Campus de São Carlos

ALGORITMOS ARRAY PARA FILTRAGEM DE SISTEMAS
SINGULARES

Autor: Antonio Carlos Padoan Junior

Orientador: Prof. Dr. Marco Henrique Terra

UNIVERSIDADE DE SÃO PAULO

**ESCOLA DE ENGENHARIA
DE SÃO CARLOS**

Serviço de Pós-Graduação EESC/USP
EXEMPLAR REVISADO
Data de entrada no Serviço.....31/08/05.....
Ass.:..... <i>leson</i>

Algoritmos Array para Filtragem de Sistemas Singulares

Antonio Carlos Padoan Junior

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

Orientador: Prof. Dr. Marco Henrique Terra

São Carlos
2005

DEDALUS - Acervo - EESC

31100052275

Class.	TESE-EESC
Cott.	4339
Tombo	T205/05
Sysno	1460836

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

P124a Padoan Junior, Antonio Carlos
Algoritmos Array para filtragem de sistemas
singulares / Antonio Carlos Padoan Junior. -- São Carlos,
2005.

Dissertação (Mestrado) -- Escola de Engenharia
de São Carlos-Universidade de São Paulo, 2005.
Área: Engenharia Elétrica.
Orientador: Prof. Dr. Marco Henrique Terra.

1. Filtro de Kalman. 2. Sistemas singulares.
3. Algoritmos Array. 4. Filtragem de informação.
I. Título.

FOLHA DE JULGAMENTO

Candidato: Engenheiro **ANTONIO CARLOS PADOAN JUNIOR**

Dissertação defendida e julgada em 24-06-2005 perante a Comissão Julgadora:

Prof. Associado **MARCO HENRIQUE TERRA (Orientador)**
(Escola de Engenharia de São Carlos/USP)

aprovado

Prof. Dr. **JOÃO YOSHIYUKI ISHIHARA**
(Universidade de Brasília/UnB)

aprovado

Prof. Dr. **EDUARDO FONTOURA COSTA**
(Instituto de Ciências Matemáticas e de Computação/USP)

aprovado

Prof. Associado **HOMERO SCHIABEL**
Vice-Coordenador em Exercício do Programa de Pós-Graduação
em Engenharia Elétrica

Profa. Titular **MARIA DO CARMO CALIJURI**
Presidente da Comissão de Pós-Graduação

Dedicatória

Dedico esta dissertação a minha família.

Agradecimentos

Agradeço a minha família, pelo suporte dado ao longo desta jornada.

Ao meu orientador, Prof. Dr. Marco Henrique Terra, principalmente por sua perseverança no desenvolvimento deste trabalho.

Ao Prof. Dr. João Ishihara cujo apoio foi muito precioso.

Também não posso deixar de agradecer todas as pessoas que fizeram parte da minha vida ao longo deste mestrado, colegas de laboratório, funcionários do departamento, colegas de república, pessoal do CAASO e ex-colegas de graduação.

Resumo

Esta dissertação apresenta novos resultados para a solução de problemas de implementação computacional na estimativa de sistemas singulares e sistemas Markovianos. São apresentados algoritmos alternativos para problemas de filtragem de maneira a minimizar problemas causados principalmente por erros de arredondamento e mal condicionamento de matrizes. O trabalho envolve basicamente algoritmos *array* e filtragem de informação para a estimativa de sistemas singulares nominais e robustos. Também é deduzido um algoritmo *array* para a filtragem de sistemas lineares sujeitos a saltos Markovianos.

Palavras-chaves: Filtro de Kalman, sistemas singulares, algoritmos *array*, filtragem de informação.

Abstract

This dissertation presents new results to solve computational implementation problems to estimate singular and Markovian systems. Alternative algorithms to handle computational filtering errors due rounding errors and ill-conditioned matrices are developed. This dissertation comprehends basically array algorithms and information filters for the estimate of nominal and robust singular systems. Also, it is developed an array algorithm for Markovian jump linear systems filtering.

Key-words: Kalman filtering, singular systems, array algorithms, information filtering.

Sumário

Resumo	iii
Abstract	iv
Lista de Figuras	vii
1 Introdução	1
1.1 Motivação	1
1.2 Estrutura do Texto	2
2 Revisão Bibliográfica	3
2.1 Filtro de Kalman no Espaço de Estados	3
2.2 Algoritmos <i>Array</i>	5
2.2.1 Algoritmo de Potter	7
2.2.2 Erros de Arredondamento	9
2.3 Exemplos de <i>Array</i>	11
2.3.1 Atualizações temporais	11
2.3.2 Atualizações da Medida	12
2.4 Sistemas Singulares	13
2.4.1 Filtro Singular Robusto	15
3 Filtragem de Sistemas Singulares	19
3.1 Estimativa Filtrada Nominal	19
3.2 Estimativa Preditora Nominal	23
3.3 Estimativa Filtrada Robusta	27
3.4 Estimativa Preditora Robusta	29
3.5 Implementação	30
4 Algoritmo Array para Sistemas Markovianos	34
4.1 Filtro Markoviano	35

4.1.1	Implementação	41
5	Algoritmo de Paige	43
5.1	Estrutura de covariância	45
5.2	O Algoritmo de Paige	48
5.3	Implementação	53
6	Conclusão	56
	Referências Bibliográficas	57
A	Resultados Matriciais Importantes	61
A.1	Complemento de Schur	61
A.2	Lema da Inversão de Matrizes	62
B	Transformações Unitárias	63
B.1	Transformações de Householder	63
B.2	Transformações de Givens	65
B.3	Rotações de Givens Hiperbólicas	66
C	Teoremas Auxiliares	68
C.1	Rotações de bases	68
C.2	Filtro Robusto para Sistemas Singulares CAMPOS (2004)	68
C.3	Transformações J-unitárias	71

Lista de Figuras

2.1	Diagrama de blocos exemplificando a utilização do Filtro de Kalman. . .	4
2.2	Ciclo de atualizações do filtro de Kalman	5
2.3	Transformação unitária (T.U.) no Algoritmo <i>Array</i>	7
3.1	Valores Singulares de $P_{i i}^{-1}$ para a implementação robusta.	33
4.1	Cadeia de Markov com $N = 2$	36
4.2	Valores Singulares de $\tilde{Z}_{i i-1}$	42
5.1	(a) Estimativa dos estados e (b) valores singulares de $P_{i,i}$, calculados através da equação nominal do filtro de Kalman para sistemas singulares e do algoritmo apresentado no Teorema 5.2.1.	55
B.1	Interpretação geométrica da rotação de Householder	64

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 com sucesso em áreas tais como aeronáutica, processamento em sinais de voz e controle de processos industriais. Pode-se definir um filtro de Kalman em três categorias distintas que consistem em suavização, filtragem e predição dos estados de um sistema.

À medida que as aplicações foram se diversificando, alguns inconvenientes para o filtro de Kalman foram surgindo. Pode-se exemplificar problemas de convergência devido a falta de fidelidade dos algoritmos numéricos ou modelagem não apropriada dos sistemas em consideração JAZWINSKI (1970). Para contornar estes problemas, foram sendo desenvolvidos novos algoritmos para diferentes implementações do filtro de Kalman. Em particular podem ser citados os algoritmos *array*, introduzidos por Potter KAILATH et al. (2000), filtros de informação ANDERSON & MOORE (1979) KAILATH et al. (2000) e o algoritmo desenvolvido em PAIGE (1985). Estes algoritmos aumentam a eficiência numérica nas implementações dos filtros devido ao uso de transformações ortogonais nos cálculos. Estas transformações ortogonais são mais estáveis numericamente se comparadas às equações de Riccati utilizadas nas implementações dos filtros de Kalman. O contraponto destes algoritmos alternativos é a carga computacional maior dos cálculos a serem feitos KAMINSKI et al. (1971).

Neste trabalho, as contribuições originais são as deduções de novos filtros de in-

formação (onde se propaga P_i^{-1}) e algoritmos *array* para filtros de Kalman de sistemas singulares ISHIHARA & TERRA (2001), ISHIHARA & TERRA (2002), ISHIHARA et al. (2004b), CAMPOS (2004), DAI (1989) e GERDIN (2004) e também um algoritmo *array* para o filtro deduzido em COSTA (1994), para sistemas lineares sujeitos a saltos Markovianos (que neste trabalho serão denominados sistemas Markovianos, apenas por facilidade de terminologia). Filtros para sistemas Markovianos são usados quando o sistema pode ter mudanças abruptas em seus parâmetros devido a falhas, mudanças ambientais ou outros motivos. Até onde pode-se identificar na literatura, este é o primeiro algoritmo *array* para filtros de sistemas Markovianos encontrado.

O uso e análise de sistemas singulares têm recebido destaque na literatura devido a frequência com que aparecem em problemas de modelagem de sistemas econômicos LUENBERGER (1977), modelamento de imagem HASSAN & SADJANI (1995), robótica MILLS & GOLDENBERG (1989) e outros. A filtragem de informação pode ser usada quando se tem pouca informação sobre as condições iniciais do sistema, o que resulta em uma matriz de covariância do erro muito grande ou até infinita.

Também é feita a extensão neste trabalho do algoritmo de filtragem para sistemas no espaço de estados deduzido em PAIGE (1985) para sistemas singulares, onde se adota duas estratégias de estimativa, filtragem de covariância e filtragem de informação.

1.2 Estrutura do Texto

Este texto está organizado da seguinte forma: No Capítulo 2 é feita uma revisão bibliográfica dos principais tópicos abordados por esta pesquisa

- *Filtragem de Kalman convencional;*
- *Algoritmos Array para filtros de Kalman;*
- *Filtragem de Kalman para Sistemas Descritores e as versões robustas.*

No Capítulo 3, são deduzidos filtros de informação, através das equações de Riccati, e os respectivos algoritmos *array* para os filtros robustos e nominais de sistemas singulares, também serão apresentados exemplos numéricos. Nos capítulos 4 e 5 são deduzidos algoritmos *array* para sistemas Markovianos e apresentada a versão para sistemas singulares do filtro deduzido em PAIGE (1985).

Capítulo 2

Revisão Bibliográfica

2.1 Filtro de Kalman no Espaço de Estados

Em 1960, R. E. Kalman publicou um artigo que descreve uma solução recursiva ao problema de filtragem linear de sinais discretos KALMAN (1960). Desde então, devido em grande parte aos avanços da computação digital, o filtro de Kalman tem sido tema de intensas pesquisas e aplicações. O filtro de Kalman é um conjunto de equações matemáticas que fornece uma solução computacional (recursiva) eficiente para a estimativa de sistemas, baseada fundamentalmente na otimização de um funcional quadrático. Este filtro é bastante eficiente em diversos aspectos: suporta estimativas de estados passados, atuais e futuros, ou seja, suavização, filtragem e predição.

De acordo com KAILATH et al. (2000), mostra-se a seguir um problema de estimativa de estados, resolvido através da minimização do erro médio quadrático da estimativa, para as equações

$$\begin{cases} x_{i+1} &= F_i x_i + G_i u_i \\ y_i &= H_i x_i + v_i \end{cases} \quad (2.1)$$

sendo as variáveis aleatórias gaussianas u_i, v_i, x_0 de média zero e variâncias

$$\mathcal{E} \left(\begin{bmatrix} u_i \\ v_i \\ x_0 \end{bmatrix} \begin{bmatrix} u_j \\ v_j \\ x_0 \end{bmatrix}^* \right) = \begin{bmatrix} Q_i \delta_{ij} & S_i \delta_{ij} & 0 \\ S_i^* \delta_{ij} & R_i \delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix} \quad (2.2)$$

e as matrizes $\{F_i, G_i, H_i, \Pi_0, Q_i, S_i, R_i\}$ são conhecidas. O objetivo é estimar os estados x_i (não diretamente observáveis) de um sistema modelado por (2.1) tendo como referência as medidas ruidosas y_i . A Figura 2.1 mostra uma utilização do filtro de Kalman.

Figura 2.1: Diagrama de blocos exemplificando a utilização do Filtro de Kalman.

O filtro de Kalman para prever os estados em um passo para x_i , com $\hat{x}_i = \hat{x}_{i|i-1}$, sendo conhecido o vetor de saída do sistema $\{y_0, \dots, y_{i-1}\}$, pode ser calculado recursivamente através das equações

$$\hat{x}_0 = 0, \quad \hat{x}_{i+1} = F_i \hat{x}_i + K_{p,i} e_i = F_{p,i} \hat{x}_i + K_{p,i} y_i, \quad i \geq 0, \quad (2.3)$$

sendo

$$\begin{aligned} e_i &= y_i - H_i \hat{x}_i, \quad F_{p,i} = F_i - K_{p,i} H_i \\ K_{p,i} &= K_i R_{e,i}^{-1}, \quad K_i = F_i P_i H_i^* + G_i S_i, \\ R_{e,i} &= R_i + H_i P_i H_i^* \end{aligned} \quad (2.4)$$

e P_i pode ser calculada via equação de diferenças de Riccati

$$\begin{aligned} P_{i+1} &= F_i P_i F_i^* + G_i Q_i G_i^* - K_{p,i} R_{e,i}^{-1} K_{p,i}^*, \\ P_0 &= \Pi_0 \geq 0, \quad i \geq 0. \end{aligned} \quad (2.5)$$

Equivalentemente, o estimador de estados $\hat{x}_i$ pode ser atualizado de maneira alternativa tal que envolva atualizações do tempo e de medida. Adiante têm-se os conjuntos de equações que completam os dois ciclos de cada iteração. A Figura 2.2 exemplifica o processo.

- Atualizações da medida

$$\hat{x}_{i|i} = \hat{x}_i + K_{f,i} e_i, \quad (2.6)$$

Figura 2.2: Ciclo de atualizações do filtro de Kalman

$$P_{i|i} = P_{i|i-1} - P_{i|i-1} H_i^* R_{e,i}^{-1} H_i P_{i|i-1}, \quad (2.7)$$

- Atualizações no tempo

$$\hat{x}_{i+1} = F_i \hat{x}_{i|i} + G_i S_i R_{e,i}^{-1} e_i \quad (2.8)$$

$$\begin{aligned} P_{i+1|i} = & F_i P_{i|i} F_i^* + G_i (Q_i - S_i R_{e,i}^{-1} S_i^*) G_i^* \\ & - F_i K_{f,i} S_i^* G_i^* - G_i S_i K_{f,i}^* F_i^* \end{aligned} \quad (2.9)$$

sendo

$$K_{f,i} = P_i H_i^* R_{e,i}^{-1}. \quad (2.10)$$

2.2 Algoritmos *Array*

Uma forma de implementar equações de Riccati para problemas de estimativa é através de métodos que utilizam algoritmos *array*. Esses métodos foram originalmente introduzidos na literatura como uma maneira de aliviar alguns problemas computacionais associados à equação recursiva de Riccati usada no filtro de Kalman. Também apresentam outras vantagens, como redução da faixa dinâmica dos números calculados em implementações por aritmética de pontos-fixos, possibilitando a utilização de uma faixa maior de parâmetros sem que os cálculos fiquem comprometidos pelos limites de representação dos pontos fixos, e cálculo mais seguro da matriz de covariância do erro de estimativa, que pode apresentar erros de arredondamento que tornam a matriz P_i não-Hermitiana. Deve-se notar que verificar a cada iteração do algoritmo tais propri-

idades é muito custoso computacionalmente. É desejável portanto, assegurar que P_i seja sempre definida não-negativa. Para controlar esta propriedade, não se propaga P_i mas um fator *raiz-quadrada* de P_i , ou seja, uma matriz A_i tal que $P_i = A_i A_i^*$. Apesar da possibilidade de haver erros de arredondamento em A_i , o produto dos fatores multiplicados $\hat{P}_i = \hat{A}_i \hat{A}_i^*$ é sempre uma matriz (semi) definida positiva.

Os algoritmos *array* têm sido apresentados na literatura para calcular filtros de Kalman de sistemas no espaço de estados. Neste trabalho, apresentam-se tais algoritmos para o cálculo da equação de Riccati para a filtragem de sistemas singulares, que é uma classe de sistemas dinâmicos onde pode existir singularidade em sua formulação. Esses filtros serão descritos nas próximas seções.

Uma equação de Riccati para o filtro de Kalman no espaço de estados é dada pela Equação (2.5) como um exemplo de equação recursiva

$$P_{i+1} = F_i P_i F_i^* + G_i Q_i G_i^* - K_{p,i} R_{e,i}^{-1} K_{p,i}^*.$$

Nesta equação a matriz de covariância P_i é propagada a cada iteração em função de valores já conhecidos F_i , G_i , Q_i , $K_{p,i}$ e $R_{e,i}$. Resolvendo por *array*, a equação não é explicitamente calculada no computador e um fator raiz quadrada $P_i^{1/2}$ é propagado a cada iteração ao invés de P_i . A propagação de $P_i^{1/2}$ evita uma série de problemas computacionais causados por erros de arredondamento.

Um *fator raiz quadrada* de uma matriz $n \times n$ semi-definida positiva P é definido como uma matriz $n \times n$ A tal que $P = AA^*$. Estas raízes quadradas não são únicas. Sendo Θ uma matriz unitária, ou seja, $\Theta\Theta^* = \Theta^*\Theta = I$, então pode-se deduzir que $A\Theta$ também é um *fator raiz quadrada* de P . Este fator pode vir a ser único se impusermos restrições adicionais como P sendo triangular ou *Hermitiana*. Na realidade, o fator *Hermitiano* é a raiz quadrada verdadeira pois $P = AA^* = A^2$ e podemos escrever $A = P^{1/2}$. Na maioria das aplicações é conveniente escolher este fator como sendo triangular inferior. Para facilitar a notação usa-se neste texto $P^{1/2}$ para denotar *fator raiz quadrada* triangular inferior a não ser quando devidamente comentado. Também têm-se as seguintes convenções

$$P^{*/2} \triangleq (P^{1/2})^*, \quad P^{-1/2} \triangleq (P^{1/2})^{-1}, \quad P^{-*/2} \triangleq (P^{-1/2})^*.$$

Pode-se escrever

$$P = P^{1/2} P^{*/2}, \quad P^{-1} = P^{-*/2} P^{-1/2}.$$

Em geral, os fatores raiz-quadrada de uma equação recursiva podem ser propagados por algoritmos *array* da seguinte forma

1. Forma-se um determinado *pre-array* de números baseados nos dados do tempo i .
2. Este *pre-array* é reduzido para uma forma específica (geralmente triangular) por uma seqüência de operações unitárias elementares (rotações e reflexões).
3. Os valores desejados do tempo $i + 1$ podem ser imediatamente lidos a partir dos chamados *post-array*.
4. Nenhuma equação explícita é necessária. O processo está ilustrado na Figura 2.3

$$\underbrace{\begin{bmatrix} \text{Matriz} \\ \text{Pre - Array} \end{bmatrix}}_{\text{Matriz triangular}} \xrightarrow{\text{T.U.}} \underbrace{\begin{bmatrix} \text{Matriz} \\ \text{Post - Array} \end{bmatrix}}_{\text{Matriz solução}}$$

Figura 2.3: Transformação unitária (T.U.) no Algoritmo *Array*.

Um exemplo apresentado na literatura para ilustrar os efeitos de erros numéricos no cálculo da equação de Riccati, pode ser encontrado na Seção 2.2.2.

2.2.1 Algoritmo de Potter

Potter (KAILATH et al. (2000)) foi o primeiro que introduziu o conceito de algoritmo *array* para o problema de atualização da medida do filtro de Kalman no espaço de estados, 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. Nesta seção aborda-se rapidamente este algoritmo. As seguintes notações serão adotadas a seguir

$$H_i \triangleq h_i, \text{ um vetor coluna, } R_i \triangleq r(i), R_{e,i} \triangleq r_e(i), \text{ ambos escalares.}$$

A equação recursiva de Riccati para atualização da medida é dada em (2.7) e pode ser reescrita para o caso unidimensional como

$$P_{i|i} = P_{i|i-1}^{1/2} (I - r_e^{-1}(i) a_i a_i^*) P_{i|i-1}^{*/2} \quad (2.11)$$

sendo,

$$a_i = P_{i|i-1}^{*/2} h_i^* \text{ e } r_e(i) = r(i) + a_i^* a_i.$$

Potter notou que se pode fatorar

$$I - r_e^{-1}(i) a_i a_i^* = (I - \gamma(i) a_i a_i^*) (I - \gamma(i) a_i a_i^*) \quad (2.12)$$

escolhendo $\gamma(i)$ tal que

$$-r_e^{-1} = -2\gamma(i) + \gamma^2(i) (a_i a_i^*). \quad (2.13)$$

Isto resulta na seguinte fórmula de atualização

$$P_{i|i}^{1/2} = P_{i|i-1}^{1/2} (I - \gamma(i) a_i a_i^*). \quad (2.14)$$

Resolvendo a equação quadrática em (2.13) tem-se

$$\begin{aligned} \gamma(i) &= \frac{1 \pm \sqrt{1 - r_e^{-1}(i) (a_i^* a_i)}}{(a_i^* a_i)} \\ &= \frac{1 \pm \sqrt{r(i) r_e^{-1}(i)}}{r_e(i) - r(i)} \\ &= \frac{1}{\sqrt{r_e(i)} (\sqrt{r_e(i)} \pm \sqrt{r(i)})}. \end{aligned} \quad (2.15)$$

Potter escolheu a resposta com o sinal de + e finalmente obteve a seguinte equação recursiva para a atualização da covariância do erro

$$P_{i|i}^{1/2} = P_{i|i-1}^{1/2} \left[I - \frac{a_i a_i^*}{\sqrt{r_e(i)} (\sqrt{r_e(i)} \pm \sqrt{r(i)})} \right]. \quad (2.16)$$

Desta forma foi elaborada a primeira equação recursiva a propagar o fator raiz quadrada de P_i e o início da utilização de algoritmos *array* em filtragem de sistemas.

2.2.2 Erros de Arredondamento

Um exemplo que tem sido utilizado na literatura para justificar o uso de algoritmos *array* será apresentado a seguir. Considere o seguinte sistema

$$F_i = 1, H_i = 1, G_i = 0, R_i = 1, S_i = 0.$$

Neste caso P é escalar, sendo definida da seguinte forma

$$P_{i|i} = P_{i|i-1} - \frac{P_{i|i-1}^2}{1 + P_{i|i-1}}, \quad (2.17)$$

que é equivalente a

$$P_{i|i} = \frac{P_{i|i-1}}{1 + P_{i|i-1}}. \quad (2.18)$$

Considerando que em um tempo particular i , o valor de $P_{i|i-1}$ é tão grande que em precisão finita vale a seguinte aproximação $1 + P_{i|i-1} \approx P_{i|i-1}$. Assumindo que o termo (2.17) seja implementado como

$$P_{i|i} = P_{i|i-1} - \frac{P_{i|i-1}^2}{1 + P_{i|i-1}}. \quad (2.19)$$

Então, o valor de $P_{i|i}$ calculado através desta última equação é *zero*. Porém com a fórmula (2.18) o cálculo fica sendo $P_{i|i} = 1$, que é o valor correto. Verifica-se que duas implementações diferentes da mesma equação levam a resultados totalmente distintos em virtude de erros de arredondamento.

Este exemplo ilustra o mérito de se estudar uma maneira alternativa de se implementar o filtro de Kalman. Em particular para este exemplo, sugere-se o método via algoritmo *array* descrito a seguir para se calcular $\sqrt{P_{i|i}}$ de $\sqrt{P_{i|i-1}}$. Como já foi dito, primeiramente define-se um *pre-array* que é triangularizado para se obter a resposta do problema na matriz resultante desta triangularização

$$A = \begin{bmatrix} 1 & \sqrt{P_{i|i-1}} \\ 0 & \sqrt{P_{i|i-1}} \end{bmatrix}.$$

Através de rotações de *Givens*¹ obtém-se a seguinte matriz

¹As rotações de *Givens* são transformações unitárias, possibilitando seu uso para algoritmos *array* para triangularizar matrizes, veja mais detalhes no Apêndice B.2

$$\Theta = \frac{1}{\sqrt{1+\rho^2}} \begin{bmatrix} 1 & -\rho \\ \rho & 1 \end{bmatrix}, \quad \rho = \sqrt{P_{i|i-1}}.$$

Assumindo que neste exemplo $P_{i|i-1}$ é grande o suficiente para que $1 + P_{i|i-1} = P_{i|i-1}$ em precisão finita, então $\sqrt{1+\rho^2}$ fica sendo $\sqrt{P_{i|i-1}}$ e a multiplicação de Θ com o *pre-array* resulta em

$$\begin{bmatrix} 1 & \sqrt{P_{i|i-1}} \\ 0 & \sqrt{P_{i|i-1}} \end{bmatrix} \Theta = \begin{bmatrix} \sqrt{P_{i|i-1}} & 0 \\ \sqrt{P_{i|i-1}} & 1 \end{bmatrix}. \quad (2.20)$$

Pode-se mostrar que a entrada (2, 2) da matriz à direita da igualdade da Equação (2.20) é igual a $\sqrt{P_{i|i}}$, usando raciocínio análogo ao da Seção 2.3.2. Portanto este método de implementação leva a $P_{i|i} = 1$, que é o valor correto.

Outra maneira de se verificar a maior robustez numérica do uso de *array* é observar o número de condição da matriz P , $K(P)$ HOUSEHOLDER (1964), definido como

$$K(P) = \sigma_1/\sigma_n, \quad (2.21)$$

sendo σ_1 o maior valor singular de P e σ_n é o menor valor singular de P . Este número de condição é usado geralmente para analisar o efeito de perturbações em equações lineares, e é relevante para analisar operações numéricas em operações com a matriz de covariância. Um exemplo dado em KAMINSKI et al. (1971) mostra que usando aritmética de base 10 com p dígitos significativos e as dificuldades numéricas surgem quando $K(P)$ se aproxima de valores próximos a 10^p . A vantagem do uso de fator raiz quadrada aparece quando se relaciona $K(P)$ com $K(P^{1/2})$

$$K(P) = K(P^{1/2}P^{*/2}) = [K(P^{1/2})]^2. \quad (2.22)$$

Pode-se concluir que o número de condição de $P^{1/2}$ é a raiz quadrada de $K(P)$. Então, para operações numéricas relacionadas a P que resultam em erros numéricos quando $K(P) = 10^p$, com o algoritmo *array* esses problemas aparecerão para $K(P) = 10^{2p}$.

2.3 Exemplos de *Array*

Antes de apresentar algoritmo *array* para sistemas singulares, nesta seção há um breve resumo do uso de *array* para o filtro de Kalman no espaço de estados baseados nas equações da Seção 2.1.

2.3.1 Atualizações temporais

A equação de atualização do tempo (2.9), assumindo $S_i = 0$, é

$$P_{i+1|i} = F_i P_{i|i} F_i^* + G_i Q_i G_i^*, \quad i \geq 0. \quad (2.23)$$

Pode-se fatorar a expressão para chegar a

$$P_{i+1|i} = \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}^*.$$

Esta expressão fornece uma fatoração para P_{i+1} , porém as dimensões de

$$\begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}$$

são muito grandes, $(n \times (n + m))$ no invés de $(n \times n)$ que é a dimensão de P_i , não sendo eficiente portanto o uso direto desta fatoração no algoritmo. Para contornar esse problema, através da não-unicidade dos fatores raiz quadrada e utilizando uma matriz unitária Θ , redefine-se $P_{i+1|i}$ da seguinte forma

$$P_{i+1|i} = \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \Theta \Theta^* \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}^*$$

e escolhe-se Θ tal que

$$\begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} X & 0_{n \times m} \end{bmatrix} \quad (2.24)$$

sendo que $0_{n \times m}$ representa uma matriz de elementos zero de tamanho $n \times m$ e X representa uma matriz de dimensão apropriada. Determinando Θ , então pode-se ter

$$\begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \underbrace{\Theta \Theta^*}_I \begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix}^* = \begin{bmatrix} X & 0_{n \times m} \end{bmatrix} \begin{bmatrix} X & 0_{n \times m} \end{bmatrix}^*$$

e então,

$$F_i P_{i|i} F_i^* + G_i Q_i G_i^* = X X^*.$$

Porém como o lado esquerdo da expressão é igual a $P_{i+1|i}$, X pode ser identificado como $P_{i+1|i}^{1/2}$, um fator raiz quadrada de $P_{i+1|i}$. Finalmente tem-se o algoritmo *array* resumido da seguinte forma

$$\begin{bmatrix} F_i P_{i|i}^{1/2} & G_i Q_i^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} P_{i+1|i}^{1/2} & 0_{n \times m} \end{bmatrix}. \quad (2.25)$$

Isto resolve o problema assim que se determina uma transformação Θ apropriada. Para encontrar tais transformações unitárias Θ pode-se usar rotações de *Givens* ou reflexões de *Householder*, descritas nos Apêndices B.1 e B.2.

2.3.2 Atualizações da Medida

O objetivo da atualização da medida é calcular $P_{i|i}^{1/2}$ a partir de $P_{i|i-1}^{1/2}$, de acordo com a Equação (2.7). Se tal equação tivesse um sinal de mais ao invés de um sinal de menos, poderia-se facilmente usar a técnica da Seção 2.3.1. Há uma formulação alternativa, que utiliza transformações unitárias, definida em KAILATH et al. (2000) para resolver este problema e fornece o seguinte algoritmo *array*. Define-se

$$\begin{bmatrix} R_i^{1/2} & H_i P_{i|i-1}^{1/2} \\ 0 & P_{i|i-1}^{1/2} \end{bmatrix} \quad (2.26)$$

como *pre-array*, e triangulariza-se através de uma transformação unitária Θ

$$\begin{bmatrix} R_i^{1/2} & H_i P_{i|i-1}^{1/2} \\ 0 & P_{i|i-1}^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} X & 0 \\ Y & Z \end{bmatrix} \quad (2.27)$$

denominada *post-array*, cujas entradas podem ser visualizadas elevando ao quadrado ambos os lados

$$\begin{bmatrix} R_i^{1/2} & H_i P_{i|i-1}^{1/2} \\ 0 & P_{i|i-1}^{1/2} \end{bmatrix} \Theta \Theta^* \begin{bmatrix} R_i^{1/2} & H_i P_{i|i-1}^{1/2} \\ 0 & P_{i|i-1}^{1/2} \end{bmatrix}^* = \begin{bmatrix} X & 0 \\ Y & Z \end{bmatrix} \begin{bmatrix} X & 0 \\ Y & Z \end{bmatrix}^* \quad (2.28)$$

obtendo assim

$$\begin{aligned}
XX^* &= R_i^{1/2} R_i^{*/2} + H_i P_{i|i-1}^{1/2} P_{i|i-1}^{*/2} H_i^* \\
&= R_i + H_i P_{i|i-1} H_i^* = R_{e,i}, \\
YX^* &= P_{i|i-1}^{1/2} P_{i|i-1}^{*/2} H_i^* = P_{i|i-1} H_i^*, \\
ZZ^* &= P_{i|i-1}^{1/2} P_{i|i-1}^{*/2} - Y Y^* \\
&= P_{i|i-1} - P_{i|i-1} H_i^* (X^{-*} X^{-1}) H_i P_{i|i-1} \\
&= P_{i|i-1} - P_{i|i-1} H_i^* R_{e,i}^{-1} H_i P_{i|i-1} = P_{i|i}.
\end{aligned} \tag{2.29}$$

Pode-se então identificar Z como fator raiz quadrada de $P_{i|i}$, e similarmente identificar X e Y como $X = R_{e,i}^{1/2}$ e $Y = \bar{K}_{f,i} = P_{i|i-1} H_i^* R_{e,i}^{-*/2}$. Então a forma final para o problema de atualização da medida é dada por

$$\begin{bmatrix} R_i^{1/2} & H_i P_{i|i-1}^{1/2} \\ 0 & P_{i|i-1}^{1/2} \end{bmatrix} \Theta = \begin{bmatrix} R_{e,i}^{1/2} & 0 \\ \bar{K}_{f,i} & P_{i|i}^{1/2} \end{bmatrix}. \tag{2.30}$$

2.4 Sistemas Singulares

Sistemas singulares (também conhecidos como sistemas descritores ou implícitos) têm recebido destaque na literatura devido a frequência com que aparecem em problemas de modelagem de sistemas econômicos LUENBERGER (1977), modelagem de imagem HASSAN & SADJANI (1995), robótica MILLS & GOLDENBERG (1989) e outros. A representação de um sistema discreto pode ser representada por

$$f(x_{i+1}, x_i, u_i, i) = 0, \tag{2.31}$$

$$g(x_i, u_i, y_i, i) = 0, \tag{2.32}$$

sendo x_i o estado do sistema composto de variáveis de estado, u_i é a entrada de controle, y_i é a medida da saída, f e g são funções vetoriais de x_{i+1} , x_i , u_i e i com dimensões apropriadas. Um caso especial de (2.31) e (2.32) ocorre quando

$$E_i x_{i+1} = \varphi_1(x_i, u_i, i), \tag{2.33}$$

$$y_i = \varphi_2(x_i, u_i, i), \tag{2.34}$$

sendo φ_1, φ_2 funções vetoriais de x_i, u_i e i . Estas equações representarão um sistema singular quando a matriz E_i for singular. Se φ_1 e φ_2 são funções lineares de x_i e u_i uma formulação possível para este sistema é descrita por

$$E_i x_{i+1} = F_i x_i + G_i u_i, \quad (2.35)$$

$$y_i = H_i x_i, \quad (2.36)$$

sendo $x_i \in R^n, u_i \in R^m, y_i \in R^r$. $E_i, F_i \in R^{n \times n}, G_i \in R^{n \times m}$ e $H_i \in R^{r \times n}$ matrizes constantes. Quando E_i é não singular, o sistema recai no espaço de estados convencional, daí a outra denominação para ele: sistema generalizado, sendo o espaço de estados um caso particular quando $E_i = I$. Note que há casos onde E_i, F_i são intrinsecamente retangulares HASSAN & SADJANI (1995) e portanto uma análise usando equações no espaço de estados usuais não é possível. Os sistemas singulares foram mencionados pela primeira vez na literatura em 1973 SINGH & LIU (1973). Em muitas publicações os sistemas singulares são chamados também de semi-estados, sistemas degenerados e sistemas algébrico-diferenciais, por envolverem equações algébricas e equações diferenciais. Para um caso de sistemas discretos onde $G_i = I$, tem-se a formulação dada pelas equações (2.37)-(2.38) onde consideram-se ruídos de sistema e de medição

$$E_i x_{i+1} = F_i x_i + u_i, \quad (2.37)$$

$$y_i = H_i x_i + v_i, \quad (2.38)$$

sendo as variáveis aleatórias u_i, v_i, x_0 de média zero tais que

$$\mathcal{E} \left(\begin{bmatrix} u_i \\ v_i \\ x_0 \end{bmatrix} \cdot \begin{bmatrix} u_j \\ v_j \\ x_0 \end{bmatrix}^* \right) = \begin{bmatrix} Q_i \delta_{ij} & S_i \delta_{ij} & 0 \\ S_i^* \delta_{ij} & R_i \delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix}, \quad (2.39)$$

sendo as matrizes $\{E_i, F_i, G_i, H_i, \Pi_0, Q_i, S_i, R_i\}$ conhecidas e os valores Q_i e R_i são as autocovariâncias de u_i e v_i respectivamente e S_i é covariância cruzada entre os ruídos. Então a estimativa filtrada para x_i , $\hat{x}_i = \hat{x}_{i|i}$ (sendo conhecida a saída do sistema, $\{y_0, \dots, y_i\}$), em conjunto com a matriz covariância do erro $P_{i|i}$, podem ser calculadas

recursivamente através das equações ISHIHARA et al. (2004b)

$$P_{i|i}^{-1} = E_i^*(Q_{i-1} + F_{i-1}P_{i-1|i-1}F_{i-1}^*)^{-1}E_i + H_i^*R_i^{-1}H_i, \quad (2.40)$$

$$\hat{x}_{i|i} = P_{i|i}E_i^*(Q_i + F_{i-1}P_{i-1|i-1}F_{i-1}^*)^{-1}F_{i-1}\hat{x}_{i-1|i-1} + P_{i|i}H_i^*R_i^{-1}y_i. \quad (2.41)$$

2.4.1 Filtro Singular Robusto

Quando o sistema singular está sujeito a incertezas paramétricas na forma

$$\begin{aligned} (E_{i+1} + \delta E_{i+1})x_{i+1} &= (F_i + \delta F_i)x_i + w_i, \quad i = 0, 1, \dots \\ y_i &= (H_i + \delta H_i)x_i + v_i, \end{aligned} \quad (2.42)$$

sendo δE_{i+1} , δF_i e δH_i perturbações nas matrizes nominais do sistema, variáveis no tempo e definidas como

$$\delta F_i = M_{f,i}\Delta_i N_{f,i}, \quad (2.43)$$

$$\delta E_{i+1} = M_{f,i}\Delta_i N_{e,i+1}, \quad (2.44)$$

$$\delta H_i = M_{h,i}\Delta_i N_{h,i}, \quad (2.45)$$

$$\|\Delta_i\| \leq 1, \quad (2.46)$$

$M_{f,i}$, $M_{h,i}$, $N_{e,i+1}$, $N_{f,i}$, $N_{h,i}$ matrizes conhecidas e Δ_i uma contração, utilizam-se filtros robustos para a filtragem deste tipo de sistema singular. Neste trabalho serão considerados os filtros desenvolvidos em ISHIHARA et al. (2004), ISHIHARA et al. (2004b) e CAMPOS (2004) descritos a seguir. A versão robusta do filtro é dada pelo seguinte teorema

Teorema 2.4.1 *Suponha que $\begin{bmatrix} \hat{E}_i \\ \hat{H}_i \end{bmatrix}$ tem posto coluna pleno para todo $i \geq 0$. A estimativa filtrada robusta ótima $\hat{x}_{i|i}$ de x_i pode ser obtida a partir do seguinte algoritmo recursivo*

Passo 0: Condições Iniciais: Se $M_{h,0} = 0$ então

$$P_{0|0} := (P_0^{-1} + H_0^* R_0^{-1} H_0)^{-1}, \quad (2.47)$$

$$\hat{x}_{0|0} := P_{0|0} H_0^* R_0^{-1} y_0. \quad (2.48)$$

Se $M_{h,0} \neq 0$ encontre o parâmetro escalar ótimo $\hat{\lambda}_{-1}$ no intervalo $\lambda > \|M_{h,0}^ R_0^{-1} M_{h,0}\|$ (vide Apêndice C.2) e considere*

$$\hat{R}_0 := R_0 - \hat{\lambda}_{-1}^{-1} M_{h,0} M_{h,0}^*, \quad (2.49)$$

$$P_{0|0} := (P_0^{-1} + H_0^* \hat{R}_0^{-1} H_0 + \hat{\lambda}_{-1} N_{h,0}^* N_{h,0})^{-1}, \quad (2.50)$$

$$\hat{x}_{0|0} := P_{0|0} H_0^* \hat{R}_0^{-1} y_0. \quad (2.51)$$

Passo 1: Se $M_{f,i} = 0$ e $M_{h,i+1} = 0$ então $\hat{\lambda}_i := 0$. Se não, encontre o parâmetro escalar ótimo $\hat{\lambda}_i$ no intervalo

$$\hat{\lambda}_i > \lambda_{l,i} := \left\| \begin{bmatrix} M_{f,i}^* & 0 \\ 0 & M_{h,i+1}^* \end{bmatrix} \begin{bmatrix} Q_i^{-1} & 0 \\ 0 & R_{i+1}^{-1} \end{bmatrix} \begin{bmatrix} M_{f,i} & 0 \\ 0 & M_{h,i+1} \end{bmatrix} \right\|. \quad (2.52)$$

Passo 2: Se $\hat{\lambda}_i \neq 0$, substitua os parâmetros $\{Q_i, R_{i+1}, P_{i|i}, E_{i+1}, F_i, H_{i+1}\}$ pelos parâmetros corrigidos

$$\hat{Q}_i := Q_i - \hat{\lambda}_i^{-1} M_{f,i} M_{f,i}^*, \quad (2.53)$$

$$\hat{R}_{i+1} := R_{i+1} - \hat{\lambda}_i^{-1} M_{h,i+1} M_{h,i+1}^*, \quad (2.54)$$

$$\hat{P}_{i|i} := (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1}, \quad (2.55)$$

$$\hat{E}_{i+1} := E_{i+1} - \hat{\lambda}_i F_i \hat{P}_{i|i} N_{f,i}^* N_{e,i+1}. \quad (2.56)$$

Passo 3: Atualize $\{P_{i|i}, \hat{x}_{i|i}\}$ para $\{P_{i+1|i+1}, \hat{x}_{i+1|i+1}\}$ do seguinte modo

$$\begin{aligned} P_{i+1|i+1} &:= \left(\hat{E}_{i+1}^* (\hat{Q}_i + F_i \hat{P}_{i|i} F_i^*)^{-1} \hat{E}_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} \right. \\ &\quad \left. + \hat{\lambda}_i [N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* (I + \hat{\lambda}_i N_{f,i} P_{i|i} N_{f,i}^*)^{-1} N_{e,i+1}] \right)^{-1}, \end{aligned} \quad (2.57)$$

$$\begin{aligned}\hat{x}_{i+1|i+1} &:= P_{i+1|i+1} \left(\left[\hat{E}_{i+1}^* (\hat{Q}_i + F_i \hat{P}_{i|i} F_i^*)^{-1} F_i + \hat{\lambda}_i N_{e,i+1}^* N_{f,i} \right] (I - \hat{\lambda}_i \hat{P}_{i|i} N_{f,i}^* N_{f,i}) \right) \hat{x}_{i|i} \\ &+ P_{i+1|i+1} H_{i+1}^* \hat{R}_{i+1}^{-1} z_{i+1}. \quad \diamond\end{aligned}\quad (2.58)$$

A versão robusta do preditor é dada pelo seguinte teorema

Teorema 2.4.2 *Suponha que a matriz $\hat{E}_{i+1}$ tem posto coluna pleno para todo $i \geq 0$ e é dada a seqüência de medidas y_i citada acima. A estimativa preditora ótima $\hat{x}_{i+1|i}$ pode ser obtida a partir do seguinte algoritmo recursivo.*

Passo 0: Condições Iniciais

$$P_{0|-1} = P_0, \quad (2.59)$$

$$\hat{x}_{0|-1} = 0. \quad (2.60)$$

Passo 1: Se $M_{f,i} = 0$ e $M_{h,i} = 0$ então $\hat{\lambda}_i := 0$. Se não, encontre o parâmetro escalar ótimo $\hat{\lambda}_i$ (vide Apêndice C.2) no intervalo

$$\hat{\lambda}_i > \lambda_{l,i} := \left\| \begin{bmatrix} M_{f,i}^* & 0 \\ 0 & M_{h,i}^* \end{bmatrix} \begin{bmatrix} Q_i^{-1} & 0 \\ 0 & R_i^{-1} \end{bmatrix} \begin{bmatrix} M_{f,i} & 0 \\ 0 & M_{h,i} \end{bmatrix} \right\|. \quad (2.61)$$

Passo 2: Se $\hat{\lambda}_i \neq 0$, substitua os parâmetros $\{Q_i, R_i, P_{i|i-1}, E_{i+1}, F_i, H_i\}$ pelos parâmetros corrigidos

$$\hat{Q}_i := \begin{bmatrix} \hat{Q}_i & 0 \\ 0 & I \end{bmatrix} \quad \text{sendo} \quad \hat{Q}_i := Q_i - \hat{\lambda}_i^{-1} M_{f,i} M_{f,i}^*, \quad (2.62)$$

$$\hat{R}_i := \begin{bmatrix} \hat{R}_i & 0 \\ 0 & I \end{bmatrix} \quad \text{sendo} \quad \hat{R}_i := R_i - \hat{\lambda}_i^{-1} M_{h,i} M_{h,i}^*, \quad (2.63)$$

$$\hat{E}_{i+1} := \begin{bmatrix} E_{i+1} \\ \sqrt{\hat{\lambda}_i} N_{e,i+1} \end{bmatrix}, \quad (2.64)$$

$$\hat{F}_i := \begin{bmatrix} F_i \\ \sqrt{\hat{\lambda}_i} N_{f,i} \end{bmatrix}, \quad (2.65)$$

$$\hat{\mathcal{H}}_i := \begin{bmatrix} H_i \\ \sqrt{\hat{\lambda}_i} N_{h,i} \end{bmatrix}. \quad (2.66)$$

Passo 3: Atualize $\{P_{i|i-1}, \hat{x}_{i|i-1}\}$ para $\{P_{i+1|i}, \hat{x}_{i+1|i}\}$ do seguinte modo:

$$P_{i+1|i}^{-1} = \hat{\mathcal{E}}_{i+1}^* (\hat{\mathcal{Q}}_i + \hat{\mathcal{F}}_i P_{i|i-1} \hat{\mathcal{F}}_i^* - \hat{\mathcal{F}}_i P_{i|i-1} \hat{\mathcal{H}}_i^* (\hat{\mathcal{R}}_i + \hat{\mathcal{H}}_i P_{i|i-1} \hat{\mathcal{H}}_i^*)^{-1} \hat{\mathcal{H}}_i P_{i|i-1} \hat{\mathcal{F}}_i^*)^{-1} \hat{\mathcal{E}}_{i+1}^*, \quad (2.67)$$

$$\begin{aligned} \hat{x}_{i+1|i} = P_{i+1|i} \mathcal{E}_{i+1}^* & \left(\mathcal{Q}_i + \mathcal{F}_i P_{i|i-1} \mathcal{F}_i^* - \mathcal{F}_i P_{i|i-1} \mathcal{H}_i^* (\mathcal{R}_i + \mathcal{H}_i P_{i|i-1} \mathcal{H}_i^*)^{-1} \times \mathcal{H}_i P_{i|i-1} \mathcal{F}_i^* \right)^{-1} \times \\ & \mathcal{F}_i \hat{x}_{i|i-1} + P_{i+1|i} \mathcal{E}_{i+1}^* \left(\mathcal{Q}_i + \mathcal{F}_i P_{i|i-1} \mathcal{F}_i^* - \mathcal{F}_i P_{i|i-1} \mathcal{H}_i^* (\mathcal{R}_i + \mathcal{H}_i P_{i|i-1} \mathcal{H}_i^*)^{-1} \mathcal{H}_i P_{i|i-1} \mathcal{F}_i^* \right)^{-1} \times \\ & \mathcal{F}_i P_{i|i-1} \mathcal{H}_i^* (\mathcal{R}_i + \mathcal{H}_i P_{i|i-1} \mathcal{H}_i^*)^{-1} (\mathcal{Y}_i - \mathcal{H}_i \hat{x}_{i|i-1}), \end{aligned} \quad (2.68)$$

sendo

$$\mathcal{Y}_i := \begin{bmatrix} y_i \\ 0 \end{bmatrix}. \quad (2.69)$$

As provas de ambos os teoremas podem ser encontradas em CAMPOS (2004).

Um dos objetivos deste trabalho é o desenvolvimento de algoritmos *array* para filtros singulares nominais e robustos.

Capítulo 3

Filtragem de Sistemas Singulares

Neste capítulo serão deduzidos filtros de informação, nominais e robustos, para sistemas singulares discretos no tempo. Também serão apresentados os respectivos algoritmos *array* para as versões de informação. Serão consideradas as expressões para as estimativas filtradas e preditoras. Para o problema de filtragem, o objetivo é deduzir a expressão para a 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 . A versão de informação pressupõe que se deseja estimar $P_{i|i}^{-1}\hat{x}_{i|i}$ calculando $P_{i|i}^{-1}$, para o filtro, e $P_{i+1|i}^{-1}\hat{x}_{i+1|i}$ calculando $P_{i+1|i}^{-1}$, para o preditor. Portanto, neste capítulo serão deduzidos os filtros de informação e os respectivos algoritmos *array* para as versões nominais filtrada e preditora bem como as versões robustas filtrada e preditora. Os algoritmos *array* serão deduzidos baseados no Lema C.1.1 do Apêndice B. Os resultados apresentados neste capítulo fazem parte das contribuições originais deste trabalho.

3.1 Estimativa Filtrada Nominal

Filtros expressos na forma de informação propagam o valor $P_{i|i}^{-1}$ ao invés do valor $P_{i|i}$. A vantagem do filtro de informação é que se pode ter um algoritmo numericamente mais robusto quando se tem pouca ou nenhuma informação da condição inicial x_0 , que corresponde a P_0 muito grande ou infinita enquanto P_0^{-1} é pequena ou nula.

Teorema 3.1.1 *Dadas as seguintes condições iniciais*

$$\begin{aligned} P_{0|0}^{-1} &= P_0^{-1} + H_0^* R_0^{-1} H_0 \\ P_{0|0}^{-1} \hat{x}_{0|0} &= H_0^* R_0^{-1} y_0 \end{aligned} \quad (3.1)$$

a versão de informação para a estimativa nominal filtrada do sistema singular (2.37)-(2.38) é dada pelas seguintes expressões

$$P_{i|i}^{-1} = E_i^* Q_{i-1}^{-1} E_i + H_i^* R_i^{-1} H_i - E_i^* Q_{i-1}^{-1} F_{i-1} (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} F_{i-1}^* Q_{i-1}^{-1} E_i \quad (3.2)$$

$$P_{i|i}^{-1} \hat{x}_{i|i} = E_i^* Q_{i-1}^{-1} F_{i-1} (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} P_{i-1|i-1}^{-1} \hat{x}_{i-1|i-1} + H_i^* R_i^{-1} y_i \quad (3.3)$$

e o correspondente algoritmo array é dado por

$$\begin{bmatrix} P_{i-1|i-1}^{-1/2} & F_{i-1}^* Q_{i-1}^{-1/2} & 0 \\ 0 & E_i^* Q_{i-1}^{-1/2} & H_i^* R_i^{-1/2} \end{bmatrix} \Theta = \begin{bmatrix} A_{i-1}^{1/2} & 0 & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} A_{i-1}^{-1/2} & P_{i|i}^{-1/2} & 0 \end{bmatrix} \quad (3.4)$$

sendo $A_{i-1} \triangleq P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1}$.

Prova: Considera-se a Equação de Riccati (2.40)

$$P_{i|i}^{-1} = E_i^* (Q_{i-1} + F_{i-1} P_{i-1|i-1} F_{i-1}^*)^{-1} E_i + H_i^* R_i^{-1} H_i. \quad (3.5)$$

Para se deduzir o filtro de informação aplica-se o lema da inversão de matrizes (A.2) em (2.40)

$$P_{i|i}^{-1} = E_i^* \underbrace{(Q_{i-1} + F_{i-1} P_{i-1|i-1} F_{i-1}^*)^{-1}} E_i + H_i^* R_i^{-1} H_i \quad (3.6)$$

para se obter

$$\begin{aligned} P_{i|i}^{-1} &= E_i^* [Q_{i-1}^{-1} - Q_{i-1}^{-1} F_{i-1} (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} F_{i-1}^* Q_{i-1}^{-1}] E_i + H_i^* R_i^{-1} H_i \\ &= E_i^* Q_{i-1}^{-1} E_i + H_i^* R_i^{-1} H_i - E_i^* Q_{i-1}^{-1} F_{i-1} (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} F_{i-1}^* Q_{i-1}^{-1} E_i. \end{aligned} \quad (3.7)$$

Esta é a versão de informação da Riccati para a estimativa filtrada. Na expressão do filtro, multiplica-se à esquerda $P_{i|i}^{-1}$ na Equação (2.41)

$$P_{i|i}^{-1}\hat{x}_{i|i} = E_i^*(Q_i + F_{i-1}P_{i-1|i-1}F_{i-1}^*)^{-1}F_{i-1}\hat{x}_{i-1|i-1} + H_i^*R_i^{-1}y_i, \quad (3.8)$$

e aplicando-se o lema da inversão de matrizes novamente em (3.8), obtém-se

$$\begin{aligned} P_{i|i}^{-1}\hat{x}_{i|i} &= E_i^*(Q_i^{-1} - Q_i^{-1}F_{i-1}(P_{i-1|i-1}^{-1} + F_{i-1}^*Q_i^{-1}F_{i-1})^{-1}F_{i-1}^*Q_i^{-1}) \\ &\quad \times F_{i-1}\hat{x}_{i-1|i-1} + H_i^*R_i^{-1}y_i, \end{aligned} \quad (3.9)$$

Pode-se colocar $E_i^*Q_i^{-1}F_{i-1}(P_{i-1|i-1}^{-1} + F_{i-1}^*Q_i^{-1}F_{i-1})^{-1}$ em evidência

$$\begin{aligned} P_{i|i}^{-1}\hat{x}_{i|i} &= E_i^*Q_i^{-1}F_{i-1}(P_{i-1|i-1}^{-1} + F_{i-1}^*Q_i^{-1}F_{i-1})^{-1}((P_{i-1|i-1}^{-1} + F_{i-1}^*Q_i^{-1}F_{i-1}) - \\ &\quad F_{i-1}^*Q_i^{-1}F_{i-1})\hat{x}_{i-1|i-1} + H_i^*R_i^{-1}y_i, \end{aligned} \quad (3.10)$$

para se obter

$$P_{i|i}^{-1}\hat{x}_{i|i} = E_i^*Q_i^{-1}F_{i-1}(P_{i-1|i-1}^{-1} + F_{i-1}^*Q_i^{-1}F_{i-1})^{-1}P_{i-1|i-1}^{-1}\hat{x}_{i-1|i-1} + H_i^*R_i^{-1}y_i. \quad (3.11)$$

Assim, a recursividade fica em função de $P_{i|i}^{-1}\hat{x}_{i|i}$. O valor de $\hat{x}_{i|i}$ pode ser calculado por métodos algébricos sem a necessidade de se calcular a inversa de $P_{i|i}^{-1}$. Para se deduzir o algoritmo *array* para essa Equação Algébrica de Riccati, deve-se notar através da última igualdade de (3.7) que $P_{i|i}^{-1}$ pode ser escrita como o complemento de *Schur*¹ de $P_{i-1|i-1}^{-1} + F_{i-1}^*Q_{i-1}^{-1}F_{i-1}$ em

$$\begin{bmatrix} P_{i-1|i-1}^{-1} + F_{i-1}^*Q_{i-1}^{-1}F_{i-1} & F_{i-1}^*Q_{i-1}^{-1}E_i \\ E_i^*Q_{i-1}^{-1}F_{i-1} & E_i^*Q_{i-1}^{-1}E_i + H_i^*R_i^{-1}H_i \end{bmatrix}. \quad (3.12)$$

Como se procura uma expressão do tipo $AA^* = BB^*$ para que o Lema C.1.1 seja aplicado, fatora-se (3.12) de modo a obter

$$\begin{bmatrix} P_{i-1|i-1}^{-1/2} & F_{i-1}^*Q_{i-1}^{-1/2} & 0 \\ 0 & E_i^*Q_{i-1}^{-1/2} & H_i^*R_i^{-1/2} \end{bmatrix} \begin{bmatrix} P_{i-1|i-1}^{-*/2} & 0 \\ Q_{i-1}^{-1/2}F_{i-1} & Q_{i-1}^{-1/2}E_i \\ 0 & R_i^{-1/2}H_i \end{bmatrix}. \quad (3.13)$$

¹Complemento de Schur de A em $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ é dado por $\Delta_A \triangleq D - CA^{-1}B$. Veja mais detalhes no Apêndice A.

Esta é uma decomposição auxiliar para se definir o *pre-array*. Para se definir o *post-array*, fatora-se (3.12) através da seguinte decomposição, definida em detalhes no Apêndice A

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & \Delta A \end{bmatrix} \begin{bmatrix} I & A^{-1}B \\ 0 & I \end{bmatrix} \quad (3.14)$$

sendo ΔA o complemento de *Schur* de A em $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$. Usando tal propriedade, pode-se escrever (3.12) como

$$\begin{bmatrix} I & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} & I \end{bmatrix} \begin{bmatrix} P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1} & 0 \\ 0 & P_{i|i}^{-1} \end{bmatrix} \begin{bmatrix} I & (P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1})^{-1} F_{i-1}^* Q_{i-1}^{-1} E_i \\ 0 & I \end{bmatrix}. \quad (3.15)$$

Pode-se ainda fatorar a matriz central de (3.15) renomeando $A_{i-1} \triangleq P_{i-1|i-1}^{-1} + F_{i-1}^* Q_{i-1}^{-1} F_{i-1}$, que é hermitiana e positiva-definida, tem-se

$$\begin{bmatrix} I & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} A_{i-1}^{-1} & I \end{bmatrix} \begin{bmatrix} A_{i-1}^{1/2} & 0 \\ 0 & P_{i|i}^{-1/2} \end{bmatrix} \begin{bmatrix} A_{i-1}^{*/2} & 0 \\ 0 & P_{i|i}^{-*/2} \end{bmatrix} \begin{bmatrix} I & A_{i-1}^{-1} F_{i-1}^* Q_{i-1}^{-1} E_i \\ 0 & I \end{bmatrix}. \quad (3.16)$$

Nota-se que (3.16) apresenta uma estrutura do tipo ABB^*A^* . Simplificando (3.16), obtém-se

$$\begin{bmatrix} A_{i-1}^{1/2} & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} A_{i-1}^{-1/2} & P_{i|i}^{-1/2} \end{bmatrix} \begin{bmatrix} A_{i-1}^{1/2} & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} A_{i-1}^{-1/2} & P_{i|i}^{-1/2} \end{bmatrix}^*. \quad (3.17)$$

Logo, igualando (3.17) com (3.13) existe, de acordo com o Lema C.1.1, uma matriz unitária Θ tal que

$$\begin{bmatrix} P_{i-1|i-1}^{-1/2} & F_{i-1}^* Q_{i-1}^{-1/2} & 0 \\ 0 & E_i^* Q_{i-1}^{-1/2} & H_i^* R_i^{-1/2} \end{bmatrix} \Theta = \begin{bmatrix} A_{i-1}^{1/2} & 0 & 0 \\ E_i^* Q_{i-1}^{-1} F_{i-1} A_{i-1}^{-1/2} & P_{i|i}^{-1/2} & 0 \end{bmatrix}. \quad (3.18)$$

Portanto este é o algoritmo *array* para a equação recursiva da Riccati (2.40). Perceba que não há nenhuma perda de generalidade em virtude da inclusão de uma coluna de zeros em (3.17). $\diamond$

3.2 Estimativa Preditora Nominal

Nesta seção será deduzido um filtro de informação na versão preditora com o correspondente algoritmo *array*.

Teorema 3.2.1 *Dadas as seguintes condições iniciais*

$$\begin{aligned} P_{0|-1}^{-1} &= P_0^{-1} \\ P_{0|-1}^{-1} \hat{x}_{0|-1} &= 0 \end{aligned} \quad (3.19)$$

a versão de informação para a estimativa nominal preditora do sistema singular (2.37)-(2.38) é dada por

$$P_{i+1|i}^{-1} = E_{i+1}^* Q_i^{-1} E_{i+1} - E_{i+1}^* Q_i^{-1} F_i \left(P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i \right)^{-1} F_i^* Q_i^{-1} E_{i+1}, \quad (3.20)$$

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ &\quad (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + H_i^* R_i^{-1} y_i), \end{aligned} \quad (3.21)$$

e o correspondente algoritmo *array* para a solução da Equação de Riccati é dado por

$$\begin{aligned} &\begin{bmatrix} P_{i|i-1}^{-1/2} & H_i^* R_i^{-1/2} & F_i^* Q_i^{-1/2} \\ 0 & 0 & E_i^* Q_i^{-1/2} \end{bmatrix} \Theta = \\ &\begin{bmatrix} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{1/2} & 0 & 0 \\ E_{i+1}^* Q_i^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{-1/2} & P_{i+1|i}^{-1} & 0 \end{bmatrix}. \end{aligned} \quad (3.22)$$

Prova: A equação da estimativa preditora é dada em CAMPOS (2004)

$$P_{i+1|i} = \left(E_{i+1}^* (Q_i + F_i P_{i|i-1} F_i^* - F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1} F_i^*)^{-1} E_{i+1} \right)^{-1}. \quad (3.23)$$

Pode-se reescrever (3.23) como

$$P_{i+1|i} = \left(E_{i+1}^* (Q_i + F_i (P_{i|i-1} - P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1}) F_i^*)^{-1} E_{i+1} \right)^{-1}. \quad (3.24)$$

Aplicando o lema da inversão de matrizes duas vezes, tem-se

$$P_{i+1|i}^{-1} = E_{i+1}^* \left(Q_i + F_i \left(P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i \right)^{-1} F_i^* \right)^{-1} E_{i+1}. \quad (3.25)$$

$$\begin{aligned}
P_{i+1|i}^{-1} &= E_{i+1}^* \left(Q_i^{-1} - Q_i^{-1} F_i \left(P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i \right)^{-1} F_i^* Q_i^{-1} \right) E_{i+1} = \\
&E_{i+1}^* Q_i^{-1} E_{i+1} - E_{i+1}^* Q_i^{-1} F_i \left(P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i \right)^{-1} F_i^* Q_i^{-1} E_{i+1}
\end{aligned} \tag{3.26}$$

A Equação (3.26) é a versão de informação para a estimativa preditora de sistemas singulares. Para a equação do filtro, tem-se que

$$\begin{aligned}
\hat{x}_{i+1|i} &= P_{i+1|i} E_{i+1}^* \left(Q_i + F_i P_{i|i-1} F_i^* - F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1} F_i^* \right)^{-1} \times \\
&F_i \hat{x}_{i|i-1} + P_{i+1|i} E_{i+1}^* \left(Q_i + F_i P_{i|i-1} F_i^* - F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1} F_i^* \right)^{-1} \times \\
&F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} (y_i - H_i \hat{x}_{i|i-1}),
\end{aligned} \tag{3.27}$$

aplica-se o lema da inversão assim como na Equação (3.25) e multiplica-se à direita por $P_{i+1|i}^{-1}$, então obtém-se

$$\begin{aligned}
P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* \left(Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^* \right)^{-1} F_i \hat{x}_{i|i-1} + \\
&E_{i+1}^* \left(Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^* \right)^{-1} F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} (y_i - H_i \hat{x}_{i|i-1}).
\end{aligned} \tag{3.28}$$

Colocando em evidência o primeiro termo e também $\hat{x}_{i|i}$, obtém-se

$$\begin{aligned}
P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^*)^{-1} \times \\
&F_i^*)^{-1} (F_i \hat{x}_{i|i-1} + F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} (y_i - H_i \hat{x}_{i|i-1}))
\end{aligned} \tag{3.29}$$

$$\begin{aligned}
P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^*)^{-1} \times \\
&((F_i - F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i) \hat{x}_{i|i-1} + F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i).
\end{aligned} \tag{3.30}$$

Pode-se também escrever da seguinte forma

$$\begin{aligned}
P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^*)^{-1} \times \\
&(F_i (P_{i|i-1} P_{i|i-1}^{-1} - P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1} P_{i|i-1}^{-1}) \hat{x}_{i|i-1} + \\
&F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i)
\end{aligned} \tag{3.31}$$

colocando-se $P_{i|i-1}^{-1}$ em evidência

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} F_i^*)^{-1} \times \\ & (F_i (P_{i|i-1} - P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} H_i P_{i|i-1}) P_{i|i-1}^{-1} \hat{x}_{i|i-1} + \\ & F_i P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i). \end{aligned} \quad (3.32)$$

Com o auxílio do lema da inversão de matrizes, obtém-se

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} \times \\ & F_i ((P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} P_{i|i-1}^{-1} \hat{x}_{i|i-1} + P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i) \end{aligned} \quad (3.33)$$

e multiplicando-se o último termo por $(P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)$ a expressão é reescrita como

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i ((P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ & P_{i|i-1}^{-1} \hat{x}_{i|i-1} + (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i). \end{aligned} \quad (3.34)$$

Com a sequência de operações algébricas definida a seguir, obtém-se o filtro de informação para a estimativa preditora

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ & (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) P_{i|i-1} H_i^* (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i) \end{aligned} \quad (3.35)$$

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ & (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + (H_i^* + H_i^* R_i^{-1} H_i P_{i|i-1} H_i^*) (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i). \end{aligned} \quad (3.36)$$

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ & (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + H_i^* R_i^{-1} (R_i + H_i P_{i|i-1} H_i^*) (R_i + H_i P_{i|i-1} H_i^*)^{-1} y_i) \end{aligned} \quad (3.37)$$

$$\begin{aligned} P_{i+1|i}^{-1} \hat{x}_{i+1|i} &= E_{i+1}^* (Q_i + F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i) F_i^*)^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i)^{-1} \times \\ & (P_{i|i-1}^{-1} \hat{x}_{i|i-1} + H_i^* R_i^{-1} y_i). \end{aligned} \quad (3.38)$$

Para se deduzir o algoritmo *array* da Riccati de informação, utiliza-se o fato de que $P_{i+1|i}$ é o complemento de Schur de $P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i$ em

$$\begin{bmatrix} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i) & F_i^* Q_i^{-1} E_{i+1} \\ E_{i+1}^* Q_i^{-1} F_i & E_{i+1}^* Q_i^{-1} E_{i+1} \end{bmatrix}. \quad (3.39)$$

Pode-se fatorar a Equação (3.39) em

$$\begin{bmatrix} P_{i|i-1}^{-1/2} & H^* R^{-1/2} & F^* Q_i^{-1/2} \\ 0 & 0 & E^* Q_i^{-1/2} \end{bmatrix} \begin{bmatrix} P_{i|i-1}^{-1/2} & H^* R^{-1/2} & F^* Q_i^{-1/2} \\ 0 & 0 & E^* Q_i^{-1/2} \end{bmatrix}^* \quad (3.40)$$

e também podem-se usar uma decomposição do complemento de *schur*, mostrada no Apêndice A, para fatorar a Equação (3.39) novamente como

$$\begin{bmatrix} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{1/2} & 0 & 0 \\ E_{i+1}^* Q_i^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{-1/2} & P_{i+1|i}^{-1} & 0 \\ (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{1/2} & 0 & 0 \\ E_{i+1}^* Q_i^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{-1/2} & P_{i+1|i}^{-1} & 0 \end{bmatrix}^*. \quad (3.41)$$

Sendo assim pode-se dizer que existe Θ unitário tal que

$$\begin{bmatrix} P_{i|i-1}^{-1/2} & H^* R^{-1/2} & F^* Q_i^{-1/2} \\ 0 & 0 & E^* Q_i^{-1/2} \end{bmatrix} \Theta = \begin{bmatrix} (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{1/2} & 0 & 0 \\ E_{i+1}^* Q_i^{-1} F_i (P_{i|i-1}^{-1} + H_i^* R_i^{-1} H_i + F_i^* Q_i^{-1} F_i)^{-1/2} & P_{i+1|i}^{-1} & 0 \end{bmatrix}. \quad (3.42)$$

A Equação 3.42 é o algoritmo *array* para a forma preditora da filtragem de sistemas singulares. $\diamond$

3.3 Estimativa Filtrada Robusta

As expressões para a versão de informação da estimativa robusta filtrada de sistemas singulares sujeitos a incertezas paramétricas na forma

$$\begin{aligned}(E_{i+1} + \delta E_{i+1})x_{i+1} &= (F_i + \delta F_i)x_i + w_i, \quad i = 0, 1, \dots \\ y_i &= (H_i + \delta H_i)x_i + v_i,\end{aligned}\tag{3.43}$$

sendo δE_{i+1} , δF_i e δH_i perturbações nas matrizes nominais do sistema, variáveis no tempo e definidas em (2.46), e a respectiva forma algoritmica *array*, são definidas de acordo com o teorema a seguir.

Teorema 3.3.1 *Dada as seguintes condições iniciais*

$$\begin{aligned}P_{0|0}^{-1} &= P_0^{-1} + H_0^* R_0^{-1} H_0 \\ P_{0|0}^{-1} \hat{x}_{0|0} &= H_0^* R_0^{-1} y_0\end{aligned}\tag{3.44}$$

a versão de informação para a estimativa robusta filtrada do sistema (3.43) com $N_{e,i+1}^* N_{f,i} = 0$ (matrizes definidas em (2.46)), é dada por

$$\begin{aligned}P_{i+1|i+1}^{-1} &= E_{i+1}^* \hat{Q}_i^{-1} E_{i+1} - E_{i+1}^* \hat{Q}_i^{-1} F_i (P_{i|i}^{-1} + \hat{\lambda}_i^{1/2} N_{f,i}^* N_{f,i} \hat{\lambda}_i^{1/2} + F_i^* \hat{Q}_i^{-1} F_i)^{-1} \times \\ &\quad F_i^* \hat{Q}_i^{-1} E_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} + \hat{\lambda}_i [N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* N_{e,i+1}], \\ P_{i+1|i+1}^{-1} \hat{x}_{i+1|i+1} &= E_{i+1}^* (\hat{Q}_i - F_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} F_i^*)^{-1} F_i \times \\ &\quad (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} P_{i|i}^{-1} \hat{x}_{i|i} + H_{i+1}^* \hat{R}_{i+1}^{-1} y_{i+1},\end{aligned}\tag{3.45}$$

e o algoritmo *array* correspondente é definido da seguinte forma

$$\begin{aligned}\begin{bmatrix} P_{i|i}^{-1/2} & F_i^* \hat{Q}_i^{-1/2} & \hat{\lambda}_i^{1/2} N_{f,i}^* & 0 & 0 & 0 \\ 0 & E_{i+1}^* \hat{Q}_i^{-1/2} & 0 & H_{i+1}^* \hat{R}_{i+1}^{-1/2} & \hat{\lambda}_i^{1/2} N_{h,i+1}^* & \hat{\lambda}_i^{1/2} N_{e,i+1}^* \end{bmatrix} \Theta = \\ \begin{bmatrix} A_i^{1/2} & 0 & 0 & 0 & 0 & 0 \\ E_{i+1}^* \hat{Q}_i^{-1} F_i A_i^{-1/2} & P_{i+1|i+1}^{-1/2} & 0 & 0 & 0 & 0 \end{bmatrix}.\end{aligned}\tag{3.46}$$

Prova: A equação do filtro robusto, com $N_{e,i+1}^* N_{f,i} = 0$, é definida por

$$\begin{aligned}P_{i+1|i+1}^{-1} &= E_{i+1}^* (\hat{Q}_i + F_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} F_i^*)^{-1} E_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} + \\ &\quad \hat{\lambda}_i [N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* N_{e,i+1}].\end{aligned}\tag{3.47}$$

Esta equação já está na forma de informação no entanto, para a dedução do algoritmo *array*, reescreve-se esta equação utilizando-se o lema da inversão de matrizes

$$P_{i+1|i+1}^{-1} = E_{i+1}^* (\hat{Q}_i^{-1} - \hat{Q}_i^{-1} F_i (P_{i|i}^{-1} + \hat{\lambda}_i^{1/2} N_{f,i}^* N_{f,i} \hat{\lambda}_i^{1/2} + F_i^* \hat{Q}_i^{-1} F_i)^{-1} F_i^* \hat{Q}_i^{-1}) E_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} + \hat{\lambda}_i^{1/2} [N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* N_{e,i+1}] \hat{\lambda}_i^{1/2}. \quad (3.48)$$

Manipulando a equação anterior, tem-se

$$P_{i+1|i+1}^{-1} = E_{i+1}^* \hat{Q}_i^{-1} E_{i+1} - E_{i+1}^* \hat{Q}_i^{-1} F_i (P_{i|i}^{-1} + \hat{\lambda}_i^{1/2} N_{f,i}^* N_{f,i} \hat{\lambda}_i^{1/2} + F_i^* \hat{Q}_i^{-1} F_i)^{-1} \times F_i^* \hat{Q}_i^{-1} E_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} + \hat{\lambda}_i^{1/2} [N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* N_{e,i+1}] \hat{\lambda}_i^{1/2}. \quad (3.49)$$

A equação do filtro é calculada manipulando-se a Equação (2.58) com $N_{e,i+1}^* N_{f,i} = 0$

$$\hat{x}_{i+1|i+1} = P_{i+1|i+1} E_{i+1}^* (\hat{Q}_i + F_i \hat{P}_{i|i} F_i^*)^{-1} F_i (I - \hat{\lambda}_i \hat{P}_{i|i} N_{f,i}^* N_{f,i}) \hat{x}_{i|i} + P_{i+1|i+1} H_{i+1}^* \hat{R}_{i+1}^{-1} y_{i+1}, \quad (3.50)$$

sendo que $\hat{P}_{i|i} = (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1}$. Multiplicando por $P_{i+1|i+1}^{-1}$

$$P_{i+1|i+1}^{-1} \hat{x}_{i+1|i+1} = E_{i+1}^* (\hat{Q}_i - F_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} F_i^*)^{-1} F_i \times (I - \hat{\lambda}_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} N_{f,i}^* N_{f,i}) \hat{x}_{i|i} + H_{i+1}^* \hat{R}_{i+1}^{-1} y_{i+1} \quad (3.51)$$

Pode-se fazer a seguinte fatoração acrescentando-se o termo $P_{i|i}^{-1} P_{i|i}$

$$P_{i+1|i+1}^{-1} \hat{x}_{i+1|i+1} = E_{i+1}^* (\hat{Q}_i - F_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} F_i^*)^{-1} F_i \times (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i} - \hat{\lambda}_i N_{f,i}^* N_{f,i}) \hat{x}_{i|i} + H_{i+1}^* \hat{R}_{i+1}^{-1} y_{i+1}. \quad (3.52)$$

Assim obtém-se o filtro de informação para sistemas singulares com incertezas

$$P_{i+1|i+1}^{-1} \hat{x}_{i+1|i+1} = E_{i+1}^* (\hat{Q}_i - F_i (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} F_i^*)^{-1} F_i \times (P_{i|i}^{-1} + \hat{\lambda}_i N_{f,i}^* N_{f,i})^{-1} P_{i|i}^{-1} \hat{x}_{i|i} + H_{i+1}^* \hat{R}_{i+1}^{-1} y_{i+1}. \quad (3.53)$$

Para a dedução do algoritmo *array*, pode-se escrever a equação de Riccati (3.49) na forma de Schur, denominando $C_i = N_{h,i+1}^* N_{h,i+1} + N_{e,i+1}^* N_{e,i+1}$, tem-se

$$\begin{bmatrix} P_{i|i}^{-1} + \hat{\lambda}_i^{1/2} N_{f,i}^* N_{f,i} \hat{\lambda}_i^{1/2} + F_i^* \hat{Q}_i^{-1} F_i & F_i^* \hat{Q}_i^{-1} E_{i+1} \\ E_{i+1}^* \hat{Q}_i^{-1} F_i & E_{i+1}^* \hat{Q}_i^{-1} E_{i+1} + H_{i+1}^* \hat{R}_{i+1}^{-1} H_{i+1} + \hat{\lambda}_i^{1/2} C_i \hat{\lambda}_i^{1/2} \end{bmatrix}. \quad (3.54)$$

Pode-se fatorar a equação acima como

$$\begin{bmatrix} P_{i|i}^{-1/2} & F_i^* \hat{Q}_i^{-1/2} & \hat{\lambda}_i^{1/2} N_{f,i}^* & 0 & 0 & 0 \\ 0 & E_{i+1}^* \hat{Q}_i^{-1/2} & 0 & H_{i+1}^* \hat{R}_{i+1}^{-1/2} & \hat{\lambda}_i^{1/2} N_{h,i+1}^* & \hat{\lambda}_i^{1/2} N_{e,i+1}^* \end{bmatrix} \begin{bmatrix} P_{i|i}^{-1/2} & F_i^* \hat{Q}_i^{-1/2} & \hat{\lambda}_i^{1/2} N_{f,i}^* & 0 & 0 & 0 \\ 0 & E_{i+1}^* \hat{Q}_i^{-1/2} & 0 & H_{i+1}^* \hat{R}_{i+1}^{-1/2} & \hat{\lambda}_i^{1/2} N_{h,i+1}^* & \hat{\lambda}_i^{1/2} N_{e,i+1}^* \end{bmatrix}^* \quad (3.55)$$

Através do complemento de Schur e denominando $A_i = P_{i|i}^{-1} + \hat{\lambda}_i^{1/2} N_{f,i}^* N_{f,i} \hat{\lambda}_i^{1/2} + F_i^* \hat{Q}_i^{-1} F_i$, obtém-se

$$\begin{bmatrix} I & 0 \\ E_{i+1}^* \hat{Q}_i^{-1} F_i A_i^{-1} & I \end{bmatrix} \begin{bmatrix} A_i & 0 \\ 0 & P_{i+1|i+1}^{-1} \end{bmatrix} \begin{bmatrix} I & A_i^{-1} F_i^* \hat{Q}_i^{-1} E_{i+1} \\ 0 & I \end{bmatrix} \quad (3.56)$$

que equivale a

$$\begin{bmatrix} A_i^{1/2} & 0 & 0 \\ E_{i+1}^* \hat{Q}_i^{-1} F_i A_i^{-1/2} & P_{i+1|i+1}^{-1/2} & 0 \end{bmatrix} \begin{bmatrix} A_i^{1/2} & 0 & 0 \\ E_{i+1}^* \hat{Q}_i^{-1} F_i A_i^{-1/2} & P_{i+1|i+1}^{-1/2} & 0 \end{bmatrix}^* \quad (3.57)$$

Portanto, o algoritmo *array* para o filtro de informação robusto de sistemas singulares com incertezas é definido como

$$\begin{bmatrix} P_{i|i}^{-1/2} & F_i^* \hat{Q}_i^{-1/2} & \hat{\lambda}_i^{1/2} N_{f,i}^* & 0 & 0 & 0 \\ 0 & E_{i+1}^* \hat{Q}_i^{-1/2} & 0 & H_{i+1}^* \hat{R}_{i+1}^{-1/2} & \hat{\lambda}_i^{1/2} N_{h,i+1}^* & \hat{\lambda}_i^{1/2} N_{e,i+1}^* \end{bmatrix} \Theta = \begin{bmatrix} A_i^{1/2} & 0 & 0 & 0 & 0 & 0 \\ E_{i+1}^* \hat{Q}_i^{-1} F_i A_i^{-1/2} & P_{i+1|i+1}^{-1/2} & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.58)$$

3.4 Estimativa Preditora Robusta

Para a estimativa preditora robusta do sistema singular (3.43) tem-se o seguinte teorema

Teorema 3.4.1 *Dadas as seguintes condições iniciais*

$$\begin{aligned} P_{0|-1}^{-1} &= P_0^{-1} \\ P_{0|-1}^{-1} \hat{x}_{0|-1} &= 0 \end{aligned} \quad (3.59)$$

a equação de Riccati e a expressão para o filtro preditor robusto do sistema singular

(3.43), na versão de informação, são dadas por

$$P_{i+1|i}^{-1} = \mathcal{E}_{i+1}^* \mathcal{Q}_i^{-1} \mathcal{E}_{i+1} - \mathcal{E}_{i+1}^* \mathcal{Q}_i^{-1} \mathcal{F}_i \left(P_{i|i-1}^{-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} \mathcal{H}_i + \mathcal{F}_i^* \mathcal{Q}_i^{-1} \mathcal{F}_i \right)^{-1} \mathcal{F}_i^* \mathcal{Q}_i^{-1} \mathcal{E}_{i+1} \quad (3.60)$$

$$P_{i+1|i}^{-1} \hat{x}_{i+1|i} = \mathcal{E}_{i+1}^* (\mathcal{Q}_i + \mathcal{F}_i (P_{i|i-1}^{-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} \mathcal{H}_i) \mathcal{F}_i^*)^{-1} \mathcal{F}_i (P_{i|i-1}^{-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} \mathcal{H}_i)^{-1} \times \quad (3.61)$$

$$(P_{i|i-1}^{-1} \hat{x}_{i|i-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} Y_i),$$

e o correspondente algoritmo array para a equação de Riccati é dado por

$$\begin{bmatrix} P_{i|i-1}^{-1/2} & \mathcal{H}^* \mathcal{R}^{-1/2} & \mathcal{F}^* \mathcal{Q}_i^{-1/2} \\ 0 & 0 & E^* \mathcal{Q}_i^{-1/2} \end{bmatrix} \Theta = \begin{bmatrix} (P_{i|i-1}^{-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} \mathcal{H}_i + \mathcal{F}_i^* \mathcal{Q}_i^{-1} \mathcal{F}_i)^{1/2} & 0 & 0 \\ \mathcal{E}_{i+1}^* \mathcal{Q}_i^{-1} \mathcal{F}_i (P_{i|i-1}^{-1} + \mathcal{H}_i^* \mathcal{R}_i^{-1} \mathcal{H}_i + \mathcal{F}_i^* \mathcal{Q}_i^{-1} \mathcal{F}_i)^{-1/2} & P_{i+1|i}^{-1} & 0 \end{bmatrix} \quad (3.62)$$

sendo estas matrizes definidas no Teorema 2.4.2.

Prova: São utilizados os mesmos passos algébricos da prova do preditor nominal robusto.

3.5 Implementação

Serão apresentados a seguir alguns exemplos de implementação para comparar o desempenho dos algoritmos *array* com relação ao uso das equações de informação em sua forma explícita. Foram utilizadas implementações por aritmética de pontos fixos devido a sua maior limitação de representação dos dados do que pontos flutuantes. Programando com pacote *simulink* do Matlab de pontos-fixos, as equações de informação em conjunto com suas correspondentes formas definidas pelos algoritmos *array*, puderam ser comparadas através da implementação por pontos flutuantes, que é interpretada como referência a ser comparada.

A idéia de fazer esta comparação usando pontos-fixos é salientar o aspecto de que o melhor número de condição que resulta da aplicação dos algoritmos *array* possibilita uma segurança maior em implementações que necessitam hardware com processamento de pontos fixos. Utilizando como exemplo o sistema singular abaixo

$$\begin{aligned}
E_i &= \begin{bmatrix} 1.1396 & 0 & 0 \\ 0 & 1.1659 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_i = \begin{bmatrix} 0.9693 & 0 & 0 \\ 0.2693 & -0.7792 & 0 \\ 0.1194 & 0.1134 & 0.6747 \end{bmatrix}, \\
H_i &= \begin{bmatrix} 0.1081 & 0 & 0 \\ 0 & 0.5149 & 0 \\ 0 & 0 & 0.3141 \end{bmatrix}, \quad Q_i = \begin{bmatrix} 0.1288 & 0 & 0 \\ 0 & 0.1768 & 0 \\ 0 & 0 & 0.1397 \end{bmatrix}, \\
R_i &= \begin{bmatrix} 0.0829 & 0 & 0 \\ 0 & 0.0317 & 0 \\ 0 & 0 & 0.0218 \end{bmatrix}, \quad Ne = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \\
Nf &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.3 \end{bmatrix}, \quad Nh = \begin{bmatrix} 0.002 & 0 & 0 \\ 0 & 0.02818 & 0 \\ 0 & 0 & 0.002 \end{bmatrix}, \\
Mf &= \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix}, \quad Mh = \begin{bmatrix} .8 & 0 & 0 \\ 0 & .8 & 0 \\ 0 & 0 & .8 \end{bmatrix}, \\
&\quad \hat{\lambda}_i = 40,
\end{aligned}$$

foram calculados os valores singulares de P_{ii}^{-1} para três diferentes implementações. Na primeira, a equação foi calculada via Matlab[®] usando processamento de ponto flutuante de precisão dupla. Como já foi dito, estes cálculos serão usados como referência para comparação. Para muitas implementações em pontos fixos, é necessário associar o tipo de dado e a informação de escala. Nestes exemplos foram utilizados arquitetura de ponto-fixe com 16-bits onde o primeiro bit determinou o sinal do número (+ ou -) e o último bit, o menos significativo foi escalonado para 2^{-10} significando que a implementação tem precisão de 2^{-10} e pode representar um número de -31.999 a 31.999 . Estas duas implementações foram feitas via Matlab[®] software usando o fix-point Simulink toolbox.

No caso nominal, onde $\hat{\lambda}_i = 0$, foram usadas as Equações (3.7) e (3.18). A tabela 3.1 mostra o erro quadrático médio das implementações de ponto fixo em relação ao ponto flutuante. O cálculo do MSE (*mean-square error*) para um valor singular VS_γ é dado por $E_\gamma = \frac{\sqrt{\sum_{i=1}^N (VS_{fi,\gamma} - VS_{i,\gamma})^2}}{N}$ onde $\gamma = 1, \dots, n$, define cada valor singular de uma

matriz, VS_f é o valor singular calculado usando ponto flutuante e i é o instante de tempo.

Tipo	Implementação	VS_1	VS_2	VS_3
Nominal ($\times 10^{-5}$)	Informação	22.9510	67.1321	0.0026
	Alg. <i>array</i>	0.3410	0.0033	0.0007
Robusto	Informação	68.6992	3.8622	1.4211
	Alg. <i>array</i>	0.0044	0.0003	0.0004

Tabela 3.1: Valores MSE para os valores singulares de P_{ii}^{-1} .

Portanto a Figura 3.1 e Tabela 3.1 mostram os valores singulares para P_{ii}^{-1} e seus erros médios quadráticos para a implementação robusta usando Equações (3.45) e (3.46).

Usando a Tabela 3.1 ou a Figura 3.1, pode-se notar a vantagem do algoritmo array em ambos os casos. Os valores singulares de P_{ii}^{-1} , calculados via algoritmo array em ponto fixo, estão próximos dos valores singulares calculados via ponto flutuante. Já a equação de informação apresentou números distantes da referência.

A razão destes resultados é que a faixa dinâmica dos valores calculados via algoritmo array está em função das raízes-quadradas dos números usados na equação de informação. Assim os números em *array* se mantiveram em um intervalo seguro para a implementação em ponto fixo. Porém, no caso das equações explícitas, ocorreram alguns casos onde os resultados dos cálculos não foram bem representados pelos pontos fixos, o que causou as deformações nos resultados.

Figura 3.1: Valores Singulares de P_{ii}^{-1} para a implementação robusta.

Capítulo 4

Algoritmo Array para Sistemas Markovianos

Vários sistemas dinâmicos são vulneráveis a mudanças abruptas em suas estruturas devido principalmente a falhas em componentes, conexões e mudanças no ambiente. Problemas de estimativa e controle desta categoria de sistemas, têm sido tratados por vários autores, veja JI & CHIZECK (1990b), JI & CHIZECK (1990), COSTA & FRAGOSO (1995), FRAGOSO & BACZYNSKI (2001), COSTA & do Val (2002), COSTA & JIMENEZ (2002), SWORDER & ROGERS (1983), ATHANS et al. (1977) e as referências contidas nesses artigos.

Em COSTA (1994) é deduzido um estimador de mínimos quadrados para sistemas lineares discretos sujeitos a saltos Markovianos, que modelam mudanças abruptas na dinâmica do sistema, para o sistema (4.1). É assumido que estas mudanças podem ser modeladas por uma cadeia de Markov no espaço de estados $\Theta(i) \in \{1, \dots, N\}$. Por simplicidade, chama-se neste texto tal filtro de filtro Markoviano.

Neste capítulo é deduzido um algoritmo *array* para o filtro Markoviano de COSTA (1994). O texto a seguir descreve tal filtro, formula o algoritmo array e apresenta um exemplo numérico ilustrativo.

4.1 Filtro Markoviano

Em um espaço de probabilidade $(\Omega, \mathcal{F}, \mathcal{P})$ onde Ω é o espaço amostral (conjunto de todos resultados ou saídas), $\mathcal{F}$ é o conjunto de eventos, e $\mathcal{P}$ é a função que associa probabilidades aos eventos, denota-se $1_{\{\cdot\}}$ como o Dirac de medição (para qualquer $\mathcal{U} \in \mathcal{F}$ e $\omega \in \Omega$, $1_{\{\mathcal{U}\}}(\omega) = 1$ se $\omega \in \mathcal{U}$, 0 caso contrário) e considera-se o seguinte sistema discreto linear Markoviano

$$x_{i+1} = F_{\Theta(i)}x_i + G_{\Theta(i)}u_i, \quad (4.1)$$

$$y_i = H_{\Theta(i)}x_i + D_{\Theta(i)}v_i, \quad (4.2)$$

sendo u_i, v_i ruídos gaussianos com média-zero e matrizes de covariância unitária, a variável aleatória $x_0 1_{\{\Theta(0)=j\}}$ com média e variância dadas por $\mathcal{E}(x_0 1_{\{\Theta(0)=j\}}) = \mu_j$ e $\mathcal{E}(x_0 x_0^* 1_{\{\Theta(0)=j\}}) = V_j$ para $\{j = 1, \dots, N\}$, o valor N é o número total de estados Markovianos. As variáveis u_i, v_i e $x_0 1_{\{\Theta(0)=j\}}$ são todas independentes entre si.

Para exemplificar, considera-se um espaço de estados Markovianos para o caso $N = 2$. Assim o sistema pode assumir duas formas: uma com F_1, G_1, H_1 e D_1 ou outra com F_2, G_2, H_2 e D_2 . A matriz $\mathcal{P}$ de probabilidade de transição dos estados Markovianos é definida como

$$\mathcal{P} = \begin{bmatrix} p_{11} & \dots & p_{1N} \\ \vdots & & \vdots \\ p_{N1} & \dots & p_{NN} \end{bmatrix}$$

que no caso $N = 2$ é escrita na forma

$$\mathcal{P} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}.$$

Na cadeia de Markov, estando o estado Markoviano no instante i em $\Theta(i) = 1$, existe uma probabilidade p_{11} do estado permanecer em $\Theta(i+1) = 1$ e p_{12} do estado saltar para $\Theta(i+1) = 2$. Note que a soma destes valores nas linhas de $\mathcal{P}$ deve ser igual a 1. A Figura 4.1 exemplifica a probabilidade de transição dos estados para o sistema com $N = 2$.

Para utilizar o filtro deduzido em COSTA (1994), deve-se fazer as seguintes de-

Figura 4.1: Cadeia de Markov com $N = 2$.

finições

$$\begin{aligned} z_{i,j} &:= x_i 1_{\{\Theta(i)=j\}}, \\ z_i &:= \begin{bmatrix} z_{i,1}^* & \dots & z_{i,N}^* \end{bmatrix}^*, \\ Z_i &:= \mathcal{E}(z_i z_i^*), \\ Z_{i,j} &:= \mathcal{E}(z_{i,j} z_{i,j}^*), \end{aligned}$$

O melhor estimador linear: $\hat{Z}_{i|l} := \mathcal{E}(\hat{z}_{i|l} \hat{z}_{i|l}^*),$

O erro de estimativa: $\tilde{Z}_{i|l} := \mathcal{E}(\tilde{z}_{i|l} \tilde{z}_{i|l}^*).$

Note que $Z_i = \text{diag}(Z_{i,j})$, ou seja, Z_i é uma matriz quadrada formada por $Z_{i,j}$, $j = 1, \dots, N$ na diagonal e nula nas outras posições. O filtro será escrito em função das seguintes matrizes

$$\mathcal{F} := \begin{bmatrix} p_{11}F_1 & \dots & p_{N1}F_N \\ \vdots & \dots & \vdots \\ p_{1N}F_1 & \dots & p_{NN}F_N \end{bmatrix}$$

$$\mathcal{D}_i := \begin{bmatrix} D_1 \pi_1(i)^{1/2} & \dots & D_N \pi_N(i)^{1/2} \end{bmatrix}$$

$$\mathcal{H} := \begin{bmatrix} H_1 & \dots & H_N \end{bmatrix}.$$

sendo o valor $\pi_j(i)$, a probabilidade de no instante i , o estado Markoviano ser j , que é calculado elevando a matriz de transição de probabilidades da cadeia de Markov à i -ésima potência.

$$\pi_j(i) = \pi_j(0)P^i \tag{4.3}$$

A estimativa $\hat{x}_{i|i}$ é dada por

$$\hat{x}_{i|i} = \sum_{j=1}^N \hat{z}_{i,j|i} \quad (4.4)$$

sendo que $\hat{z}_{i|i}$ satisfaz a seguinte equação de recursão

$$\hat{z}_{i|i} = \hat{z}_{i|i-1} + \hat{Z}_{i|i-1} H^* (H \hat{Z}_{i|i-1} H^* + D_i D_i^*)^{-1} (y_i - H \hat{z}_{i|i-1}), \quad (4.5)$$

$$\hat{z}_{i|i-1} = \mathcal{F} \hat{z}_{i-1|i-1}, \quad (4.6)$$

com $\hat{z}_{0|-1} = \zeta(0)$, $\zeta(i) = \mathcal{E}[z_i] = (\mu_1^*, \dots, \mu_N^*)^*$. Temos que a equação de Riccati é dada por

$$\tilde{Z}_{i|i-1} = Z_i - \hat{Z}_{i|i-1}, \quad \tilde{Z}_{i|i} = Z_i - \hat{Z}_{i|i}, \quad Z_i = \text{diag}(Z_{i,k}) \quad (4.7)$$

com

$$Z_{i+1,k} = \sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^* + \sum_{j=1}^N p_{jk} \pi_j(i) H_j H_j^*, \quad (4.8)$$

$$Z_{0,k} = V_k, \quad k \in \{1, \dots, N\} \quad (4.9)$$

$$\hat{Z}_{i|i} = \hat{Z}_{i|i-1} + \hat{Z}_{i|i-1} \mathcal{H}^* (\mathcal{H} \hat{Z}_{i|i-1} \mathcal{H}^* + \mathcal{D}_i \mathcal{D}_i^*)^{-1} \mathcal{H} \hat{Z}_{i|i-1}, \quad (4.10)$$

$$\hat{Z}_{i|i-1} = \mathcal{F} \hat{Z}_{i-1|i-1} \mathcal{F}^*, \quad (4.11)$$

$$\hat{Z}_{0|-1} = \zeta(0) \zeta(0)^*. \quad (4.12)$$

A forma para a Riccati em algoritmo *array* deduzida neste trabalho para o filtro acima é dada pelo seguinte teorema

Teorema 4.1.1 *A propagação dos erros de estimativa definida por (4.7) do filtro Markoviano calculado através do algoritmo array é definida pelo 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 $\hat{Z}_{0|-1}^{1/2} = (\zeta(0) \zeta(0)^*)^{1/2}$.

Passo 2: Calcular $\tilde{Z}_{i|i-1}^{1/2}$ utilizando uma matriz J -unitária λ_{J_1} de dimensões apropriadas (ver mais detalhes nos Apêndices C.3 e B.3):

$$\begin{bmatrix} Z_i & \mathcal{F} \hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix} \lambda_{J_1} = \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix}, \quad (4.13)$$

sendo que Z_i é dada por

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \quad (4.14)$$

sendo $\mathcal{L}_k = [L_{1k} \ L_{2k} \ \cdots \ L_{Nk}]$ e $\mathcal{M}_k = [M_{1k} \ M_{2k} \ \cdots \ M_{Nk}]$ com $L_{jk} = p_{jk}^{1/2} F_j Z_{i,j}^{1/2}$ e $M_{jk} = p_{jk}^{1/2} \pi_j^{1/2}(i) H_j$. $Z_{i,j}^{1/2}$ pode ser calculada usando uma matriz unitária λ_z tal que

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \lambda_z = \begin{bmatrix} Z_{i,1}^{1/2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & Z_{i,2}^{1/2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Z_{i,N}^{1/2} & 0 \end{bmatrix} \quad (4.15)$$

Passo 3: Calcular $\hat{Z}_{i|i}^{1/2}$ utilizando uma matriz unitária λ tal que

$$\begin{bmatrix} \tilde{Z}_{i|i-1}^{-1/2} & \mathcal{H}^* G_i^{1/2} & 0 \\ 0 & \hat{Z}_{i|i-1} \mathcal{H}^* \mathcal{D}_i^{1/2} & \hat{Z}_{i|i-1}^{1/2} \end{bmatrix} \lambda = \begin{bmatrix} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^* (\mathcal{D}_i \mathcal{D}_i^*)^{-1} \mathcal{H})^{1/2} & 0 \\ \hat{Z}_{i|i-1} \mathcal{H}^* (\mathcal{D}_i \mathcal{D}_i^*)^{-1} \mathcal{H} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^* (\mathcal{D}_i \mathcal{D}_i^*)^{-1} \mathcal{H})^{-1/2} & \hat{Z}_{i|i}^{1/2} \end{bmatrix} \quad (4.16)$$

e calcular $\tilde{Z}_{i|i}^{1/2}$ utilizando uma matriz J -unitária λ_{J_2} de dimensões apropriadas tal que

$$\begin{bmatrix} Z_i & \hat{Z}_{i|i}^{1/2} \end{bmatrix} \lambda_{J_2} = \begin{bmatrix} \tilde{Z}_{i|i}^{1/2} & 0 \end{bmatrix}. \quad (4.17)$$

e ainda calcular $\hat{Z}_{i|i-1}^{1/2}$ usando a transformação unitária λ_2

$$\mathcal{F} \hat{Z}_{i-1|i-1}^{1/2} \lambda_2 = \hat{Z}_{i|i-1}^{1/2}. \quad (4.18)$$

Prova: Tem-se que para a equação de Riccati da atualização da medida definida

em (4.10), pode-se usar o lema da inversão de matrizes

$$(\mathcal{H}\tilde{Z}_{i|i-1}\mathcal{H}^* + \mathcal{D}_i\mathcal{D}_i^*)^{-1} = (\mathcal{D}_i\mathcal{D}_i^*)^{-1} - (\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}. \quad (4.19)$$

A equação $\hat{Z}_{i|i}$ fica sendo

$$\begin{aligned} \hat{Z}_{i|i} &= \hat{Z}_{i|i-1} + \hat{Z}_{i|i-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}\hat{Z}_{i|i-1} - \\ &\hat{Z}_{i|i-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}\hat{Z}_{i|i-1}, \end{aligned} \quad (4.20)$$

escrevendo $\hat{Z}_{i|i}$ como complemento de Schur do termo (2, 2) da matriz

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

Definindo $\mathfrak{D}_i^{1/2} := (\mathcal{D}_i\mathcal{D}_i^*)^{-1/2}$, pode-se fatorar a matriz acima como

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

e usando as propriedades do complemento de Schur pode-se fatorar também como

$$\begin{bmatrix} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{1/2} & 0 \\ \hat{Z}_{i|i-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{-1/2} & \hat{Z}_{i|i}^{1/2} \\ (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{1/2} & 0 \\ \hat{Z}_{i|i-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{-1/2} & \hat{Z}_{i|i}^{1/2} \end{bmatrix} \times \quad (4.23)$$

que então define o algoritmo array para a Equação (4.10) dada pela Equação (4.16)

$$\begin{aligned} &\begin{bmatrix} \tilde{Z}_{i|i-1}^{-1/2} & \mathcal{H}^*\mathfrak{D}_i^{1/2} & 0 \\ 0 & \hat{Z}_{i|i-1}\mathcal{H}^*\mathfrak{D}_i^{1/2} & \hat{Z}_{i|i-1}^{1/2} \end{bmatrix} \lambda = \\ &\begin{bmatrix} (\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{1/2} & 0 \\ \hat{Z}_{i|i-1}\mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H}(\tilde{Z}_{i|i-1}^{-1} + \mathcal{H}^*(\mathcal{D}_i\mathcal{D}_i^*)^{-1}\mathcal{H})^{-1/2} & \hat{Z}_{i|i}^{1/2} \end{bmatrix}. \end{aligned} \quad (4.24)$$

Para calcular $\tilde{Z}_{i|i}$ na Equação (4.7) é necessário calcular Z_i . Deve-se notar que se

pode escrever $\tilde{Z}_{i|i}$ como $\text{diag}(Z_{i,k})$

$$Z_{i+1,k} = \sum_{j=1}^N p_{jk} F_j Z_{i,j} F_j^* + \sum_{j=1}^N p_{jk} \pi_j(i) H_j H_j^* \quad (4.25)$$

sendo que $Z_{0,k} = V_k$, $k \in \{1, \dots, N\}$. Pode-se escrever a equação de $\tilde{Z}_{i|i}$ como

$$\begin{bmatrix} Z_i^{1/2} & \mathcal{F} \hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix} J \begin{bmatrix} Z_i^{1/2} & \mathcal{F} \hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix}^* = \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix} J \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix}^T \quad (4.26)$$

sendo que Z_i é dada por

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \quad (4.27)$$

e ainda $\mathcal{L}_k = \begin{bmatrix} L_{1k} & L_{2k} & \cdots & L_{Nk} \end{bmatrix}$ e $\mathcal{M}_k = \begin{bmatrix} M_{1k} & M_{2k} & \cdots & M_{Nk} \end{bmatrix}$ com $L_{jk} = p_{jk}^{1/2} F_j Z_{i,j}^{1/2}$ e $M_{jk} = p_{jk}^{1/2} \pi_j^{1/2}(i) H_j$. $J = (I \oplus -I)$ (matriz cuja diagonal é formada por I e $-I$) tem dimensões apropriadas para que a identidade negativa $-I$ multiplique somente os termos de $\hat{Z}_{i-1|i-1}^{1/2}$. Usando o lema C.1.1, $Z_{i,j}^{1/2}$ pode ser calculada usando uma matriz unitária λ_z tal que

$$\begin{bmatrix} \mathcal{L}_1 & \mathcal{M}_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \mathcal{L}_2 & \mathcal{M}_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathcal{L}_N & \mathcal{M}_N \end{bmatrix} \lambda_z = \begin{bmatrix} Z_{i,1}^{1/2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & Z_{i,2}^{1/2} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & Z_{i,N}^{1/2} & 0 \end{bmatrix} \quad (4.28)$$

pois se cada matriz acima for multiplicada por sua tranposta, chega-se ao conjunto de equações dado em (4.8) que caracteriza Z_i , ou seja, uma relação do tipo $AA^* = BB^*$. Portanto, existe uma transformação J-ortogonal λ_{J_2} que resulta no seguinte algoritmo array

$$\begin{bmatrix} Z_i & \hat{Z}_{i|i}^{1/2} \end{bmatrix} \lambda_{J_2} = \begin{bmatrix} \tilde{Z}_{i|i}^{1/2} & 0 \end{bmatrix}. \quad (4.29)$$

Para a atualização no tempo, tem-se $\hat{Z}_{i|i-1} = \mathcal{F} \hat{Z}_{i-1|i-1} \mathcal{F}^*$ como

$$\hat{Z}_{i|i-1}^{1/2} = \mathcal{F} \hat{Z}_{i-1|i-1}^{1/2} \lambda_2. \quad (4.30)$$

É possível calcular $\tilde{Z}_{i|i-1}$ em (4.7) usando uma transformação J-unitária originária da seguinte expressão

$$\begin{bmatrix} Z_i & \mathcal{F}\hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix} J \begin{bmatrix} Z_i & \mathcal{F}\hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix}^* = \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix} J \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix}^* . \quad (4.31)$$

Portanto, o algoritmo *array* para o cálculo de $\tilde{Z}_{i|i-1}$, utilizando uma transformação J-unitária λ_{J_2} , é dado por

$$\begin{bmatrix} Z_i & \mathcal{F}\hat{Z}_{i-1|i-1}^{1/2} \end{bmatrix} \lambda_{J_2} = \begin{bmatrix} \tilde{Z}_{i|i-1}^{1/2} & 0 \end{bmatrix} . \quad \diamond \quad (4.32)$$

4.1.1 Implementação

Para o cálculo do algoritmo *array* desenvolvido neste capítulo, utiliza-se como exemplo o sistema Markoviano de dois estados apresentado em COSTA (1994)

Estado 1:

$$F_1 = \begin{bmatrix} .7 & 0 \\ .1 & .1 \end{bmatrix}, H_1 = \begin{bmatrix} 2 & 0 \end{bmatrix}, G_1 = \begin{bmatrix} 3.9 & 0 \\ 0 & 5.4 \end{bmatrix}, D_1 = \begin{bmatrix} 1.6 \end{bmatrix}$$

Estado 2:

$$F_2 = \begin{bmatrix} .6 & 0 \\ .1 & .2 \end{bmatrix}, H_2 = \begin{bmatrix} 2 & 0 \end{bmatrix}, G_2 = \begin{bmatrix} 3.9 & 0 \\ 0 & 5.4 \end{bmatrix}, D_2 = \begin{bmatrix} 1.6 \end{bmatrix}$$

Matriz de probabilidade de transição:

$$P = \begin{bmatrix} .9 & .1 \\ .1 & .9 \end{bmatrix} .$$

Foram calculados os valores singulares de $\tilde{Z}_{i|i-1}$ para três diferentes implementações assim como no caso robusto: ponto-flutuante, ponto-fixo implementação por equações explícitas e por algoritmos *array*. Nestes exemplos também foram utilizados arquitetura de ponto-fixo com 16-bits onde o primeiro bit determinou o sinal do número (+ ou -) porém, o último bit o menos significativo foi escalonado para 2^{-9} significando que a implementação tem precisão de 0.002 e pode representar um número de -65.543 a 65.543. Estas duas implementações foram feitas via Matlab® software usando o *fix-point Simulink toolbox*.

Tipo	Implementação	VS_1	VS_2	VS_3	VS_4
Markoviano	Equações	533.0976	14.5409	47.2012	0.0733
	Alg. <i>array</i>	0.0348	0.0030	0.0083	0.0004

Tabela 4.1: Valores MSE para os valores singulares de $\tilde{Z}_{i|i-1}$

Usando a Tabela 4.1 ou a Figura 4.2, pode-se notar novamente a vantagem do algoritmo *array* perante a implementação por equações explícitas. O cálculo do MSE (*mean-square error*) para um valor singular VS_γ é dado por $E_\gamma = \frac{\sqrt{\sum_{i=1}^N (VS_{fi,\gamma} - VS_{i,\gamma})^2}}{N}$ onde γ define um dos quatro valores singulares calculados para cada simulação, VS_f é o valor singular calculado usando ponto flutuante e i é o instante de tempo.

Em alguns cálculos, o ponto fixo não conseguiu representar bem os números, ocasionando erros numéricos. Já na implementação em *array*, por causa do número de condição das matrizes ser menor e por causa da menor faixa dinâmica dos números, o cálculo efetuado através do ponto fixo foi satisfatório.

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

Capítulo 5

Algoritmo de Paige

Simulações e algumas análises matemáticas mostram que o desempenho das diferentes variações das equações recursivas dos filtros de Kalman em implementação com precisão finita depende muito do condicionamento das matrizes R_i e em menos grau da dupla $\{Q_i, F_i\}$ VERHAEGEN & DOOREN (1986). Quando as matrizes são bem condicionadas todas as implementações são satisfatórias e o usuário deve usar a que mais lhe convém. A melhor maneira de lidar com mal condicionamento é rever o modelo assumido e a precisão com que os parâmetros são conhecidos ou determinados. Esta precisão geralmente determina o nível de esforço a ser gasto na escolha do algoritmo.

Com isto em mente, será introduzida uma outra variação das equações recursivas do filtro de Kalman. Esta variação evita uma perda de precisão potencial quando se faz os produtos matriciais $\{F_i P_i^{1/2}, H_i P_i^{1/2}\}$ usados em algoritmos *array* e também é facilmente adaptável para utilizar em sistemas singulares, que é o objetivo principal deste capítulo. Esta formulação foi desenvolvida por Paige PAIGE (1985) para problemas de filtragem no espaço de estados.

Como já foi visto nos capítulos anteriores, este trabalho tem considerado duas abordagens para se obter a solução para este problema de estimativa. A primeira abordagem, chamada de *filtragem de covariância*, obtém a matriz de covariância $P_{i|i}$ de $P_{i|i-1}$ incorporando o passo de medida (5.2) e então obtém $P_{i+1|i}$ de $P_{i|i}$ usando o passo temporal (5.1). A outra abordagem obtém $P_{i|i}^{-1}$ de $P_{i|i-1}^{-1}$ usando (5.2) e $P_{i+1|i}^{-1}$ de $P_{i|i}^{-1}$ usando (5.1). Esta segunda abordagem é normalmente chamada de *filtragem de informação* por causa da relação da inversa da covariância com a *Matriz de Informação*

de Fisher, veja MAYBECK (1979). Uma outra terminologia para este caso é *filtragem de covariância inversa*. Considera-se o seguinte sistema singular

$$E_i x_{i+1} = F_i x_i + G_i w_i \quad (5.1)$$

$$y_i = H_i x_i + v_i, \quad (5.2)$$

sendo as variáveis aleatórias u_i, v_i, x_0 de média zero e

$$\mathcal{E} \left(\begin{bmatrix} w_i \\ v_i \\ x_0 \end{bmatrix} \begin{bmatrix} w_j \\ v_j \\ x_0 \end{bmatrix}^T \right) = \begin{bmatrix} Q_i \delta_{ij} & S_i \delta_{ij} & 0 \\ S_i^* \delta_{ij} & R_i \delta_{ij} & 0 \\ 0 & 0 & \Pi_0 \end{bmatrix} \quad (5.3)$$

as matrizes $\{F_i, G_i, H_i, \Pi_0, Q_i, S_i, R_i\}$ são conhecidas, os valores Q_i e R_i são as autocovariâncias de w_i e v_i respectivamente e S_i é covariância cruzada entre os ruídos. O objetivo é usar a saída y_i para estimar x_i . A abordagem adotada é achar a melhor estimativa linear não polarizada, que expressa a estimativa de mínima variância, veja ANDERSON & MOORE (1979). Seja $\hat{x}_{i|j}$ a estimativa de mínima variância de x_i dado $y_1, \dots, y_j$, o erro da estimativa com a respectiva covariância são definidos como

$$\tilde{x}_{i|j} \triangleq \hat{x}_{i|j} - x_i, \quad (5.4)$$

$$P_{i|j} \triangleq \mathcal{E}[(\tilde{x}_{i|j} - \mathcal{E}[\tilde{x}_{i|j}])(\tilde{x}_{i|j} - \mathcal{E}[\tilde{x}_{i|j}])^*]. \quad (5.5)$$

Normalmente busca-se definir estimativas não polarizadas, portanto

$$\mathcal{E}[\tilde{x}_{i|j}] = 0. \quad (5.6)$$

No algoritmo de Paige também são utilizados fatores raiz quadrada nos cálculos do filtro

$$P_{i|i-1} = P_{i|i-1}^{1/2} P_{i|i-1}^{*/2}. \quad (5.7)$$

Assume-se que x_0 tem as seguintes características

$$x_0 \text{ com } \mathcal{E}[x_0] = 0, \quad \mathcal{E}[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^*] = \Pi_0 \quad (5.8)$$

sendo que a média e a covariância são conhecidas. Na abordagem definida em Paige,

é utilizado o argumento de que o uso de matrizes de covariância apenas, ou o uso de inversas de matrizes de covariância apenas, é desnecessariamente restritiva. Para isto, Paige usa duas representações para descrever a estrutura de covariância do vetor de estados x_i , uma baseada no filtro de covariância e outra baseada na inversa da covariância, que serão descritas a seguir.

5.1 Estrutura de covariância

Para resolver o problema das estimativas iniciais e prosseguir na busca de um algoritmo alternativo robusto numericamente, define-se uma estrutura de representação do sistema a partir da definição de covariância. São utilizadas duas maneiras distintas de representação de um vetor aleatório x qualquer

$$x = \bar{x} + Nu, \quad (5.9)$$

ou seja, x tem média $\mathcal{E}[x] = \bar{x}$ e variância $\mathcal{E}[xx^*] = NN^*$. Para isto define-se u com $\mathcal{E}[u] = 0$ e $\mathcal{E}[uu^*] = I$. Ainda pode-se representar um vetor aleatório x qualquer como

$$Ux = b + u \quad (5.10)$$

sendo que as informações de média e variância são estabelecidas de maneira implícita. Quando $N = U^{-1}$ elas são equivalentes porém, (5.9) permite que a combinação linear dos elementos de x seja constante (conhecidos *a priori*) fazendo com que N não tenha posto linha pleno, enquanto (5.10) permite a possibilidade, quando U não tem posto coluna pleno, de que não se tenha nenhuma informação sobre a parte de x que está no espaço nulo de U .

As duas representações são matematicamente distintas, exceto quando N e U são quadradas e não singulares. A primeira representação é originária do filtro de covariância, enquanto que a segunda é originária do filtro de informação. Define-se uma estrutura de representação do vetor utilizando-se as definições dos filtros de covariância e de informação juntas

$$Ux = b + Nu, \quad (5.11)$$

sendo que $\mathcal{E}[u] = 0$, $\mathcal{E}[uu^*] = I$. A estrutura da Equação (5.11) pode ser usada

para representar condições iniciais em filtragem de covariância, filtragem de covariância inversa e qualquer combinação possível de ambas. Com algumas pequenas mudanças para facilitar seu uso mais adiante, a condição inicial pode ser expressa como

$$\hat{b}_0 = \hat{U}_0 x_0 + \hat{N}_0 \hat{u}_0. \quad (5.12)$$

Os vetores de ruídos w_i e v_i também têm representações diferentes em filtros de covariância e em filtragem de informação PAIGE (1985). De maneira geral, considerando-se as combinações entre essas representações, pode-se escrever

$$\begin{bmatrix} R_i^l & 0 \\ S_i^l & Q_i^l \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = - \begin{bmatrix} R_i^r & 0 \\ S_i^r & Q_i^r \end{bmatrix} \begin{bmatrix} \tilde{v}_i \\ \tilde{w}_i \end{bmatrix}, \quad (5.13)$$

sendo que $\begin{bmatrix} \tilde{v}_i \\ \tilde{w}_i \end{bmatrix}$ são ruídos normalizados com média nula e variância unitária, l e r são índices que representam *esquerda* e *direita*, respectivamente. Diferentes representações de (5.13) levam a algoritmos de filtragem diferentes. Assim, mais adiante, será apresentado um algoritmo para um tipo de representação da estrutura (5.13). Um modelo dinâmico discreto em uma janela de tempo $i = 1, \dots, N$ pode ser formulado levando em consideração (5.1), (5.2) e (5.13)

$$\begin{bmatrix} b_0 \\ \eta_0 \\ \eta_1 \\ \vdots \\ \eta_N \end{bmatrix} = \begin{bmatrix} \mathfrak{U}_0 & 0 & 0 & \dots & 0 & 0 \\ \mathfrak{R}_0 & \mathfrak{E}_0 & 0 & \dots & 0 & 0 \\ 0 & \mathfrak{R}_1 & \mathfrak{E}_1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \mathfrak{R}_N & \mathfrak{E}_N \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} \mathfrak{N}_0 & 0 & 0 & \dots & 0 \\ 0 & \mathfrak{S}_0 & 0 & \dots & 0 \\ 0 & 0 & \mathfrak{S}_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mathfrak{S}_N \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} \quad (5.14)$$

$$\text{sendo } b_0 = \begin{bmatrix} \hat{b}_0 \end{bmatrix}, \eta_i = \begin{bmatrix} y_i \\ 0 \\ 0 \\ d_i \end{bmatrix}, \mathfrak{U}_0 = \begin{bmatrix} \hat{U}_0 & 0 & 0 \end{bmatrix}, \mathfrak{R}_i = \begin{bmatrix} H_i & I & 0 \\ 0 & R_i^l & 0 \\ 0 & S_i^l & Q_i^l \\ F_i & 0 & G_i \end{bmatrix}, \mathfrak{E}_i = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ E_{i+1} & 0 & 0 \end{bmatrix},$$

$$\mathfrak{x}_i = \begin{bmatrix} x_i \\ v_i \\ w_i \end{bmatrix}, \mathfrak{N}_0 = [\hat{N}_0], \mathfrak{S}_i = \begin{bmatrix} 0 & 0 \\ R_i^r & 0 \\ S_i^r & Q_i^r \\ 0 & 0 \end{bmatrix}, \hat{u}_0 = [\hat{u}_0], u_i = \begin{bmatrix} \tilde{v}_i \\ \tilde{w}_i \end{bmatrix}.$$

Pode-se expressar (5.14) de maneira compacta como

$$y = Ax + Lu, \text{ sendo } \mathcal{E}[u] = 0 \text{ e } \mathcal{E}[uu^*] = I. \quad (5.15)$$

Note que (5.14) inclui uma entrada conhecida d_i , que é opcional para o problema de filtragem. O sistema está definido para N instantes de tempo. A Equação (5.14) mostra a estrutura na qual o algoritmo de filtragem será formulado, que é de natureza recursiva. Deve-se notar que a primeira linha mostra a condição inicial do problema e o restante das linhas mostra a estrutura do problema para cada instante i . Como não existem produtos de matrizes em (5.14), que tem como sub-blocos os dados originais, qualquer algoritmo estável para achar a estimativa ótima para x em (5.15) será numericamente estável para o problema original de acordo com PAIGE (1985).

Em geral, a formulação (5.14) e (5.15) facilita o entendimento do problema e facilita a formulação de algoritmos estáveis e eficientes numericamente. É o que será apresentado na próxima seção. Note que (5.15) tem a mesma forma de (5.12). Para ilustrar uma outra maneira de se resolver (5.15), é mostrado em KOUROUKLIS & PAIGE (1981) que a estimativa de variância mínima para x em (5.15) é a solução do problema de mínimos quadrados generalizados

$$\min_{x,u} u^* u \text{ sujeito a } y = Ax + Lu, \quad (5.16)$$

e é mostrado como resolvê-lo de maneira estável. Como (5.12) tem a forma de (5.15), a estimativa $\hat{x}_0$ para x_0 é solução de

$$\min_{x_0, \hat{u}_0} \hat{u}_0^* \hat{u}_0 \text{ sujeito a } \hat{b}_0 = \hat{U}_0 x_0 + \hat{N}_0 \hat{u}_0, \quad (5.17)$$

se $\hat{U}_0$ tem posto linha pleno

$$\hat{U}_0 \hat{x}_0 = \hat{b}_0. \quad (5.18)$$

Em PAIGE (1985) são apresentadas outras alternativas para resolver o sistema (5.15).

5.2 O Algoritmo de Paige

O modelo (5.15) com estrutura (5.14), apresenta mais informações que as necessárias para a definição da estimativa filtrada. Será apresentado nesta seção um algoritmo para apenas algumas das especializações de (5.14). É um algoritmo para o modelo com todas condições iniciais de (5.12), mas com apenas os fatores da matriz de covariância dos ruídos v_i e w_i fornecidos. Isto equivale a fazer $R_i^l = I$, $S_i^l = 0$ e $Q_i^l = I$ em (5.14) e então estas matrizes unitárias podem ser eliminadas através de combinações lineares chegando a um modelo redefinido abaixo, deduzido em PAIGE (1985).

$$\begin{bmatrix} \mathcal{Y}_0 \\ \mathcal{Y}_1 \\ \vdots \\ \mathcal{Y}_N \end{bmatrix} = \begin{bmatrix} \mathcal{U}_0 & 0 & 0 & \dots & 0 & 0 \\ \mathcal{R}_0 & \mathcal{E}_0 & 0 & \dots & 0 & 0 \\ 0 & \mathcal{R}_1 & \mathcal{E}_1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \mathcal{R}_N & \mathcal{E}_N \end{bmatrix} \begin{bmatrix} \mathcal{X}_0 \\ \mathcal{X}_1 \\ \vdots \\ \mathcal{X}_N \end{bmatrix} + \begin{bmatrix} \mathcal{N}_0 & 0 & 0 & \dots & 0 \\ 0 & \mathcal{S}_0 & 0 & \dots & 0 \\ 0 & 0 & \mathcal{S}_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mathcal{S}_N \end{bmatrix} \begin{bmatrix} \hat{\mathcal{U}}_0 \\ \mathcal{U}_0 \\ \mathcal{U}_1 \\ \vdots \\ \mathcal{U}_N \end{bmatrix} \quad (5.19)$$

$$\text{sendo } B_0 = \begin{bmatrix} \hat{b}_0 \end{bmatrix}, \mathcal{Y}_i = \begin{bmatrix} y_i \\ d_i \end{bmatrix}, \mathcal{U}_0 = \begin{bmatrix} \hat{U}_0 \end{bmatrix}, \mathcal{R}_i = \begin{bmatrix} H_i \\ F_i \end{bmatrix}, \mathcal{E}_i = \begin{bmatrix} 0 \\ -E_{i+1} \end{bmatrix}, \mathcal{X}_i = \begin{bmatrix} x_i \end{bmatrix},$$

$$\mathcal{N}_0 = \begin{bmatrix} \hat{N}_0 \end{bmatrix}, \mathcal{S}_i = \begin{bmatrix} R_i^{1/2} & 0 \\ S_i & Q_i^{1/2} \end{bmatrix}, \hat{\mathcal{U}}_0 = \begin{bmatrix} \hat{u}_0 \end{bmatrix}, \mathcal{U}_i = \begin{bmatrix} v_i \\ w_i \end{bmatrix} \text{ ou}$$

$$y = Ax + Lu \quad (5.20)$$

sendo que por simplicidade foram feitas as seguintes mudanças de notação

$$\tilde{v}_i \Rightarrow v_i, \quad \tilde{w}_i \Rightarrow w_i, \quad -R_i^r \Rightarrow R_i^{1/2}, \quad -G_i S_i^r \Rightarrow S_i, \quad -G_i Q_i^r \Rightarrow Q_i^{1/2}. \quad (5.21)$$

O Teorema 5.2.1 apresenta a versão modificada do algoritmo de filtragem definido por Paige que calcula a estimativa filtrada do estado. Para que se possa acompanhar melhor o teorema e a respectiva prova, a Equação (5.19) é reescrita em (5.22) de maneira

explícita

$$\begin{bmatrix} \hat{b}_0 \\ y_0 \\ d_0 \\ y_1 \\ d_1 \\ \vdots \\ y_N \\ d_N \end{bmatrix} = \begin{bmatrix} \hat{U}_0 & 0 & 0 & 0 & \dots & 0 & 0 \\ H_0 & 0 & 0 & 0 & \dots & 0 & 0 \\ F_0 & E_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & H_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & F_1 & E_2 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & H_N & 0 \\ 0 & 0 & 0 & 0 & \dots & F_N & E_{N+1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} \hat{N}_0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & R_0^{1/2} & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & S_0 & Q_0^{1/2} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & R_1^{1/2} & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & S_1 & Q_1^{1/2} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & R_N^{1/2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & S_N & Q_N^{1/2} \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ v_0 \\ w_0 \\ v_1 \\ w_1 \\ \vdots \\ v_N \\ w_N \end{bmatrix} \quad (5.22)$$

Teorema 5.2.1 *A estimativa filtrada do sistema (5.1), reescrito de acordo com (5.22), é dada pelo seguinte algoritmo*

1. Dadas as condições iniciais $\hat{U}_0 = I$, $\hat{b}_0 = 0$ e $\hat{N}_0 = (P_0^{-1} + H_0^* R_0^{-1} H_0)^{-1/2}$, se $\begin{bmatrix} E_i \\ H_i \end{bmatrix}$ for posto coluna pleno, o que garante que $\hat{U}_i$ seja posto coluna pleno, definir uma transformação ortogonal $\hat{T}_i$ que triangularize pela esquerda o seguinte array

$$\hat{T}_i \begin{bmatrix} \hat{U}_i \\ H_i \end{bmatrix} = \begin{bmatrix} 0 \\ \bar{U}_i \end{bmatrix}, \quad (5.23)$$

e calcular

$$\hat{T}_i \begin{bmatrix} \hat{b}_i \\ y_i \end{bmatrix} =: \begin{bmatrix} r_i \\ \tilde{b}_i \end{bmatrix}, \quad (5.24)$$

$$\hat{T}_i \begin{bmatrix} \hat{N}_i & 0 \\ 0 & R_i^{1/2} \end{bmatrix} =: \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix}. \quad (5.25)$$

2. Achar uma matriz unitária $\hat{P}_i$ que triangularize

$$\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \\ 0 & S_0 \end{bmatrix} \hat{P}_i = \begin{bmatrix} \tilde{N}_i & 0 \\ \tilde{R}_i & \tilde{R}_i \\ \tilde{S}_i & \tilde{S}_i \end{bmatrix} \quad (5.26)$$

sendo $\tilde{N}_i$ posto coluna pleno, calcular

$$\tilde{v}_i = \tilde{N}_i^{-1} r_i, \quad (5.27)$$

$$\tilde{b}_i = \tilde{b}_i - \tilde{R}_i \tilde{v}_i, \quad (5.28)$$

$$\tilde{d}_i = \tilde{d}_i - \tilde{S}_i \tilde{v}_i, \quad (5.29)$$

$$(5.30)$$

3. Definir uma transformação ortogonal $\bar{T}_i$ tal que triangularize

$$\bar{T}_i \begin{bmatrix} \bar{U}_i & 0 \\ \bar{F}_i & E_{i+1} \end{bmatrix} = \begin{bmatrix} U_i & U_{0,i} \\ 0 & \hat{U}_{i+1} \end{bmatrix}, \quad (5.31)$$

e calcular

$$\bar{T}_i \begin{bmatrix} \bar{R}_i & 0 \\ \bar{S}_i & Q_i^{1/2} \end{bmatrix} =: \begin{bmatrix} \bar{X}_{11} & \bar{X}_{12} \\ \bar{X}_{21} & \bar{X}_{22} \end{bmatrix} \quad (5.32)$$

$$\bar{T}_i \begin{bmatrix} \bar{b}_i \\ \bar{d}_i \end{bmatrix} =: \begin{bmatrix} c_i \\ \hat{b}_{i+1} \end{bmatrix}. \quad (5.33)$$

4. Encontrar uma transformação $\bar{P}_i$ tal que triangularize

$$\begin{bmatrix} \bar{X}_{11} & \bar{X}_{12} \\ \bar{X}_{21} & \bar{X}_{22} \end{bmatrix} \bar{P}_i =: \begin{bmatrix} N_i & \hat{N}_{0,i} \\ 0 & \hat{N}_{i+1} \end{bmatrix}. \quad (5.34)$$

calcular

$$P_{i|i}^{1/2} = \bar{U}_i^{-1} \bar{R}_i, \quad (5.35)$$

$$\hat{x}_{i|i} = \bar{U}_i^{-1} \bar{b}_i. \quad (5.36)$$

e retornar ao passo 1 para calcular as estimativas da próxima iteração.

Prova: Para o instante $i = 0$, a transformação para o primeiro passo de medida usando as matrizes $\hat{T}_0$ e $\hat{P}_0$ deve ter a forma

$$\begin{bmatrix} \hat{T}_0 & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{U}_0 & \hat{N}_0 & 0 \\ H_0 & 0 & R_0^{1/2} \\ F_0 & 0 & S_0 \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \hat{P}_0 \end{bmatrix} = \begin{bmatrix} 0 & * & * \\ \bar{U}_0 & * & * \\ F_0 & 0 & S_0 \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \hat{P}_0 \end{bmatrix} = \begin{bmatrix} 0 & \tilde{N}_0 & 0 \\ \bar{U}_0 & \tilde{R}_0 & \bar{R}_0 \\ F_0 & \tilde{S}_0 & \bar{S}_0 \end{bmatrix}, \quad (5.37)$$

sendo que $\bar{U}_0$ tem posto linha pleno e $\tilde{N}_0$ tem posto coluna pleno. $\hat{T}_0$ pode ser uma matriz ortogonal ou o produto de matrizes de eliminação qualquer. Já $\hat{P}_0$ deve ser somente ortogonal, de forma a manter a igualdade do sistema aplicando $\hat{T}_0$ a y e $\hat{P}_0^*$ ao ruído

$$\hat{T}_0 \begin{bmatrix} \hat{b}_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} r_0 \\ \tilde{b}_0 \end{bmatrix}, \quad \hat{P}_0^* \begin{bmatrix} \hat{u}_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} \tilde{v}_0 \\ \bar{u}_1 \end{bmatrix} \quad (5.38)$$

sendo $\begin{bmatrix} \tilde{v}_0 \\ \bar{u}_1 \end{bmatrix}$ variáveis aleatórias com média zero e variância unitária. As três primeiras linhas do modelo transformado são definidas como (note que $\hat{P}_0 \hat{P}_0^* = I$)

$$\begin{bmatrix} r_0 \\ \tilde{b}_0 \\ d_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ \bar{U}_0 & 0 \\ F_0 & E_1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} + \begin{bmatrix} \tilde{N}_0 & 0 & 0 \\ \tilde{R}_0 & \bar{R}_0 & 0 \\ \tilde{S}_0 & \bar{S}_0 & Q_0^{1/2} \end{bmatrix} \begin{bmatrix} \tilde{v}_0 \\ \bar{u}_0 \\ w_0 \end{bmatrix}. \quad (5.39)$$

Pode-se calcular $\tilde{v}_0$ em

$$\tilde{N}_0 \tilde{v}_0 = r_0, \quad (5.40)$$

sendo $\tilde{N}_0$ posto coluna pleno. Calculado $\tilde{v}_0$, pode-se eliminá-lo se forem definidos

$$\bar{b}_0 = \tilde{b}_0 - \tilde{R}_0 \tilde{v}_0, \quad \bar{d}_0 = d_0 - \tilde{S}_0 \tilde{v}_0. \quad (5.41)$$

Tem-se o seguinte modelo com a linha de $\tilde{v}_0$ descartada

$$\begin{bmatrix} \bar{b}_0 \\ \bar{d}_0 \end{bmatrix} = \begin{bmatrix} \bar{U}_0 & 0 \\ F_0 & E_1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} + \begin{bmatrix} \bar{R}_0 & 0 \\ \bar{S}_0 & Q_0^{1/2} \end{bmatrix} \begin{bmatrix} \bar{u}_0 \\ w_0 \end{bmatrix}. \quad (5.42)$$

A estimativa para o primeiro passo segue da primeira linha. Como $\bar{U}_0$ é invertível, $\hat{x}_{0|0}$ deve satisfazer

$$\bar{b}_0 = \bar{U}_0 \hat{x}_{0|0} \quad (5.43)$$

pois isolando x_0 na primeira linha

$$\begin{aligned}\bar{b}_0 &= \bar{U}_0 x_0 + \bar{R}_0 \bar{u}_0 \\ x_0 &= \bar{U}_0^{-1} \bar{b}_0 - \bar{U}_0^{-1} \bar{R}_0 \bar{u}_0\end{aligned}\tag{5.44}$$

tem-se uma estrutura como em (5.9), ou seja, a variável aleatória x_0 tem média $\bar{U}_0^{-1} \bar{b}_0$ e variância $\bar{U}_0^{-1} \bar{R}_0 \bar{R}_0^* \bar{U}_0^{-*}$. A estimativa ótima para x_0 é a sua média, ou seja, a estimativa filtrada $\hat{x}_{0|0}$ é igual a $\bar{U}_0^{-1} \bar{b}_0$. A estimativa preditora $\hat{x}_{1|0}$ obedece a seguinte igualdade se E_1 tiver posto linha pleno

$$\bar{d}_0 = F_0 \hat{x}_{0|0} + E_1 \hat{x}_{1|0}.\tag{5.45}$$

De (5.12), tem-se

$$\bar{b}_0 = \bar{U}_0 x_0 + \bar{R}_0 \bar{u}_0, \text{ sendo } \mathcal{E}[\bar{u}_0] = 0 \text{ e } \mathcal{E}[\bar{u}_0 \bar{u}_0^*] = I,\tag{5.46}$$

que inclui a primeira medida. Com (5.43) a matriz de covariância do erro é representada por

$$\bar{U}_0(\hat{x}_{0|0} - x_0) = \bar{R}_0 \bar{u}_0.\tag{5.47}$$

ou seja, a variância $\bar{U}_0^{-1} \bar{R}_0 \bar{R}_0^* \bar{U}_0^{-*}$ calculada em (5.44) é a covariância do erro $P_{0|0}$. Na atualização do tempo, deve-se eliminar F_0 e $\bar{S}_0$ em (5.42) usando as transformações $\bar{T}_0$ e $\bar{P}_0$ em

$$\begin{aligned}& \bar{T}_0 \begin{bmatrix} \bar{U}_0 & 0 & | & \bar{R}_0 & 0 \\ F_0 & E_1 & | & \bar{S}_0 & Q_0^{1/2} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \bar{P}_0 \end{bmatrix} = \\ &= \begin{bmatrix} U_0 & U_{01} & | & * & * \\ 0 & \hat{U}_1 & | & * & * \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \bar{P}_0 \end{bmatrix} = \begin{bmatrix} U_0 & U_{01} & | & N_0 & \hat{N}_{01} \\ 0 & \hat{U}_1 & | & 0 & \hat{N}_1 \end{bmatrix}\end{aligned}\tag{5.48}$$

$$\bar{T}_0 \begin{bmatrix} \bar{b}_0 \\ \bar{d}_0 \end{bmatrix} = \begin{bmatrix} \hat{b}_{0|0} \\ \hat{b}_1 \end{bmatrix}, \quad \bar{P}_0^* \begin{bmatrix} \bar{u}_0 \\ w_0 \end{bmatrix} = \begin{bmatrix} u_0 \\ \hat{u}_1 \end{bmatrix}, \quad U_0 \text{ posto linha pleno}\tag{5.49}$$

O sistema então se torna

$$\begin{bmatrix} \hat{b}_{0|0} \\ \hat{b}_1 \\ y_1 \\ d_1 \\ \vdots \\ y_N \\ d_N \end{bmatrix} = \begin{bmatrix} U_0 & U_{01} & 0 & \dots & 0 & 0 \\ 0 & \hat{U}_1 & 0 & \dots & 0 & 0 \\ 0 & H_1 & 0 & \dots & 0 & 0 \\ 0 & F_1 & E_2 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & H_N & 0 \\ 0 & 0 & 0 & \dots & F_N & E_{N+1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} N_0 & \hat{N}_{01} & 0 & 0 & \dots & 0 & 0 \\ 0 & \hat{N}_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & R_1^{1/2} & 0 & \dots & 0 & 0 \\ 0 & 0 & S_1 & Q_1^{1/2} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & R_N^{1/2} & 0 \\ 0 & 0 & 0 & 0 & \dots & S_N & Q_N^{1/2} \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ \hat{u}_1 \\ v_1 \\ w_1 \\ \vdots \\ v_N \\ w_N \end{bmatrix}. \quad (5.50)$$

Como U_0 tem posto linha pleno, pode-se desconsiderar a primeira linha e a primeira coluna de (5.50) e resolver o problema resultante que é similar ao problema (5.22), e então o processo pode ser continuado da mesma maneira com valores atualizados do tempo. A condição inicial para x_1 é dada por

$$\hat{b}_1 = \hat{U}_1 x_1 + \hat{N}_1 \hat{u}_1. \quad \diamond \quad (5.51)$$

Salienta-se aqui que a estimativa filtrada não estava explicitamente definida nos textos originais, mesmo para a filtragem no espaço de estados. A estimativa filtrada foi definida a partir da dedução do algoritmo apresentado no Teorema 5.2.1, para sistemas singulares, sendo portanto mais uma contribuição dos estudos de algoritmos apresentados nesse trabalho para a estimativa de sistemas singulares.

5.3 Implementação

Para exemplificar o uso da estimativa de sistemas singulares são apresentadas as Figuras 5.1(a) e 5.1(b) que mostram as estimativas filtradas e os respectivos valores sin-

gulares da matriz P , calculados através das equações nominais do filtro de Kalman para sistemas singulares definidas em (2.40)-(2.41) e do algoritmo apresentado no Teorema 5.2.1 para o sistema

$$E_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, F_i = \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.6 & 0 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}, H_i = \begin{bmatrix} 1.4 & 0.8 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

$$Q_i = \begin{bmatrix} 1.2 & 0 & 0 \\ 0 & 1.6 & 0 \\ 0 & 0 & 2 \end{bmatrix}, R_i = \begin{bmatrix} 1.5 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1.2 \end{bmatrix}.$$

Pode-se observar que ambos os algoritmos forneceram os mesmos resultados. Apesar dos resultados similares, o algoritmo apresentado no Teorema 5.2.1, de acordo com PAIGE (1985b) é mais robusto numericamente pois além de ter propriedades de robustez parecidas com os algoritmos *array*, não existe perda de informação por arredondamento das multiplicações $F_{i-1}^* Q_{i-1}^{-1/2}$ ou $H_i^* R_i^{-1/2}$ existentes no pre-*array* da estimativa nominal filtrada nominal.

Figura 5.1: (a) Estimativa dos estados e (b) valores singulares de $P_{i,i}$, calculados através da equação nominal do filtro de Kalman para sistemas singulares e do algoritmo apresentado no Teorema 5.2.1.

Capítulo 6

Conclusão

Nesta dissertação foram feitos estudos sobre implementações computacionais alternativas para problemas de estimativa utilizando algoritmos *array* e filtros de informação. Estas alternativas minimizam problemas causados por erros de arredondamento e mal condicionamento de matrizes pois reduzem a faixa dinâmica das matrizes calculadas, ou seja, os valores calculados variam dentro de uma faixa reduzida. As simulações apresentadas no texto mostram a viabilidade dos algoritmos propostos.

As principais contribuições originais deste trabalho estão resumidas a seguir.

- Foram deduzidos filtros de informação, nominais e robustos, para sistemas singulares discretos no tempo. Também foram apresentados os respectivos algoritmos *array* para as versões de informação. Em KAILATH et al. (2000), uma condição suficiente para a existência de P_i^{-1} para os filtros de Kalman na forma de informação é a invertibilidade da matriz de parâmetros F_i (veja Lema 9.5.1, p. 343). Os filtros de informação desenvolvidos neste trabalho não têm esta restrição.
- Neste trabalho também foi deduzido um algoritmo *array* para um filtragem de um sistema linear sujeito a saltos Markovianos que estima sistemas dinâmicos vulneráveis a mudanças abruptas em suas estruturas. Não foi encontrada na literatura uma formulação similar utilizando algoritmos *array* para a filtragem deste tipo de sistema. Resultados de simulação mostram as vantagens de se usar este tipo de abordagem também para este caso.

Referências Bibliográficas

ANDERSON & MOORE (1979) ANDERSON, B. & MOORE, J. (1979). *Optimal Filtering*. Prentice-Hall, New Jersey.

ATHANS et al. (1977) ATHANS, M., CASTANON, D., DUNN, K. P., GREENE, C. S., LEE, W. H., SANDELL, N. R., & WILLSKY, A. S. (1977). The stochastic control of the F-8C aircraft using a multiple model adaptive control (MMAC) method - Part I: Equilibrium flight. *IEEE Transactions on Automatic Control*, 22:768–780.

→ CAMPOS (2004) CAMPOS, J. (2004). *Filtragem robusta para sistemas singulares discretos no tempo*. Tese de Doutorado, Escola de Engenharia de São Carlos - USP.

CHISCI & ZAPPA (1992) CHISCI & ZAPPA (1992). Square Root Kalman Filtering of Descriptor Systems. *System and Control Letters*, 19:325–334.

COSTA & do Val (2002) COSTA, E. F. & do Val, J. B. R. (2002). Weak Detectability and the Linear Quadratic Control Problem of Discrete-Time Markov Jump Linear Systems. *International Journal of Control. Special issue on Switched and Polytopic Linear Systems*, 75:1282–1292.

COSTA (1994) COSTA, O. L. V. (1994). Linear Minimum Mean Square Error Estimation for Discrete-Time Markovian Jump Linear Systems. *IEEE Trans. on Automatic Control*, 39(8):1685–1689.

✎ COSTA & FRAGOSO (1995) COSTA, O. L. V. & FRAGOSO, M. (1995). Discrete-Time LQ-Optimal Control Problems for Infinite Markov Jump Parameter Systems. *IEEE Transactions on Automatic Control*, AC-40:2076–2088.

COSTA & JIMENEZ (2002) COSTA, O. L. V. & JIMENEZ, S. G. (2002). Stationary Filter

- for Linear Minimum Mean Square Error Estimator of Discrete-Time Markovian Jump Systems. *International Journal of Control*, 75:712–727.
- DAI (1989) DAI, L. (1989). Filtering and LQG Problems for Discrete-Time Stochastic Singular Systems. *IEEE Trans. Automat. Contr.*, 34(10):1105–1108.
- FRAGOSO & BACZYNSKI (2001) FRAGOSO, M. & BACZYNSKI, J. (2001). Optimal Control for Continuous Time LQ Problems with infinite Markov Jump Parameters. *SIAM Journal on Control and Optimization*, 40(1):270–297.
- GERDIN (2004) GERDIN, M. (2004). *Parameter Estimation in Linear Descriptor Systems*. Tese de Doutorado, Linköping University.
- ⇒ HASSAN & SADJANI (1995) HASSAN, M. A. & SADJANI, M. R. A. (1995). Noncasual imagem modeling using. *IEEE Trans. Circuits Syst. II*, 42(8):536–540.
- HOUSEHOLDER (1964) HOUSEHOLDER, A. (1964). *The theory of Matrices in Numerical Analysis*, chapter 5. Waltham.
- ISHIHARA & TERRA (2001) ISHIHARA, J. Y. & TERRA, M. H. (2001). Impulsive Controllability and Observability of Rectangular Descriptor Systems. *IEEE Trans. Automat. Contr.*, 46(6):991–994.
- ISHIHARA & TERRA (2002) ISHIHARA, J. Y. & TERRA, M. H. (2002). On the Lyapunov Theorem for Singular Systems. *IEEE Trans. on Automatic Control*, 47(11):1926–1930.
- ISHIHARA et al. (2004) ISHIHARA, J. Y., TERRA, M. H., & CAMPOS, J. C. T. (2004). Optimal recursive estimation for discrete-time descriptor systems. In *Proceedings of American Control Conference, Boston, MA, June/July 2004*.
- ISHIHARA et al. (2004b) ISHIHARA, J. Y., TERRA, M. H., & CAMPOS, J. C. T. (2004). Robust Kalman Filter for Descriptor Systems. In *Proceedings of American Control Conference, Boston, MA, June/July 2004*.
- JAZWINSKI (1970) JAZWINSKI, A. (1970). *Stochastic processes and Filtering Theory*. New York: Academic.

- JI & CHIZECK (1990) JI, Y. & CHIZECK, H. J. (1990). Controlability, Stabilizability and Continuous Time Markovian Jump Linear Quadratic Control. *IEEE Transactions on Automatic Control*, 35(7):777–788.
- JI & CHIZECK (1990b) JI, Y. & CHIZECK, H. J. (1990). Jump Linear Quadratic Gaussian Control: Steady-State Solution and Testable Conditions. *Control Theory and Advanced Technology*, 6(3):289–319.
- KAILATH et al. (2000) KAILATH, T., SAYED, A. H., & HASSIBI, B. (2000). *Linear Estimation*. Prentice Hall.
- KALMAN (1960) KALMAN, R. (1960). A new approach to linear filtering and Prediction Problems. *Trans. of the ASME (J. Basic En.)*, 82D:35–45.
- KAMINSKI et al. (1971) KAMINSKI, P., BRYSON, A., & SCHMIDT, S. (1971). Discrete square root filtering: a survey of current techniques. *IEEE TRans. Automat. Contr.*, AC-16:727–736.
- KAYLATH et al. (2000) KAYLATH, T., SAYED, A. H., & HASSIBI, B. (2000). *Linear Estimation*. Prentice-Hall.
- KOUROUKLIS & PAIGE (1981) KOUROUKLIS, S. & PAIGE, C. C. (1981). A constrained least squares approach to the general Gauss-Markov linear model. *J. Amer. Statist. Assoc.*, 76(375):620–625.
- ⇒ LUENBERGER (1977) LUENBERGER, D. V. (1977). Dynamic equations in descriptor systems. *IEEE Trans. Automat. Control*, 22(3):312–321.
- MAYBECK (1979) MAYBECK, P. (1979). *Stochastic Models, Estimation and Control*, volume 1. Academic Press, New York.
- ⇒ MILLS & GOLDENBERG (1989) MILLS, J. K. & GOLDENBERG, A. A. (1989). Force and position control of manipulators during constrained motion tasks. *IEEE Trans. Robot. Automat.*, 68:30–46.
- PAIGE (1985) PAIGE, C. (1985). Covariance Matrix Representation in Linear Filtering. In *Proc. 1985 Joint Summer Research Conference on Linear Algebra and its Role in System Theory*.

- PAIGE (1985b) PAIGE, C. C. (1985). Covariance matrix representation in linear filtering. In *Linear algebra and its role in systems theory (Brunswick, Maine, 1984)*, volume 47 of *Contemp. Math.*, pages 309–321. Amer. Math. Soc., Providence, RI.
- ✂ SINGH & LIU (1973) SINGH, S. P. & LIU, R. W. (1973). Existence of state equation representation if linear large-scale dynamic systems. *IEEE Trans. on Circuit Theory*, 20(5):239–246.
- SWORDER & ROGERS (1983) SWORDER, D. D. & ROGERS, R. O. (1983). An LQ-solution to a control problem associated with a solar thermal central receiver. *IEEE Transactions on Automatic Control*, 28(10):971–978.
- VERHAEGEN & DOOREN (1986) VERHAEGEN, M. & DOOREN, P. V. (1986). Numerical aspects of different Kalman filter implementations. *IEEE Trans. Automat. Control*, 31(10):907–917.

Apêndice A

Resultados Matriciais Importantes

A.1 Complemento de Schur

Considere o bloco de matrizes M

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

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 $\Delta A \triangleq D - CA^{-1}B$ 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

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

pode-se achar a fatoração

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

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

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

Apêndice B

Transformações Unitárias

B.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

$$\begin{bmatrix} x_1 & x_2 & \dots & x_{n-1} \end{bmatrix} \Theta = \alpha e_0 \quad (\text{B.1})$$

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

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 de Θ e de (B.1) segue que $x\Theta\Theta^T x^T = xx^T = \|x\|^2$. Então deve-se ter $\alpha = \pm\|x\|$. Ambos valores de α são possíveis. Uma maneira de calcular Θ é através da reflexão de Householder. A expressão será derivada geometricamente.

Considera-se Figura B.1. Assume-se que α já foi determinado, busca-se transformar x através de uma matriz unitária Θ tal que o vetor resultante $x\Theta$ seja alinhado com o vetor base e_0 . Esta transformação deve manter a norma de x . O vetor alinhado com e_0 fica sendo então αe_0 . O triângulo com os lados x e αe_0 e base $g = x - \alpha e_0$ é portanto um triângulo isósceles. Tem-se então que $\alpha e_0 = x - g$. Mas pode-se expressar g de uma forma alternativa. Definindo-se uma reta perpendicular da origem de x ao vetor g , o

Figura B.1: Interpretação geométrica da rotação de Householder

vetor será dividido em duas partes iguais. A parte de cima é a projeção do vetor x em g , que é igual a $\langle x, g \rangle \|g\|^{-2} g$. Portanto

$$\alpha e_0 = x - 2 \langle x, g \rangle \|g\|^{-2} g. \quad (\text{B.2})$$

Esta transformação é denominada reflexão porque reflete o vetor x pela perpendicular de g . Quando x e g são vetores puros, tem-se que $\langle x, g \rangle = xg^T$, e a equação é reduzida a

$$\alpha e_0 = x - 2xg^T(gg^T)^{-1}g = x \left[I - 2 \frac{g^T g}{gg^T} \right] \quad (\text{B.3})$$

sendo que se pode definir Θ como

$$\Theta \triangleq \left[I - 2 \frac{g^T g}{gg^T} \right] \quad (\text{B.4})$$

que é ortogonal como se deseja. Uma sequência de transformações Householder pode ser usada para triangularizar uma matriz A $m \times n$. Para isto deve-se achar a primeira transformação Θ_0 para rotacionar a primeira linha tal que temos $A\Theta_0$ da forma abaixo

$$A\Theta_0 = \begin{bmatrix} \alpha \\ \times \\ A_1 \end{bmatrix} \quad (\text{B.5})$$

sendo que $\times$ denota entradas cujos valores não são importantes para este procedimento. Aplica-se então uma transformação do tipo $(1 \oplus \Theta_1)$, sendo que Θ_1 rotaciona a primeira linha de A_1 tal que ela caia em e_0 sobre o espaço $(n-1)$ dimensional, e assim repete-se o processo.

B.2 Transformações de Givens

Se a matriz a ser triangularizada já possui muitos zeros, o método de Householder pode ser muito custoso. As rotações de Givens podem ser uma boa alternativa quando existem muitos zeros. Uma rotação elementar de Givens Θ , ortogonal de tamanho 2×2 transforma um vetor linha 1×2 , $x = \begin{bmatrix} a & b \end{bmatrix}$, e rotaciona para levá-la ao vetor de base $e_0 = \begin{bmatrix} 1 & 0 \end{bmatrix}$, ou seja, a rotação faz a seguinte transformação

$$\begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} \alpha & 0 \end{bmatrix} \quad (\text{B.6})$$

sendo que α é um número real a ser determinado. O valor de Θ que transforma (B.6) é dado por

$$\Theta = \frac{1}{\sqrt{1+\rho^2}} \begin{bmatrix} 1 & -\rho \\ \rho & 1 \end{bmatrix} \quad (\text{B.7})$$

sendo $\rho = \frac{b}{a}$, $a \neq 0$. Pode-se verificar que Θ faz realmente a seguinte transformação

$$\begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} \pm\sqrt{a^2+b^2} & 0 \end{bmatrix} \quad (\text{B.8})$$

sendo que os sinais de mais ou de menos dependem do sinal considerado em (B.7). As triangularizações de matrizes podem ser feitas através de um produto de transformações de Givens. Por exemplo, supondo que se queira zerar x_j usando x_i em

$\begin{bmatrix} \times & \times & x_i & \times & \times & x_j & \times & \times \end{bmatrix}$. Seja então $\rho = x_j/x_i$ e definindo-se

$$\Theta = \begin{bmatrix} 1 & & & & & & & \\ & 1 & & & & & & \\ & & \frac{1}{\sqrt{1+\rho^2}} & \frac{-\rho}{\sqrt{1+\rho^2}} & & & & \\ & & & 1 & & & & \\ & & \frac{\rho}{\sqrt{1+\rho^2}} & \frac{1}{\sqrt{1+\rho^2}} & & & & \\ & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & 1 & \end{bmatrix} \quad (\text{B.9})$$

tem-se então que $x\Theta = \begin{bmatrix} \times & \times & \alpha & \times & \times & 0 & \times & \times \end{bmatrix}$. Para triangularizar uma matriz A qualquer, deve-se realizar uma série de transformações como em (B.9) para zerar os elementos de interesse da matriz A linha a linha.

B.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

$$\Theta J \Theta^* = \Theta^* J \Theta = J$$

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

$$J = (I_p \oplus -I_q), \quad p \geq 1, \quad q \geq 1$$

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

$$\|y\|_J^2 \triangleq y J y^* = x \Theta J \Theta^* x^* = x J x^* \triangleq \|x\|_J^2$$

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

$$x = \begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} \alpha & 0 \end{bmatrix}, \quad (\text{B.10})$$

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

$$x = \begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} 0 & \alpha \end{bmatrix}. \quad (\text{B.11})$$

Assim, pode-se distinguir distinguido dois casos

1. $|a| > |b|$: a expressão para calcular a matriz de rotação hiperbólica Θ que satisfaz (B.10) é dada por

$$\Theta = \sqrt{1 - \rho^2} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \quad \text{onde } \rho = \frac{b}{a}, \quad a \neq 0 \quad (\text{B.12})$$

o que gera

$$\begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} \pm\sqrt{a^2 - b^2} & 0 \end{bmatrix} e \quad (\text{B.13})$$

2. $|b| > |a|$: a expressão para calcular a matriz de rotação hiperbólica Θ que satisfaz (B.11) é dada por

$$\Theta = \frac{1}{\sqrt{1 - \rho^2}} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \quad \text{onde } \rho = \frac{a}{b}, \quad b \neq 0 \quad (\text{B.14})$$

o que gera

$$\begin{bmatrix} a & b \end{bmatrix} \Theta = \begin{bmatrix} 0 & \pm\sqrt{b^2 - a^2} \end{bmatrix}. \quad (\text{B.15})$$

Apêndice C

Teoremas Auxiliares

C.1 Rotações de bases

Lema C.1.1 *KAILATH et al. (2000)* Dadas duas matrizes $n \times m$ com $n \leq m$ A e B , a igualdade $AA^* = BB^*$ é válida se e apenas se existir uma matriz unitária $m \times m$ Θ ($\Theta\Theta^* = I = \Theta^*\Theta$) tal que $A = B\Theta$.

C.2 Filtro Robusto para Sistemas Singulares CAMPOS (2004)

Lema C.2.1 *Considere o seguinte problema de otimização*

$$\min_x \max_{\{\delta A, \delta b\}} [\|x\|_Q^2 + \|(A + \delta A)x - (b + \delta b)\|_W^2] \quad (\text{C.1})$$

sendo A uma matriz de dados, b um vetor de medida que assume-se conhecido, x um vetor desconhecido, $Q = Q^T \geq 0$ e $W = W^T > 0$ matrizes de ponderação dadas, $\{\delta A, \delta b\}$ perturbações modeladas por

$$\begin{bmatrix} \delta A & \delta b \end{bmatrix} = H\Delta \begin{bmatrix} N_a & N_b \end{bmatrix}, \quad \|\Delta\| \leq 1. \quad (\text{C.2})$$

A solução do problema de otimização (C.1) é dada por

$$\hat{x} = [\hat{Q} + A^T \hat{W} A]^{-1} [A^T \hat{W} b + \hat{\lambda} N_a^T N_b] \quad (\text{C.3})$$

sendo que as matrizes de ponderação modificadas $\{\hat{Q}, \hat{W}\}$ são definidas como

$$\hat{Q} := Q + \hat{\lambda} N_a^T N_a; \quad (C.4)$$

$$\hat{W} := W + WH(\hat{\lambda}I - H^TWH)^\dagger H^TW \quad (C.5)$$

e $\hat{\lambda}$ é o parâmetro escalar não negativo calculado pelo seguinte problema de otimização

$$\hat{\lambda} = \arg \min_{\lambda \geq \|H^TW\|} G(\lambda) \quad (C.6)$$

sendo

$$G(\lambda) := \|x(\lambda)\|_Q^2 + \lambda \|N_a x(\lambda) - N_b\|^2 + \|Ax(\lambda) - b\|_{W(\lambda)}^2. \quad (C.7)$$

As funções auxiliares são definidas como

$$x(\lambda) := [Q(\lambda) + A^TW(\lambda)A]^{-1}[A^TW(\lambda)b + \hat{\lambda}N_a^TN_b]; \quad (C.8)$$

$$Q(\lambda) := Q + \lambda N_a^TN_a; \quad (C.9)$$

$$W(\lambda) := W + WH(\lambda I - H^TW H)^\dagger H^TW. \quad \diamond$$

Para calcular as estimativas robustas filtradas ótimas, as seguintes definições devem ser aplicadas no Lema C.2.1 para o cálculo de $\hat{\lambda}_i$

$$\begin{aligned}
A &\leftarrow \begin{bmatrix} -F_i & E_{i+1} \\ 0 & H_{i+1} \end{bmatrix}; \quad b \leftarrow \begin{bmatrix} F_i \hat{x}_{i|i} \\ z_{i+1} \end{bmatrix} \\
\delta A &\leftarrow \begin{bmatrix} -\delta F_i & \delta E_{i+1} \\ 0 & \delta H_{i+1} \end{bmatrix}; \quad \delta b \leftarrow \begin{bmatrix} \delta F_i \hat{x}_{i|i} \\ 0 \end{bmatrix} \\
Q &\leftarrow \begin{bmatrix} P_{i|i}^{-1} & 0 \\ 0 & 0 \end{bmatrix}; \quad W \leftarrow \begin{bmatrix} Q_i^{-1} & 0 \\ 0 & R_{i+1}^{-1} \end{bmatrix} \\
N_a &\leftarrow \begin{bmatrix} -N_{f,i} & N_{e,i+1} \\ 0 & N_{h,i+1} \end{bmatrix}; \quad N_b \leftarrow \begin{bmatrix} N_{f,i} \hat{x}_{i|i} \\ 0 \end{bmatrix} \\
H &\leftarrow \begin{bmatrix} M_{f,i} & 0 \\ 0 & M_{h,i} \end{bmatrix}
\end{aligned} \tag{C.10}$$

e para as condições iniciais, as seguintes definições são válidas

$$\begin{aligned}
A &\leftarrow H_0; \quad b \leftarrow z_0; \\
\delta A &\leftarrow \delta H_0; \quad \delta b \leftarrow 0; \\
Q &\leftarrow P_0^{-1}; \quad W \leftarrow R_0^{-1}; \\
H &\leftarrow M_{h,0}; \quad N_a \leftarrow N_{h,0}; \\
N_b &\leftarrow 0.
\end{aligned} \tag{C.11}$$

Para calcular as estimativas robustas preditoras ótimas, as seguintes definições devem ser aplicadas no Lema C.2.1 para o cálculo de $\hat{\lambda}_i$

$$\begin{aligned}
A &\leftarrow \begin{bmatrix} -F_i & E_{i+1} \\ H_i & 0 \end{bmatrix}; b \leftarrow \begin{bmatrix} F_i \hat{x}_{i|i-1} \\ z_i - H_i \hat{x}_{i|i-1} \end{bmatrix}; \\
\delta A &\leftarrow \begin{bmatrix} -\delta F_i & \delta E_{i+1} \\ \delta H_i & 0 \end{bmatrix}; \delta b \leftarrow \begin{bmatrix} \delta F_i \\ \delta H_i \end{bmatrix} \hat{x}_{i|i-1} \\
Q &\leftarrow \begin{bmatrix} P_{i|i-1}^{-1} & 0 \\ 0 & 0 \end{bmatrix}; W \leftarrow \begin{bmatrix} Q_i^{-1} & 0 \\ 0 & R_i^{-1} \end{bmatrix}; \\
H &\leftarrow \begin{bmatrix} M_{f,i} & 0 \\ 0 & M_{h,i+1} \end{bmatrix}; N_b \leftarrow \begin{bmatrix} N_{f,i} \\ N_{h,i} \end{bmatrix} \hat{x}_{i|i-1}. \\
N_a &\leftarrow \begin{bmatrix} -N_{f,i} & N_{e,i+1} \\ N_{h,i} & 0 \end{bmatrix}.
\end{aligned} \tag{C.12}$$

C.3 Transformações J-unitárias

Lema C.3.1 *KAILATH et al. (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 $AJA^* = BJB^*$ permanece se e apenas se existir uma única matriz J -unitária $m \times m$ Θ tal que $A = B\Theta$.