F_i &= M_{f,i} \Delta_i N_{f,i}, & \delta E_{i+1} &= M_{f,i} \Delta_i N_{e,i+1} \\ \delta H_i &= M_{h,i} \Delta_i N_{h,i}, & \|\Delta_i\| &\leq 1 \end{aligned}$$

sendo $M_{f,i}$, $M_{h,i}$, $N_{e,i+1}$, $N_{f,i}$, $N_{h,i}$ matrizes conhecidas e Δ_i uma contração. A condição inicial, os ruídos de processo e de medida, $\{x_0, w_i, v_i\}$, são assumidas variáveis aleatórias com média

zero e estatísticas de segunda ordem

$$\mathbb{E} \left(\begin{bmatrix} x_0 \\ w_i \\ v_i \end{bmatrix} \begin{bmatrix} x_0 \\ w_j \\ v_j \end{bmatrix}^T \right) = \begin{bmatrix} P_0 & 0 & 0 \\ 0 & Q_i \delta_{ij} & 0 \\ 0 & 0 & R_i \delta_{ij} \end{bmatrix} > 0 \quad (2.20)$$

sendo $\{R_i, Q_i\}$ matrizes conhecidas, $\delta_{ij} = 1$ se $i = j$ e $\delta_{ij} = 0$ caso contrário.

Capítulo 3

Filtro para SLSSM na Forma de Informação

Neste capítulo será deduzida a versão na forma de informação de um filtro para SLSSM. Filtros expressos na forma de informação propagam o valor do inverso da variância do erro de estimativa $\tilde{Z}_{i|i-1}^{-1}$ ao invés da própria variância $\tilde{Z}_{i|i-1}$. A vantagem do filtro de informação é que se pode ter algoritmos numericamente mais robustos quando há pouca ou nenhuma informação da condição inicial, por exemplo, para $\tilde{Z}_{0|-1}$ muito grande ou infinita, $\tilde{Z}_{0|-1}^{-1}$ é pequena ou nula. Os resultados apresentados neste capítulo fazem parte das contribuições originais deste trabalho.

3.1 Resultados Preliminares

Serão apresentados nesta seção resultados auxiliares que serão utilizados na dedução do filtro para SLSSM na forma de informação para o sistema definido na Seção 2.3. Considere as seguintes definições para $i \geq 0$ e $k \in \{1, \dots, N\}$,

$$\begin{aligned} z_i &:= \begin{bmatrix} z_{i,1}^T & \dots & z_{i,N}^T \end{bmatrix}^T \in \mathbb{R}^{Nn}, \\ z_{i,k} &:= x_i 1_{\{\Theta_i=k\}} \in \mathbb{R}^n \end{aligned}$$

e $\hat{z}_{i|i-1}$ a projeção de z_i em $\mathcal{L}(y^{i-1})$ (subespaço linear medido por $y^{i-1} := (y_{i-1}^T \dots y_0^T)^T$) com

$$\tilde{z}_{i|i-1} := z_i - \hat{z}_{i|i-1}.$$

Estas variáveis são associadas com as seguintes matrizes de segundo momento

$$\begin{aligned} Z_i &:= \mathbb{E} \{ z_i z_i^T \} = \text{diag}(Z_{i,k}) \in \mathbb{R}^{Nn \times Nn}, \\ Z_{i,k} &:= \mathbb{E} \{ z_{i,k} z_{i,k}^T \} \in \mathbb{R}^{n \times n}, \quad k = 1, \dots, N \\ \hat{Z}_{i|l} &:= \mathbb{E} \{ \hat{z}_{i|l} \hat{z}_{i|l}^T \} \in \mathbb{R}^{Nn \times Nn}, \quad l \leq i \\ \tilde{Z}_{i|l} &:= \mathbb{E} \{ \tilde{z}_{i|l} \tilde{z}_{i|l}^T \} \in \mathbb{R}^{Nn \times Nn}, \quad l \leq i \end{aligned}$$

e interagem com as seguintes matrizes aumentadas

$$\mathcal{F} := \begin{bmatrix} p_{11}F_1 & \dots & p_{N1}F_N \\ \vdots & \dots & \vdots \\ p_{1N}F_1 & \dots & p_{NN}F_N \end{bmatrix} \in \mathbb{R}^{Nn \times Nn}, \quad (3.1)$$

$$\mathcal{H} := \begin{bmatrix} H_1 & \dots & H_N \end{bmatrix} \in \mathbb{R}^{m \times Nn}, \quad (3.2)$$

$$\mathcal{D}_i := \begin{bmatrix} D_1 \pi_{i,1}^{1/2} & \dots & D_N \pi_{i,N}^{1/2} \end{bmatrix} \in \mathbb{R}^{m \times Nq}, \quad (3.3)$$

$$\pi_{i,k} := \mathcal{P}(\Theta(i) = k), \quad (3.4)$$

sendo que $\text{diag}[Z_{i,k}]$ denota uma matriz formada por $Z_{i,k}$, $k = 1, \dots, N$ na diagonal e zero nas outras posições. Considere o estado estimado $\hat{x}_{i|i}$ como

$$\hat{x}_{i|i} = \sum_{j=1}^N \hat{z}_{i,j|i} \quad (3.5)$$

sendo que $\hat{z}_{i|i}$ é obtido do seguinte algoritmo recursivo

$$\hat{z}_{i|i} = \hat{z}_{i|i-1} + \tilde{Z}_{i|i-1} \mathcal{H}^T (\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T)^{-1} (y_i - \mathcal{H} \hat{z}_{i|i-1}), \quad (3.6)$$

$$\hat{z}_{i|i-1} = \mathcal{F} \hat{z}_{i-1|i-1}, \quad (3.7)$$

$$\hat{z}_{0|-1} = E(z(0)) = \begin{bmatrix} \mu_1^T & \dots & \mu_N^T \end{bmatrix}^T, \quad (3.8)$$

e $\tilde{Z}_{i|i-1} \in \mathbb{R}^{Nn \times Nn}$ são matrizes semidefinidas positivas dadas por

$$\tilde{Z}_{i|i-1} = Z_i - \hat{Z}_{i|i-1}, \quad (3.9)$$

$$\tilde{Z}_{i|i} = Z_i - \hat{Z}_{i|i}, \quad (3.10)$$

$$Z_i = \text{diag}[Z_{i,k}] \quad (3.11)$$

$$Z_{i+1,k} = \sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^T + \sum_{j=1}^N p_{jk} \pi_{i,j} G_j G_j^T, \quad (3.12)$$

$$Z_{0,k} = V_k, \quad k \in \{1, \dots, N\}. \quad (3.13)$$

A vantagem do filtro para SLSSM deduzido em (Costa e Guerra (2002)), e apresentado a seguir, é que $\tilde{Z}_{i|i-1}$ pode ser calculada diretamente como uma equação recursiva de Riccati, com um termo adicional que depende das matrizes de segundo momento $Z_{i,k}$

$$\hat{z}_{i|i} = \hat{z}_{i|i-1} + \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} (y_i - \mathcal{H} \hat{z}_{i|i-1}), \quad (3.14)$$

$$\hat{z}_{i|i-1} = \mathcal{F} \hat{z}_{i-1|i-1},$$

$$\begin{aligned} \tilde{Z}_{i+1|i} &= \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{F}^T + \mathcal{B}(Q(i)) + \text{diag} \left[\sum_{j=1}^N p_{jk} \pi_{i,j} G_j G_j^T \right] - \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \\ &\times \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T. \end{aligned} \quad (3.15)$$

Denotando

$$\mathcal{Q}_i := \mathcal{B}(Q(i)) + \text{diag} \left[\sum_{j=1}^N \pi_{i,j} p_{jk} G_j G_j^T \right] \quad (3.16)$$

sendo

$$\mathcal{B}(Q(i)) = \text{diag} \left[\sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^T \right] - \mathcal{F} (\text{diag}[Z_{i,k}]) \mathcal{F}^T \quad (3.17)$$

pode-se reescrever a equação algébrica de Riccati (3.15) como

$$\tilde{Z}_{i+1|i} = \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{F}^T + \mathcal{Q}_i - \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T. \quad (3.18)$$

A seguir serão apresentados alguns lemas auxiliares que serão utilizados na dedução do filtro na forma de informação para SLSSM.

Lema 3.1.1 *Considere matrizes A, B, C e D de dimensões apropriadas com A e D invertíveis. A seguinte igualdade é válida*

$$(A + BDC)^{-1} BD = A^{-1} B (D^{-1} + CA^{-1}B)^{-1}. \quad (3.19)$$

Prova: Utilizando o Lema (A.2) da inversão de matrizes, as seguintes igualdades são válidas

$$\begin{aligned}
(A + BDC)^{-1} BD &= A^{-1}BD - A^{-1}B(D^{-1} + CA^{-1}B)^{-1}CA^{-1}BD \\
&= A^{-1}BD - A^{-1}BD(I + CA^{-1}BD)^{-1}CA^{-1}BD \\
&= A^{-1}BD \left(I - (I + CA^{-1}BD)^{-1}CA^{-1}BD \right) \\
&= A^{-1}BD(I + CA^{-1}BD)^{-1} \\
&= A^{-1}B(D^{-1} + CA^{-1}B)^{-1}. \quad \diamond
\end{aligned}$$

Lema 3.1.2 *A equação algébrica de Riccati do filtro para SLSSM*

$$\tilde{Z}_{i+1|i} = \mathcal{F}\tilde{Z}_{i|i-1}\mathcal{F}^T + \mathcal{Q}_i - \mathcal{F}\tilde{Z}_{i|i-1}\mathcal{H}^T \left(\mathcal{H}\tilde{Z}_{i|i-1}\mathcal{H}^T + \mathcal{D}_i\mathcal{D}_i^T \right)^{-1} \mathcal{H}\tilde{Z}_{i|i-1}\mathcal{F}^T \quad (3.20)$$

pode ser reescrita na forma de informação como

$$\tilde{Z}_{i+1|i}^{-1} = \mathcal{Q}_i^{-1} - \mathcal{Q}_i^{-1}\mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1} \mathcal{F}^T \mathcal{Q}_i^{-1}. \quad (3.21)$$

Prova: Pode-se reescrever a equação (3.20) da seguinte forma

$$\tilde{Z}_{i+1|i} = \mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1} - \tilde{Z}_{i|i-1}\mathcal{H}^T \left(\mathcal{H}\tilde{Z}_{i|i-1}\mathcal{H}^T + \mathcal{D}_i\mathcal{D}_i^T \right)^{-1} \mathcal{H}\tilde{Z}_{i|i-1} \right) \mathcal{F}^T, \quad (3.22)$$

usando o Lema (A.2) da inversão de matrizes tem-se

$$\tilde{Z}_{i+1|i} = \mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T \quad (3.23)$$

e aplicando novamente o Lema da inversão (A.2) obtém-se

$$\tilde{Z}_{i+1|i}^{-1} = \mathcal{Q}_i^{-1} - \mathcal{Q}_i^{-1}\mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1} \mathcal{F}^T \mathcal{Q}_i^{-1}. \quad \diamond \quad (3.24)$$

Lema 3.1.3 *O filtro para SLSSM na forma preditora-corretora*

$$\hat{z}_{i|i} = \hat{z}_{i|i-1} + \tilde{Z}_{i|i-1}\mathcal{H}^T \left(\mathcal{H}\tilde{Z}_{i|i-1}\mathcal{H}^T + \mathcal{D}_i\mathcal{D}_i^T \right)^{-1} (y_i - \mathcal{H}\hat{z}_{i|i-1}) \quad (3.25)$$

$$\hat{z}_{i+1|i} = \mathcal{F}\hat{z}_{i|i} \quad (3.26)$$

pode ser escrito na forma preditora como

$$\begin{aligned} \hat{z}_{i+1|i} &= \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + \mathcal{F} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T \\ &\times (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i. \end{aligned} \quad (3.27)$$

Prova: Seja o filtro para SLSSM na sua forma preditora

$$\begin{aligned} \hat{z}_{i+1|i} &= \mathcal{F} \left(I - \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \right) \hat{z}_{i|i-1} + \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \\ &\times \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} y_i \end{aligned} \quad (3.28)$$

o termo de (3.28),

$$\mathcal{F} \left(I - \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \right) \hat{z}_{i|i-1} \quad (3.29)$$

pode ser reescrito como

$$\mathcal{F} \left(\tilde{Z}_{i|i-1} - \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \right) \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1}. \quad (3.30)$$

Usando o Lema (A.2) da inversão de matrizes obtém-se

$$\mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} \quad (3.31)$$

substituindo o termo (3.29) pelo (3.31) em (3.28) tem-se

$$\begin{aligned} \hat{z}_{i+1|i} &= \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \\ &\times \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} y_i \end{aligned} \quad (3.32)$$

e considere também o outro termo de (3.28),

$$\tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} y_i \quad (3.33)$$

que pode ser reescrito como

$$\tilde{Z}_{i|i-1} \mathcal{H}^T \left(I + (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T \right)^{-1} (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i \quad (3.34)$$

ou,

$$\left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i. \quad (3.35)$$

Substituindo (3.33) pelo (3.35) em (3.32) tem-se portanto

$$\begin{aligned} \hat{z}_{i+1|i} &= \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + \mathcal{F} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T \\ &\times (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i. \quad \diamond \end{aligned} \quad (3.36)$$

3.2 Filtro para SLSSM na Forma de Informação

Com os resultados apresentados na seção anterior, o filtro para SLSSM na forma de informação pode então ser deduzido.

Teorema 3.2.1 *O filtro para SLSSM é dado por*

$$\begin{aligned} \hat{z}_{i+1|i} &= \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \\ &\times \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} y_i \end{aligned} \quad (3.37)$$

e pode ser reescrito na forma de informação como

$$\begin{aligned} \tilde{Z}_{i+1|i}^{-1} \hat{z}_{i+1|i} &= \mathcal{Q}_i^{-1} \mathcal{F} \left(\mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} + \tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + \mathcal{Q}_i^{-1} \mathcal{F} \\ &\times \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1} \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i. \end{aligned} \quad (3.38)$$

Prova: Lembrando que, $\tilde{Z}_{i+1|i} = \mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T$ e multiplicando a esquerda de (3.37) por $\tilde{Z}_{i+1|i}^{-1}$ tem-se

$$\begin{aligned} \tilde{Z}_{i+1|i}^{-1} \hat{z}_{i+1|i} &= (\mathcal{Q}_i + \mathcal{F} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \mathcal{F}^T)^{-1} \mathcal{F} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \\ &\times \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} + (\mathcal{Q}_i + \mathcal{F} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \mathcal{F}^T)^{-1} \\ &\times \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T (\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i. \end{aligned} \quad (3.39)$$

Considerando o termo de (3.39),

$$\begin{aligned} & \left(\mathcal{Q}_i + \mathcal{F}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H})^{-1} \mathcal{F}^T \right)^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \\ & \times \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} \end{aligned} \quad (3.40)$$

e usando o Lema (3.1.1), pode-se reescrever o termo (3.40) como

$$\mathcal{Q}_i^{-1} \mathcal{F} \left(\mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} + \tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1}. \quad (3.41)$$

Substituindo o termo (3.40) pelo (3.41) em (3.39) tem-se

$$\begin{aligned} \tilde{Z}_{i+1|i}^{-1} \hat{z}_{i+1|i} &= \mathcal{Q}_i^{-1} \mathcal{F} \left(\mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} + \tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \tilde{Z}_{i|i-1}^{-1} \hat{z}_{i|i-1} \\ &+ \left(\mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T \right)^{-1} \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \\ &\times \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} y_i. \end{aligned} \quad (3.42)$$

Considerando o termo de (3.39),

$$\begin{aligned} & \left(\mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T \right)^{-1} \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \\ & \times y_i \end{aligned} \quad (3.43)$$

e usando novamente o Lema (3.1.1) obtém-se

$$\begin{aligned} & \left(\mathcal{Q}_i + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T \right)^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{H}^T \\ & \times (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i \end{aligned} \quad (3.44)$$

que pode ser reescrito como

$$\begin{aligned} &= \mathcal{Q}_i^{-1} \left(I + \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \mathcal{F}^T \mathcal{Q}_i^{-1} \right)^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \\ &\times \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i \\ &= \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \left(I + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} \right)^{-1} \right)^{-1} \\ &\times \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i \\ &= \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1} \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} y_i \end{aligned} \quad (3.45)$$

e substituindo novamente em (3.42), obtém-se

$$\begin{aligned}\tilde{Z}_{i+1|i}^{-1}\hat{z}_{i+1|i} &= \mathcal{Q}_i^{-1}\mathcal{F}\left(\mathcal{F}^T\mathcal{Q}_i^{-1}\mathcal{F} + \tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T(\mathcal{D}_i\mathcal{D}_i^T)^{-1}\mathcal{H}\right)^{-1}\tilde{Z}_{i|i-1}^{-1}\hat{z}_{i|i-1} + \mathcal{Q}_i^{-1}\mathcal{F} \\ &\times \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T(\mathcal{D}_i\mathcal{D}_i^T)^{-1}\mathcal{H} + \mathcal{F}^T\mathcal{Q}_i^{-1}\mathcal{F}\right)^{-1}\mathcal{H}^T(\mathcal{D}_i\mathcal{D}_i^T)^{-1}y_i. \quad \diamond \quad (3.46)\end{aligned}$$

Capítulo 4

Algoritmos *Array* para Filtragem de SLSSM

Neste capítulo serão apresentados os algoritmos *array* do filtro para SLSSM e para sua forma de informação, bem como exemplos numéricos ilustrativos. O objetivo como já foi dito é apresentar alternativas mais robustas para o cálculo desses filtros de SLSSMs. Os resultados apresentados neste capítulo fazem parte das contribuições originais deste trabalho.

4.1 Algoritmo *Array* do Filtro para SLSSM

Segue abaixo a dedução do algoritmo *array* do filtro para SLSSM.

Algoritmo 4.1.1 *A equação de Riccati (3.15), utilizada para o cálculo do erro de estimativa do filtro para SLSSM, pode ser calculada de maneira alternativa através do algoritmo array definido de acordo com o seguinte procedimento*

Passo 1: Calcular as condições iniciais $Z_{0,j}^{1/2} = V_j^{1/2}$ com $j = 1, \dots, N$; $Z_0^{1/2} = \text{diag}[Z_{0,j}]^{1/2}$ e $\tilde{Z}_{0,j}^{1/2} = (\xi(0)\xi(0)^T)^{1/2}$.

Passo 2: Calcular $\tilde{Z}_{i+1|i}^{1/2}$ utilizando uma matriz J -unitária Λ_{J_1} (matriz que satisfaz a seguinte condição: $\Lambda_{J_1} J \Lambda_{J_1}^T = J$ sendo J uma matriz diagonal cujos elementos são $+1$ e -1) de dimensões

apropriadas

$$\begin{aligned}
 & \begin{bmatrix} \mathcal{H}\tilde{Z}_{i|i-1}^{1/2} & (\mathcal{D}_i\mathcal{D}_i^T)^{1/2} & 0 & 0 \\ \mathcal{F}\tilde{Z}_{i|i-1}^{1/2} & 0 & \mathcal{Z}_{i+1} & \mathcal{F}\mathcal{Z}_i \end{bmatrix} \Lambda_{J_1} \\
 &= \begin{bmatrix} \left(\mathcal{H}^T\tilde{Z}_{i|i-1}\mathcal{H} + \mathcal{D}_i\mathcal{D}_i^T\right)^{1/2} & 0 & 0 & 0 \\ \mathcal{F}\tilde{Z}_{i|i-1}\mathcal{H}^T \left(\mathcal{H}\tilde{Z}_{i|i-1}\mathcal{H}^T + \mathcal{D}_i\mathcal{D}_i^T\right)^{-1/2} & \tilde{Z}_{i+1|i}^{1/2} & 0 & 0 \end{bmatrix}
 \end{aligned} \tag{4.1}$$

sendo que $Z_i = \mathcal{Z}_i\mathcal{Z}_i^T = \text{diag}[Z_{i,k}]$ e $\mathcal{Z}_i$ é dada por

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \tag{4.2}$$

e

$$\begin{aligned}
 \mathcal{L}_k &= \begin{bmatrix} L_{1k} & L_{2k} & \cdots & L_{Nk} \end{bmatrix}, \quad \mathcal{M}_k = \begin{bmatrix} M_{1k} & M_{2k} & \cdots & M_{Nk} \end{bmatrix} \\
 L_{jk} &= p_{jk}^{1/2} F_j Z_{i,j}^{1/2}, \quad M_{jk} = p_{jk}^{1/2} \pi_{i,j}^{1/2} G_j.
 \end{aligned}$$

$Z_{i,j}^{1/2}$ pode ser calculada usando uma matriz unitária Λ_z tal que

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \Lambda_z = \begin{bmatrix} Z_{i+1,1}^{1/2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & Z_{i+1,2}^{1/2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Z_{i+1,N}^{1/2} & 0 \end{bmatrix}. \tag{4.3}$$

Prova: Para calcular $\tilde{Z}_{i+1|i}$ é necessário calcular $Z_i = \mathcal{Z}_i\mathcal{Z}_i^T = \text{diag}[Z_{i,k}]$ com,

$$Z_{i+1,k} = \sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^T + \sum_{j=1}^N \pi_{i,j} p_{jk} G_j G_j^T. \tag{4.4}$$

Considere Z_i calculada em (4.2), assim usando o Lema (A.4.1), $Z_{i,j}^{1/2}$ pode ser calculada através de uma matriz unitária Λ_z tal que

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \Lambda_z = \begin{bmatrix} Z_{i+1,1}^{1/2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & Z_{i+1,2}^{1/2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Z_{i+1,N}^{1/2} & 0 \end{bmatrix} \quad (4.5)$$

pois se cada matriz acima for multiplicada pela sua transposta, chega-se ao conjunto de equações dado em (4.4) que caracteriza Z_i , ou seja, uma relação do tipo $AA^T = BB^T$. O algoritmo *array* para o cálculo de $\tilde{Z}_{i+1|i}$ é definido da seguinte maneira. Considere a equação de Riccati do filtro para SLSSM

$$\begin{aligned} \tilde{Z}_{i+1|i} &= \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{F}^T + \text{diag} \left[\sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^T + \sum_{j=1}^N \pi_{i,j} p_{jk} G_j G_j^T \right] - \mathcal{F} (\text{diag} (Z_{i,k})) \mathcal{F}^T \\ &\quad - \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T \end{aligned} \quad (4.6)$$

ou,

$$\tilde{Z}_{i+1|i} = \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{F}^T + Z_{i+1} - \mathcal{F} Z_i \mathcal{F}^T - \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T. \quad (4.7)$$

O complemento de Schur de (4.7) é dado por

$$\begin{bmatrix} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T & \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T \\ \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T & \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{F}^T + Z_{i+1} - \mathcal{F} Z_i \mathcal{F}^T \end{bmatrix}. \quad (4.8)$$

Pode-se fatorar (4.8) utilizando uma matriz assinatura J como

$$\begin{bmatrix} \mathcal{H} \tilde{Z}_{i|i-1}^{1/2} & (\mathcal{D}_i \mathcal{D}_i^T)^{1/2} & 0 & 0 \\ \mathcal{F} \tilde{Z}_{i|i-1}^{1/2} & 0 & Z_{i+1} & \mathcal{F} Z_i \end{bmatrix} J \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} \mathcal{H}^T & \tilde{Z}_{i|i-1}^{1/2} \mathcal{F}^T \\ (\mathcal{D}_i \mathcal{D}_i^T)^{T/2} & 0 \\ 0 & Z_{i+1}^T \\ 0 & Z_i^T \mathcal{F}^T \end{bmatrix} \quad (4.9)$$

sendo,

$$J = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & -I \end{bmatrix}. \quad (4.10)$$

Vamos usar a propriedade do complemento de Schur mostrada no Apêndice A.1 para encontrar uma outra fatoração para (4.8)

$$\begin{aligned} & \begin{bmatrix} \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{1/2} & 0 \\ \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1/2} & \tilde{Z}_{i+1|i}^{1/2} \end{bmatrix} \\ & \times \begin{bmatrix} \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{T/2} & 0 \\ \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-T/2} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T & \tilde{Z}_{i+1|i}^{T/2} \end{bmatrix}^T. \end{aligned} \quad (4.11)$$

Pode-se reescrever a fatoração (4.11) utilizando a matriz assinatura (4.10) da seguinte forma

$$\begin{aligned} & \begin{bmatrix} \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{1/2} & 0 & 0 \\ \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1/2} & \tilde{Z}_{i+1|i}^{1/2} & 0 \end{bmatrix} J \\ & \times \begin{bmatrix} \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{T/2} & 0 & 0 \\ \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-T/2} \mathcal{H} \tilde{Z}_{i|i-1} \mathcal{F}^T & \tilde{Z}_{i+1|i}^{T/2} & 0 \end{bmatrix}^T. \end{aligned} \quad (4.12)$$

Sendo assim pode-se dizer que existe uma transformação J -unitária Λ_{J_1} tal que

$$\begin{aligned} & \begin{bmatrix} \mathcal{H} \tilde{Z}_{i|i-1}^{1/2} & (\mathcal{D}_i \mathcal{D}_i^T)^{1/2} & 0 & 0 \\ \mathcal{F} \tilde{Z}_{i|i-1}^{1/2} & 0 & \mathcal{Z}_{i+1} & \mathcal{F} \mathcal{Z}_i \end{bmatrix} \Lambda_{J_1} \\ & = \begin{bmatrix} \left(\mathcal{H}^T \tilde{Z}_{i|i-1} \mathcal{H} + \mathcal{D}_i \mathcal{D}_i^T \right)^{1/2} & 0 & 0 & 0 \\ \mathcal{F} \tilde{Z}_{i|i-1} \mathcal{H}^T \left(\mathcal{H} \tilde{Z}_{i|i-1} \mathcal{H}^T + \mathcal{D}_i \mathcal{D}_i^T \right)^{-1/2} & \tilde{Z}_{i+1|i}^{1/2} & 0 & 0 \end{bmatrix}. \quad \diamond \end{aligned} \quad (4.13)$$

4.2 Algoritmo *Array* do Filtro para SLSSM na Forma de Informação

A seguir será apresentada a dedução do algoritmo *array* do filtro para SLSSM na forma de informação.

Algoritmo 4.2.1 *A equação de Riccati (3.21), utilizada para o cálculo do erro de estimativa do filtro para SLSSM na forma de informação, pode ser calculada de maneira alternativa através do algoritmo array definido de acordo com o seguinte procedimento:*

Passo 1: Calcular as condições iniciais $Z_{0,j}^{-1/2} = V_j^{-1/2}$ com $j = 1, \dots, N$; $Z_0^{-1/2} = \text{diag}[Z_{0,j}]^{-1/2}$ e $\tilde{Z}_{0|-1}^{-1/2} = (\varsigma(0)\varsigma(0)^T)^{-1/2}$.

Passo 2: Calcular $\tilde{Z}_{i+1|i}^{-1/2}$ utilizando uma matriz unitária Λ tal que:

$$\begin{aligned} & \begin{bmatrix} \tilde{Z}_{i|i-1}^{-1/2} & \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1/2} & \mathcal{F}^T \mathcal{Q}_i^{-1/2} \\ 0 & 0 & \mathcal{Q}_i^{-1/2} \end{bmatrix} \Lambda \\ &= \begin{bmatrix} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{1/2} & 0 & 0 \\ \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1/2} & \tilde{Z}_{i+1|i}^{-1/2} & 0 \end{bmatrix} \end{aligned} \quad (4.14)$$

sendo, $\mathcal{Q}_i = Z_{i+1} - \mathcal{F}^T Z_i \mathcal{F}$, pode ser calculado através de Z_i que é dado como no Passo 2 do algoritmo (4.1.1).

Prova: Para calcular $\tilde{Z}_{i+1|i}^{-1}$ é necessário calcular $\mathcal{Q}_i$. Seja,

$$\mathcal{Q}_i = \mathcal{B}(Q(i)) + \text{diag} \left[\sum_{j=1}^N \pi_{i,j} p_{jk} G_j G_j^T \right] \quad (4.15)$$

que pode ser reescrito como

$$\mathcal{Q}_i = \text{diag} \left[\sum_{j=1}^N p_{ij} F_j Z_{i,k} F_j^T + \sum_{j=1}^N \pi_{i,j} p_{jk} G_j G_j^T \right] - \mathcal{F} (\text{diag}(Z_{i,k})) \mathcal{F}^T, \quad (4.16)$$

$$\mathcal{Q}_i = \text{diag}(Z_{i,k+1}) - \mathcal{F} (\text{diag}(Z_{i,k})) \mathcal{F}^T, \quad (4.17)$$

$$\mathcal{Q}_i = Z_{i+1} - \mathcal{F}Z_i\mathcal{F}^T. \quad (4.18)$$

sendo $Z_i = \text{diag}(Z_{i,k}) = \mathcal{Z}_i\mathcal{Z}_i^T$.

Considere $\mathcal{Z}_i$ como em (4.2), assim usando o Lema (A.4.1) $Z_{i,j}^{1/2}$ pode ser calculada usando uma matriz unitária Λ_z tal que

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \Lambda_z = \begin{bmatrix} Z_{i+1,1}^{1/2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & Z_{i+1,2}^{1/2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Z_{i+1,N}^{1/2} & 0 \end{bmatrix} \quad (4.19)$$

pois se cada matriz acima for multiplicada pela sua transposta, chega-se ao conjunto de equações dado em (4.4) que caracteriza Z_i , ou seja, uma relação do tipo $AA^* = BB^*$. Agora será construído o algoritmo *array* para o cálculo de $\tilde{Z}_{i+1|i}^{-1}$. Considere a equação de Riccati do filtro para SLSSM abaixo:

$$\tilde{Z}_{i+1|i}^{-1} = \mathcal{Q}_i^{-1} - \mathcal{Q}_i^{-1}\mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{-1} \mathcal{F}^T \mathcal{Q}_i^{-1}. \quad (4.20)$$

O complemento de Schur da equação de Riccati do filtro para SLSSM na forma de informação (4.20) é dado por

$$\begin{bmatrix} \tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} & \mathcal{F}^T \mathcal{Q}_i^{-1} \\ \mathcal{Q}_i^{-1} \mathcal{F} & \mathcal{Q}_i^{-1} \end{bmatrix} \quad (4.21)$$

fatorando o complemento de Schur (4.21) tem-se

$$\begin{bmatrix} \tilde{Z}_{i|i-1}^{-1/2} & \mathcal{H}^T (\mathcal{D}_i\mathcal{D}_i^T)^{-1/2} & \mathcal{F}^T \mathcal{Q}_i^{-1/2} \\ 0 & 0 & \mathcal{Q}_i^{-1/2} \end{bmatrix} \begin{bmatrix} \tilde{Z}_{i|i-1}^{-T/2} & 0 \\ (\mathcal{D}_i\mathcal{D}_i^T)^{-T/2} \mathcal{H} & 0 \\ \mathcal{Q}_i^{-T/2} \mathcal{F} & \mathcal{Q}_i^{-T/2} \end{bmatrix}. \quad (4.22)$$

Usando a propriedade do complemento de Schur mostrada no Apêndice A.1 será realizada a seguinte fatoração alternativa:

$$\begin{aligned}
 & \begin{bmatrix} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{1/2} & 0 & 0 \\ \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i \mathcal{F} \right)^{-1/2} & \tilde{Z}_{i+1|i}^{-1/2} & 0 \end{bmatrix} \\
 & \times \begin{bmatrix} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{T/2} & 0 & 0 \\ \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i \mathcal{F} \right)^{-T/2} \mathcal{F}^T \mathcal{Q}_i^{-T} & \tilde{Z}_{i+1|i}^{-T/2} & 0 \end{bmatrix}^T.
 \end{aligned} \tag{4.23}$$

Sendo assim pode-se dizer que existe uma transformação unitária Λ tal que

$$\begin{aligned}
 & \begin{bmatrix} \tilde{Z}_{i|i-1}^{-1/2} & \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1/2} & \mathcal{F}^T \mathcal{Q}_i^{-1/2} \\ 0 & 0 & \mathcal{Q}_i^{-1/2} \end{bmatrix} \Lambda \\
 & = \begin{bmatrix} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i^{-1} \mathcal{F} \right)^{1/2} & 0 & 0 \\ \mathcal{Q}_i^{-1} \mathcal{F} \left(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^T (\mathcal{D}_i \mathcal{D}_i^T)^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}_i \mathcal{F} \right)^{-1/2} & \tilde{Z}_{i+1|i}^{-1/2} & 0 \end{bmatrix}. \quad \diamond
 \end{aligned} \tag{4.24}$$

4.3 Exemplos Numéricos

Para o cálculo dos algoritmos *array* desenvolvidos neste capítulo, utiliza-se como exemplo o SLSSM de dois estados:

Estado 1:

$$F_1 = \begin{bmatrix} 0.7 & 0 \\ 0.1 & 0.1 \end{bmatrix}, H_1 = \begin{bmatrix} 10^{-5} & 0 \end{bmatrix}, G_1 = \begin{bmatrix} 1.95 \times 10^{-4} & 0 \\ 0 & 2.7 \times 10^{-4} \end{bmatrix}, D_1 = 1.6$$

Estado 2:

$$F_2 = \begin{bmatrix} 0.6 & 0 \\ 0.1 & 0.2 \end{bmatrix}, H_2 = \begin{bmatrix} 10^{-5} & 0 \end{bmatrix}, G_2 = \begin{bmatrix} 1.95 \times 10^{-4} & 0 \\ 0 & 2.7 \times 10^{-4} \end{bmatrix}, D_2 = 1.6$$

Matriz de Probabilidade de Transição:

$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$$

e $\pi_{i,1} = 0.9$, $\pi_{i,2} = 0.1$ para $i = 1, \dots, 16$ e $\pi_{i,1} = 0.1$, $\pi_{i,2} = 0.9$ para $i = 17, \dots, 30$. $\pi_{i,k}$ foram redefinidos em $i = 17$ de maneira a testar a robustez do algoritmo *array*, veja Figura 4.1. Foram calculados os valores singulares de $\tilde{Z}_{i|i-1}$ e $\tilde{Z}_{i|i-1}^{-1}$ para três diferentes implementações: ponto flutuante, ponto fixo para equações de Riccati explícitas e para algoritmos *array*. O cálculo com ponto flutuante foi utilizado como referência e a precisão dos outros dois cálculos foi definida através do cálculo do erro médio quadrático definido da seguinte forma

$$E_\gamma = \sqrt{\sum_{i=1}^N (VSf_{i,\gamma} - VS_{i,\gamma})^2 / N} \quad (4.25)$$

sendo que γ define um dos quatro valores singulares calculados para cada simulação, VSf é o valor singular calculado usando ponto flutuante, VS é o valor singular calculado usando ponto fixo, tanto para as equações de Riccati quanto para os algoritmos *array*, e i é o instante de tempo. Foi usada uma arquitetura de ponto fixo em 16-bits que pode representar um número de -65.543 a 65.543 . Estas implementações foram feitas via MatLab através do *fix-point Simulink toolbox*. Pode-se notar na figura (4.1), a vantagem do algoritmo *array* perante a implementação por equações de Riccati explícitas. Na implementação em ponto fixo das equações de Riccati ocorreram erros numéricos. Com o algoritmo *array*, o resultado do cálculo em ponto fixo foi equivalente ao resultado obtido para ponto flutuante. A Figura (4.2) mostra os resultados obtidos para o filtro de informação. São calculadas as matrizes inversas da covariância do erro de estimativa para as três implementações definidas acima. Veja que em virtude da redução dos valores calculados, estão na faixa de -65.543 a 65.543 , os resultados para as três implementações são equivalentes. As equações utilizadas neste exercício são (3.20), (3.21), (4.1), (4.3) e (4.14).

Tipo	Implementação	E ₁	E ₂	E ₃	E ₄
SLSSM	Equações	0.8827	0.9530	0.3415	0.1419
	Array	0.0307	0.0197	0.0123	0.0075
SLSSM	Equações	5.2866×10^{-4}	6.6319×10^{-4}	1.5901×10^{-4}	1.0968×10^{-4}
Informação	Array	7.7242×10^{-4}	0.0013	1.8468×10^{-4}	3.1219×10^{-4}

Tabela 4.1: Erro médio quadrático para o cálculo dos valores singulares de $\tilde{Z}_{i|i-1}$ e $\tilde{Z}_{i|i-1}^{-1}$.

Figura 4.1: Valores Singulares de $\tilde{Z}_{i|i-1}$.

Figura 4.2: Valores Singulares de $\tilde{Z}_{i|i-1}^{-1}$.

Capítulo 5

Algoritmos *Array* Rápidos para Filtragem de Sistemas Singulares

Neste capítulo serão deduzidos os algoritmos *array* rápidos para filtros nominais e robustos de sistemas singulares discretos no tempo. Para o problema de filtragem, o objetivo é deduzir a expressão de estimativa ótima $\hat{x}_{i|i}$ em conjunto com a solução de uma equação recursiva de Riccati $P_{i|i}$, sendo que a estimativa linear ótima $\hat{x}_{i|i}$ é calculada levando-se em consideração as medidas de y_0 até y_i . Para o problema de predição, o objetivo é deduzir $\hat{x}_{i+1|i}$ em conjunto com $P_{i+1|i}$, sendo que $\hat{x}_{i+1|i}$ é calculada levando-se em consideração as medidas de y_0 até y_i . Portanto, neste capítulo serão deduzidos os respectivos algoritmos *array* rápidos para as versões nominais filtrada e preditora bem como as versões robustas filtrada e preditora de sistemas singulares na forma de informação. Os resultados apresentados neste capítulo fazem parte das contribuições originais deste trabalho.

5.1 Estimativa Filtrada Nominal

Nesta seção será deduzido o algoritmo *array* rápido para estimativa filtrada nominal na forma de informação.

Algoritmo 5.1.1 *Seja a equação de Riccati*

$$P_{i|i}^{-1} = E^T Q^{-1} E + H^T R^{-1} H - E^T Q^{-1} F (P_{i-1|i-1}^{-1} + F^T Q^{-1} F)^{-1} F^T Q^{-1} E \quad (5.1)$$

utilizada para o cálculo do erro de estimativa do filtro singular na forma de informação

$$P_{i|i}^{-1} \hat{x}_{i|i} = E^T Q^{-1} F (P_{i-1|i-1}^{-1} + F^T Q^{-1} F)^{-1} P_{i-1|i-1}^{-1} \hat{x}_{i-1|i-1} + H^T R^{-1} y_i \quad (5.2)$$

sendo E , Q , F , H e R matrizes conhecidas de dimensões apropriadas do sistema singular nominal definido na Seção 2.4. Esta equação pode ser calculada de maneira alternativa através do algoritmo *array rápido* definido pelo seguinte procedimento

Passo 1: Calcular as condições iniciais

$$\begin{aligned} P_{0|0}^{-1} &= \Pi_0 \\ P_{1|1}^{-1} &= E^T Q^{-1} E + H^T R^{-1} H - E^T Q^{-1} F (\Pi_0 + F^T Q^{-1} F)^{-1} F^T Q^{-1} E. \end{aligned} \quad (5.3)$$

Passo 2: Calcular M_i utilizando uma matriz unitária θ_i de dimensões apropriadas

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix} \quad (5.4)$$

sendo

$$\begin{aligned} M_i S M_i^T &= P_{i+1|i+1}^{-1} - P_{i|i}^{-1} \quad (S = I) \\ R_{e,i} &= P_{i|i}^{-1} + F^T Q^{-1} F \\ \bar{K}_{p,i} &= E^T Q^{-1} F R_{e,i}^{-1/2}. \end{aligned} \quad (5.5)$$

Prova: A prova do filtro (5.2) pode ser vista em detalhes na referência (Terra et al. (2007)). O objetivo desta prova é mostrar a dedução do algoritmo *array rápido* (5.4). Considere a seguinte equação

$$P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1} = M_{i+1} S M_{i+1}^T \quad (5.6)$$

sendo S uma matriz identidade, é denominada matriz assinatura. Reescrevendo (5.6), tem-se

$$\begin{aligned} P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1} &= -P_{i+1|i+1}^{-1} + E^T Q^{-1} E + H^T R^{-1} H - E^T Q^{-1} F (P_{i+1|i+1}^{-1} + F^T Q^{-1} F)^{-1} \\ &\times F^T Q^{-1} E. \end{aligned} \quad (5.7)$$

O complemento de Schur de $P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1}$ é dado por

$$\begin{bmatrix} P_{i+1|i+1}^{-1} + F^T Q^{-1} F & F^T Q^{-1} E \\ E^T Q^{-1} F & -P_{i+1|i+1}^{-1} + E^T Q^{-1} E + H^T R^{-1} H \end{bmatrix} \quad (5.8)$$

fatorando a matriz (5.8) tem-se

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \begin{bmatrix} R_{e,i}^{T/2} & \bar{K}_{p,i}^T \\ M_i^T & 0 \end{bmatrix}. \quad (5.9)$$

Pode-se de maneira alternativa fatorar (5.8) (veja o Apêndice A.1) da seguinte forma

$$\begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ E^T Q^{-1} F R_{e,i+1}^{-1/2} & M_{i+1} \end{bmatrix} \begin{bmatrix} R_{e,i+1}^{T/2} & F^T Q^{-1} E R_{e,i+1}^{-T/2} \\ 0 & M_{i+1}^T \end{bmatrix}. \quad (5.10)$$

Assim pode-se afirmar que existe uma matriz unitária θ_i tal que

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix}. \quad \diamond \quad (5.11)$$

5.2 Estimativa Preditora Nominal

Nesta seção será deduzido o algoritmo *array* rápido para estimativa preditora nominal na forma de informação.

Algoritmo 5.2.1 *Seja a equação de Riccati*

$$P_{i+1|i}^{-1} = E^T Q^{-1} E - E^T Q^{-1} F (P_{i|i-1}^{-1} + H^T R^{-1} H + F^T Q^{-1} F)^{-1} F^T Q^{-1} E \quad (5.12)$$

utilizada para o cálculo do erro de estimativa do filtro singular na forma de informação

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E^T (Q^{-1} + F (P_{i|i-1}^{-1} + H^T R^{-1} H) F^T)^{-1} F (P_{i|i-1}^{-1} + H^T R^{-1} H)^{-1} \\ &\times (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + H R^{-1} y_i) \end{aligned} \quad (5.13)$$

sendo E , Q , F , H e R matrizes conhecidas de dimensões apropriadas do sistema singular nominal definido na Seção 2.4. Esta equação pode ser calculada de maneira alternativa através do algoritmo *array* rápido definido pelo seguinte procedimento

Passo 1: Calcular as condições iniciais

$$\begin{aligned} P_{0|-1}^{-1} &= \Pi_0 \\ P_{1|0}^{-1} &= E^T Q^{-1} E - E^T Q^{-1} F (\Pi_0 + H^T R^{-1} H + F^T Q^{-1} F)^{-1} F^T Q^{-1} E. \end{aligned} \quad (5.14)$$

Passo 2: Calcular M_i utilizando uma matriz unitária θ_i de dimensões apropriadas

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix} \quad (5.15)$$

sendo

$$\begin{aligned} M_i S M_i^T &= P_{i+1|i}^{-1} - P_{i|i-1}^{-1} \quad (S = I) \\ R_{e,i} &= P_{i|i-1}^{-1} + H^T R^{-1} H + F^T Q^{-1} F \\ \bar{K}_{p,i} &= E^T Q^{-1} F R_{e,i}^{-1/2}. \end{aligned} \quad (5.16)$$

Prova: A prova do filtro (5.13) pode ser vista em detalhes na referência (Terra et al. (2007)). O objetivo desta prova é mostrar a dedução do algoritmo *array* rápido (5.15). Considere a seguinte equação

$$P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1} = M_{i+1} S M_{i+1}^T \quad (5.17)$$

sendo S uma matriz identidade, é denominada matriz assinatura. Reescrevendo (5.17), tem-se

$$\begin{aligned} P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1} &= -P_{i+1|i}^{-1} + E^T Q^{-1} E - E^T Q^{-1} F (P_{i+1|i}^{-1} + H^T R^{-1} H + F^T Q^{-1} F)^{-1} \\ &\quad \times F^T Q^{-1} E. \end{aligned} \quad (5.18)$$

O complemento de Schur de $P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1}$ é dado por

$$\begin{bmatrix} P_{i+1|i}^{-1} + H^T R^{-1} H + F^T Q^{-1} F & F^T Q^{-1} E \\ E^T Q^{-1} F & -P_{i+1|i}^{-1} + E^T Q^{-1} E \end{bmatrix} \quad (5.19)$$

fatorando a matriz (5.19) tem-se

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \begin{bmatrix} R_{e,i}^{T/2} & \bar{K}_{p,i}^T \\ M_i^T & 0 \end{bmatrix}. \quad (5.20)$$

De maneira alternativa pode-se fatorar (5.19) da seguinte maneira (veja o Apêndice A.1)

$$\begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ E^T Q^{-1} F R_{e,i+1}^{-1/2} & M_{i+1} \end{bmatrix} \begin{bmatrix} R_{e,i+1}^{T/2} & F Q^{-1} F^T R_{e,i+1}^{T/2} \\ 0 & M_{i+1}^T \end{bmatrix}. \quad (5.21)$$

Assim pode-se afirmar que existe uma matriz unitária θ_i tal que

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix}. \quad \diamond \quad (5.22)$$

5.3 Estimativa Filtrada Robusta

Nesta seção vamos deduzir um algoritmo *array* rápido para a estimativa filtrada robusta na forma de informação de sistemas singulares sujeitos a incertezas paramétricas na forma

$$(E + \delta E)x_{i+1} = (F + \delta F)x_i + w_i, \quad i = 0, 1, \dots \quad (5.23)$$

$$y_i = (E + \delta E)x_i + v_i \quad (5.24)$$

sendo δE , δF e δH perturbações nas matrizes nominais do sistema. Veja detalhes deste sistema na Seção 2.4.

Algoritmo 5.3.1 *Seja a equação de Riccati*

$$\begin{aligned} P_{i+1|i+1}^{-1} &= E^T \hat{Q}^{-1} E - E^T \hat{Q}^{-1} F (P_{i|i}^{-1} + \hat{\lambda}^{T/2} N_f^T N_f \hat{\lambda}^{1/2} + F^T \hat{Q}^{-1} F)^{-1} F^T \hat{Q}^{-1} E \\ &+ H^T \hat{R}^{-1} H + \hat{\lambda} [N_h^T N_h + N_e^T N_e] \end{aligned} \quad (5.25)$$

utilizada para o cálculo do erro de estimativa do filtro robusto para sistemas singulares na forma de informação

$$\begin{aligned} P_{i+1|i+1}^{-1} \hat{x}_{i+1|i+1} &= E^T (\hat{Q}^{-1} - F (P_{i|i}^{-1} + \hat{\lambda} N_f^T N_f)^{-1} F^T)^{-1} F (P_{i|i}^{-1} + \hat{\lambda} N_f^T N_f)^{-1} P_{i|i}^{-1} \hat{x}_{i|i} \\ &+ H^T \hat{R}^{-1} y_{i+1} \end{aligned} \quad (5.26)$$

sendo o parâmetro $\hat{\lambda}$ escolhido no intervalo

$$\hat{\lambda} > \lambda_l := \left\| \begin{bmatrix} M_f^T & 0 \\ 0 & M_h^T \end{bmatrix} \begin{bmatrix} Q^{-1} & 0 \\ 0 & R^{-1} \end{bmatrix} \begin{bmatrix} M_f & 0 \\ 0 & M_h \end{bmatrix} \right\| \quad (5.27)$$

$$\hat{Q}^{-1} := Q^{-1} + Q^{-1} M_f (\hat{\lambda} I - M_f^T Q^{-1} M_f)^{-1} M_f^T Q^{-1} \quad (5.28)$$

$$\hat{R}^{-1} := R^{-1} + R^{-1} M_h (\hat{\lambda} I - M_h^T R^{-1} M_h)^{-1} M_h^T R^{-1} \quad (5.29)$$

e M_f , M_h , N_f , N_h , N_e , E , F , H são matrizes de dimensões apropriadas. Esta equação pode ser calculada de maneira alternativa através do algoritmo *array rápido* definido pelo seguinte procedimento

Passo 1: Calcular as condições iniciais

$$\begin{aligned} P_{0|0}^{-1} &= \Pi_0 \\ P_{1|1}^{-1} &= E^T \hat{Q} E - E^T \hat{Q}^{-1} F (\Pi_0 + \hat{\lambda}^{T/2} N_f^T N_f \hat{\lambda}^{1/2} + F^T \hat{Q}^{-1} F)^{-1} F^T \hat{Q}^{-1} E \\ &\quad + H^T \hat{R}^{-1} H + \hat{\lambda} [N_h^T N_h + N_e^T N_e]. \end{aligned} \quad (5.30)$$

Passo 2: Calcular M_i utilizando uma matriz unitária θ_i de dimensões apropriadas

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix} \quad (5.31)$$

sendo

$$\begin{aligned} M_i S M_i^T &= P_{i+1|i+1}^{-1} - P_{i|i}^{-1} \quad (S = I) \\ R_{e,i} &= P_{i|i}^{-1} + \hat{\lambda}^{T/2} N_f^T N_f \hat{\lambda}^{1/2} + F^T \hat{Q}^{-1} F \\ \bar{K}_{p,i} &= E^T \hat{Q}^{-1} F (P_{i|i}^{-1} + \hat{\lambda}^{1/2} N_f^T N_f \hat{\lambda}^{T/2} + F^T \hat{Q}^{-1} F)^{-1/2}. \end{aligned} \quad (5.32)$$

Prova: A prova do filtro (5.26) pode ser vista em detalhes na referência (Terra et al. (2007)). O objetivo desta prova é mostrar a dedução do algoritmo *array rápido* (5.31). Considere a seguinte

equação

$$P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1} = M_{i+1} S M_{i+1}^T \quad (5.33)$$

sendo S uma matriz identidade, é denominada matriz assinatura. Reescrevendo (5.33), tem-se

$$\begin{aligned} P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1} &= -P_{i+1|i+1}^{-1} + E^T \hat{Q} E - E^T \hat{Q}^{-1} F (P_{i+1|i+1}^{-1} + \hat{\lambda}^{1/2} N_f^T N_f \hat{\lambda}^{T/2} + F^T \hat{Q}^{-1} F)^{-1} \\ &\times F^T \hat{Q}^{-1} E + H^T \hat{R}^{-1} H + \hat{\lambda} [N_h^T N_h + N_e^T N_e]. \end{aligned} \quad (5.34)$$

O complemento de Schur de $P_{i+2|i+2}^{-1} - P_{i+1|i+1}^{-1}$ é dado por

$$\begin{bmatrix} P_{i+1|i+1}^{-1} + \hat{\lambda}^{T/2} N_f^T N_f \hat{\lambda}^{1/2} + F^T \hat{Q}^{-1} F & F^T \hat{Q}^{-1} E \\ E^T \hat{Q}^{-1} F & -P_{i+1|i+1} + X \end{bmatrix} \quad (5.35)$$

sendo

$$X = E^T \hat{Q}^{-1} E + H^T \hat{R}^{-1} H + \hat{\lambda} [N_h^T N_h + N_e^T N_e] \quad (5.36)$$

fatorando a matriz (5.35) tem-se

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \begin{bmatrix} R_{e,i}^{T/2} & \bar{K}_{p,i}^T \\ M_i^T & 0 \end{bmatrix}. \quad (5.37)$$

Fatorando (5.35) de maneira alternativa (veja o Apêndice A.1), obtém-se

$$\begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ E^T \hat{Q}^{-1} F R_{e,i+1}^{-1/2} & M_{i+1} \end{bmatrix} \begin{bmatrix} R_{e,i+1}^{T/2} & R_{e,i+1}^{-T/2} F^T \hat{Q}^{-1} E \\ 0 & M_{i+1}^T \end{bmatrix}. \quad (5.38)$$

Assim pode-se afirmar que existe uma matriz unitária θ_i tal que

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix}. \quad \diamond \quad (5.39)$$

5.4 Estimativa Preditora Robusta

Nesta seção será deduzido o algoritmo *array* rápido para estimativa preditora robusta na forma de informação.

Algoritmo 5.4.1 *Seja a equação de Riccati*

$$P_{i+1|i}^{-1} = \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{E} - \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} (P_{i|i-1}^{-1} + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F})^{-1} \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{E} \quad (5.40)$$

utilizada para o cálculo do erro de estimativa do filtro para sistemas singulares na forma de informação

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= \mathcal{E}^T (\mathcal{Q}^{-1} + \mathcal{F} (P_{i|i-1}^{-1} + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H}) \mathcal{F}^T)^{-1} \mathcal{F} (P_{i|i-1}^{-1} + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H})^{-1} \\ &\times (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + \mathcal{H} \mathcal{R}^{-1} y_i) \end{aligned} \quad (5.41)$$

sendo o parâmetro $\hat{\lambda}$ escolhido no intervalo

$$\hat{\lambda} > \lambda_l := \left\| \begin{bmatrix} M_f^T & 0 \\ 0 & M_h^T \end{bmatrix} \begin{bmatrix} Q^{-1} & 0 \\ 0 & R^{-1} \end{bmatrix} \begin{bmatrix} M_f & 0 \\ 0 & M_h \end{bmatrix} \right\|$$

$$\mathcal{Q}^{-1} := \begin{bmatrix} \hat{Q}^{-1} & 0 \\ 0 & I \end{bmatrix}, \mathcal{R}^{-1} := \begin{bmatrix} \hat{R}^{-1} & 0 \\ 0 & I \end{bmatrix},$$

$$\mathcal{E} := \begin{bmatrix} E \\ \sqrt{\hat{\lambda}} N_e \end{bmatrix}, \mathcal{F} := \begin{bmatrix} F \\ \sqrt{\hat{\lambda}} N_f \end{bmatrix},$$

$$\mathcal{H} := \begin{bmatrix} H \\ \sqrt{\hat{\lambda}} N_h \end{bmatrix} \quad (5.42)$$

$\hat{Q}^{-1}$ e $\hat{R}^{-1}$ são descritos pelas equações (5.28) e (5.29) respectivamente, da Seção 5.3 e M_f , M_h , N_f , N_h , N_e , E , F , H são matrizes de dimensões apropriadas. Esta equação pode ser calculada de maneira alternativa através do algoritmo *array* rápido definido pelo seguinte procedimento

Passo 1: Calcular as condições iniciais

$$\begin{aligned} P_{0|-1}^{-1} &= \Pi_0 \\ P_{1|0}^{-1} &= \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{E} - \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} (\Pi_0 + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F})^{-1} \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{E}. \end{aligned} \quad (5.43)$$

Passo 2: Calcular M_i utilizando uma matriz unitária θ_i de dimensões apropriadas

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix} \quad (5.44)$$

sendo

$$\begin{aligned} M_i S M_i^T &= P_{i+1|i}^{-1} - P_{i|i-1}^{-1} \quad (S = I) \\ R_{e,i} &= P_{i|i-1}^{-1} + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F} \\ \bar{K}_{p,i} &= \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} R_{e,i}^{-1/2}. \end{aligned} \quad (5.45)$$

Prova: A prova do filtro (5.41) pode ser vista em detalhes na referência (Terra et al. (2007)). O objetivo desta prova é mostrar a dedução do algoritmo *array* rápido (5.44). Considere a seguinte equação

$$P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1} = M_{i+1} S M_{i+1}^T \quad (5.46)$$

sendo S uma matriz identidade, é denominada matriz assinatura. Reescrevendo (5.46), tem-se

$$\begin{aligned} P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1} &= -P_{i+1|i} + \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{E} - \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} (P_{i+1|i}^{-1} + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H} + \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F})^{-1} \\ &\times \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{E}. \end{aligned} \quad (5.47)$$

O complemento de Schur de $P_{i+2|i+1}^{-1} - P_{i+1|i}^{-1}$ é dado por

$$\begin{bmatrix} R_{e,i+1} & \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{E} \\ \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} & -P_{i+1|i}^{-1} + \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{E} \end{bmatrix} \quad (5.48)$$

fatorando a matriz (5.48) tem-se que

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \begin{bmatrix} R_{e,i}^{T/2} & \bar{K}_{p,i}^T \\ M_i^T & 0 \end{bmatrix}. \quad (5.49)$$

De maneira alternativa pode-se fatorar (5.48) (veja o Apêndice A.1) da seguinte forma

$$\begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \mathcal{E}^T \mathcal{Q}^{-1} \mathcal{F} R_{e,i+1}^{-1/2} & M_{i+1} \end{bmatrix} \begin{bmatrix} R_{e,i+1}^{T/2} & R_{e,i+1}^{-T/2} \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{E} \\ 0 & M_{i+1}^T \end{bmatrix}. \quad (5.50)$$

Assim pode-se afirmar que existe uma matriz unitária θ_i tal que

$$\begin{bmatrix} R_{e,i}^{1/2} & M_i \\ \bar{K}_{p,i} & 0 \end{bmatrix} \theta_i = \begin{bmatrix} R_{e,i+1}^{1/2} & 0 \\ \bar{K}_{p,i+1} & M_{i+1} \end{bmatrix}. \quad \diamond \quad (5.51)$$

5.5 Exemplos Numéricos

Para o cálculo dos algoritmos *array* rápidos desenvolvidos neste capítulo, utiliza-se como exemplo o sistema singular dado pelas seguintes matrizes

$$E = \begin{bmatrix} 1.14 & 0 & 0 \\ 0 & 1.17 & 0 \\ 0 & 0 & 0 \end{bmatrix}, F = \begin{bmatrix} 0.97 & 0 & 0 \\ 0.27 & -0.78 & 0 \\ 0.12 & 0.12 & 0.68 \end{bmatrix}, H = \begin{bmatrix} 0.11 & 0 & 0 \\ 0 & 0.56 & 0 \\ 0 & 0 & 0.32 \end{bmatrix},$$

$$Q = \begin{bmatrix} 0.04 & 0 & 0 \\ 0 & 0.06 & 0 \\ 0 & 0 & 0.05 \end{bmatrix}, R = \begin{bmatrix} 0.09 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 0.03 \end{bmatrix}, N_e = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$N_f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.3 \end{bmatrix}, N_h = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.03 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}, M_f = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix},$$

$$M_h = \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.8 & 0 \\ 0 & 0 & 0.8 \end{bmatrix}, \hat{\lambda} = 40.$$

Foram calculados os valores singulares de $P_{i|i}^{-1}$ para três diferentes implementações: ponto flutuante, ponto fixo para equações de Riccati explícitas e para algoritmos *array* rápidos. O filtro robusto utilizado neste exemplo é calculado através da equação de Riccati (5.25) e pelo algoritmo *array* (5.31). O cálculo com ponto flutuante foi utilizado como referência e a precisão dos outros dois cálculos foi definida através do cálculo do erro médio quadrático definido da seguinte forma

$$E_\gamma = \sqrt{\sum_{i=1}^N (VSf_{i,\gamma} - VS_{i,\gamma})^2 / N} \quad (5.52)$$

sendo que γ define um dos três valores singulares calculados para cada simulação, VSf é o valor singular calculado usando ponto flutuante, VS é o valor singular calculado usando ponto fixo, tanto para as equações de Riccati quanto para os algoritmos *array* rápidos, e i é o instante de tempo. Foi usada uma arquitetura de ponto fixo em 16-bits que pode representar um número de -65.543 a 65.543 . Essas implementações foram feitas via MatLab através do *fix-point Simulink toolbox*. Pode-se notar na Figura (5.1) a vantagem do algoritmo *array* rápido em comparação com a equação de Riccati explícita. Na implementação em ponto fixo utilizando equação de Riccati ocorreram erros numéricos. Com o algoritmo *array* rápido, o resultado do cálculo em ponto fixo foi equivalente ao resultado obtido para ponto flutuante.

Tipo	Implementação	E ₁	E ₂	E ₃
Robusta	Equações	0.4770	0.3262	0.1371
	Array	0.4765	0.1778	0.0650

Tabela 5.1: Erro médio quadrático para o cálculo dos valores singulares de $P_{i|i}^{-1}$.

Figura 5.1: Valores Singulares da Estimativa Filtrada Robusta $P_{i|i}^{-1}$.

Capítulo 6

Conclusão

Neste trabalho foi apresentado o estudo feito sobre implementações computacionais alternativas para o problema de estimativa utilizando algoritmos *array*, algoritmos *array* rápidos e filtros na forma de informação. Tais alternativas foram usadas para o cálculo do filtro para SLSSM deduzido em (Costa e Guerra (2002)) e para o filtros para sistemas singulares convencionais formulados em (Terra et al. (2007)). O objetivo foi usar as estimativas *array* e *array* rápido, bem como filtros de informação, como uma maneira de aliviar alguns problemas computacionais associados à equação recursiva de Riccati. Os algoritmos *array* minimizam problemas causados por erros de arredondamento que podem tornar a matriz de covariância P_i não-Hermitiana. Tanto os algoritmos *array* quanto os filtros de informação reduzem a faixa dinâmica dos cálculos feitos em implementações por aritmética de ponto fixo.

As implementações dos algoritmos mostraram através de gráficos a robustez numérica destas alternativas de implementação.

Referências Bibliográficas

- Anderson, N. e J. Moore (1979). *Optimal Filtering*. Prentice-Hall, New Jersey.
- Babak Hassibi, T. K. e A. H. Sayed (2000). Array algorithms for h^∞ estimation. *IEEE Transactions on Automatic Control* 45(4), 702–706.
- Costa, O., M. Fragoso, e R. Marques (2005). *Discrete-Time Markov Jump Linear Systems*. Springer.
- Costa, O. e S. Guerra (2002). Stationary filter for linear minimum mean square error estimation of discrete-time markovian jump systems. *IEEE Transaction on Automatic Control* 47(8), 1351–1356.
- Dyer, P. e S. McReynolds (1969). Extension of square-root filtering to include process noise. *J. of Optimization Theory and Applications* 3, 444–459.
- Hassibi, B., A. Sayed, e T. Kailath (1999). *Indefinite-Quadratic Estimation and Control: A Unified Approach to H_2 and H_∞ Theories*, Volume I. SIAM Studies in Applied and Numerical Mathematics.
- Jazwinski, A. (1970). *Stochastic processes and Filtering Theory*. New York: Academic.
- Kailath, T., S. A. e B. Hassibi (2000). *Linear Estimation*. Prentice-Hall.
- Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Trans. of the ASME Series D*(82), 35–45.
- Kaminski, P.G., B. A. e S. Schmidt (1971). Discrete square-root filtering: A survey of current techniques. *IEEE Transactions on Automatic Control* 16(6), 727–735.
- Morf, M. e T. Kailath (1975). Square-root algorithms for least-squares estimation. *IEEE Transactions on Automatic Control* 20(4), 487–497.

- Morf, M., S. G. e T. Kailath (1974). Some new algorithms for recursive estimation in constant, linear, discrete-time systems. *IEEE Transactions on Automatic Control* 19(4), 315–323.
- Potter, J. E. e R. G. Stern (1963). Statistical filtering of space navigation measurements. *Proc. AIAA Guidance Cintr. Conf.*
- Sayed, A. H. e T. Kailath (1994). Extend chandrasekhar recursions. *IEEE Transactions on Automatic Control* 39(3), 619–623.
- Singh, S. P. e R. W. Liu (1973). Existence of state equation representation if linear large-scale dynamic systems. *IEEE Transactions on Circuit Theory* 20(3), 239–246.
- Terra, M. H., J. Y. Ishihara, e A. Padoan (2007). Information filtering and array algorithms for descriptor systems subject to parameter uncertainties. *IEEE Transactions on Signal Processing* 55(1), 1–9.
- Verhaegen, M. e P. Van Dooren (1986). numerical aspects of different kalman filter implemeta-tions. *IEEE Transactions on Automatic Control* 31(10), 907–917.

Apêndice A

Resultados Matriciais Importantes

A.1 Complemento de Schur

Considere o bloco de matrizes M:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad (\text{A.1})$$

e a seguinte matriz multiplicada à esquerda de M

$$\begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A & B \\ XA + C & XB + D \end{bmatrix}. \quad (\text{A.2})$$

Para $X = -CA^{-1}$ tem-se

$$\begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & \Delta A \end{bmatrix} \quad (\text{A.3})$$

sendo $\Delta A = D - CA^{-1}B$ denominado complemento de Schur de A em M. Similarmente pode-se obter:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} I & -A^{-1}B \\ 0 & I \end{bmatrix} = \begin{bmatrix} A & 0 \\ C & \Delta A \end{bmatrix}. \quad (\text{A.4})$$

Estes resultados podem ser combinados para diagonalizar M em blocos

$$\begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} I & -A^{-1}B \\ 0 & I \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & \Delta A \end{bmatrix} \quad (\text{A.5})$$

e utilizando a seguinte propriedade de invertibilidade,

$$\begin{bmatrix} I & 0 \\ P & I \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ -P & I \end{bmatrix} \quad (\text{A.6})$$

pode-se achar a fatoraçaõ,

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & \Delta A \end{bmatrix} \begin{bmatrix} I & A^{-1}B \\ 0 & I \end{bmatrix}. \quad (\text{A.7})$$

A.2 Lema da Inversão de Matrizes

O lema da inversão de matrizes é dado pelas seguintes relações:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \quad (\text{A.8})$$

ou, para C não invertível,

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(I + CDA^{-1}B)^{-1}CDA^{-1}. \quad (\text{A.9})$$

A.3 Transformações Unitárias

A.3.1 Transformações de Householder

As transformações de Householder zeram várias entradas de uma linha de uma só vez. Considera-se neste texto apenas matrizes com entradas de números reais. Para um vetor linha n -dimensional x , supõe-se que se queira zerar várias entradas utilizando uma matriz simétrica e ortogonal θ , ou seja, transformar x na forma:

$$\begin{bmatrix} x_1 & x_2 & \cdots & x_{n-1} \end{bmatrix} \theta = \alpha e_0 \quad (\text{A.10})$$

sendo α um escalar real a ser determinado e e_0 o primeiro vetor base $e_0 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$.

O escalar α não pode ser arbitrário e de fato ele deve ser escolhido *a priori* e antes de determinar θ . Por causa da ortogonalidade e de (A.10) segue que $x\theta\theta^Tx^T = xx^T = \|x\|^2$. Então deve-se ter $\alpha = \pm \|x\|$. Ambos os valores de α são possíveis. Uma maneira de calcular θ é através

$$\theta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/\sqrt{1+\rho^2} & 0 & -\rho/\sqrt{1+\rho^2} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \rho/\sqrt{1+\rho^2} & 0 & 1/\sqrt{1+\rho^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.14})$$

tem-se então que $x\theta = \begin{bmatrix} \times & \times & \alpha & \times & \times & 0 & \times & \times \end{bmatrix}$. Para tringularizar uma matriz A qualquer, deve-se realizar uma série de transformações como em (A.14) para zerar os elementos de interesse da matriz A linha a linha.

A.3.3 Rotações de Givens Hiperbólicas

Rotações de Givens Hiperbólicas são matrizes de rotação 2×2 que preservam a "norma-J", sendo portanto uma transformação J-ortogonal. Uma transformação J-ortogonal θ é uma matriz que satisfaz:

$$\theta J \theta^T = \theta^T J \theta = J \quad (\text{A.15})$$

sendo J uma matriz assinatura, ou seja, uma matriz diagonal com entradas ± 1

$$J = (I_p \oplus -I_q), p \geq 1, q \geq 1. \quad (\text{A.16})$$

Transformações J-unitárias preservam a "norma quadrática J" de um vetor, ou seja, se $y = x\theta$, então

$$\|y\|_J^2 = y J y^T = x \theta J \theta^T x^T = x J x^T = \|x\|_J^2. \quad (\text{A.17})$$

No caso de uma matriz de rotação hiperbólica $\theta_{2 \times 2}$, J é dado por $J = (1 \oplus -1)$. Dado um vetor real 1×2 $x = \begin{bmatrix} a & b \end{bmatrix}$, deseja-se encontrar tal rotação hiperbólica que faça a seguinte transformação

$$x = \begin{bmatrix} a & b \end{bmatrix} \theta = \begin{bmatrix} \alpha & 0 \end{bmatrix}, \quad (\text{A.18})$$

sendo α um número real a ser determinado. Neste caso nem sempre é possível fazer tal transformação, pois pode ocorrer de $\|x\|_J^2 = a^2 - b^2$ ser negativo. Assim quando a norma-J for negativa será possível fazer a seguinte transformação

$$x\theta = \begin{bmatrix} a & b \end{bmatrix} \theta = \begin{bmatrix} 0 & \alpha \end{bmatrix}. \quad (\text{A.19})$$

Assim, pode-se distinguir dois casos:

1. $|a| > |b|$: A expressão para calcular a matriz de rotação hiperbólica θ que satisfaz (A.18)

é dada por:

$$\theta = \sqrt{1 - \rho^2} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \quad (\text{A.20})$$

sendo $\rho = b/a, a \neq 0$ o que gera

$$\begin{bmatrix} a & b \end{bmatrix} \theta = \begin{bmatrix} \pm\sqrt{a^2 - b^2} & 0 \end{bmatrix}. \quad (\text{A.21})$$

2. $|b| > |a|$: a expressão para calcular a matriz de rotação hiperbólica θ que satisfaz (A.19) é dado por:

$$\theta = 1/\sqrt{1 - \rho^2} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \quad (\text{A.22})$$

sendo $\rho = a/b, b \neq 0$ o que gera

$$\begin{bmatrix} a & b \end{bmatrix} \theta = \begin{bmatrix} 0 & \pm\sqrt{b^2 - a^2} \end{bmatrix}. \quad (\text{A.23})$$

A.4 Teoremas Auxiliares

A.4.1 Rotações de Bases

Lema A.4.1 *Kailath e Hassibi (2000)* Dadas duas matrizes $n \times m$ com $n \leq m$ A e B , a igualdade $AA^T = BB^T$ é válida se e apenas se existir uma matriz unitária $m \times m$ θ ($\theta\theta^T = I = \theta^T\theta$) tal que $A = B\theta$.

A.4.2 Transformações J-unitárias

Lema A.4.2 *Kailath e Hassibi (2000)* Sejam duas matrizes $n \times m$ com $n \leq m$ A e B de posto pleno, e seja $J = (I_p \oplus -I_q)$ uma matriz onde $p + q = m$. A relação $AJA^T = BJB^T$ permanece se e apenas se existir uma única matriz J-unitária $m \times m$ θ tal que $A = B\theta$.