

MARIA D. MIRANDA

**SOBRE ALGORITMOS DOS MÍNIMOS QUADRADOS  
RÁPIDOS, RECURSIVOS NA ORDEM, QUE UTILIZAM  
TRIANGULARIZAÇÃO ORTOGONAL**

Tese apresentada à Escola Politécnica da  
Universidade de São Paulo para obtenção  
do título de Doutor em Engenharia.

São Paulo, abril de 1996.

MARIA D. MIRANDA

**SOBRE ALGORITMOS DOS MÍNIMOS QUADRADOS  
RÁPIDOS, RECURSIVOS NA ORDEM, QUE UTILIZAM  
TRIANGULARIZAÇÃO ORTOGONAL**

Tese apresentada à Escola Politécnica da  
Universidade de São Paulo para obtenção  
do título de Doutor em Engenharia.

**Área de Concentração:**  
Sistemas Eletrônicos

**Orientador:**  
Prof. Dr. Max Gerken

São Paulo, abril de 1996.

*De certo modo, muitas atividades humanas,  
as artes, as ciências, as técnicas, a política,  
são apenas maneiras peculiares,  
cada uma com suas regras próprias,  
de jogar o jogo dos possíveis.*

*François Jacob*

## **Agradecimentos**

Ao Prof. Dr. Max Gerken pela orientação, críticas e valiosas sugestões que muito influenciaram no conteúdo da tese; ao Prof. Dr. Jacyntho J. Angerami pelo incentivo e por ter proporcionado excelentes condições de trabalho; aos professores, funcionários e alunos do LCS, em especial ao Philip M. S. Burt, ao Euvaldo F. Cabral Jr. e ao Leonardo Aguayo; aos professores João Marcos T. Romano e André F. Kohn que participaram da banca de qualificação; ao analista de sistemas R. Mengato; a minha família, em especial a Eunice, a Rosângela e ao Renato pela simpatia e compreensão; ao CNPq pelo suporte financeiro.

## Resumo

Com o objetivo de relacionar os principais métodos de resolução do problema dos mínimos quadrados, faz-se uma interpretação dos elementos da matriz de sistema que descreve as operações de adaptação e filtragem dos algoritmos RLS (Recursive Least Squares) recorrentes na ordem. Baseado nesta interpretação, apresenta-se um novo algoritmo RLS rápido que é híbrido entre os que usam decomposição QR e os que usam estrutura em treliça com erros de predição *a priori*, denominado algoritmo QR-LSL *a priori*. Como vantagens este novo algoritmo apresenta, não somente um número de operações aritméticas ligeiramente reduzido em relação ao algoritmo QR-LSL *a posteriori* mas também, um paralelismo inerente que pode ser explorado em implementações rápidas. Demonstra-se que este novo algoritmo apresenta a propriedade de estabilidade retrógrada na presença de excitações persistentes, e se corretamente implementado um comportamento estável é garantido mesmo na presença de seqüências mal condicionadas. Ao contrário do algoritmo QR-LSL *a posteriori*, rotações passivas não são necessárias para garantir a estabilidade retrógrada. Resultados de simulações são apresentados, confirmando o excelente comportamento numérico do algoritmo.

## Abstract

The adaptation and filtering operations of Recursive Least Squares (RLS) algorithms can be described by a system matrix. With the purpose of clarifying the relationship between different order recursive RLS algorithms, an interpretation of the elements of the corresponding system matrix is presented. Using this interpretation a new normalized lattice filter, based on *a priori* prediction errors, is obtained. It is used to derive a new hybrid QR-LSL algorithm based on normalized *a priori* prediction errors. This algorithm presents not only a slight reduced computational complexity when compared with the hybrid QR-LSL algorithm based on *a posteriori* prediction errors, but also inherent parallelism that can be advantageously exploited in fast implementations. It is shown to be backward consistent and backward stable under persistent excitation. In contrast to the QR-LSL *a posteriori* algorithm, passive rotations are not necessary to guarantee these properties. Hence, rounding of all internal variables can be employed without sacrificing stability. Also shown, is that, even in presence of very ill conditioned signals, the algorithm presents a stable behavior and produce meaningful results. Some simulation results illustrate this robust numerical behavior, guaranteed if the algorithm is properly implemented under very mild conditions.

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Equivalência entre métodos de resolução do problema dos mínimos quadrados</b>	<b>6</b>
2.1	Estimação dos mínimos quadrados . . . . .	7
2.2	Algoritmo RLS convencional . . . . .	17
2.3	Triangularização ortogonal . . . . .	20
2.4	Ortogonalização de Gram-Schmidt . . . . .	29
2.5	Considerações sobre os vetores de estados . . . . .	37
2.6	Considerações sobre a matriz de sistema . . . . .	40
2.7	Comentários . . . . .	43
<b>3</b>	<b>Sobre algoritmos recorrentes no tempo e na ordem</b>	<b>45</b>
3.1	Atualizações na ordem . . . . .	46
3.2	Atualizações no tempo . . . . .	54
3.3	Representações para a estrutura em treliça . . . . .	58
3.4	Atualização no tempo dos coeficientes da estrutura em treliça . . . . .	63
3.5	Sobre as matrizes de sistema $\mathbf{S}_M^k(n)$ e $\mathbf{S}_M^g(n)$ . . . . .	66
3.6	Atualizações com rotações de Givens . . . . .	70
3.7	Algoritmos recorrentes no tempo e na ordem . . . . .	77
3.8	Um novo algoritmo QR-LSL . . . . .	81
3.9	Comentários . . . . .	86
<b>4</b>	<b>Estabilidade numérica dos algoritmos QR-LSL</b>	<b>87</b>
4.1	Propagação de erro e consistência retrógrada . . . . .	88
4.2	Algoritmo QR-LSL com erros a posteriori . . . . .	100
4.2.1	Minimalidade . . . . .	101
4.2.2	Caracterização do domínio de estabilidade $\mathcal{S}_e(n)$ . . . . .	101
4.2.3	Demonstração da Propriedade 2 . . . . .	108
4.2.4	Consistência retrógrada . . . . .	112
4.3	Algoritmo QR-LSL com erros a priori . . . . .	116
4.3.1	Minimalidade . . . . .	116
4.3.2	Caracterização do domínio de estabilidade . . . . .	116
4.3.3	Propriedade 2 para o algoritmo QR-LSL a priori . . . . .	119
4.3.4	Consistência retrógrada . . . . .	121
4.4	Considerações sobre representação finita . . . . .	126
4.5	Resultados de simulação . . . . .	131

## Errata

- ◇ Página 4, 2<sup>a</sup> §, 2<sup>a</sup> linha, ao invés de *RLS que utilizam triangularização ortogonal*, leia *RLS recorrentes na ordem*.
- ◇ Página 9, 4<sup>a</sup> linha depois da equação (2.13). ao invés de *ortogonal ao vetor de estimado  $\hat{d}(n)$* , leia *ortogonal ao vetor estimado  $\hat{d}(n)$* .
- ◇ Página 12, equação (2.26), ao invés de  $\mathbf{d}^d(n) = \mathbf{A}_{M-1}(n)\mathbf{w}_{M-1}^b(n)$  leia  $\mathbf{d}^b(n) = \mathbf{A}_{M-1}(n)\mathbf{w}_{M-1}^b(n)$ .
- ◇ Página 22, 1<sup>a</sup> linha, depois da equação (2.75). ao invés de *definido por (2.29)*, leia *definido por (2.17)*.
- ◇ Página 24, Tabela 2.1, Coluna do algoritmo RLS convencional, ao invés de  $(\mathbf{I}_{n \times n} - \mathbf{P}_M(n))\mathbf{d}(n)$  e  $\mathbf{I}_{n \times n} - \mathbf{P}_M(n)$ , leia  $(\mathbf{I}_{(n+1) \times (n+1)} - \mathbf{P}_M(n))\mathbf{d}(n)$  e  $\mathbf{I}_{(n+1) \times (n+1)} - \mathbf{P}_M(n)$ .
- ◇ Página 27, 2<sup>a</sup> linha, ao invés de *o algoritmo QRD-RLS*, leia *o algoritmo QR-RLS*.
- ◇ Página 28, item Predição progressiva, 1<sup>a</sup> linha depois da equação (2.95), ao invés de *progressiva de  $M - 1$  e no tempo  $n$* , leia *progressiva de ordem  $M - 1$  e no instante  $n$* .
- ◇ Página 31, equação (2.106), ao invés de  $e_{i-1}^b(\ell) = b_{i-1}^b(\ell) = u(\ell - M + 1) - \mathbf{u}_{M-1}^T(\ell)\mathbf{w}_{i-1}^b(n)$ , leia  $e_{i-1}^b(\ell) = b_{i-1}^b(\ell) = u(\ell - i + 1) - \mathbf{u}_{i-1}^T(\ell)\mathbf{w}_{i-1}^b(n)$ .
- ◇ Página 43, 2<sup>a</sup> §, 2<sup>a</sup> linha, ao invés de *por exemplo [12, 14, 13, 21]*, leia *por exemplo [12, 13, 21]*.
- ◇ Página 56, item e) Atualização no tempo da energia de erro da predição progressiva, 3<sup>a</sup> linha, ao invés de *no tempo e na ordem para*, leia *no tempo para*.
- ◇ Página 60, item 4) Estrutura em treliça com erros de predição a priori normalizados, última equação não numerada, ao invés de

$$\begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \frac{1}{a_i(n-1)} \begin{bmatrix} 1 & -\bar{\Delta}_{i-1}(n) \\ -\bar{\Delta}_{i-1}(n) & 1 \end{bmatrix} \begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_i(n) \end{bmatrix}.$$



Sendo que esta última equação, com em, leia

$$\begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \frac{1}{a_i(n-1)} \begin{bmatrix} 1 & -\bar{\Delta}_{i-1}(n) \\ -\bar{\Delta}_{i-1}(n) & 1 \end{bmatrix} \begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_{i-1}(n) \end{bmatrix}.$$

Sendo que esta última equação, como em.

- ◊ Página 61, item 5) Estrutura em treliça com erros de predição normalizados em ângulo, 3ª linha, ao invés de *a equação (3.52)*, leia *a equação (3.56)*.
- ◊ Página 77, 1º § , nas duas equações ao invés de  $k_i^b(n)$  e  $k_i^b(n-1)$  leia  $k_i^d(n)$  e  $k_i^d(n-1)$ .
- ◊ Página 77, item Algoritmo RLS rápido recursivo na ordem com erros a priori, 2ª linha, ao invés de *por Ling em [12]*. leia *por Ling em [14]*.
- ◊ Página 79, item Algoritmo QR-RLS rápido recursivo na ordem com erros normalizados em ângulo, Tabela 3.2. ao invés de *Algoritmo QR-RSL com erros normalizados em ângulo*, leia *Algoritmo QR-RLS com erros normalizados em ângulo*.
- ◊ Página 88, seção 4.1 Propagação de erro e consistência retrógrada, 3ª linha depois da equação (4.3), ao invés de *o algoritmo QR-LSL a priori*, leia *o algoritmo QR-LSL a posteriori*.
- ◊ Página 91, 8ª linha, ao invés de *convergem exponencialmente rápido em* leia, *convergem de forma exponencial em*.
- ◊ Página 94, 3ª linha, ao invés de *de forma exponencialmente rápida [6, 5]*, leia *de forma exponencial [5, 6]*.
- ◊ Página 120, Prova da propriedade 2a.ii, 1ª equação, ao invés de  $\|\bar{\mathbf{b}}_M(n)\| = 1 - \alpha$ , leia  $\|\bar{\mathbf{b}}_M(n)\| \leq 1 - \alpha$ .
- ◊ Página 129, 2º §, 9ª linha, ao invés de *no algoritmo QR-LSL com a posteriori*, leia *no algoritmo QR-LSL a posteriori*.
- ◊ Página 145, Referência [36], ao invés de *IEEE Trans. on Circuits and Systems, vol. 42*, leia *IEEE Trans. on Circuits and Systems - II, Analog and Digital Signal Processing, vol. 42*.

- ◊ Página 149, Seção A.3 Representação para a inversa da matriz de autocorrelação, item a, equação (A.17), ao invés de  $\Phi_M^{-1}(n) = \lambda^{-1}[\mathbf{I} - \mathbf{g}(n)\mathbf{u}_M^T(n)]\Phi_M^{-1}(n-1)$ , leia  $\Phi_M^{-1}(n) = \lambda^{-1}[\mathbf{I} - \mathbf{g}_M(n)\mathbf{u}_M^T(n)]\Phi_M^{-1}(n-1)$ .
- ◊ Página 160, item Interpretação geométrica, 3ª linha, ao invés *observar e o vetor*, leia *observar que o vetor*.
- ◊ Página 161, equação (B.13), ao invés de

$$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ \sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}.$$

leia

$$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}.$$

- ◊ Página 175, linha 13, ao invés de  $\Lambda(n) = \text{diag}(\lambda^n, \lambda^{n-1}, \dots, \lambda^n, 1)$ , leia  $\Lambda(n) = \text{diag}(\lambda^n, \lambda^{n-1}, \dots, \lambda, 1)$ .

4.6	Comentários . . . . .	140
<b>Referências bibliográficas</b>		<b>142</b>
<b>A</b>	<b>Propriedades estruturais resultantes da utilização de dados seqüenciais</b>	<b>146</b>
A.1	Representações para a matriz de dados . . . . .	146
A.2	Representações para matriz de autocorrelação . . . . .	148
A.3	Representações para a inversa da matriz de autocorrelação . . . . .	149
A.4	Correlações cruzadas . . . . .	151
A.5	Fator de conversão . . . . .	153
<b>B</b>	<b>Erro normalizado em ângulo e estrutura de McWhirter</b>	<b>157</b>
B.1	Erro normalizado em ângulo . . . . .	157
B.2	Estrutura de McWhirter . . . . .	160
<b>C</b>	<b>Sobre a atualização na ordem dos coeficientes do filtro de erro de predição</b>	<b>162</b>
C.1	Atualização dos coeficientes do filtro transversal . . . . .	162
C.2	Atualização dos coeficientes do filtro transversal normalizado . . . . .	166
<b>D</b>	<b>Condição de excitação persistente e condicionamento da matriz de dados</b>	<b>168</b>
<b>E</b>	<b>Lista dos principais símbolos</b>	<b>172</b>

# Capítulo 1

## Introdução

Nos últimos anos surgiram várias versões de algoritmos tentando resolver o problema dos mínimos quadrados de forma eficiente. A convergência dos algoritmos RLS (Recursive Least Squares) independe do espalhamento dos valores singulares da matriz de dados, fato que torna os algoritmos RLS atrativos quando comparados aos do gradiente estocástico [21, 27, 38]. Por outro lado, os primeiros algoritmos RLS propostos apresentavam uma complexidade computacional significativamente superior. No âmbito da filtragem adaptativa o desenvolvimento dos algoritmos RLS rápidos, que exploram a natureza seqüencial dos dados de entrada, reduziu de forma expressiva a diferença entre os dois tipos de algoritmos no que concerne a complexidade computacional [38, 27]. Estes algoritmos apresentam uma complexidade computacional proporcional a  $M$ , sendo  $M$  o número de parâmetros ajustáveis do filtro adaptativo.

As diferentes versões de algoritmos RLS apresentadas na literatura podem ser divididas basicamente em quatro grupos não excludentes:

### 1) Algoritmos baseados no filtro transversal

Neste grupo, onde os coeficientes do filtro transversal são obtidos diretamente, estão o algoritmo RLS convencional [38, cap. 13] e as correspondentes versões rápidas [15, 23, 26]. O primeiro algoritmo rápido é atribuído a Ljung, Morf e Falconer [26], sendo que outras versões apresentadas em [15, 23] reduzem ainda mais a complexidade computacional. Variações e combinações destes algoritmos podem ser encontradas na literatura especializada. Apesar destas versões apresentarem complexidade computacional proporcional a  $M$ , apresentam um sério inconveniente que é a instabilidade numérica, quando implementadas em aritmética de precisão finita [11, 24, 38]. Em [24] foi apresentado um método de estabilização, cujo o desempenho é superior em relação aos métodos de esta-

bilização conhecidos, e.g. [11]. Entretanto, como verificado em [24], os métodos de estabilização apresentam regiões de funcionamento, não constituindo uma solução geral para o problema.

2) Algoritmos rápidos baseados no filtro preditor em treliça

Neste grupo existem quatro versões básicas de algoritmos LSL (Least Square Lattice): os baseados no erro *a posteriori*, os baseados no erro *a priori*, os com realimentação de erro *a posteriori* e os com realimentação de erro *a priori*. Estas versões estão apresentadas em [38, cap. 17] e em [14]. Estes algoritmos são recorrentes no tempo e na ordem, resolvendo todos os problemas de ordem inferior a ordem considerada. Entretanto, também apresentam problemas de instabilidade numérica quando implementados com um número reduzido de bits, porém as versões com realimentação de erro apresentam um comportamento numérico bastante favorável em relação as versões baseadas no filtro transversal.

3) Algoritmos baseados no procedimento de ortogonalização de Gram-Schmidt

Estes algoritmos são obtidos a partir do procedimento de ortogonalização de GS (Gram-Schmidt) da matriz de dados. Como eles a princípio não impõem restrições à estrutura dos dados apresentam uma complexidade computacional proporcional a  $M^2$ . Em [21] são apresentadas diferentes versões de algoritmos dos mínimos quadrados que utilizam o procedimento de ortogonalização de GS. O algoritmo recursivo modificado de GS com realimentação de erro, é um algoritmo recursivo no tempo e na ordem que se destaca por apresentar um melhor desempenho numérico [13, 14].

4) Algoritmos que usam rotações de Givens

Rotações de Givens podem ser usadas para atualizar as variáveis de qualquer algoritmo recursivo na ordem, podendo este ser rápido ou não. Em [12], Ling propõe versões de algoritmos de GS baseados em rotações de Givens. Em [7, 19, 22, 33] e em [38, cap. 18], são apresentadas versões de algoritmos RLS rápidos que usam rotações ortogonais. Estes algoritmos são denominados QR-RLS rápidos, pois aplicam a decomposição QR na matriz de dados para resolver o problema dos mínimos quadrados. Em [40, 1] são apresentadas versões de algoritmos, baseados na decomposição QR-inversa, que utilizam operações de

rotações ortogonais para atualizar diretamente os coeficientes do filtro transversal. Os algoritmos rápidos resultantes do método QR com rotações ortogonais, são menos susceptíveis a efeitos de implementação em precisão finita, quando comparados a outros algoritmos rápidos [19, 21, 38].

Problemas de instabilidade numérica têm impedido o uso mais amplo dos algoritmos de filtragem dos mínimos quadrados rápidos. Nos últimos anos, um grande número de publicações tem sido direcionada à resolução deste problema, e.g. [10, 11, 24, 25, 31, 32, 36, 37]. Recentemente Regalia e Slock, de forma independente, combinaram o conceito de consistência retrógrada ('backward consistency') da análise numérica com os conceitos de minimalidade e alcançabilidade da teoria de sistemas. Com isso estabeleceram condições suficientes para propagação estável do erro aritmético nos algoritmos RLS rápidos. Em [31, 32] Regalia estabeleceu condições suficientes para garantir a propriedade de estabilidade retrógrada ('backward stability') nos algoritmos RLS rápidos. A estabilidade retrógrada assegura, que no caso de excitação persistente, a diferença entre a solução afetada de erros numéricos e a solução exata tem uma variância limitada. No caso, a solução afetada de erros numéricos é a solução exata do mesmo problema considerando um sinal de entrada perturbado. As condições estabelecidas por Regalia podem, a princípio, ser utilizadas em qualquer tipo de algoritmo RLS rápido. Este resultado é bastante importante, pois mostra que os algoritmos RLS rápidos não necessariamente são instáveis numericamente quando implementados em precisão finita, como foi considerado até recentemente na literatura. Desta forma, incentivou-se a investigação de algoritmos que apresentem esta interessante propriedade.

Os vários algoritmos RLS rápidos existentes que foram surgindo de forma isolada, tentam ao mesmo tempo, resolver os problemas numéricos e aumentar a eficiência computacional. Diferentes técnicas foram empregadas para obter diferentes versões de algoritmos que resolvem o problema dos mínimos quadrados. Com isso, torna-se muitas vezes difícil relacionar as soluções propostas. Por essa razão, alguns autores têm buscado formular uma teoria unificada para caracterizar os filtros adaptativos dos mínimos quadrados, e.g. [3, 4, 7, 8, 12, 18]. Em [18] Kailath fez uma interpretação geométrica dos algoritmos LSL, relacionando-os aos do filtro transversal. Em [12] Ling fez uma conexão entre o procedimento de Gram-Schmidt

e a decomposição QR, evidenciando que as variáveis internas dos algoritmos rápidos podem ser interpretadas usando os mesmos princípios geométricos encontrados nos filtros em treliça. Em [4] Sayed e Kailath descreveram o problema de filtragem adaptativa no espaço de estados, tentando assim, deduzir os vários algoritmos adaptativos diretamente. Em [33] Regalia e Bellanger mostraram que quando se aplica a decomposição QR para resolver o problema dos mínimos quadrados, os coeficientes de regressão da estimação conjunta e os coeficientes de reflexão de um filtro em treliça normalizado aparecem como variáveis intermediárias. Isto ocorre porque os elementos do vetor de resposta desejada rotacionado no caso da estimação estão relacionados com os coeficientes de regressão da estimação conjunta. No caso da predição regressiva, os elementos do vetor de resposta desejada rotacionado estão relacionados com os coeficientes de reflexão de um filtro em treliça normalizado. A partir destas observações, foi apresentado em [33] um algoritmo rápido QR-LSL *a posteriori*, assim denominado porque usa decomposição QR baseada em rotações de Givens e a estrutura em treliça baseada nos erros de predição *a posteriori*. Este algoritmo obedece aos critérios estabelecidos em [31, 32], sendo mínimo e satisfazendo a propriedade de estabilidade retrógrada. Neste caso, para assegurar a estabilidade retrógrada é necessário implementar as rotações do filtro em treliça de forma passiva.

O trabalho apresentado nesta tese tem duplo objetivo. O primeiro é obter um melhor entendimento não só das relações entre os principais algoritmos RLS que utilizam triangularização ortogonal, mas também dos conceitos introduzidos por Regalia em [31, 32]. O outro objetivo é introduzir um novo algoritmo QR-LSL que garante a propriedade de estabilidade retrógrada e apresenta vantagens em relação ao apresentado em [33]. Este algoritmo foi desenvolvido a partir de interpretações do significado dos elementos da matriz de sistema que descreve os algoritmos QR-RLS. Os resultados aqui apresentados foram parcialmente publicados em [28, 29, 30].

Com o objetivo de obter uma visão geral dos principais algoritmos recursivos na ordem, faz-se no capítulo 2 uma análise conceitual do método de triangularização ortogonal aplicado à resolução do problema dos mínimos quadrados. Deste modo chega-se a uma interpretação dos elementos do vetor de estados e da matriz de sistema, que descrevem as operações de adaptação e filtragem em termos dos erros de predição regressiva *a priori* e *a posteriori*. Algumas destas interpretações são contribuições originais deste trabalho [30].

São apresentadas no capítulo 3 as equações recorrentes na ordem e no tempo que são comuns aos métodos de ortogonalização que resolvem o problema dos mínimos quadrados de forma recorrente na ordem. Mostra-se que cada um dos elementos da matriz de sistema dos algoritmos que usam triangularização ortogonal pode ser representado em função dos erros da predição regressiva e/ou fator de conversão. Uma nova estrutura em treliça é apresentada usando a definição de erro de predição regressiva *a priori* normalizado introduzida no capítulo 2. Esta estrutura é fundamental para o desenvolvimento do novo algoritmo QR-LSL [29], denominado QR-LSL *a priori*. Uma das vantagens deste novo algoritmo consiste em um paralelismo inerente, capaz de tornar possível a implementação em  $4M + 3$  ciclos de CORDIC (COordinate Rotation Digital Computer), sendo  $M$  a ordem do problema de estimação.

Demonstra-se no capítulo 4 que o novo algoritmo tem a propriedade de estabilidade retrógrada [30]. Como o conceito de estabilidade retrógrada é recente e pouco difundido na literatura sobre filtros adaptativos, faz-se uma revisão das definições e propriedades mais importantes associadas a este conceito. Em seguida, baseado em [31, 32], demonstra-se que tanto o algoritmo QR-LSL *a posteriori*, como o QR-LSL *a priori*, possuem a propriedade de estabilidade retrógrada. Além disso, para o caso de sinais estacionários, é feita uma análise dos valores médios quadráticos que algumas variáveis dos algoritmos QR-LSL podem assumir. Tal análise permite obter informações sobre a faixa dinâmica, e conseqüentemente sobre a precisão necessária para armazenamento das variáveis consideradas. Finalmente, são apresentados resultados de simulações para comprovar o bom desempenho do algoritmo proposto aplicado às configurações de equalização adaptativa e identificação de sistemas. São também consideradas seqüências de observação mal condicionadas (ausência de excitação persistente).

No decorrer dos capítulos serão utilizadas várias expressões que já são bem conhecidas na literatura, porém para facilitar a apresentação do assunto estas são apresentadas nos apêndices *A* e *B*. Os apêndices *C* e *D* consistem em demonstrações de alguns resultados usados no capítulo 4. O apêndice *E* apresenta uma lista dos principais símbolos usados. Procurou-se, na medida do possível, adotar a mesma notação usada em [38], que é uma referência clássica na área.



## Capítulo 2

### Equivalência entre métodos de resolução do problema dos mínimos quadrados

Neste capítulo faz-se uma análise conceitual do método de triangularização ortogonal aplicado a resolução do problema dos mínimos quadrados. O objetivo é esclarecer as equivalências de algumas formas de resolver o problema dos mínimos quadrados. O procedimento de ortogonalização de Gram-Schmidt é usado como uma ponte para relacionar o algoritmo RLS convencional e os algoritmos RLS que usam decomposição QR baseada em rotações de Givens. Isto é feito porque o significado das variáveis envolvidas no procedimento de ortogonalização de Gram-Schmidt para resolver o problema dos mínimos quadrados é bem conhecido, sendo que no caso da seqüência de observação ser formada por dados seqüenciais, resulta nos filtros em treliça [21, 38].

Na seção 2.1 são apresentados os fundamentos da estimação dos mínimos quadrados. Nas seções seguintes são apresentados o algoritmo RLS convencional, o do método de triangularização ortogonal baseado em rotações de Givens e os que usam o procedimento de ortogonalização de Gram-Schmidt. Finalmente, faz-se uma interpretação do vetor de estados e da matriz de sistema que descrevem as operações de filtragem e adaptação dos algoritmos que usam triangularização ortogonal.

## 2.1 Estimação dos mínimos quadrados

O problema de filtragem dos mínimos quadrados, no tempo  $n$  e na ordem  $M$ , consiste em determinar o vetor de coeficientes  $\mathbf{w}_M(n) = [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T$  que minimiza a função

$$\xi_M(n) = \|\mathbf{e}_M(n)\|^2, \quad (2.1)$$

sendo  $\mathbf{e}_M(n) = \Lambda^{1/2}(n)[e_M(0) \dots e_M(n)]^T$  o vetor dos erros de estimação *a posteriori* ponderado. Este vetor é definido como a diferença entre o vetor de respostas desejadas ponderado  $\mathbf{d}(n) = \Lambda^{1/2}(n)[d(0) \dots d(n)]^T$  e o vetor de estimação destas respostas,

$$\hat{\mathbf{d}}(n) = \mathbf{A}_M(n)\mathbf{w}_M(n), \quad (2.2)$$

ou seja,

$$\mathbf{e}_M(n) = \mathbf{d}(n) - \mathbf{A}_M(n)\mathbf{w}_M(n). \quad (2.3)$$

A matriz  $\mathbf{A}_M(n) = [A(i, j)] = \lambda^{(n-i+1)/2}[u(i-j)]$  é uma matriz de dimensão  $(n+1) \times M$  com  $u(i-j)$  representando a seqüência de dados pré-janelada, isto é,  $u(i-j) = 0$  para  $(i-j) < 0$ . A matriz  $\Lambda(n) = \text{diag}\{\lambda^n, \lambda^{n-1}, \dots, 1\}$  é a matriz de ponderação sendo  $\lambda$  o fator de ponderação,  $0 \ll \lambda < 1$ . Utilizando a representação da matriz de dados apresentada na equação (A.2) do apêndice A, os elementos do vetor de estimação da resposta desejada ficam definidos como

$$\hat{d}(\ell) = \mathbf{u}_M^T(\ell)\mathbf{w}_M(n) = \sum_{k=1}^M u(\ell - k + 1)w_{k-1}(n), \quad (2.4)$$

para  $0 \leq \ell \leq n$ . A função (2.1) é minimizada quando o vetor de coeficientes satisfaz

$$\mathbf{w}_M(n) = \Phi_M^{-1}(n)\Omega_M(n), \quad (2.5)$$

onde

$$\Phi_M(n) = \mathbf{A}_M^T(n)\mathbf{A}_M(n) \quad (2.6)$$

é a matriz de autocorrelação de dados de entrada e o vetor

$$\Omega_M(n) = \mathbf{A}_M^T(n)\mathbf{d}(n) \quad (2.7)$$

de dimensão  $M \times 1$  representa a correlação cruzada entre os dados de entrada  $\mathbf{A}_M(n)$  e a resposta desejada  $\mathbf{d}(n)$ . A matriz de autocorrelação  $\Phi_M(n)$  de dimensão  $M \times M$  é admitida como não singular. A demonstração deste resultado clássico pode ser encontrada, e.g., em [38]. Para as finalidades do presente trabalho, mais importante que a demonstração, é fazer uma interpretação geométrica deste resultado.

### Interpretação geométrica

Considere-se a representação da matriz de dados em termos de  $M$  colunas, conforme equação (A.4) do apêndice A. Geometricamente, o vetor  $\hat{\mathbf{d}}(n)$  pode ser interpretado como uma projeção do vetor  $\mathbf{d}(n)$  no subespaço gerado pelas colunas  $\mathbf{a}(n - i + 1)$ , para  $1 \leq i \leq M$ , da matriz  $\mathbf{A}_M(n)$ :

$$\hat{\mathbf{d}}(n) = \mathbf{A}_M(n) \mathbf{w}_M(n) = \sum_{i=1}^M \mathbf{a}(n - i + 1) w_{i-1}(n). \quad (2.8)$$

Os coeficientes  $w_{i-1}(n)$  do vetor  $\mathbf{w}_M(n)$ , indicam em que proporção cada uma das colunas de  $\mathbf{A}_M(n)$  devem ser combinadas para se obter

$$\left\| \mathbf{d}(n) - \sum_{i=1}^M \mathbf{a}(n - i + 1) w_{i-1}(n) \right\| \stackrel{!}{=} \text{mínimo}. \quad (2.9)$$

Como os vetores  $\mathbf{a}(n - i + 1)$  são linearmente independentes, mas **não** são ortogonais, **não** é possível calcular  $\mathbf{w}_M(n)$  diretamente através de projeções na base formada pelas colunas  $\mathbf{a}(n - i + 1)$  da matriz  $\mathbf{A}_M(n)$  [16]. Para fazer um cálculo deste tipo, pode-se lançar mão de uma transformação que ortonormalize as colunas da matriz  $\mathbf{A}_M(n)$ . Esta transformação pode ser obtida através da matriz de autocorrelação (2.6). Como  $\Phi_M(n)$  é simétrica e positiva definida, é possível decompô-la no produto

$$\Phi_M(n) = \mathbf{L}_M^T(n) \mathbf{L}_M(n), \quad (2.10)$$

onde  $\mathbf{L}_M(n)$  é uma matriz não singular. A matriz  $\mathbf{L}_M(n)$  pode assumir por exemplo, a forma

$$\mathbf{L}_M(n) = \Phi_M^{1/2}(n),$$

ou considerando a representação de  $\Phi_M(n)$  através dos autovalores e dos autovetores como

$$\mathbf{L}_M(n) = \mathbf{\Upsilon}^T(n) \mathbf{\Xi}^{1/2}(n) \mathbf{\Upsilon}(n),$$

onde  $\Upsilon(n)$  é a matriz unitária dos autovetores e  $\Xi(n)$  é uma matriz diagonal positiva definida dos autovalores. Ou ainda, considerando a decomposição de Cholesky de  $\Phi_M(n)$ , em uma matriz  $L_M(n)$  triangular inferior ou superior, casos em que  $L_M(n)$  é chamada fator de Cholesky. Igualando (2.6) com (2.10) obtemos

$$\mathbf{I} = \mathbf{L}_M^{-T}(n) \mathbf{A}_M^T(n) \mathbf{A}_M(n) \mathbf{L}_M^{-1}(n),$$

o que demonstra que as colunas da matriz

$$\mathbf{A}'_M(n) = \mathbf{A}_M(n) \mathbf{L}_M^{-1}(n) \quad (2.11)$$

são ortonormais. Com esta transformação conclui-se facilmente que os vetores ótimos devem satisfazer

$$\left\| \mathbf{d}(n) - \underbrace{\mathbf{A}_M(n) \mathbf{L}_M^{-1}(n)}_{\mathbf{A}'_M(n)} \underbrace{\mathbf{L}_M(n) \mathbf{w}_M(n)}_{\mathbf{w}'_M(n)} \right\| \stackrel{!}{=} \text{mínimo}, \quad (2.12)$$

Neste caso, o vetor

$$\hat{\mathbf{d}}(n) = \mathbf{A}'_M(n) \mathbf{w}'_M(n) \quad (2.13)$$

é dado por uma combinação linear das colunas ortonormais da matriz  $\mathbf{A}'_M(n)$ . Geometricamente é bastante simples concluir que o vetor  $\hat{\mathbf{d}}(n)$  ótimo é a projeção do vetor  $\mathbf{d}(n)$  no espaço gerado pelas colunas da matriz  $\mathbf{A}'_M(n)$  sendo o erro de estimação ótimo  $\mathbf{e}_M(n)$  ortogonal ao vetor de estimado  $\hat{\mathbf{d}}(n)$ . O vetor de coeficientes  $\mathbf{w}'_M(n)$  é a representação de  $\hat{\mathbf{d}}(n)$  na base ortonormal formada pelas colunas da matriz  $\mathbf{A}'_M(n)$  sendo os seus elementos iguais ao produto escalar do vetor  $\mathbf{d}(n)$  pelos vetores da base:

$$\mathbf{w}'_M(n) = \mathbf{A}'_M{}^T(n) \mathbf{d}(n). \quad (2.14)$$

Consequentemente, o vetor de estimação ótima  $\hat{\mathbf{d}}(n)$  e o vetor do erro da estimação ótima  $\mathbf{e}_M(n)$  podem ser calculados por

$$\hat{\mathbf{d}}(n) = \mathbf{A}'_M(n) \mathbf{w}'_M(n) = \underbrace{\mathbf{A}'_M(n)}_{\text{(II)}} \underbrace{\mathbf{A}'_M{}^T(n) \mathbf{d}(n)}_{\text{(I)}} \quad (2.15)$$

e

$$\mathbf{e}_M(n) = \mathbf{d}(n) - \hat{\mathbf{d}}(n) = (\mathbf{I} - \mathbf{A}'_M(n) \mathbf{A}'_M{}^T(n)) \mathbf{d}(n). \quad (2.16)$$

Na expressão (2.15) estão explicitadas as operações envolvidas: em (I) calcula-se as projeções de  $\mathbf{d}(n)$  nas direções dos vetores da base ortonormal e em (II) contrõe-se o vetor  $\hat{\mathbf{d}}(n)$  dentro do espaço gerado pelas colunas da matriz  $\mathbf{A}'_M(n)$ . Cabe observar que a partir de (2.15) e (2.16) é possível confirmar que  $\hat{\mathbf{d}}(n)$  e  $\mathbf{e}_M(n)$  são realmente ortogonais:

$$\hat{\mathbf{d}}^T(n)\mathbf{e}_M(n) = \mathbf{d}^T(n)(\mathbf{A}'_M(n)\mathbf{A}'_M{}^T(n) - \mathbf{A}'_M(n) \underbrace{\mathbf{A}'_M{}^T(n)\mathbf{A}'_M(n)}_{=\mathbf{I}_{M \times M}}\mathbf{A}'_M{}^T(n))\mathbf{d}(n) = 0.$$

O operador

$$\mathbf{P}_M(n) = \mathbf{A}'_M(n)\mathbf{A}'_M{}^T(n) \quad (2.17)$$

de ordem  $(n+1) \times (n+1)$  que aparece em (2.15) é responsável pela projeção de  $\mathbf{d}(n)$  no espaço gerado pelas colunas da matriz  $\mathbf{A}'_M(n)$ , espaço que é ortogonal ao erro de estimação ótimo  $\mathbf{e}_M(n)$ . Por esse motivo ele é chamado de operador de projeção. O termo  $\mathbf{I} - \mathbf{P}_M(n)$  em (2.16) projeta  $\mathbf{d}(n)$  ortogonalmente ao espaço gerado pelas colunas da matriz  $\mathbf{A}'_M(n)$ , razão pela qual é chamado de operador complementar.

Lembrando que

$$\mathbf{A}'_M(n) = \mathbf{A}_M(n)\mathbf{L}_M^{-1}(n)$$

e

$$\mathbf{w}'_M(n) = \mathbf{L}_M(n)\mathbf{w}_M(n),$$

pode-se obter de (2.14) e (2.7), a expressão (2.5) para o vetor de coeficientes ótimo:

$$\begin{aligned} \mathbf{w}_{\mathcal{M}}(n) &= \mathbf{L}_M^{-1}(n)\mathbf{L}_M^{-T}(n)\mathbf{A}_M^T(n)\mathbf{d}(n) = \mathbf{\Phi}_M^{-1}(n)\mathbf{A}_M^T(n)\mathbf{d}(n) \\ &= \mathbf{\Phi}_M^{-1}(n)\mathbf{\Omega}_M^T(n). \end{aligned}$$

Além disso, resulta ainda a seguinte expressão para o operador de projeção

$$\mathbf{P}_M(n) = \mathbf{A}_M(n)\mathbf{L}_M^{-1}(n)\mathbf{L}_M^{-T}(n)\mathbf{A}_M^T(n) = \mathbf{A}_M(n)\mathbf{\Phi}_M^{-1}(n)\mathbf{A}_M^T(n), \quad (2.18)$$

que pode ser utilizada para o cálculo de

$$\hat{\mathbf{d}}(n) = \mathbf{P}_M(n)\mathbf{d}(n) \quad (2.19)$$

e

$$\mathbf{e}_M(n) = (\mathbf{I} - \mathbf{P}_M(n))\mathbf{d}(n), \quad (2.20)$$

sem recorrer à ortogonalização das colunas da matriz  $\mathbf{A}_M(n)$ .

Finalmente, a norma ao quadrado do vetor de erro ótimo  $\mathbf{e}_M(n)$ , que é a energia da seqüência de erro de estimação ótima, pode ser calculada por:

$$\xi_M(n) = \mathbf{e}_M^T(n)\mathbf{e}_M(n) = \mathbf{d}^T(n)\mathbf{e}_M(n) = \mathbf{d}^T(n)(\mathbf{I} - \mathbf{P}_M(n))\mathbf{d}(n). \quad (2.21)$$

Utilizando (2.18), (2.5) e (2.7), a equação (2.21) pode então ser reescrita como

$$\xi_M(n) = \xi_0(n) - \boldsymbol{\Omega}_M^T(n)\mathbf{w}_M(n), \quad (2.22)$$

onde  $\xi_0(n) = \|\mathbf{d}(n)\|^2$  representa a energia do erro da estimação de ordem zero que é a própria energia do sinal de entrada.

A seguir, os casos de predição progressiva e regressiva são considerados como casos de estimação para uma resposta desejada particular.

### Predição regressiva

Com a predição linear regressiva procura-se prever um valor passado de um processo a partir de observações subsequentes. No caso da predição regressiva de ordem  $M - 1$  procura-se prever o valor  $u(\ell - M + 1)$ , para  $\ell \geq M - 1$ , a partir do conjunto de  $M - 1$  amostras subsequentes

$$U_\ell = \{u(\ell - M + 2), u(\ell - M + 3), \dots, u(\ell)\}.$$

Define-se o erro de predição regressiva, no instante  $\ell$  como

$$b_{M-1}(\ell) = u(\ell - M + 1) - \hat{u}(\ell - M + 1|U_\ell). \quad (2.23)$$

As amostras estimadas  $\hat{u}(\ell - M + 1|U_\ell)$ , são obtidas como

$$\hat{u}(\ell - M + 1|U_\ell) = \sum_{k=1}^{M-1} u(\ell - k + 1)w_{k-1}^b(\ell) = \mathbf{u}_{M-1}^T(\ell)\mathbf{w}_{M-1}^b(\ell),$$

onde  $\mathbf{w}_{M-1}^b(\ell)$  é o vetor de coeficientes da predição regressiva. Considerando a expressão (2.23), porém com as amostras estimadas obtidas com o vetor de coeficientes do instante  $n$ . isto é

$$\sum_{k=1}^{M-1} u(\ell - k + 1)w_{k-1}^b(n) = \mathbf{u}_{M-1}^T(\ell)\mathbf{w}_{M-1}^b(n), \quad \text{para } 0 \leq \ell \leq n, \quad (2.24)$$

resulta em notação vetorial

$$\mathbf{e}_{M-1}^b(n) = \mathbf{a}(n - M + 1) - \mathbf{A}_{M-1}(n)\mathbf{w}_{M-1}^b(n) \quad (2.25)$$

onde  $\mathbf{e}_{M-1}^b(n) = \mathbf{\Lambda}^{1/2}(n) [ e_{M-1}^b(0) \ e_{M-1}^b(1) \ \dots \ e_{M-1}^b(n) ]^T$  é o vetor do erro de predição regressiva ponderado de ordem  $M - 1$ . O vetor  $\mathbf{e}_{M-1}^b(n)$  é a diferença entre o vetor de respostas desejadas ponderado, que neste caso é representado pela última coluna da matriz  $\mathbf{A}_M(n)$ , definida na equação (A.7) do apêndice A como  $\mathbf{a}(n - M + 1)$ , e o vetor estimado destas respostas

$$\hat{\mathbf{d}}^d(n) = \mathbf{A}_{M-1}(n)\mathbf{w}_{M-1}^b(n). \quad (2.26)$$

Desta forma, o problema de predição regressiva, no tempo  $n$  e na ordem  $M - 1$ , consiste em determinar o vetor de coeficientes  $\mathbf{w}_{M-1}^b(n)$  que minimiza a função

$$\xi_{M-1}^b(n) = \|\mathbf{e}_{M-1}^b(n)\|^2. \quad (2.27)$$

Trata-se portanto, de um problema de estimação em que o vetor do sinal desejado é  $\mathbf{a}(n - M + 1)$  e o vetor estimado é dado por uma combinação linear dos elementos de  $U_\ell$ . Sendo assim, os resultados já apresentados para o problema de estimação, bem como a interpretação geométrica, são válidos, desde que sejam feitas as devidas adaptações. Consequentemente, a função (2.27) é minimizada quando o vetor de coeficientes satisfaz

$$\mathbf{w}_{M-1}^b(n) = \mathbf{\Phi}_{M-1}^{-1}(n)\mathbf{\Omega}_{M-1}^b(n), \quad (2.28)$$

onde  $\mathbf{\Omega}_{M-1}^b(n)$  é o vetor de correlação cruzada de dimensão  $(M - 1) \times 1$ , conforme definido em (A.13). Neste caso, o operador de projeção

$$\mathbf{P}_{M-1}(n) = \mathbf{A}_{M-1}(n)\mathbf{\Phi}_{M-1}^{-1}(n)\mathbf{A}_{M-1}^T(n), \quad (2.29)$$

projeta o vetor  $\mathbf{a}(n - M + 1)$  no subespaço gerado pelas colunas da matriz  $\mathbf{A}_{M-1}(n)$ , isto é, pelos vetores  $[ \mathbf{a}(n) \ \mathbf{a}(n - 1) \ \dots \ \mathbf{a}(n - M + 2) ]$ , que dependem das amostras passadas  $u(\ell - M + 3)$ , com  $0 \leq \ell \leq n - 1$ . Observa-se que o operador de projeção da predição regressiva coincide com o operador de projeção do problema de estimação de ordem  $M - 1$ , pois  $M - 1$  vetores no instante  $n$  que geram o espaço onde ocorre a projeção coincidem nos dois casos. Os vetores da predição regressiva e do erro da predição regressiva ótimos satisfazem

$$\hat{\mathbf{d}}^b(n) = \mathbf{P}_{M-1}(n)\mathbf{a}(n - M + 1) \quad (2.30)$$

e

$$\mathbf{e}_{M-1}^b(n) = (\mathbf{I} - \mathbf{P}_{M-1}(n)) \mathbf{a}(n - M + 1). \quad (2.31)$$

Considerando (2.31), a energia mínima do erro da predição regressiva *a posteriori* pode ser expressa como

$$\xi_{M-1}^b(n) = \mathbf{e}_{M-1}^{bT}(n) \mathbf{a}(n - M + 1) = \mathbf{a}^T(n - M + 1) (\mathbf{I} - \mathbf{P}_{M-1}(n)) \mathbf{a}(n - M + 1). \quad (2.32)$$

Utilizando (A.12), (2.30), (2.26) e (A.13), a equação (2.32) pode então ser reescrita como

$$\xi_{M-1}^b(n) = E_o^b(n) - \boldsymbol{\Omega}_{M-1}^{bT}(n) \mathbf{w}_{M-1}^b(n). \quad (2.33)$$

Com isso, é estabelecida uma interpretação às grandezas das igualdades (A.12) e (A.13):  $\boldsymbol{\Omega}_{M-1}^b(n)$  é o vetor de correlação cruzada entre a resposta desejada e a matriz de dados de um problema de predição de ordem  $M - 1$  e  $E_o^b(n)$  é a energia do vetor de resposta desejada  $\mathbf{a}(n - M + 1)$  de um problema de predição regressiva de ordem zero, que é a própria energia do sinal de entrada ponderado no instante  $n - M + 1$ .

### Predição progressiva

Com a predição linear progressiva, procura-se prever um valor futuro de um processo a partir de observações passadas. No caso da predição progressiva de ordem  $M - 1$  e no tempo  $\ell$ , procura-se prever o valor  $u(\ell)$  para  $\ell \geq M - 1$ , a partir do conjunto de  $M - 1$  amostras passadas

$$U_{\ell-1} = \{u(\ell - M + 1), \dots, u(\ell - 2), u(\ell - 1)\}.$$

Define-se o erro de predição progressiva, no instante  $\ell$  como

$$f_{M-1}(\ell) = u(\ell) - \hat{u}(\ell|U_{\ell-1}). \quad (2.34)$$

As amostras estimadas  $\hat{u}(\ell|U_{\ell-1})$  são obtidas como

$$\hat{u}(\ell|U_{\ell-1}) = \sum_{k=1}^{M-1} w_{k-1}^f(\ell) u(\ell - k) = \mathbf{u}_{M-1}^T(\ell - 1) \mathbf{w}_{M-1}^f(\ell),$$

onde  $\mathbf{w}_{M-1}^f(\ell)$  é o vetor de coeficientes da predição progressiva. Considerando a expressão (2.34), porém com as amostras estimadas obtidas com o vetor de coeficientes no instante  $n$ , isto é

$$\sum_{k=1}^{M-1} w_{k-1}^f(n) u(\ell - k) = \mathbf{u}_{M-1}^T(\ell - 1) \mathbf{w}_{M-1}^f(n) \quad \text{para } 0 \leq \ell \leq n, \quad (2.35)$$



resulta a seguinte expressão vetorial

$$\mathbf{e}_{M-1}^f(n) = \mathbf{a}(n) - \begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} \mathbf{w}_{M-1}^f(n), \quad (2.36)$$

onde,  $\mathbf{e}_{M-1}^f(n) = \mathbf{\Lambda}^{1/2}(n) \left[ e_{M-1}^f(0) \ e_{M-1}^f(1) \ \dots \ e_{M-1}^f(n) \right]^T$  é o vetor do erro de predição progressiva ponderado de ordem  $M-1$ . O vetor  $\mathbf{e}_{M-1}^f(n)$  é a diferença entre o vetor de respostas desejadas ponderado, que neste caso é representado pela primeira coluna da matriz  $\mathbf{A}_M(n)$ , definida na equação (A.8) do apêndice A como  $\mathbf{a}(n)$ , e o vetor estimado destas respostas

$$\hat{\mathbf{d}}^f(n) = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} \mathbf{w}_{M-1}^f(n). \quad (2.37)$$

Desta forma, o problema de predição progressiva, no tempo  $n$  e na ordem  $M-1$ , consiste em determinar o vetor de coeficientes  $\mathbf{w}_{M-1}^f(n)$  que minimiza a função

$$\xi_{M-1}^f(n) = \left\| \mathbf{e}_{M-1}^f(n) \right\|^2. \quad (2.38)$$

Trata-se portanto, de um problema de estimação em que o vetor do sinal desejado é  $\mathbf{a}(n)$  e o vetor estimado é dado por uma combinação linear dos elementos de  $U_{\ell-1}$ . Desde que sejam feitas as devidas adaptações, os resultados já apresentados para o problema de estimação são também válidos para o caso da predição progressiva. A função (2.38) é minimizada quando o vetor de coeficientes satisfaz

$$\mathbf{w}_{M-1}^f(n) = \mathbf{\Phi}_{M-1}^{-1}(n-1) \mathbf{\Omega}_{M-1}^f(n), \quad (2.39)$$

onde  $\mathbf{\Omega}_{M-1}^f(n)$  é o vetor da correlação cruzada de dimensão  $(M-1) \times 1$  conforme definido na equação (A.16) do apêndice A. Neste caso, o operador de projeção

$$\mathbf{P}_{M-1}^f(n) = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{P}_{M-1}(n-1) \end{bmatrix}, \quad (2.40)$$

projeta a resposta desejada  $\mathbf{a}(n)$ , que consiste das amostras futuras  $u(\ell)$ ,  $0 \leq \ell \leq n$ , no subespaço gerado pelos vetores  $\{ \mathbf{a}(n-1) \ \mathbf{a}(n-2) \ \dots \ \mathbf{a}(n-M+1) \}$ . Cabe observar que no problema de predição progressiva a projeção do vetor desejado é feita no espaço gerado pelas últimas  $M-1$  colunas da matriz  $\mathbf{A}_M(n)$  (equação (A.4)). Como estas colunas podem ser representadas como em (2.37), envolvendo somente

amostras de sinais anteriores ao instante  $n$ , é bastante razoável que  $\mathbf{P}_{M-1}^f(n)$  esteja diretamente relacionado com  $\mathbf{P}_{M-1}(n-1)$ . Os vetores da predição progressiva e de erro da predição progressiva ótimos satisfazem

$$\hat{\mathbf{d}}^f(n) = \mathbf{P}_{M-1}^f(n)\mathbf{a}(n) \quad (2.41)$$

e

$$\mathbf{e}_{M-1}^f(n) = (\mathbf{I} - \mathbf{P}_{M-1}^f(n))\mathbf{a}(n). \quad (2.42)$$

Considerando (2.42) a energia mínima do erro da predição progressiva pode ser expressa como

$$\xi_{M-1}^f(n) = \mathbf{e}_{M-1}^{fT}(n)\mathbf{a}(n) = \mathbf{a}^T(n) (\mathbf{I} - \mathbf{P}_{M-1}^f(n))\mathbf{a}(n). \quad (2.43)$$

Considerando (A.15), (2.41), (2.37) e (A.16), a equação (2.43) pode então ser reescrita como

$$\xi_{M-1}^f(n) = \xi_0^f(n) - \boldsymbol{\Omega}_{M-1}^{fT}(n)\mathbf{w}_{M-1}^f(n). \quad (2.44)$$

Deste modo, neste item foi dada uma interpretação às grandezas das igualdades (A.15) e (A.16), onde  $\boldsymbol{\Omega}_{M-1}^f(n)$  é o vetor de correlação cruzada entre a resposta desejada e a matriz de dados de um problema de predição progressiva de ordem  $M-1$  e  $\xi_0^f(n)$  é a energia do vetor de respostas desejadas  $\mathbf{a}(n)$  de um problema de predição progressiva de ordem zero, que é a própria energia do sinal de entrada, ponderado no instante  $n$ .

### Conseqüências do princípio da ortogonalidade

A seguir, são apresentadas as relações em que os vetores dos erros das predições regressiva e progressiva satisfazem como conseqüência do princípio da ortogonalidade.

- 1) Como o problema da predição regressiva é um problema de estimação para um vetor de resposta desejada particular, o vetor do erro da predição regressiva  $\mathbf{e}_{M-1}^b(n)$  é ortogonal ao espaço das colunas da matriz  $\mathbf{A}_{M-1}(n)$ . Desta forma, vale

$$\mathbf{e}_{M-1}^{bT}(n)\mathbf{A}_{M-1}(n) = \mathbf{0} \quad (2.45)$$

e

$$\mathbf{e}_{M-1}^{bT}(n)\mathbf{P}_{M-1}(n) = \mathbf{0} \quad (2.46)$$

Reescrevendo a equação (2.20) para ordem  $M - 1$  e usando (2.46), resulta a seguinte igualdade de produto interno de vetores

$$\mathbf{e}_{M-1}^{bT}(n)\mathbf{e}_{M-1}(n) = \mathbf{e}_{M-1}^{bT}(n)\mathbf{d}(n). \quad (2.47)$$

Reescrevendo a equação (2.46) para o instante de tempo  $n - 1$  e usando (2.42) e (2.40), resulta na seguinte igualdade de produto interno de vetores:

$$\begin{bmatrix} 0 & \mathbf{e}_{M-1}^{bT}(n-1) \end{bmatrix} \mathbf{e}_{M-1}^f(n) = \begin{bmatrix} 0 & \mathbf{e}_{M-1}^{bT}(n-1) \end{bmatrix} \mathbf{a}(n). \quad (2.48)$$

Conclui-se portanto, que as componentes do sinal desejado e do erro de estimação na direção do erro de predição regressivo são iguais. Resultado que se aplica tanto no caso do problema da estimação, (2.47), como no caso da predição progressiva, (2.48).

- 2) Como o problema da predição progressiva também é um problema de estimação para um vetor de resposta desejada particular, o vetor do erro da predição progressiva  $\mathbf{e}_{M-1}^f(n)$  é ortogonal ao espaço das colunas da matriz  $\mathbf{A}_{M-1}(n-1)$ . Desta forma vale

$$\mathbf{e}_{M-1}^{fT}(n) \begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} = \mathbf{0}. \quad (2.49)$$

A equação (2.31) pode ser reescrita para o instante de tempo  $n - 1$  e para ordem  $M - 2$ , como

$$\begin{bmatrix} 0 \\ \mathbf{e}_{M-2}^b(n-1) \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{I} - \mathbf{P}_{M-2}(n-1) \end{bmatrix} \mathbf{a}(n-M+1). \quad (2.50)$$

Esta equação é válida para  $M \geq 2$ , situação em que o primeiro elemento do vetor  $\mathbf{a}(n - M + 1)$  é nulo devido ao pré-janelamento. Cabe observar que  $\mathbf{a}(n - M + 1)$  é o vetor dos dados ponderados no instante  $n - M + 1$  e tem  $n + 1$  elementos. Usando (2.50) e (2.49) para ordem  $M - 2$  resulta na seguinte igualdade de produto interno de vetores:

$$\mathbf{e}_{M-2}^{fT}(n) \begin{bmatrix} 0 \\ \mathbf{e}_{M-2}^b(n-1) \end{bmatrix} = \mathbf{e}_{M-2}^{fT}(n)\mathbf{a}(n-M+1). \quad (2.51)$$

As equações (2.45), (2.47), (2.48), (2.49) e (2.51) são válidas para qualquer ordem  $i - 1$  com  $1 \leq i \leq M$ . Lembrando que os dados são reais então vale a seguinte relação

$$\Delta_{i-1}(n) = \mathbf{e}_{i-1}^{fT}(n) \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{e}_{i-1}^{bT}(n-1) \end{bmatrix} \mathbf{e}_{i-1}^f(n) \quad (2.52)$$

para  $1 \leq i \leq M$ . Cabe observar ainda que  $\Delta_{i-1}(n)$  representa a correlação cruzada entre os vetores de erro da predição progressiva e regressiva, quando as estimações das respostas desejadas  $\mathbf{a}(n)$  e  $\mathbf{a}(n-i)$  são feitas no mesmo subespaço, isto é, no subespaço definido pelas colunas da matriz  $\begin{bmatrix} \mathbf{0} & \mathbf{A}_{i-1}^T(n-1) \end{bmatrix}$ .

## 2.2 Algoritmo RLS convencional

A solução exata do problema dos mínimos quadrados pode ser obtida resolvendo-se a equação (2.5) de forma recorrente no tempo. Um exemplo é o algoritmo RLS (Recursive Least Squares) convencional, que calcula o vetor de coeficientes ótimos do instante de tempo  $n$  a partir do vetor de coeficientes do instante de tempo  $n-1$  e da observação no instante  $n$ . Conforme [38, cap. 13] a operação de adaptação deste algoritmo é obtida substituindo-se na expressão do vetor de coeficientes  $\mathbf{w}_M(n)$  ótimos definido em (2.5), as expressões de atualização no tempo da inversa da matriz de autocorrelação (A.17) e do vetor de correlação cruzada (A.33), resultando em

$$\mathbf{w}_M(n) = \mathbf{w}_M(n-1) + \mathbf{g}_M(n)\alpha_M(n). \quad (2.53)$$

Em (2.53),  $\alpha_M(n)$  é o erro de estimação *a priori*, que é obtido através da operação de filtragem

$$\alpha_M(n) = d(n) - \mathbf{u}_M^T(n)\mathbf{w}_M(n-1), \quad (2.54)$$

onde  $\mathbf{u}_M(n) = [u(n) \ u(n-1) \ \dots \ u(n-M+1)]^T$  é o vetor dos dados de entrada no instante  $n$  para ordem  $M$ . O vetor  $\mathbf{g}_M(n)$ , conforme demonstrado no apêndice A, equações (A.20) e (A.21), pode ser obtido como

$$\mathbf{g}_M(n) = \Phi_M^{-1}(n)\mathbf{u}_M(n) = \lambda^{-1}\gamma_M(n)\Phi_M^{-1}(n-1)\mathbf{u}_M(n). \quad (2.55)$$

Por sua vez, o fator de conversão  $\gamma_M(n)$ , definido como o quociente entre os erros de estimação *a posteriori* e *a priori* [38, 15], equações (A.19) e (A.22) do apêndice A,

satisfaz

$$\gamma_M(n) = 1 - \mathbf{u}_M^T(n) \Phi_M^{-1}(n) \mathbf{u}_M(n) = \frac{\lambda}{\lambda + \mathbf{u}_M^T(n) \Phi_M^{-1}(n-1) \mathbf{u}_M(n)}. \quad (2.56)$$

Por conveniência, a operação de filtragem será representada aqui, através do erro normalizado em ângulo. Usando a relação entre os erros de estimação *a priori* e *a posteriori* (A.40), define-se o erro normalizado em ângulo como [38, 20]

$$\epsilon_M(n) := \gamma_M^{1/2}(n) \alpha_M(n) = \gamma_M^{-1/2}(n) e_M(n). \quad (2.57)$$

Reunindo (2.53) e (2.57), resulta no seguinte conjunto de equações recursivas no tempo, as quais caracterizam as operações de adaptação e de filtragem do algoritmo RLS convencional:

$$\begin{bmatrix} \mathbf{w}_M(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^c(n) \begin{bmatrix} \mathbf{w}_M(n-1) \\ d(n) \end{bmatrix}, \quad (2.58)$$

sendo  $\mathbf{S}_M^c(n)$  a matriz de sistema, com dimensão  $(M+1) \times (M+1)$ , definida como

$$\mathbf{S}_M^c(n) = \begin{bmatrix} \mathbf{I} - \mathbf{g}_M(n) \mathbf{u}_M^T(n) & \mathbf{g}_M(n) \\ -\gamma_M^{1/2}(n) \mathbf{u}_M^T(n) & \gamma_M^{1/2}(n) \end{bmatrix}. \quad (2.59)$$

A equação (2.58) resolve o problema de estimação definido no item 2.1. Entretanto, o algoritmo RLS convencional é susceptível ao ruído numérico em aritmética de precisão finita [38, cap. 19]. O maior problema dos erros ocorrem na atualização da inversa da matriz de autocorrelação [21]. Uma forma de amenizar este problema é efetuar uma triangularização ortogonal conveniente na matriz de dados, o que significa efetuar uma troca de coordenadas em (2.58). Na verdade, o sistema (2.58) pode ser transformado em um número ilimitado de sistemas com o mesmo comportamento entrada-saída, e resolvendo o mesmo problema dos mínimos quadrados. Apesar dos sistemas resultantes serem teoricamente equivalentes, o comportamento numérico pode variar conforme o sistema de coordenadas usado. A equivalência entre eles é exata somente no caso de aritmética de precisão infinita [39]. Além dos problemas numéricos, o algoritmo RLS convencional apresenta uma complexidade computacional proporcional a  $M^2$ . Entretanto, utilizando as predições progressiva e regressiva para resolver o problema de estimação, é possível obter algoritmos com complexidade computacional proporcional a  $M$ . As versões com complexidade computacional proporcional a  $M$  são conhecidas como algoritmos RLS rápidos.

Os problemas de predição regressiva e progressiva são casos particulares do problema de estimação em que se considera sinais desejados bem específicos. No caso da predição regressiva de ordem  $M - 1$ , o sinal desejado no instante  $n$  é  $u(n - M + 1)$ . Sendo que a matriz de sistema, por não depender do sinal desejado é idêntica ao problema de estimação de ordem  $M - 1$  no instante  $n$ . O conjunto de equações que caracterizam as operações de adaptação e de filtragem são:

$$\begin{bmatrix} \mathbf{w}_{M-1}^b(n) \\ \epsilon_{M-1}^b(n) \end{bmatrix} = \mathbf{S}_{M-1}^c(n) \begin{bmatrix} \mathbf{w}_{M-1}^b(n-1) \\ u(n-M+1) \end{bmatrix}. \quad (2.60)$$

Neste caso, o erro de predição regressiva normalizado em ângulo  $\epsilon_{M-1}^b(n)$ , é definido a partir do erro de predição regressiva *a priori*

$$\psi_{M-1}(n) := u(n - M + 1) - \mathbf{u}_{M-1}^T(n) \mathbf{w}_{M-1}^b(n - 1), \quad (2.61)$$

ou do erro de predição regressiva *a posteriori* (A.41)

$$b_{M-1}(n) = u(n - M + 1) - \mathbf{u}_{M-1}^T(n) \mathbf{w}_{M-1}^b(n),$$

como sendo

$$\epsilon_{M-1}^b(n) := \gamma_{M-1}^{1/2}(n) \psi_{M-1}(n) = \gamma_{M-1}^{-1/2}(n) b_{M-1}(n). \quad (2.62)$$

Cabe notar que  $\gamma_{M-1}(n)$ , quociente entre os erros de estimação *a priori* e *a posteriori*, não depende do particular sinal desejado do problema de estimação.

Na predição progressiva do instante de tempo  $n$ , o sinal desejado é  $u(n)$  e a projeção do vetor de resposta desejada para ordem  $M - 1$  é feita no espaço definido pela matriz de dados  $\mathbf{A}_{M-1}(n - 1)$ . Neste caso, a matriz de sistema é a mesma que no caso de estimação para a ordem  $M - 1$  e instante  $n - 1$ . O conjunto de equações que caracterizam as operações de adaptação e de filtragem são:

$$\begin{bmatrix} \mathbf{w}_{M-1}^f(n) \\ \epsilon_{M-1}^f(n) \end{bmatrix} = \mathbf{S}_{M-1}^c(n-1) \begin{bmatrix} \mathbf{w}_{M-1}^f(n-1) \\ u(n) \end{bmatrix}. \quad (2.63)$$

O erro de predição progressiva normalizado em ângulo  $\epsilon_{M-1}^f(n)$  é definido a partir do erro de predição progressiva *a priori*

$$\eta_{M-1}(n) := u(n) - \mathbf{u}_{M-1}^T(n-1) \mathbf{w}_{M-1}^f(n-1) \quad (2.64)$$

e do erro de predição progressiva *a posteriori* (A.44)

$$f_{M-1}(n) = u(n) - \mathbf{u}_{M-1}^T(n-1)\mathbf{w}_{M-1}^f(n),$$

como sendo

$$\epsilon_{M-1}^f(n) := \gamma_{M-1}^{1/2}(n-1)\eta_{M-1}(n) = \gamma_{M-1}^{-1/2}(n-1)f_{M-1}(n). \quad (2.65)$$

## 2.3 Triangularização ortogonal

Conforme mencionado na introdução a triangularização ortogonal da matriz de dados pode ser utilizada com vantagens para resolver o problema dos mínimos quadrados [19, 21, 38]. Genericamente, fala-se em aplicar a decomposição QR na matriz de dados, onde  $Q$  representa uma matriz ortogonal e  $R$  uma matriz triangular. Tal decomposição da matriz de dados pode ser efetuada usando-se por exemplo, o procedimento de ortogonalização de Gram-Schmidt ou o procedimento de ortogonalização baseado em rotações de Givens. Como consequência da aplicação da decomposição QR ser efetuada diretamente na matriz de dados obtém-se um melhor comportamento numérico dos algoritmos [21, 38]. Apresenta-se a seguir o procedimento de ortogonalização baseado em rotações de Givens e o procedimento de ortogonalização de Gram-Schmidt.

A triangularização da matriz de dados pode ser convenientemente descrita por

$$\mathbf{Q}_M(n)\mathbf{A}_M(n) = \begin{bmatrix} \mathbf{R}_M(n) \\ \mathbf{O} \end{bmatrix}, \quad (2.66)$$

onde  $\mathbf{Q}_M(n)$  é uma matriz unitária de dimensão  $(n+1) \times (n+1)$ ,  $\mathbf{R}_M(n)$  é uma matriz triangular superior de dimensão  $M \times M$  e  $\mathbf{O}$  é uma matriz de zeros de ordem  $(n-M+1) \times M$ . Como  $\mathbf{A}_M(n)$  é uma matriz de dimensão  $(n+1) \times M$ , é interessante considerar a partição da matriz  $\mathbf{Q}_M(n)$  em duas matrizes, uma de dimensão  $M \times (n+1)$  representada por  $\mathbf{Q1}_M(n)$ , e outra de dimensão  $(n-M+1) \times (n+1)$  representada por  $\mathbf{Q2}_M(n)$ , isto é

$$\mathbf{Q}_M(n) = \begin{bmatrix} \mathbf{Q1}_M(n) \\ \mathbf{Q2}_M(n) \end{bmatrix}.$$

Decorrente do fato da matriz  $\mathbf{Q}_M(n)$  ser uma matriz unitária, ou seja

$$\mathbf{Q}_M^T(n)\mathbf{Q}_M(n) = \mathbf{Q}_M(n)\mathbf{Q}_M^T = \mathbf{I}_{(n+1) \times (n+1)}, \quad (2.67)$$

onde  $\mathbf{I}$  é a matriz identidade, tem-se

$$\mathbf{Q}\mathbf{1}_M(n)\mathbf{Q}\mathbf{1}_M^T(n) = \mathbf{I}_{M \times M}, \quad (2.68)$$

$$\mathbf{Q}\mathbf{2}_M(n)\mathbf{Q}\mathbf{2}_M^T(n) = \mathbf{I}_{(n+1-M) \times (n+1-M)} \quad (2.69)$$

e da equação (2.66), obtém-se a igualdade

$$\mathbf{A}_M(n) = \mathbf{Q}\mathbf{1}_M^T(n)\mathbf{R}_M(n). \quad (2.70)$$

Esta última equação representa a matriz de dados através do produto da matriz ortogonal  $\mathbf{Q}\mathbf{1}_M^T(n)$  e da matriz triangular superior  $\mathbf{R}_M(n)$ .

A função custo do problema dos mínimos quadrados, equação (2.1), fica inalterada se incluímos nos cálculos a matriz unitária  $\mathbf{Q}_M(n)$ , isto é,

$$\|\mathbf{Q}_M(n)\mathbf{e}_M(n)\|^2 = \|\mathbf{Q}_M(n)\mathbf{d}(n) - \mathbf{Q}_M(n)\mathbf{A}_M(n)\mathbf{w}_M(n)\|^2. \quad (2.71)$$

Introduzindo a notação

$$\begin{bmatrix} \mathbf{Q}\mathbf{1}_M(n)\mathbf{d}(n) \\ \mathbf{Q}\mathbf{2}_M(n)\mathbf{d}(n) \end{bmatrix} = \begin{bmatrix} \mathbf{d}\mathbf{q}\mathbf{1}(n) \\ \mathbf{d}\mathbf{q}\mathbf{2}(n) \end{bmatrix} = \mathbf{d}\mathbf{q}(n) \quad (2.72)$$

obtém-se com (2.66) a seguinte expressão para o vetor dos erro da estimação *a posteriori* rotacionado

$$\mathbf{Q}_M(n)\mathbf{e}_M(n) = \begin{bmatrix} \mathbf{d}\mathbf{q}\mathbf{1}(n) \\ \mathbf{d}\mathbf{q}\mathbf{2}(n) \end{bmatrix} - \begin{bmatrix} \mathbf{R}_M(n) \\ \mathbf{O} \end{bmatrix} \mathbf{w}_M(n). \quad (2.73)$$

De (2.73) conclui-se que a norma Euclidiana do vetor dos erros da estimação *a posteriori* é minimizada quando a seguinte equação é satisfeita

$$\mathbf{d}\mathbf{q}\mathbf{1}(n) = \mathbf{R}_M(n)\mathbf{w}_M(n), \quad (2.74)$$

condição em que

$$\xi_M(n) = \|\mathbf{e}_M(n)\|^2 = \|\mathbf{d}\mathbf{q}\mathbf{2}(n)\|^2. \quad (2.75)$$

Como  $\mathbf{R}_M(n)$  é uma matriz triangular superior, o vetor de coeficientes ótimos pode ser obtido de (2.74) pelo método de substituições sucessivas.

Com (2.70), (2.74) e (2.72), conclui-se que o vetor estimado ótimo satisfaz

$$\hat{\mathbf{d}}(n) = \mathbf{A}_M(n)\mathbf{w}_M(n) = \mathbf{Q}\mathbf{1}_M^T(n)\mathbf{Q}\mathbf{1}_M(n)\mathbf{d}(n). \quad (2.76)$$



De (2.73), (2.74) e (2.72) resulta que o erro de estimação mínimo satisfaz

$$\mathbf{Q}_M(n)\mathbf{e}_M(n) = \begin{bmatrix} \mathbf{0} \\ \mathbf{Q}\mathbf{2}_M(n)\mathbf{d}(n) \end{bmatrix} \quad (2.77)$$

de modo que

$$\mathbf{e}_M(n) = \mathbf{Q}\mathbf{2}_M^T(n)\mathbf{Q}\mathbf{2}_M(n)\mathbf{d}(n). \quad (2.78)$$

Desta forma, obtém-se uma nova expressão para o operador de projeção, definido por (2.29), agora em função da matriz  $\mathbf{Q}\mathbf{1}_M(n)$ :

$$\mathbf{P}_M(n) = \mathbf{Q}\mathbf{1}_M^T(n)\mathbf{Q}\mathbf{1}_M(n). \quad (2.79)$$

A expressão (2.78) mostra por sua vez, o operador de projeção complementar como função da matriz  $\mathbf{Q}\mathbf{2}_M(n)$ ,

$$\mathbf{I} - \mathbf{P}_M(n) = \mathbf{Q}\mathbf{2}_M^T(n)\mathbf{Q}\mathbf{2}_M(n). \quad (2.80)$$

A partir de (2.70) e (2.72) resulta que o vetor de correlação cruzada da equação normal é dado por

$$\mathbf{\Omega}_M(n) = \mathbf{A}_M^T(n)\mathbf{d}(n) = \mathbf{R}_M^T(n)\mathbf{d}\mathbf{q}\mathbf{1}(n). \quad (2.81)$$

Finalmente, a matriz de autocorrelação satisfaz

$$\mathbf{\Phi}_M(n) = \mathbf{R}_M^T(n)\mathbf{R}_M(n), \quad (2.82)$$

o que decorre das relações (2.70) e (2.68). A matriz  $\mathbf{R}_M(n)$  é uma das representações possíveis para a matriz  $\mathbf{L}_M(n)$ , apresentada em (2.10), e  $\mathbf{Q}\mathbf{1}_M^T(n)$  é uma das representações possíveis para  $\mathbf{A}'_M(n)$ , apresentada em (2.11).

A matriz  $\mathbf{Q}_M(n)$  pode ser interpretada como uma rotação do subespaço gerado pelas colunas da matriz  $\mathbf{A}_M(n)$  e pelo erro de estimação ótimo  $\mathbf{e}_M(n)$ , o qual é ortogonal as colunas da matriz  $\mathbf{A}_M(n)$ . No caso, a matriz  $\mathbf{Q}_M(n)$  faz coincidir o subespaço gerado pelas colunas da matriz  $\mathbf{A}_M(n)$ , de dimensão  $M$ , com o subespaço correspondente aos  $M$  primeiros vetores da base do espaço euclidiano, de dimensão  $(n+1) \times (n+1)$ , utilizado na representação. O subespaço de dimensão  $n+1-M$ , que contém o erro de estimação ótimo  $\mathbf{e}_M(n)$  e é ortogonal ao espaço das colunas de  $\mathbf{A}_M(n)$ , coincide então com o subespaço gerado pelos demais  $n+1-M$  vetores da base

euclidiana. A figura 2.1 exemplifica estas interpretações. Tal transformação afeta o sinal de entrada, o sinal desejado e o operador de projeção, sem alterar a matriz de autocorrelação, o vetor de correlação cruzada e o vetor de coeficientes ótimos. A relação entre os algoritmos RLS convencional e o QR-RLS está apresentada na tabela 2.1, nas últimas três linhas está indicado como as grandezas que não se alteram com a transformação podem ser calculadas a partir de grandezas convencionais ou rotacionadas. Cabe destacar que no domínio dos sinais rotacionados pela matriz  $Q_M(n)$ , o operador de projeção e o seu complementar assumem uma forma muito simples.

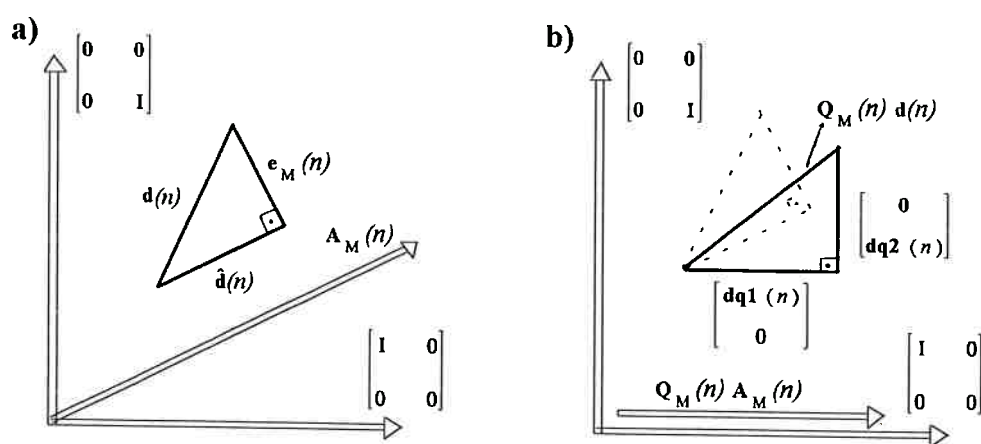


Figura 2.1: a) Estimativa de  $d(n)$  projetada no subespaço definido pelas colunas de  $A_M(n)$ ; b) estimativa de  $dq(n)$  projetada na direção do subespaço da base euclidiana.

Tabela 2.1

Relações entre o algoritmo RLS convencional e o baseado em rotações		
Algoritmo	RLS convencional	QR-RLS
Matriz de dados	$\mathbf{A}_M(n)$	$\begin{bmatrix} \mathbf{R}_M(n) \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_M(n)\mathbf{A}_M(n)$
Resp. desejada	$\mathbf{d}(n)$	$\begin{bmatrix} \mathbf{dq1}(n) \\ \mathbf{dq2}(n) \end{bmatrix} = \mathbf{Q}_M(n)\mathbf{d}(n)$
Resp. estimada	$\mathbf{P}_M(n)\mathbf{d}(n)$	$\begin{bmatrix} \mathbf{dq1}(n) \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{Q1}_M(n) \\ \mathbf{0} \end{bmatrix} \mathbf{d}(n)$
Erro de estimação	$(\mathbf{I}_{n \times n} - \mathbf{P}_M(n))\mathbf{d}(n)$	$\begin{bmatrix} \mathbf{0} \\ \mathbf{dq2}(n) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{Q2}_M(n) \end{bmatrix} \mathbf{d}(n)$
Op. de projeção	$\mathbf{A}_M(n)\Phi_M^{-1}(n)\mathbf{A}_M^T(n)$	$\begin{bmatrix} \mathbf{I}_{M \times M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$
Op. proj. compl.	$\mathbf{I}_{n \times n} - \mathbf{P}_M(n)$	$\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{(n+1-M) \times (n+1-M)} \end{bmatrix}$
Matriz de autocor.	$\Phi_M(n) = \mathbf{A}_M^T(n)\mathbf{A}_M(n) = \mathbf{R}_M^T(n)\mathbf{R}_M(n)$	
Correl. cruzada	$\Omega_M(n) = \mathbf{A}_M^T(n)\mathbf{d}(n) = \mathbf{R}_M^T(n)\mathbf{dq1}(n)$	
Coef. ótimos	$\mathbf{w}_M(n) = \Phi_M^{-1}(n)\Omega_M(n) = \mathbf{R}_M^{-1}(n)\mathbf{dq1}(n)$	

### Atualizações no tempo através de rotações de Givens

Neste item é apresentado um procedimento para calcular a matriz  $\mathbf{R}_M(n)$  e o vetor  $\mathbf{dq1}(n)$  de forma recorrente no tempo. Considere que no instante  $n$  a matriz  $\mathbf{A}_M(n-1)$  já tenha sofrido uma triangularização ortogonal através da decomposição QR, isto é

$$\mathbf{Q}_M(n-1)\mathbf{A}_M(n-1) = \begin{bmatrix} \mathbf{R}_M(n-1) \\ \mathbf{0} \end{bmatrix}, \quad (2.83)$$

onde  $\mathbf{R}_M(n-1)$  é uma matriz triangular superior de ordem  $M \times M$  e  $\mathbf{0}$  é uma matriz de zeros de ordem  $(n-M) \times M$ . Para calcular a decomposição QR de  $\mathbf{A}_M(n)$ , supõe-se que o valor atualizado da matriz unitária  $\mathbf{Q}_M(n)$  está relacionado com  $\mathbf{Q}_M(n-1)$

pela seguinte expressão

$$\mathbf{Q}_M(n) = \mathbf{T}(n) \begin{bmatrix} \mathbf{Q}_M(n-1) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix}, \quad (2.84)$$

onde  $\mathbf{T}(n)$  é necessariamente uma matriz unitária, já que as matrizes da expressão (2.84) também o são. Representando a matriz de dados conforme (A.6), a matriz unitária conforme (2.84) e a triangularização ortogonal da matriz de dados  $\mathbf{A}_M(n-1)$  conforme (2.83), tem-se

$$\mathbf{Q}_M(n)\mathbf{A}_M(n) = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{R}_M(n-1) \\ \mathbf{0} \\ \mathbf{u}_M^T(n) \end{bmatrix} = \begin{bmatrix} \mathbf{R}_M(n) \\ \mathbf{0} \\ \mathbf{0}^T \end{bmatrix}. \quad (2.85)$$

Desta última expressão, é possível concluir que a matriz  $\mathbf{T}(n)$  deve anular os elementos do vetor  $\mathbf{u}_M^T(n)$ . Isto pode ser feito, como é mostrado a seguir, através de  $M$  rotações de Givens de modo que  $\mathbf{T}(n)$  será formada pelo produto de  $M$  rotações de Givens.

Considere que no instante  $n$  o vetor de respostas desejada  $\mathbf{d}(n-1)$  já tenha sofrido uma rotação

$$\mathbf{Q}_M(n-1)\mathbf{d}(n-1) = \begin{bmatrix} \mathbf{dq1}(n-1) \\ \mathbf{dq2}(n-1) \end{bmatrix}.$$

Representando o vetor de respostas desejadas como

$$\mathbf{d}(n) = \begin{bmatrix} \lambda^{1/2}\mathbf{d}(n-1) \\ d(n) \end{bmatrix}$$

e usando (2.84) é possível obter a seguinte expressão para atualização no tempo do vetor de respostas desejadas rotacionado

$$\begin{bmatrix} \mathbf{dq1}(n) \\ \mathbf{dq2}(n) \end{bmatrix} = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{dq1}(n-1) \\ \lambda^{1/2}\mathbf{dq2}(n-1) \\ d(n) \end{bmatrix}.$$

Assim, as atualizações no tempo da matriz de dados triangularizada  $\mathbf{R}_M(n)$  e do vetor de respostas desejadas rotacionado, podem ser convenientemente descritas por

$$\begin{aligned} & \mathbf{Q}_M(n) \cdot \begin{bmatrix} \mathbf{A}_M(n) & \mathbf{d}(n) \end{bmatrix} = \\ & = \mathbf{T}(n) \begin{bmatrix} \lambda^{1/2}\mathbf{R}_M(n-1) & \lambda^{1/2}\mathbf{dq1}(n-1) \\ \mathbf{0} & \lambda^{1/2}\mathbf{dq2}(n-1) \\ \mathbf{u}_M^T(n) & d(n) \end{bmatrix} = \begin{bmatrix} \mathbf{R}_M(n) & \mathbf{dq1}(n) \\ \mathbf{0} & \lambda^{1/2}\mathbf{dq2}(n-1) \\ \mathbf{0}^T & \epsilon_M(n) \end{bmatrix}. \end{aligned} \quad (2.86)$$

De acordo com (2.86) fica claro que a matriz  $\mathbf{T}(n)$  é formada por  $M$  rotações de Givens, isto é

$$\mathbf{T}(n) = \prod_{i=1}^M \mathbf{T}_i(n),$$

onde  $\mathbf{T}_i(n)$  tem a seguinte forma

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & c & 0 & \cdots & 0 & s \\ 0 & \cdots & 0 & 0 & 1 & & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & & \ddots & \vdots & \\ 0 & \cdots & 0 & 0 & 0 & & 1 & 0 \\ 0 & \cdots & 0 & -s & 0 & \cdots & 0 & c \end{bmatrix} \leftarrow i.$$

↑  
 $i$

O índice  $i$ ,  $1 \leq i \leq M$ , indica a coluna do elemento de  $\mathbf{u}_M^T(n)$  a ser anulado. Destas operações decorre que o elemento não nulo da última linha da matriz resultante em (2.86) é o erro de estimação *a posteriori* normalizado em ângulo  $\epsilon_M(n)$  definido em (2.57), conforme pode ser verificado na demonstração que se encontra no apêndice B. Eliminando-se da matriz  $\mathbf{T}(n)$  as linhas e colunas de índices  $M + 1$  até  $n$ , obtém-se uma matriz unitária de dimensão  $(M + 1) \times (M + 1)$ , aqui denominada de  $\mathbf{S}_M^g(n)$ . Com o auxílio desta matriz, pode-se reescrever parte da equação (2.86) como sendo

$$\mathbf{S}_M^g(n) \begin{bmatrix} \lambda^{1/2} \mathbf{R}_M(n-1) & \lambda^{1/2} \mathbf{dq1}(n-1) \\ \mathbf{u}_M^T(n) & d(n) \end{bmatrix} = \begin{bmatrix} \mathbf{R}_M(n) & \mathbf{dq1}(n) \\ \mathbf{0}^T & \epsilon_M(n) \end{bmatrix}. \quad (2.87)$$

Esta igualdade mostra que a matriz ortogonal  $\mathbf{S}_M^g(n)$ , além de atualizar a matriz  $\mathbf{R}_M(n)$  no tempo, também faz o papel de matriz de sistema que descreve as operações de adaptação e filtragem do algoritmo QR-RLS. Cabe observar que em (2.87) o vetor  $\mathbf{dq1}(n)$  tem o mesmo papel que  $\mathbf{w}_M(n)$  em (2.58). A transformação similar que relaciona ambos os sistemas é dada na expressão (2.74), aqui repetida por conveniência:

$$\mathbf{dq1}(n) = \mathbf{R}_M(n) \mathbf{w}_M(n).$$

A equação (2.87) descreve o algoritmo QR-RLS. Em [17] é demonstrado que o algoritmo QRD-RLS é estável em precisão finita no sentido BIBO ('bounded input bounded output'), isto é, entradas com magnitudes limitadas garantem saídas também com magnitudes limitadas. Porém, estabilidade no sentido BIBO não garante que os valores obtidos tenham significado, se comparados com o resultado esperado. Além disso, estes algoritmos apresentam uma complexidade computacional proporcional a  $M^2$ . Entretanto, utilizando as predições progressiva e regressiva para resolver o problema de estimação, é possível obter versões com uma complexidade computacional proporcional a  $M$ . As versões com complexidade computacional proporcional a  $M$ , são conhecidas como algoritmo QR-RLS rápidos. Uma análise detalhada da estabilidade nos algoritmos QR-RLS rápidos será apresentada no capítulo 4. A seguir, aplica-se a decomposição QR para os casos de predição progressiva e regressiva.

### Predição regressiva

A função custo do problema da predição regressiva na ordem  $M - 1$  e no tempo  $n$  fica inalterada se incluirmos nos cálculos a matriz unitária  $\mathbf{Q}_{M-1}(n)$ , isto é

$$\|\mathbf{Q}_{M-1}(n)\mathbf{e}_{M-1}^b(n)\|^2 = \|\mathbf{Q}_{M-1}(n)\mathbf{a}(n-M+1) - \mathbf{Q}_{M-1}(n)\mathbf{A}_{M-1}(n)\mathbf{w}_{M-1}^b(n)\|^2. \quad (2.88)$$

Introduzindo a notação

$$\begin{bmatrix} \mathbf{Q1}_{M-1}(n) \\ \mathbf{Q2}_{M-1}(n) \end{bmatrix} \mathbf{a}(n-M+1) = \begin{bmatrix} \mathbf{dq1}^b(n) \\ \mathbf{dq2}^b(n) \end{bmatrix}, \quad (2.89)$$

o vetor do erro da predição regressiva na ordem  $M - 1$  e no tempo  $n$ , rotacionado por  $\mathbf{Q}_{M-1}(n)$  pode então ser expresso como

$$\mathbf{Q}_{M-1}(n)\mathbf{e}_{M-1}^b(n) = \begin{bmatrix} \mathbf{dq1}^b(n) \\ \mathbf{dq2}^b(n) \end{bmatrix} - \begin{bmatrix} \mathbf{R}_{M-1}(n) \\ \mathbf{O} \end{bmatrix} \mathbf{w}_{M-1}^b(n). \quad (2.90)$$

De (2.90), conclui-se que a norma euclidiana do vetor dos erros de predição regressiva é minimizada quando

$$\mathbf{dq1}^b(n) = \mathbf{R}_{M-1}(n)\mathbf{w}_{M-1}^b(n) \quad (2.91)$$

condição em que

$$\xi_{M-1}^b(n) = \|\mathbf{e}_{M-1}^b(n)\|^2 = \|\mathbf{dq2}^b(n)\|^2. \quad (2.92)$$

De forma análoga à forma como foi obtida a expressão (2.87) para o problema de estimação, resulta para as atualizações no tempo da matriz  $\mathbf{R}_{M-1}(n)$  e do vetor  $\mathbf{dq1}^b(n)$  a seguinte expressão:

$$\begin{bmatrix} \mathbf{R}_{M-1}(n) & \mathbf{dq1}^b(n) \\ \mathbf{0} & \epsilon_{M-1}^b(n) \end{bmatrix} = \mathbf{S}_{M-1}^g(n) \begin{bmatrix} \lambda^{1/2} \mathbf{R}_{M-1}(n-1) & \lambda^{1/2} \mathbf{dq1}^b(n-1) \\ \mathbf{u}_{M-1}^T(n) & u(n-M+1) \end{bmatrix}. \quad (2.93)$$

### Predição progressiva

Na predição progressiva de ordem  $M-1$  e no instante de tempo  $n$ , a projeção do vetor de resposta desejada é feita no espaço definido pela matriz de dados  $\mathbf{A}_{M-1}(n-1)$ . Como a matriz a ser triangularizada é  $\mathbf{A}_{M-1}(n-1)$ , inclui-se a matriz unitária

$$\begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q}_{M-1}(n-1) \end{bmatrix}$$

na função custo do problema da predição progressiva de ordem  $M-1$  e no tempo  $n$ , isto é

$$\begin{aligned} & \left\| \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q}_{M-1}(n-1) \end{bmatrix} \mathbf{e}_{M-1}^f(n) \right\|^2 = \\ & \left\| \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q}_{M-1}(n-1) \end{bmatrix} \mathbf{a}(n) - \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q}_{M-1}(n-1) \end{bmatrix} \begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} \mathbf{w}_{M-1}^f(n) \right\|^2. \end{aligned} \quad (2.94)$$

Introduzindo a notação

$$\begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q1}_{M-1}(n-1) \\ \mathbf{0} & \mathbf{Q2}_{M-1}(n-1) \end{bmatrix} \mathbf{a}(n) = \begin{bmatrix} \lambda^{n/2} u(0) \\ \mathbf{dq1}^f(n) \\ \mathbf{dq2}^f(n) \end{bmatrix}, \quad (2.95)$$

o vetor de erro de predição progressiva de  $M-1$  e no tempo  $n$ , rotacionado por  $\mathbf{Q}_{M-1}(n-1)$ , pode então ser expresso como

$$\begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{Q1}_{M-1}(n-1) \\ \mathbf{0} & \mathbf{Q2}_{M-1}(n-1) \end{bmatrix} \mathbf{e}_{M-1}^f(n) = \begin{bmatrix} \lambda^{n/2} u(0) \\ \mathbf{dq1}^f(n) \\ \mathbf{dq2}^f(n) \end{bmatrix} - \begin{bmatrix} \mathbf{0}^T \\ \mathbf{R}_{M-1}(n-1) \\ \mathbf{0} \end{bmatrix} \mathbf{w}_{M-1}^f(n). \quad (2.96)$$

De (2.96), conclui-se que a norma euclidiana dos erros de predição progressiva é minimizada quando

$$\mathbf{dq1}^f(n) = \mathbf{R}_{M-1}(n-1)\mathbf{w}_{M-1}^f(n), \quad (2.97)$$

condição em que

$$\xi_{M-1}^f(n) = \|\mathbf{e}_{M-1}^f(n)\|^2 = \lambda^n |u(0)|^2 + \|\mathbf{dq2}^f(n)\|^2. \quad (2.98)$$

Analogamente à forma como foi obtida a expressão (2.87) para o problema de estimação, resulta para as atualizações no tempo da matriz  $\mathbf{R}_{M-1}(n-1)$  e do vetor  $\mathbf{dq1}^f(n)$  a seguinte expressão:

$$\begin{bmatrix} \mathbf{dq1}^f(n) & \mathbf{R}_{M-1}(n-1) \\ \mathbf{e}_{M-1}^f(n) & \mathbf{0}^T \end{bmatrix} = \mathbf{S}_{M-1}^g(n-1) \begin{bmatrix} \lambda^{1/2} \mathbf{dq1}^f(n-1) & \lambda^{1/2} \mathbf{R}_{M-1}(n-2) \\ u(n) & \mathbf{u}_{M-1}^T(n-1) \end{bmatrix}. \quad (2.99)$$

Os elementos da matriz de sistema que descrevem as operações de adaptação e filtragem no algoritmo RLS convencional, equação (2.59), estão bem definidos em termos do vetor de entrada  $\mathbf{u}_M(n)$ , do vetor de ganho de Kalman  $\mathbf{g}_M(n)$  e do fator de conversão  $\gamma_M(n)$ . Entretanto, não existe uma interpretação semelhante para os elementos da matriz de sistema que descreve as operações de adaptação e filtragem do algoritmo QR-RLS baseado em rotações de Givens. Uma interpretação deste tipo facilitaria a equivalência entre os métodos de resolução do problema dos mínimos quadrados, além de permitir uma visão geral das várias versões de algoritmos disponíveis. Visando interpretar os elementos da matriz de sistema  $\mathbf{S}_M^g(n)$  relaciona-se na seção 2.5, o método QR baseado em rotações de Givens com o procedimento de ortogonalização de Gram-Schmidt para resolver o problema de estimação dos mínimos quadrados. Antes porém, faz-se uma breve revisão do procedimento de ortogonalização de Gram-Schmidt aplicado a resolução dos problemas de estimação e de predição.

## 2.4 Ortogonalização de Gram-Schmidt

Na ortogonalização de Gram-Schmidt determina-se uma base de  $M$  vetores mutuamente ortogonais a partir de um conjunto de  $M$  vetores linearmente independentes [21]. No caso de dados seqüenciais esse conjunto é formado pelas colunas

$$\mathbf{a}(n-i+1) = \Lambda^{1/2}(n)[u(1-i) \dots u(n-i) u(n-i+1)]^T,$$



$1 \leq i \leq M$ , da matriz de dados  $\mathbf{A}_M(n)$ , equação (A.4) do apêndice A. O procedimento básico é formar vetores  $\mathbf{e}_{i-1}^b(n)$ , de dimensão  $(n+1) \times 1$  a partir das colunas  $\mathbf{a}(n-i+1)$  subtraindo de cada coluna  $\mathbf{a}(n-i+1)$  as suas componentes no espaço gerado pelo vetores  $\mathbf{e}_{j-1}^b(n)$ ,  $1 \leq j \leq i-1$ , isto é,

$$\mathbf{e}_{i-1}^b(n) = \mathbf{a}(n-i+1) - \hat{\mathbf{a}}(n-i+1), \quad \text{para } 2 \leq i \leq M \quad (2.100)$$

e

$$\mathbf{e}_0^b(n) = \mathbf{a}(n), \quad \text{para } i = 1.$$

O vetor  $\hat{\mathbf{a}}(n-i+1)$  representa a projeção da coluna  $\mathbf{a}(n-i+1)$  no subespaço descrito pelos vetores  $\mathbf{e}_{j-1}^b(n)$ :

$$\hat{\mathbf{a}}(n-i+1) = \sum_{j=1}^{i-1} k_{ji}(n) \mathbf{e}_{j-1}^b(n), \quad \text{para } i = 2, 3, \dots, M, \quad (2.101)$$

com

$$k_{ji}(n) = \frac{\mathbf{e}_{j-1}^{bT}(n) \mathbf{a}(n-i+1)}{\|\mathbf{e}_{j-1}^b(n)\|^2}, \quad j < i. \quad (2.102)$$

Combinando as equações (2.100) e (2.101), obtém-se

$$\mathbf{a}(n-i+1) = \mathbf{e}_{i-1}^b(n) + \sum_{j=1}^{i-1} k_{ji}(n) \mathbf{e}_{j-1}^b(n), \quad \text{para } 2 \leq i \leq M;$$

ou ainda, definindo  $k_{ii}(n) = 1$ ,  $1 \leq i \leq M$ ,

$$\mathbf{a}(n-i+1) = \sum_{j=1}^i k_{ji}(n) \mathbf{e}_{j-1}^b(n), \quad \text{para } 1 \leq i \leq M. \quad (2.103)$$

Considerando as  $M$  colunas da matriz  $\mathbf{A}_M(n)$ , chega-se a seguinte expressão matricial:

$$\underbrace{[\mathbf{a}(n) \ \dots \ \mathbf{a}(n-M+1)]}_{\mathbf{A}_M(n)} = \underbrace{[\mathbf{e}_0^b(n) \ \dots \ \mathbf{e}_{M-1}^b(n)]}_{\mathbf{G}_M(n)} \mathbf{K}_M(n), \quad (2.104)$$

onde  $\mathbf{K}_M(n)$  uma matriz triangular superior, de dimensão  $M \times M$ , com os elementos da diagonal principal unitários,  $k_{ji}(n) = 1$  para  $i = j$ . Os elementos da coluna  $i$  de  $\mathbf{K}_M(n)$  para  $i > j$ , são os coeficientes  $k_{ji}(n)$  obtidos das projeções sucessivas da coluna  $\mathbf{a}(n-i+1)$  sobre os vetores  $\mathbf{e}_{j-1}^b(n)$ ,  $1 \leq j \leq M-1$ . As  $M$  colunas ortogonais da matriz  $\mathbf{G}_M(n)$  não são normalizadas. Desta forma, o produto  $\mathbf{G}_M^T(n) \mathbf{G}_M(n) = \mathbf{D}_M(n)$  é uma matriz diagonal, cujos elementos representam a norma ao quadrado dos vetores ortogonais  $\mathbf{e}_{i-1}^b(n)$ , para  $1 \leq i \leq M$ , obtidos no procedimento de Gram-Schmidt.

A estimação de  $\mathbf{a}(n-i+1)$ , obtida em (2.101), está por construção no espaço gerado pelas  $i-1$  primeiras colunas da matriz  $\mathbf{A}_M(n)$ , que formam a matriz  $\mathbf{A}_{i-1}(n)$ . Como o vetor estimado é também ortogonal ao erro de estimação  $\mathbf{e}_{i-1}^b(n)$ , conclui-se que  $\mathbf{e}_{i-1}^b(n)$  representa o mesmo vetor de erros *a posteriori* da predição regressiva, conforme (2.25), porém para uma ordem  $i-1$ . Fazendo  $1 \leq i \leq M$  a matriz  $\mathbf{G}_M(n)$  pode ser reescrita como

$$\mathbf{G}_M(n) = [\mathbf{e}_0^b(n) \ \dots \ \mathbf{e}_{M-1}^b(n)] = [\mathbf{b}'(0) \ \dots \ \mathbf{b}'(n)]^T, \quad (2.105)$$

sendo  $b'_{i-1}(\ell) = e_{i-1}^b(\ell)$ . Os elementos  $\lambda^{(n-\ell)/2} b_{i-1}(\ell)$  do vetor coluna  $\mathbf{b}'(\ell)$ , para  $1 \leq i \leq M$  e  $0 \leq \ell \leq n$ , são erros de predição regressiva definidos para uma ordem  $i-1$ , onde a amostra estimada é obtida conforme equação (2.24) para um vetor de coeficientes de ordem  $i-1$  no tempo  $n$ , isto é

$$e_{i-1}^b(\ell) = b'_{i-1}(\ell) = u(\ell - M + 1) - \mathbf{u}_{M-1}^T(\ell) \mathbf{w}_{i-1}^b(n). \quad (2.106)$$

para  $0 \leq \ell \leq n$ . A partir da equação (2.104), obtém-se

$$[\mathbf{b}'(0) \ \dots \ \mathbf{b}'(n)] = \mathbf{K}_M^{-T}(n) [\mathbf{u}_M(0) \ \dots \ \mathbf{u}_M(n)] \mathbf{\Lambda}^{1/2}(n). \quad (2.107)$$

Para o instante  $\ell = n$  o erro  $b'_{i-1}(n)$  é o mesmo erro  $b_{i-1}(n)$ , conforme definido em (2.23), porém para ordem  $i-1$ . Sendo assim, considera-se a seguir  $\mathbf{b}'(n) = \mathbf{b}(n)$ . Da equação (2.107) cabe observar que

$$\mathbf{b}(n) = \mathbf{K}_M^{-T}(n) \mathbf{u}_M(n), \quad (2.108)$$

o que permite interpretar a matriz triangular inferior  $\mathbf{K}_M^{-T}(n)$  como uma pilha de filtros de erro da predição regressiva [38, 21], conforme figura 2.2. A partir dos resultados obtidos, é possível fazer as seguintes interpretações:

- ◊ os elementos não nulos da linha  $i$  da matriz  $\mathbf{K}_M^{-T}(n)$  são os coeficientes do filtro de erro de predição regressiva de ordem  $i-1$ ;
- ◊ a coluna  $i$  de  $\mathbf{G}_M(n)$ ,  $1 \leq i \leq M$ , é a seqüência de erros de predição regressiva correspondente ao filtro da  $i$ ésima linha de  $\mathbf{K}_M^{-T}(n)$ ;
- ◊ os elementos da matriz diagonal  $\mathbf{D}_M(n)$  correspondem a energia de erro de predição regressiva,

$$\mathbf{D}_M(n) = \text{diag}\{\xi_0^b(n), \xi_1^b(n), \dots, \xi_{M-1}^b(n)\}, \quad (2.109)$$

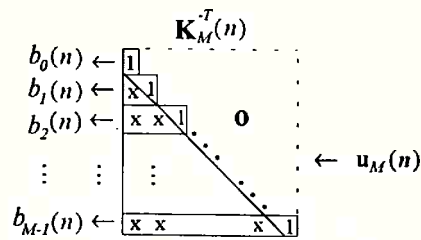


Figura 2.2: Conexão de uma pilha de filtros de erro da predição regressiva de ordem 0 a ordem  $M - 1$ .

sendo que a energia de erro da predição regressiva  $\xi_{i-1}^b(n)$ , está associada ao filtro de erro de predição da  $i$ ésima linha de  $\mathbf{K}_M^{-T}(n)$ . Desta forma, tem-se então,  $\xi_{i-1}^b(n) = \|\mathbf{e}_{i-1}^b(n)\|^2$ .

A seguir considera-se a aplicação do procedimento de ortogonalização de Gram-Schmidt aos problemas de estimação, predição regressiva e predição progressiva, onde os vetores a serem ortogonalizados são respectivamente  $\mathbf{d}(n)$ ,  $\mathbf{a}(n - M + 1)$  e  $\mathbf{a}(n)$ .

### 1) Estimação:

Aplicando o procedimento de ortogonalização de Gram-Schmidt à matriz

$$[\mathbf{A}_M(n) \mathbf{d}(n)],$$

resulta a matriz ortogonal

$$[\mathbf{G}_M(n) \mathbf{e}_M(n)],$$

sendo  $\mathbf{e}_M(n) = \mathbf{d}(n) - \hat{\mathbf{d}}(n)$ . O vetor de respostas desejadas estimado  $\hat{\mathbf{d}}(n)$ , pode ser obtido a partir dos vetores ortogonais  $\mathbf{e}_{i-1}^b(n)$ ,  $1 \leq i \leq M$ :

$$\hat{\mathbf{d}}(n) = \sum_{i=1}^M k_i^d(n) \mathbf{e}_{i-1}^b(n) = \mathbf{G}_M(n) \mathbf{k}_M^d(n), \quad (2.110)$$

onde

$$k_i^d(n) = \frac{\mathbf{e}_{i-1}^{bT}(n) \mathbf{d}(n)}{\xi_{i-1}^b(n)} \quad (2.111)$$

e

$$\mathbf{k}_M^d(n) = \begin{bmatrix} k_1^d(n) \\ \vdots \\ k_M^d(n) \end{bmatrix} = \mathbf{D}_M^{-1}(n) \mathbf{G}_M^T(n) \mathbf{d}(n). \quad (2.112)$$

Neste caso, o vetor  $\mathbf{e}_M(n)$ , que é ortogonal a  $\hat{\mathbf{d}}(n)$ , é o vetor dos erros de estimação *a posteriori* que minimiza a função custo definida em (2.1). O vetor de estimação ótimo é dado por

$$\hat{\mathbf{d}}(n) = \mathbf{G}_M(n)\mathbf{D}_M^{-1}(n)\mathbf{G}_M^T(n)\mathbf{d}(n). \quad (2.113)$$

Conseqüentemente, para o operador de projeção resulta a seguinte expressão

$$\mathbf{P}_M(n) = (\mathbf{G}_M(n)\mathbf{D}_M^{-1/2}(n))(\mathbf{D}_M^{-1/2}(n)\mathbf{G}_M^T(n)) = \sum_{i=1}^M \bar{\mathbf{e}}_{i-1}^b(n)\bar{\mathbf{e}}_{i-1}^{bT}(n), \quad (2.114)$$

sendo

$$\bar{\mathbf{e}}_{i-1}^b(n) = \frac{\mathbf{e}_{i-1}^b(n)}{\xi_{i-1}^{b/2}(n)}, \quad (2.115)$$

o vetor dos erros das predições regressivas *a posteriori* normalizados. A estimação do vetor  $\mathbf{d}(n)$ , obtida utilizando os erros de predição regressiva  $\mathbf{e}_{i-1}^b(n)$ ,  $1 \leq i \leq M$ , é conhecida como estimação do processo conjunto, além da estimação de  $\hat{\mathbf{d}}(n)$ , efetua-se implicitamente a estimação de  $\mathbf{a}(n - M + 1)$ . O vetor  $\mathbf{k}_M^d(n)$  representa o vetor de coeficientes do processo de estimação conjunta, também chamados de coeficientes de regressão [21]. Como

$$\hat{\mathbf{d}}(n) = \mathbf{A}_M(n)\mathbf{w}_M(n) = \mathbf{G}_M(n)\mathbf{k}_M^d(n), \quad (2.116)$$

o vetor de coeficientes ótimos  $\mathbf{w}_M(n)$  satisfaz

$$\mathbf{K}_M(n)\mathbf{w}_M(n) = \mathbf{k}_M^d(n), \quad (2.117)$$

pois conforme (2.104),  $\mathbf{A}_M(n) = \mathbf{G}_M(n)\mathbf{K}_M(n)$ . Usando (2.47) para ordem  $i-1$  com  $1 \leq i \leq M$ , o vetor  $\mathbf{k}_{M-1}^d(n)$ , definido em (2.112), pode ser ainda expresso como

$$\mathbf{k}_M^d(n) = \mathbf{D}_M^{-1}(n) \begin{bmatrix} \mathbf{e}_0^{bT}(n)\mathbf{e}_0(n) \\ \mathbf{e}_1^{bT}(n)\mathbf{e}_1(n) \\ \vdots \\ \mathbf{e}_{M-1}^{bT}(n)\mathbf{e}_{M-1}(n) \end{bmatrix}. \quad (2.118)$$

Para finalizar este item, cabe observar que com o procedimento de Gram-Schmidt é efetuada a triangularização de dados através de uma matriz  $\mathbf{G}_M(n)$  com colunas ortogonais, mas não ortonormais. Entretanto, como

$$\mathbf{G}_M^T(n)\mathbf{G}_M(n) = \mathbf{D}_M(n)$$

a matriz  $\mathbf{G}_M(n)\mathbf{D}_M^{-1/2}(n)$  tem colunas ortogonais. Sendo assim de (2.104) resulta a seguinte representação para a matriz de dados:

$$\mathbf{A}_M(n) = \left( \mathbf{G}_M(n)\mathbf{D}_M^{-1/2}(n) \right) \left( \mathbf{D}_M^{1/2}(n)\mathbf{K}_M(n) \right). \quad (2.119)$$

Conseqüentemente a matriz  $\mathbf{D}_M^{1/2}(n)\mathbf{K}_M(n)$  é uma das representações possíveis para a matriz  $\mathbf{L}_M(n)$ , apresentada em (2.10), e  $\mathbf{G}_M(n)\mathbf{D}_M^{-1/2}(n)$  é uma das representações possíveis para  $\mathbf{A}'_M(n)$ , apresentada em (2.11). Além disso, a comparação de (2.119) com (2.70) mostra que o procedimento de Gram-Schmidt pode ser utilizado para efetuar a decomposição QR. Neste caso,

$$\mathbf{Q}\mathbf{1}_M(n) = \mathbf{D}_M^{-1/2}(n)\mathbf{G}_M^T(n) \quad (2.120)$$

e

$$\mathbf{R}_M(n) = \mathbf{D}_M^{1/2}(n)\mathbf{K}_M(n). \quad (2.121)$$

## 2) Predição regressiva:

Aplicando o procedimento de ortogonalização de Gram-Schmidt à matriz

$$[\mathbf{A}_{M-1}(n) \mathbf{a}(n - M + 1)],$$

resulta a matriz ortogonal

$$[\mathbf{G}_{M-1}(n) \mathbf{e}_{M-1}^b(n)],$$

onde  $\mathbf{e}_{M-1}^b(n) = \mathbf{a}(n - M + 1) - \hat{\mathbf{a}}(n - M + 1)$ . O vetor  $\hat{\mathbf{a}}(n - M + 1)$  pode ser obtido a partir dos vetores ortogonais  $\mathbf{e}_{i-1}^b(n)$ ,  $1 \leq i \leq M - 1$ :

$$\hat{\mathbf{a}}(n - M + 1) = \sum_{i=1}^{M-1} k_i(n)\mathbf{e}_{i-1}^b(n) = \mathbf{G}_{M-1}(n)\mathbf{k}_{M-1}(n), \quad (2.122)$$

onde

$$k_i(n) = \frac{\mathbf{e}_{i-1}^{bT}(n)\mathbf{a}(n - M + 1)}{\xi_{i-1}^b(n)} \quad (2.123)$$

e

$$\mathbf{k}_{M-1}(n) = \begin{bmatrix} k_1(n) \\ \vdots \\ k_{M-1}(n) \end{bmatrix} = \mathbf{D}_{M-1}^{-1}(n)\mathbf{G}_{M-1}^T(n)\mathbf{a}(n - M + 1). \quad (2.124)$$

Os  $k_i(n)$  são os elementos da linha  $i$  e da coluna  $M$  da matriz  $\mathbf{K}_M(n)$ . Neste caso, o vetor  $\mathbf{e}_{M-1}^b(n)$ , que é ortogonal a  $\hat{\mathbf{a}}(n - M + 1)$ , é o vetor dos erros da

predição regressiva que minimiza a função custo definida em (2.27). O vetor de estimação ótimo é dado por

$$\hat{\mathbf{a}}(n - M + 1) = \mathbf{G}_{M-1}(n) \mathbf{D}_{M-1}^{-1}(n) \mathbf{G}_{M-1}^T(n) \mathbf{a}(n - M + 1). \quad (2.125)$$

Conseqüentemente, para o operador de projeção resulta a seguinte expressão

$$\mathbf{P}_{M-1}(n) = (\mathbf{G}_{M-1}(n) \mathbf{D}_{M-1}^{-1/2}(n)) (\mathbf{D}_{M-1}^{-1/2}(n) \mathbf{G}_{M-1}^T(n)) = \sum_{i=1}^{M-1} \bar{\mathbf{e}}_{i-1}^b(n) \bar{\mathbf{e}}_{i-1}^{bT}(n), \quad (2.126)$$

a qual corresponde a expressão (2.114) para uma ordem inferior. Como

$$\hat{\mathbf{a}}(n - M + 1) = \mathbf{A}_{M-1}(n) \mathbf{w}_{M-1}^b(n) = \mathbf{G}_{M-1}(n) \mathbf{k}_{M-1}(n), \quad (2.127)$$

o vetor de coeficientes ótimos  $\mathbf{w}_{M-1}^b(n)$  satisfaz

$$\mathbf{K}_{M-1}(n) \mathbf{w}_{M-1}^b(n) = \mathbf{k}_{M-1}(n). \quad (2.128)$$

### 3) Predição progressiva:

Aplicando o procedimento de ortogonalização de Gram-Schmidt à matriz

$$\begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} \mathbf{a}(n),$$

resulta a matriz ortogonal

$$\begin{bmatrix} \mathbf{0}^T \\ \mathbf{G}_{M-1}(n-1) \end{bmatrix} \mathbf{e}_{M-1}^f(n),$$

onde  $\mathbf{e}_{M-1}^f(n) = \mathbf{a}(n) - \hat{\mathbf{a}}(n)$ . O vetor  $\hat{\mathbf{a}}(n)$  pode ser obtido a partir dos vetores ortogonais  $\mathbf{e}_{i-1}^b(n-1)$ ,  $1 \leq i \leq M-1$ :

$$\hat{\mathbf{a}}(n) = \sum_{i=1}^{M-1} k_i^f(n) \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix} = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{G}_{M-1}(n-1) \end{bmatrix} \mathbf{k}_{M-1}^f(n), \quad (2.129)$$

onde

$$k_i^f(n) = \frac{\begin{bmatrix} 0 & \mathbf{e}_{i-1}^{bT}(n-1) \end{bmatrix} \mathbf{a}(n)}{\xi_{i-1}^b(n-1)} \quad (2.130)$$

e

$$\mathbf{k}_{M-1}^f(n) = \begin{bmatrix} k_1^f(n) \\ \vdots \\ k_{M-1}^f(n) \end{bmatrix} = \mathbf{D}_{M-1}^{-1}(n-1) \begin{bmatrix} \mathbf{0} & \mathbf{G}_{M-1}^T(n-1) \end{bmatrix} \mathbf{a}(n). \quad (2.131)$$

Neste caso, o vetor  $\mathbf{e}_{M-1}^f(n)$ , que é ortogonal a  $\hat{\mathbf{a}}(n)$ , é o vetor dos erros da predição progressiva que minimiza a função custo definida em (2.38). O vetor de estimação ótimo é dado por

$$\hat{\mathbf{a}}(n) = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{G}_{M-1}(n-1) \end{bmatrix} \mathbf{D}_{M-1}^{-1}(n-1) \begin{bmatrix} \mathbf{0} & \mathbf{G}_{M-1}^T(n-1) \end{bmatrix} \mathbf{a}(n). \quad (2.132)$$

Consequentemente, para o operador de projeção resulta a seguinte expressão

$$\begin{bmatrix} \mathbf{0} & \mathbf{0}^T \\ \mathbf{0} & \mathbf{P}_{M-1}(n-1) \end{bmatrix}$$

sendo

$$\mathbf{P}_{M-1}(n-1) = \sum_{i=1}^{M-1} \bar{\mathbf{e}}_{i-1}^b(n-1) \bar{\mathbf{e}}_{i-1}^{bT}(n-1). \quad (2.133)$$

Como

$$\hat{\mathbf{a}}(n) = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{A}_{M-1}(n-1) \end{bmatrix} \mathbf{w}_{M-1}^f(n) = \begin{bmatrix} \mathbf{0}^T \\ \mathbf{G}_{M-1}(n-1) \end{bmatrix} \mathbf{k}_{M-1}^f(n), \quad (2.134)$$

o vetor de coeficientes ótimos  $\mathbf{w}_{M-1}^f(n)$  satisfaz

$$\mathbf{K}_{M-1}(n-1) \mathbf{w}_{M-1}^f(n) = \mathbf{k}_{M-1}^f(n). \quad (2.135)$$

Usando (2.48), para ordem  $i-1$  com  $1 \leq i \leq M-1$ , o vetor  $\mathbf{k}_{M-1}^f(n)$  pode ser expresso como

$$\mathbf{k}_{M-1}^f(n) = \mathbf{D}_{M-1}^{-1}(n-1) \begin{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{e}_0^{bT}(n-1) \end{bmatrix} \mathbf{e}_0^f(n) \\ \begin{bmatrix} \mathbf{0} & \mathbf{e}_1^{bT}(n-1) \end{bmatrix} \mathbf{e}_1^f(n) \\ \vdots \\ \begin{bmatrix} \mathbf{0} & \mathbf{e}_{M-2}^{bT}(n-1) \end{bmatrix} \mathbf{e}_{M-2}^f(n) \end{bmatrix}. \quad (2.136)$$

Por fim, nas próximas seções são feitas interpretações dos vetores de estados e da matriz de sistema, os quais descrevem as operações de adaptação e filtragem dos algoritmos que utilizam triangularização ortogonal.

## 2.5 Considerações sobre os vetores de estados

Utilizando os resultados apresentados até agora, verifica-se ser possível relacionar os vetores de estados dos sistemas variantes no tempo, que descrevem as operações de adaptação e filtragem do método RLS convencional e dos métodos que utilizam a decomposição QR. Como observado em (2.121), as matrizes  $\mathbf{R}_M(n)$  e  $\mathbf{D}_M^{1/2}(n)\mathbf{K}_M(n)$  são iguais. Este fato pode também ser justificado observando que ambas são fator de Cholesky da matriz de autocorrelação  $\mathbf{A}_M^T(n)\mathbf{A}_M(n)$ . Isto resulta das equações (2.70) e (2.104) e das propriedades das matrizes envolvidas. Para estabelecer as interpretações desejadas é necessário estabelecer convenientemente a estrutura da matriz  $\mathbf{Q}\mathbf{1}_M(n)$ . Da equação (2.120), tem-se

$$\mathbf{Q}\mathbf{1}_M^T(n) = \begin{bmatrix} \mathbf{e}_0^b(n) & \dots & \mathbf{e}_{M-1}^b(n) \end{bmatrix} \mathbf{D}_M^{-1/2}(n) = \begin{bmatrix} \bar{\mathbf{e}}_0^b(n) & \dots & \bar{\mathbf{e}}_{M-1}^b(n) \end{bmatrix}. \quad (2.137)$$

Ou seja, as colunas de  $\mathbf{Q}\mathbf{1}_M^T(n)$  são os vetores dos erros da predição regressiva *a posteriori* normalizados, conforme equação (2.115). É interessante observar que a matriz  $\mathbf{Q}\mathbf{1}_M^T(n)$  pode também ser representada em termos dos vetores  $\mathbf{b}'(\ell)$  para  $0 \leq \ell \leq n$ , isto é

$$\mathbf{Q}\mathbf{1}_M^T(n) = \begin{bmatrix} \mathbf{b}'(0) & \mathbf{b}'(1) & \dots & \mathbf{b}'(n-1) & \mathbf{b}(n) \end{bmatrix}^T \mathbf{D}_M^{-1/2}(n). \quad (2.138)$$

Apresenta-se a seguir, interpretações para os vetores de estados, os quais descrevem as operações de adaptação do algoritmo QR-RLS nos casos de estimação e das predições progressiva e regressiva.

- a) Usando  $\mathbf{R}_M(n) = \mathbf{D}_M^{1/2}(n)\mathbf{K}_M(n)$  em (2.74) e reconhecendo (2.117), conclui-se que os vetores  $\mathbf{k}^d(n)$  e  $\mathbf{dq}\mathbf{1}(n)$ , resultantes dos procedimentos de ortogonalização de Gram-Schmidt e de Givens da matriz  $\mathbf{A}_M(n)$ , estão relacionados pela matriz diagonal  $\mathbf{D}_M^{1/2}(n)$ :

$$\mathbf{dq}\mathbf{1}(n) = \mathbf{D}_M^{1/2}(n)\mathbf{k}^d(n). \quad (2.139)$$

Consequentemente  $\mathbf{dq}\mathbf{1}(n)$  tem uma interpretação equivalente ao vetor dos coeficientes de regressão de estimação conjunta apresentado na equação (2.112). Como  $\mathbf{dq}\mathbf{1}(n) = \mathbf{Q}\mathbf{1}_M(n)\mathbf{d}(n)$ , os elementos de  $\mathbf{dq}\mathbf{1}(n)$  são os coeficientes das projeções do vetor de respostas desejadas sobre os vetores de erro da predição regressiva *a posteriori* normalizados.



- b) Considerando que as linhas da matriz unitária  $\mathbf{Q}\mathbf{1}_{M-1}(n)$  representam os vetores  $\bar{\mathbf{e}}_{i-1}^{bT}(n)$  para  $1 \leq i \leq M$ , então o vetor  $\mathbf{dq}\mathbf{1}^b(n)$ , definido em (2.89), pode ser expresso como

$$\mathbf{dq}\mathbf{1}^b(n) = \mathbf{Q}\mathbf{1}_{M-1}(n)\mathbf{a}(n - M + 1)$$

$$\mathbf{dq}\mathbf{1}^b(n) = \begin{bmatrix} \bar{\mathbf{e}}_0^{bT}(n) \\ \bar{\mathbf{e}}_1^{bT}(n) \\ \vdots \\ \bar{\mathbf{e}}_{M-2}^{bT}(n) \end{bmatrix} \mathbf{a}(n - M + 1). \quad (2.140)$$

Os elementos de  $\mathbf{dq}\mathbf{1}^b(n)$  são expressos como

$$dq\mathbf{1}_i^b(n) = \frac{\mathbf{e}_{i-1}^{bT}(n)\mathbf{a}(n - M + 1)}{\xi_{i-1}^{b/2}(n)} = \bar{\mathbf{e}}_{i-1}^{bT}(n)\mathbf{a}(n - M + 1)$$

com  $1 \leq i \leq M - 1$ . Comparando com a equação (2.102), tem-se

$$dq\mathbf{1}_i^b(n) = k_{i,M}(n)\xi_{i-1}^{b/2}(n),$$

isto é, os  $dq\mathbf{1}_i^b(n)$  são os elementos da linha  $i$  e da coluna  $M$  da matriz  $\mathbf{K}_M(n)$  ponderados por  $\xi_{i-1}^{b/2}(n)$ . Comparando com (2.89) e (2.124), tem-se

$$\mathbf{dq}\mathbf{1}^b(n) = \mathbf{D}_{M-1}^{1/2}(n)\mathbf{k}_{M-1}(n). \quad (2.141)$$

A interpretação dos elementos do vetor  $\mathbf{dq}\mathbf{1}^b(n)$  é completamente análoga aos dos elementos de  $\mathbf{dq}\mathbf{1}(n)$ : são coeficientes das projeções do vetor de resposta desejada  $\mathbf{a}(n - M + 1)$  sobre os vetores de erro de predição regressiva normalizados.

- c) Considerando que as linhas da matriz unitária  $\mathbf{Q}\mathbf{1}_{M-1}(n - 1)$  representam os vetores  $\bar{\mathbf{e}}_{i-1}^{bT}(n - 1)$  para  $1 \leq i \leq M$ , então o vetor  $\mathbf{dq}\mathbf{1}^f(n)$  definido em (2.95) pode ser expresso como

$$\mathbf{dq}\mathbf{1}^f(n) = \begin{bmatrix} \mathbf{0} & \mathbf{Q}\mathbf{1}_{M-1}(n - 1) \end{bmatrix} \mathbf{a}(n) =$$

$$= \mathbf{D}_{M-1}^{-1/2}(n - 1) \begin{bmatrix} 0 & \mathbf{e}_0^{bT}(n - 1) \\ 0 & \mathbf{e}_1^{bT}(n - 1) \\ \vdots & \vdots \\ 0 & \mathbf{e}_{M-2}^{bT}(n - 1) \end{bmatrix} \mathbf{a}(n). \quad (2.142)$$

Com base em (2.48), para ordem  $i - 1$  com  $2 \leq i \leq M$ , o vetor  $\mathbf{dq1}^f(n)$  pode ser reescrito como

$$\mathbf{dq1}^f(n) = \mathbf{D}_{M-1}^{-1/2}(n-1) \begin{bmatrix} \left[ 0 \quad \mathbf{e}_0^{bT}(n-1) \right] \mathbf{e}_0^f(n) \\ \left[ 0 \quad \mathbf{e}_1^{bT}(n-1) \right] \mathbf{e}_1^f(n) \\ \vdots \\ \left[ 0 \quad \mathbf{e}_{M-1}^{bT}(n-1) \right] \mathbf{e}_{M-1}^f(n) \end{bmatrix}. \quad (2.143)$$

Comparando com (2.136), tem-se

$$\mathbf{dq1}^f(n) = \mathbf{D}_{M-1}^{1/2}(n-1) \mathbf{k}_{M-1}^f(n) \quad (2.144)$$

Os elementos de  $\mathbf{dq1}^f(n)$  representam então, as correlações cruzadas entre os vetores dos erros das predições regressivas  $\mathbf{e}_{i-1}^{bT}(n-1)$  com os vetores dos erros das predições progressivas  $\mathbf{e}_{i-1}^f(n)$  ponderados por  $\xi_{i-1}^{-b/2}(n-1)$ , com  $1 \leq i \leq M-1$ .

Os vetores que descrevem as operações de adaptação nos algoritmos RLS convencional, QR-RLS baseado em rotações de Givens e os baseados no procedimento de ortogonalização de Gram-Schmidt, respectivamente  $\mathbf{w}_M(n)$ ,  $\mathbf{dq1}(n)$  e  $\mathbf{k}^d(n)$ , são interpretados como vetores de estados em sistemas de coordenadas distintos. Das equações (2.74), (2.117) e (2.139) é possível interpretar as matrizes  $\mathbf{R}_M(n)$ ,  $\mathbf{K}_M(n)$  e  $\mathbf{D}_M^{1/2}(n)$  como transformações similares. A figura 2.3 ilustra estas transformações. Do

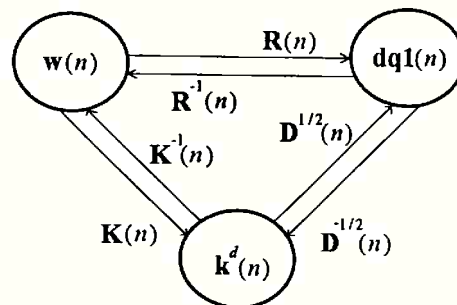


Figura 2.3: Relações entre os vetores de estados do algoritmos RLS convencional e dos algoritmos que utilizam triangularização ortogonal.

mesmo modo, das equações (2.91), (2.128) e (2.141) é possível interpretar as matrizes  $\mathbf{R}_{M-1}(n)$ ,  $\mathbf{K}_{M-1}(n)$  e  $\mathbf{D}_{M-1}^{1/2}(n)$  como transformações similares sobre os vetores de

estado  $\mathbf{w}_M^b(n)$ ,  $\mathbf{dq}\mathbf{1}^b(n)$  e  $\mathbf{k}(n)$ , e das equações (2.97), (2.135) e (2.144) é possível interpretar as matrizes  $\mathbf{R}_{M-1}(n-1)$ ,  $\mathbf{K}_{M-1}(n-1)$  e  $\mathbf{D}_{M-1}^{1/2}(n-1)$  como transformações similares sobre os vetores de estado  $\mathbf{w}_M^f(n)$ ,  $\mathbf{dq}\mathbf{1}^f(n)$  e  $\mathbf{k}^f(n)$ .

## 2.6 Considerações sobre a matriz de sistema

A equação (2.87) mostra que as operações de adaptação e de filtragem dos algoritmos QR-RLS podem ser descritas pelo seguinte sistema variante no tempo:

$$\begin{bmatrix} \mathbf{dq}\mathbf{1}(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^g(n) \begin{bmatrix} \lambda^{1/2} \mathbf{dq}\mathbf{1}(n-1) \\ d(n) \end{bmatrix}. \quad (2.145)$$

Neste caso a matriz unitária  $\mathbf{S}_M^g(n)$  é a matriz de sistema. O sistema (2.145) pode também ser obtido através de uma transformação similar no sistema (2.58). Para isto, substitui-se em (2.58) a equação (2.74) reescrita como  $\mathbf{w}_M(n) = \mathbf{R}_M^{-1}(n) \mathbf{dq}\mathbf{1}(n)$  e (A.17) reescrita como  $\mathbf{I} - \mathbf{g}(n) \mathbf{u}_M^T(n) = \lambda \Phi_M^{-1}(n) \Phi_M(n-1)$  e considera-se  $\Phi_M(n)$  com em (2.82), isto é  $\Phi_M(n) = \mathbf{R}_M^T(n) \mathbf{R}_M(n)$ . Após algumas manipulações algébricas, o sistema resultante é (2.145) sendo que agora é possível explicitar os elementos da matriz  $\mathbf{S}_M^g(n)$  como

$$\mathbf{S}_M^g(n) = \gamma_M^{1/2}(n) \begin{bmatrix} \lambda^{1/2} \gamma_M^{-1/2}(n) \mathbf{R}_M^{-T}(n) \mathbf{R}_M^T(n-1) & \gamma_M^{-1/2}(n) \mathbf{R}_M^{-T}(n) \mathbf{u}_M(n) \\ -\lambda^{-1/2} \mathbf{u}_M^T(n) \mathbf{R}_M^{-1}(n-1) & 1 \end{bmatrix}. \quad (2.146)$$

Para facilitar a manipulação dos blocos da matriz  $\mathbf{S}_M^g(n)$ , adota-se intencionalmente a seguinte notação

$$\Gamma(n) = \lambda^{1/2} \mathbf{R}_M^{-T}(n) \mathbf{R}_M^T(n-1), \quad (2.147)$$

$$\bar{\mathbf{b}}_M(n) = \mathbf{R}_M^{-T}(n) \mathbf{u}_M(n), \quad (2.148)$$

$$\bar{\boldsymbol{\psi}}_M(n) = \lambda^{-1/2} \mathbf{R}_M^{-T}(n-1) \mathbf{u}_M(n). \quad (2.149)$$

Substituindo  $\mathbf{R}_M(n) = \mathbf{D}_M^{1/2}(n) \mathbf{K}_M(n)$  em (2.148) e (2.149), obtém-se

$$\bar{\mathbf{b}}_M(n) = \mathbf{D}_M^{-1/2}(n) \underbrace{\mathbf{K}_M^{-T}(n) \mathbf{u}_M(n)}_{\mathbf{b}_M(n)} \quad (2.150)$$

e

$$\bar{\boldsymbol{\psi}}_M(n) = \lambda^{-1/2} \mathbf{D}_M^{-1/2}(n-1) \underbrace{\mathbf{K}_M^{-T}(n-1) \mathbf{u}_M(n)}_{\boldsymbol{\psi}_M(n)}. \quad (2.151)$$

A partir das observações sobre as matrizes  $\mathbf{K}_M^{-T}(n)$ ,  $\mathbf{D}_M$  e  $\mathbf{G}_M(n)$ , efetuadas na seção 2.4, é possível interpretar adequadamente as equações (2.150) e (2.151) da seguinte forma:

- ◊ Na equação (2.150) o vetor  $\mathbf{b}_M(n)$  representa as saídas no tempo  $n$  dos filtros definidos por  $\mathbf{K}_M^{-T}(n)$ , isto é, o vetor dos erros de predição regressivos *a posteriori* no tempo  $n$ . Consequentemente os elementos do vetor

$$\bar{\mathbf{b}}_M(n) = [\bar{b}_0(n), \bar{b}_1(n), \dots, \bar{b}_{M-1}(n)]^T$$

são os erros de predição regressivos *a posteriori* normalizados

$$\bar{b}_{i-1}(n) = \frac{b_{i-1}(n)}{\xi_{i-1}^{b/2}(n)}, \quad (2.152)$$

para  $1 \leq i \leq M$ .

- ◊ Da equação (2.151), conclui-se que os elementos do vetor  $\boldsymbol{\psi}_M(n)$  são saídas no tempo  $n$  dos filtros de erro de predição regressiva, definidos por  $\mathbf{K}_M^{-T}(n-1)$ , isto é, dos filtros de erro de predição regressiva com os coeficientes no tempo  $n-1$ . Portanto,  $\boldsymbol{\psi}_M(n)$  é o vetor dos erros de predição regressiva *a priori* no tempo  $n$ . Os elementos de

$$\bar{\boldsymbol{\psi}}_M(n) = [\bar{\psi}_0(n), \bar{\psi}_1(n), \dots, \bar{\psi}_{M-1}(n)]^T$$

são os erros de predição regressiva *a priori* normalizados pela energia do erro de predição regressiva *a posteriori* no tempo  $n-1$  e ponderados por  $\lambda^{1/2}$

$$\bar{\psi}_{i-1}(n) = \frac{\psi_{i-1}(n)}{\lambda^{1/2} \xi_{i-1}^{b/2}(n-1)}, \quad (2.153)$$

para  $1 \leq i \leq M$ .

Como a matriz  $\mathbf{S}_M^g(n)$  é o resultado do produto de uma seqüência de rotações de Givens, tem-se  $\mathbf{S}_M^{gT}(n)\mathbf{S}_M^g(n) = \mathbf{I}$  e  $\mathbf{S}_M^g(n)\mathbf{S}_M^{gT}(n) = \mathbf{I}$ . Destas igualdades resultam as seguintes propriedades:

$$\|\bar{\boldsymbol{\psi}}_M(n)\|^2 = \gamma_M^{-1}(n) - 1, \quad (2.154)$$

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \gamma_M(n), \quad (2.155)$$

$$\bar{\boldsymbol{\psi}}_M(n) = [\gamma_M^{-1}(n)\boldsymbol{\Gamma}^T(n)]\bar{\mathbf{b}}_M(n), \quad (2.156)$$

$$\bar{\mathbf{b}}_M(n) = \mathbf{\Gamma}(n) \bar{\boldsymbol{\psi}}_M(n), \quad (2.157)$$

$$\bar{\boldsymbol{\psi}}_M(n) \bar{\boldsymbol{\psi}}_M^T(n) = \gamma_M^{-1}(n) [\mathbf{I} - \mathbf{\Gamma}^T(n) \mathbf{\Gamma}(n)], \quad (2.158)$$

e

$$\bar{\mathbf{b}}_M(n) \bar{\mathbf{b}}_M^T(n) = \mathbf{I} - \mathbf{\Gamma}(n) \mathbf{\Gamma}^T(n). \quad (2.159)$$

As equações (2.154) e (2.155) mostram-se úteis tanto para o desenvolvimento de novos algoritmos QR-RLS (que serão tratados no capítulo 3), como para a análise de propriedades de estabilidade (que serão tratadas no capítulo 4). As equações (2.156) e (2.157) mostram que  $\bar{\boldsymbol{\psi}}_M(n)$  pode ser obtido a partir de  $\bar{\mathbf{b}}_M(n)$  e vice-versa. Assim, é possível interpretar a matriz  $\mathbf{\Gamma}(n)$  como uma matriz de conversão entre o vetor do erro da predição regressiva *a priori* e o vetor do erro da predição regressiva *a posteriori*. Uma outra interpretação para  $\mathbf{\Gamma}(n)$  pode ser obtida considerando equação (2.147) reescrita como

$$\mathbf{R}_M^{-T}(n) = \lambda^{-1/2} \mathbf{\Gamma}(n) \mathbf{R}_M^{-T}(n-1),$$

a qual pode ser interpretada como uma matriz que atualiza os coeficientes do banco de filtros de predição regressiva.

Complementando as relações entre as várias formas de resolver o problema dos mínimos quadrados, considere-se as expressões que definem o vetor do ganho de Kalman  $\mathbf{g}_M(n)$ , equação (2.55). Representando nesta equação a matriz de autocorrelação em termos da matriz triangular superior  $\mathbf{R}_M(n)$ , isto é

$$\mathbf{g}_M(n) = \mathbf{R}_M^{-1}(n) \mathbf{R}_M^{-T}(n) \mathbf{u}_M(n) = \lambda^{-1} \gamma_M(n) \mathbf{R}_M^{-1}(n-1) \mathbf{R}_M^{-T}(n-1) \mathbf{u}_M(n)$$

e reconhecendo os vetores  $\bar{\mathbf{b}}_M(n)$  e  $\bar{\boldsymbol{\psi}}_M(n)$  conforme definidos em (2.148) e (2.149), resulta em

$$\mathbf{g}_M(n) = \mathbf{R}_M^{-1}(n) \bar{\mathbf{b}}_M(n) = \lambda^{-1/2} \gamma_M(n) \mathbf{R}_M^{-1}(n-1) \bar{\boldsymbol{\psi}}_M(n).$$

Desta última expressão, é possível concluir que a matriz  $\mathbf{R}_M^{-1}(n)$  transforma o vetor dos erros da predição regressiva  $\bar{\mathbf{b}}_M(n)$  no vetor de ganho de Kalman. Da mesma forma, a matriz  $\lambda^{-1/2} \gamma_M(n) \mathbf{R}_M^{-1}(n-1)$  transforma o vetor dos erros da predição regressiva  $\bar{\boldsymbol{\psi}}_M(n)$  no vetor de ganho de Kalman. A transformação similar  $\mathbf{R}_M(n)$  transforma o ganho de Kalman no vetor dos erros da predição regressiva  $\bar{\mathbf{b}}_M(n)$ , o qual no domínio transformado, faz então o papel do ganho de Kalman.

## Equações de atualização dos coeficientes de regressão $\mathbf{k}^d(n)$

Substituindo (2.139) em (2.145), após algumas manipulações algébricas, obtém-se como resultado a seguinte equação recorrente no tempo para os coeficientes de regressão

$$\begin{bmatrix} \mathbf{k}^d(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^k(n) \begin{bmatrix} \lambda^{1/2} \mathbf{k}^d(n-1) \\ d(n) \end{bmatrix}, \quad (2.160)$$

onde

$$\mathbf{S}_M^k(n) = \begin{bmatrix} \mathbf{D}_M^{-1/2}(n) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{S}_M^g(n) \begin{bmatrix} \mathbf{D}_M^{1/2}(n-1) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix}, \quad (2.161)$$

sendo  $\mathbf{S}_M^k(n)$  a matriz de sistema no procedimento de ortogonalização de Gram-Schmidt.

Existem várias versões de algoritmos RLS que utilizam o procedimento de ortogonalização de Gram-Schmidt, como por exemplo [12, 14, 13, 21]. Estes algoritmos não fazem restrições quanto a natureza seqüencial dos dados de entrada, apresentando uma complexidade computacional elevada. Cabe observar entretanto, que apesar dos elementos da matriz de sistema  $\mathbf{S}_M^k(n)$  não representarem mais os erros de predição regressiva, continuam a representar os erros *a posteriori* e *a priori* de algum problema de estimação. Quando os dados de entrada são seqüenciais, as interpretações da matriz  $\mathbf{S}_M^k(n)$  permanecem, sendo possível então, obter os filtros em treliça [14, 21].

## 2.7 Comentários

Neste capítulo, fez-se uma equivalência entre as operações de adaptação e filtragem do algoritmo RLS convencional e o algoritmo RLS que usa decomposição QR baseada em rotações de Givens em termos de representação no espaço de estados. Tal equivalência pode ser resumida conforme a tabela 2.2. Nesta tabela,  $A$  representa a matriz de transição de estados (realimentação);  $B$  é a matriz de entrada, que é o vetor de ganho de Kalman representado por  $\mathbf{g}_M(n)$  ou então, por  $\bar{\mathbf{b}}_M(n)$  no domínio transformado;  $C$  é a matriz de observação representada por  $-\gamma_M^{-1/2}(n)\mathbf{u}_M^T(n)$  ou por  $-\gamma_M^{-1/2}(n)\bar{\boldsymbol{\psi}}_M^T(n)$  no domínio transformado e  $D$  é a matriz de transmissão direta, representada por  $\gamma_M^{1/2}(n)$  nos dois domínios.

Tabela 2.2

Representação no espaço de estados		
Algoritmo	RLS convencional	QR-RLS
Estados	$\mathbf{w}_M(n)$	$\mathbf{dq1}(n) = \mathbf{R}_M(n)\mathbf{w}_M(n)$
A	$\mathbf{I} - \mathbf{g}_M(n)\mathbf{u}_M^T(n) =$ $\lambda^{-1}\Phi_M^{-1}(n)\Phi_M(n-1)$	$\Gamma(n) = \lambda^{1/2}\mathbf{R}_M^{-T}(n)\mathbf{R}_M^T(n-1)$
B	$\mathbf{g}_M(n)$	$\bar{\mathbf{b}}_M(n) = \mathbf{R}_M(n)\mathbf{g}_M(n)$
C	$-\gamma_M^{1/2}(n)\mathbf{u}_M^T(n)$	$-\gamma_M^{1/2}(n)\bar{\psi}_M^T(n) =$ $-\gamma_M^{1/2}(n)\mathbf{u}_M^T(n)\mathbf{R}_M^{-1}(n-1)$
D	$\gamma_M^{1/2}(n)$	$\gamma_M^{1/2}(n)$

Os elementos da matriz de sistema  $\mathbf{S}_M^c(n)$  que descrevem o algoritmo RLS convencional tem uma interpretação bem conhecida em termos da inversa da matriz de autocorrelação, do vetor de ganho de Kalman e do vetor de entrada. A interpretação dos elementos da matriz de sistema  $\mathbf{S}_M^g(n)$ , que descrevem as operações de adaptação e filtragem dos algoritmos QR-RLS em termos de  $\bar{\mathbf{b}}_M(n)$ ,  $\bar{\psi}_M(n)$ ,  $\Gamma(n)$  e  $\gamma_M^{1/2}(n)$ , pretende ser uma contribuição original deste trabalho.

A interpretação das variáveis envolvidas nos métodos de triangularização ortogonal da matriz de dados para resolver o problema dos mínimos quadrados de ordem  $M$ , confirma que as soluções dos problemas de ordem 0 a ordem  $M - 1$  são obtidas como sub-produto. Sendo assim, chega-se a conclusão bem conhecida que é possível resolver o problema dos mínimos quadrados de forma recorrente na ordem, o que pode ser feito tanto em função dos erros *a priori*, como dos erros *a posteriori* e dos erros normalizados em ângulo.

Os algoritmos RLS convencional e o QR-RLS apresentam uma complexidade computacional de  $M^2$ . As versões rápidas utilizam as predições progressivas e regressivas para obter o erro de estimação de forma conveniente. Apresenta-se no capítulo seguinte as equações de atualização no tempo e na ordem comuns às várias versões dos algoritmos rápidos existentes recorrentes na ordem.

## Capítulo 3

### Sobre algoritmos recorrentes no tempo e na ordem

As diferentes versões de algoritmos RLS rápidos recorrentes na ordem existentes na literatura são decorrentes dos métodos de triangularização ortogonal (e. g. [7, 14, 19, 33]). Tais versões se caracterizam, a primeira vista, não só por utilizar diferentes variáveis mas também diferentes formas de atualizar estas variáveis tanto no tempo como na ordem.

Neste capítulo são apresentadas várias equações que podem ser usadas para resolver o problema dos mínimos quadrados de forma recorrente na ordem e no tempo e também como estas várias equações estão relacionadas entre si. Nas seções 3.1 e 3.2 são apresentadas respectivamente as equações de atualização na ordem e no tempo do operador de projeção, utilizadas para obter as expressões de atualização na ordem e no tempo de diversas variáveis envolvidas na resolução do problemas dos mínimos quadrados. Dessa maneira, as formas para se atualizar diferentes variáveis de interesse podem ser entendidas como consequência da atualização no tempo e na ordem do operador de projeção. Na seção 3.3 são apresentadas as várias representações para a estrutura em treliça e como os erros de predição destas representações estão relacionados. Nesta seção é introduzida uma nova estrutura em treliça, obtida a partir de uma normalização dos erros de predição *a priori*. Esta normalização é sugerida pelas interpretações dos elementos da matriz de sistema efetuada no capítulo 2. A atualização no tempo para os coeficientes das estruturas em treliças são apresentadas na seção 3.4. Com as equações de atualização no tempo dos coeficientes das estruturas em treliça (vetores de estado), complementam-se na seção 3.5, as interpretações para os elementos da matriz de transição de estados dos algoritmos QR-RLS.

Dado um vetor de dois elementos, é possível calcular a sua norma e o seu ângulo através de uma redução de Givens ('Givens annihilation'). Este fato é usado



na seção 3.6 para calcular as energias de predição e o fator de conversão através de rotações. Os ângulos obtidos nestes cálculos são usados nas estruturas em treliça normalizadas e nas estruturas de McWhirter que aparecem nos algoritmos QR-RLS. Na seção 3.7 apresenta-se três algoritmos recorrentes no tempo e na ordem [14, 38, 33]. Finalmente, na seção 3.8 apresenta-se o novo algoritmo QR-LSL, que utiliza a estrutura em treliça introduzida na seção 3.3.

### 3.1 Atualizações na ordem

São apresentadas a seguir as equações de atualização na ordem do operador de projeção e suas conseqüências para as demais variáveis envolvidas na resolução do problema dos mínimos quadrados.

#### Atualização na ordem do operador de projeção

Incluindo uma coluna na matriz de dados aumenta-se de uma dimensão o subespaço associado, resultando em um novo operador de projeção. No caso de dados seqüenciais, o operador de projeção  $\mathbf{P}_M(n)$  definido em (2.18), pode ser atualizado na ordem através dos vetores dos erros das predições regressivas ou através dos vetores dos erros das predições progressivas, como demonstrado a seguir.

- a) Considerando a matriz de dados, conforme (A.7), e a inversa da matriz de autocorrelação, conforme (A.24), o operador de projeção (2.18) pode ser expresso como

$$\mathbf{P}_M(n) = \mathbf{P}_{M-1}(n) + \bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n), \quad (3.1)$$

onde

$$\mathbf{P}_{M-1}(n) = \mathbf{A}_{M-1}(n) \Phi_{M-1}^{-1}(n) \mathbf{A}_{M-1}^T(n)$$

e

$$\mathbf{e}_{M-1}^{bT}(n) = \begin{bmatrix} \mathbf{w}_{M-1}^{bT}(n) & -1 \end{bmatrix} \begin{bmatrix} \mathbf{A}_{M-1}^T(n) \\ \mathbf{a}^T(n - M + 1) \end{bmatrix} \quad (3.2)$$

é o vetor do erro da predição regressiva, conforme definido em (2.25), sendo  $\bar{\mathbf{e}}_{M-1}^b(n)$  o vetor do erro da predição regressiva normalizado de forma a assegu-

rar  $\|\bar{\mathbf{e}}_{M-1}^b(n)\|^2 = 1$ , isto é

$$\bar{\mathbf{e}}_{M-1}^b(n) = \frac{\mathbf{e}_{M-1}^b(n)}{\xi_{M-1}^{b/2}(n)}. \quad (3.3)$$

A equação (3.1) mostra que a atualização na ordem do operador de projeção pode ser feita a partir do conjunto de vetores ortonormais  $\bar{\mathbf{e}}_{i-1}^b(n)$ , para  $1 \leq i \leq M$ . Este resultado pode ser obtido também diretamente de (2.114), como uma consequência da aplicação do procedimento de ortogonalização de Gram-Schmidt nas colunas da matriz de dados.

- b) Considerando a matriz de dados conforme (A.8) e a inversa da matriz de autocorrelação conforme (A.26), o operador de projeção (2.18) pode ser expresso como

$$\mathbf{P}_M(n) = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{P}_{M-1}(n-1) \end{bmatrix} + \bar{\mathbf{e}}_{M-1}^f(n) \bar{\mathbf{e}}_{M-1}^{fT}(n) \quad (3.4)$$

onde

$$\mathbf{P}_{M-1}(n-1) = \mathbf{A}_{M-1}(n-1) \Phi_{M-1}^{-1}(n-1) \mathbf{A}_{M-1}^T(n-1)$$

e

$$\mathbf{e}_{M-1}^{fT}(n) = \begin{bmatrix} 1 & -\mathbf{w}_{M-1}^{fT}(n) \end{bmatrix} \begin{bmatrix} \mathbf{a}^T(n) \\ \mathbf{0} \quad \mathbf{A}_{M-1}^T(n-1) \end{bmatrix}$$

é o vetor do erro da predição progressiva conforme definido em (2.38), sendo

$$\bar{\mathbf{e}}_{M-1}^f(n) = \frac{\mathbf{e}_{M-1}^f(n)}{\xi_{M-1}^{f/2}(n)}, \quad (3.5)$$

o vetor do erro da predição progressiva normalizado de forma a assegurar  $\|\bar{\mathbf{e}}_{M-1}^f(n)\|^2 = 1$ . A equação (3.4) mostra que a atualização na ordem do operador de projeção pode também ser feita através dos vetores dos erros da predição progressiva.

As equações (3.1) e (3.4) serão usadas para obter os erros de estimação e os das predições progressiva e regressiva de forma recorrente na ordem.

### Atualização na ordem do erro de estimação e da estimativa da resposta desejada

O vetor do erro de estimação  $\mathbf{e}_M(n)$  e a estimativa da resposta desejada  $\hat{\mathbf{d}}(n)$  podem ser obtidos de forma recorrente na ordem a partir do conjunto de vetores ortonormais  $\bar{\mathbf{e}}_{i-1}^b(n)$  com  $1 \leq i \leq M$ , como demonstrado a seguir.

#### a) Atualização na ordem do erro de estimação:

Considerando (3.1) o vetor do erro de estimação (2.16) pode ser reescrito como

$$\mathbf{e}_M(n) = \mathbf{e}_{M-1}(n) - \bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n) \mathbf{d}(n). \quad (3.6)$$

O termo  $\bar{\mathbf{e}}_{M-1}^{bT}(n) \mathbf{d}(n)$  representa o coeficiente de projeção do vetor de respostas desejadas  $\mathbf{d}(n)$  no vetor do erro da predição regressiva  $\bar{\mathbf{e}}_{M-1}^b(n)$ . Desta forma, a equação (3.6) mostra que o vetor do erro da estimação  $\mathbf{e}_M(n)$  é atualizado na ordem subtraindo-se de  $\mathbf{e}_{M-1}(n)$  a projeção do vetor da resposta desejada  $\mathbf{d}(n)$  na direção do erro de predição regressivo  $\bar{\mathbf{e}}_{M-1}^b(n)$ . Usando (2.47), a equação (3.6) pode ser rescrita como

$$\mathbf{e}_M(n) = \left( \mathbf{I} - \bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n) \right) \mathbf{e}_{M-1}(n). \quad (3.7)$$

A matriz  $\bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n)$  atua como operador de projeção ortogonal na atualização na ordem de  $\mathbf{e}_M(n)$ .

#### b) Atualização na ordem do vetor estimado:

Considerando (3.1), o vetor de estimação da resposta desejada (2.19) pode ser reescrito como

$$\hat{\mathbf{d}}_M(n) = \hat{\mathbf{d}}_{M-1}(n) + \bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n) \mathbf{d}(n), \quad (3.8)$$

sendo  $\hat{\mathbf{d}}_{M-1}(n) = \mathbf{P}_{M-1}(n) \mathbf{d}(n)$ . Usando novamente (2.47) resulta

$$\hat{\mathbf{d}}_M(n) = \hat{\mathbf{d}}_{M-1}(n) + \bar{\mathbf{e}}_{M-1}^b(n) \bar{\mathbf{e}}_{M-1}^{bT}(n) \mathbf{e}_{M-1}(n). \quad (3.9)$$

A estimativa ótima do vetor do sinal desejado é atualizada na ordem adicionando à estimativa de ordem  $M-1$  a projeção do vetor do erro de estimação  $\mathbf{e}_{M-1}(n)$  na direção do erro de predição regressivo  $\bar{\mathbf{e}}_{M-1}^b(n)$ . Cabe observar que é exatamente o mesmo fator subtraído na atualização do erro de estimação.

As expressões (3.7) e (3.9) são úteis em aplicações onde desejamos expressar a resposta desejada  $\mathbf{d}(n)$  como uma combinação linear de ordem finita de uma seqüência de observação  $\{ u(0) \ u(1) \ \dots \ u(n) \}$ . Estas equações são válidas para qualquer ordem  $i$ , com  $1 \leq i \leq M$ , podendo ser reescritas como

$$\mathbf{e}_i(n) = \mathbf{e}_{i-1}(n) - \mathbf{e}_{i-1}^b(n)k_i^d(n) \quad (3.10)$$

e

$$\hat{\mathbf{d}}_i(n) = \hat{\mathbf{d}}_{i-1}(n) + \mathbf{e}_{i-1}^b(n)k_i^d(n), \quad (3.11)$$

onde  $k_i^d(n)$  é o coeficiente de regressão do processo de estimação conjunta definido em (2.111), isto é

$$k_i^d(n) = \frac{\Delta_{i-1}^d(n)}{\xi_{i-1}^b(n)}, \quad (3.12)$$

sendo

$$\Delta_{i-1}^d(n) = \mathbf{e}_{i-1}^{bT}(n)\mathbf{d}(n) = \mathbf{e}_{i-1}^{bT}(n)\mathbf{e}_{i-1}(n). \quad (3.13)$$

### Atualização na ordem do erro da predição regressiva e sua energia

Utilizando a equação de atualização do operador de projeção em função dos vetores dos erros das predições progressivas (3.4), é possível representar o vetor do erro da predição regressiva (2.31) como sendo

$$\mathbf{e}_{M-1}^b(n) = \begin{bmatrix} 0 \\ \mathbf{e}_{M-2}^b(n-1) \end{bmatrix} - \bar{\mathbf{e}}_{M-2}^f(n)\bar{\mathbf{e}}_{M-2}^{fT}(n)\mathbf{a}(n-M+1). \quad (3.14)$$

Com base em (2.51), a atualização na ordem do vetor do erro da predição regressiva pode ser assim reescrita

$$\mathbf{e}_{M-1}^b(n) = \begin{bmatrix} 0 \\ \mathbf{e}_{M-2}^b(n-1) \end{bmatrix} - \bar{\mathbf{e}}_{M-2}^f(n)\bar{\mathbf{e}}_{M-2}^{fT}(n) \begin{bmatrix} 0 \\ \mathbf{e}_{M-2}^b(n-1) \end{bmatrix}. \quad (3.15)$$

A equação (3.15) mostra que o vetor de erro da predição regressiva é atualizado na ordem e no tempo subtraindo a projeção do erro de estimação regressivo da ordem imediatamente inferior e no instante de tempo anterior sobre o vetor de erro da predição progressiva da ordem imediatamente inferior e no mesmo instante de tempo. A matriz  $\bar{\mathbf{e}}_{M-2}^f(n)\bar{\mathbf{e}}_{M-2}^{fT}(n)$  atua como operador de projeção ortogonal na atualização

na ordem e no tempo do vetor  $\mathbf{e}_{M-1}^b(n)$ . A equação (3.15) é válida para qualquer ordem  $i$  com  $1 \leq i \leq M$ , podendo ser expressa como

$$\mathbf{e}_i^b(n) = \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix} - \mathbf{e}_{i-1}^f(n) k_i^b(n), \quad (3.16)$$

onde  $k_i^b(n)$  é definido como

$$k_i^b(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^f(n)}, \quad (3.17)$$

sendo

$$\Delta_{i-1}(n) = \mathbf{e}_{i-1}^{fT}(n) \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix}. \quad (3.18)$$

Para  $i = 0$ ,  $\mathbf{e}_0^b(n) = \mathbf{a}(n)$ .

Usando (3.16), (3.17) e (3.18), é possível após algumas manipulações algébricas, expressar a atualização na ordem e no tempo da energia do erro da predição regressiva  $\xi_i^b(n) = \|\mathbf{e}_i^b(n)\|^2$  como

$$\xi_i^b(n) = \xi_{i-1}^b(n-1) - \xi_{i-1}^f(n) |k_i^b(n)|^2, \quad \text{para } 1 \leq i \leq M. \quad (3.19)$$

### Atualização na ordem do erro da predição progressiva e da sua energia

Utilizando a equação de atualização do operador de projeção em função dos vetores dos erros das predições regressivas (3.1), o vetor do erro de predição progressiva (2.42) pode ser expresso como

$$\mathbf{e}_{M-1}^f(n) = \mathbf{e}_{M-2}^f(n) - \begin{bmatrix} 0 \\ \bar{\mathbf{e}}_{M-2}^b(n-1) \end{bmatrix} \begin{bmatrix} 0 & \bar{\mathbf{e}}_{M-2}^{bT}(n-1) \end{bmatrix} \mathbf{a}(n) \quad (3.20)$$

sendo

$$\mathbf{e}_{M-2}^f(n) = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{I} - \mathbf{P}_{M-2}(n-1) \end{bmatrix} \mathbf{a}(n).$$

Usando (2.48), a atualização na ordem do vetor do erro da predição progressiva pode ser expressa como

$$\mathbf{e}_{M-1}^f(n) = \mathbf{e}_{M-2}^f(n) - \begin{bmatrix} 0 \\ \bar{\mathbf{e}}_{M-2}^b(n-1) \end{bmatrix} \begin{bmatrix} 0 & \bar{\mathbf{e}}_{M-2}^{bT}(n-1) \end{bmatrix} \mathbf{e}_{M-2}^f(n). \quad (3.21)$$

A equação (3.21) mostra que o vetor de erro de predição progressiva é atualizado na ordem, subtraindo-se a projeção do erro de predição progressivo da ordem imediatamente inferior sobre o vetor de erro da predição regressiva da ordem imediatamente inferior e instante de tempo anterior. Esta é a situação dual de (3.15). A matriz

$$\begin{bmatrix} 0 \\ \bar{\mathbf{e}}_{M-2}^b(n-1) \end{bmatrix} \begin{bmatrix} 0 & \bar{\mathbf{e}}_{M-2}^{bT}(n-1) \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \bar{\mathbf{e}}_{M-2}^b(n-1)\bar{\mathbf{e}}_{M-2}^{bT}(n-1) \end{bmatrix}$$

atua como operador de projeção ortogonal na atualização na ordem do vetor  $\mathbf{e}_{M-1}^f(n)$ . A equação (3.21) é válida para qualquer ordem  $i$  com  $1 \leq i \leq M$ , podendo ser expressa como

$$\mathbf{e}_i^f(n) = \mathbf{e}_{i-1}^f(n) - \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix} k_i^f(n), \quad (3.22)$$

onde  $k_i^f(n)$  é o mesmo coeficiente definido em (2.130), isto é

$$k_i^f(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^b(n-1)}, \quad (3.23)$$

sendo que  $\Delta_{i-1}(n)$  é a mesma correlação cruzada definida em (3.18). Para  $i = 0$ ,  $\mathbf{e}_0^f(n) = \mathbf{a}(n)$ .

Usando (3.22) e (3.23), após algumas manipulações algébricas é possível expressar a atualização na ordem da energia do erro da predição progressiva  $\xi_i^f(n) = \|\mathbf{e}_i^f(n)\|^2$  como

$$\xi_i^f(n) = \xi_{i-1}^f(n) - \xi_{i-1}^b(n-1) |k_i^f(n)|^2 \quad \text{para } 1 \leq i \leq M. \quad (3.24)$$

### Estrutura em treliça

Utilizando o fato de que para atualizar na ordem o erro de predição regressiva é necessário conhecer o erro de predição progressiva do mesmo instante de tempo e que para atualizar o erro de predição progressiva é necessário conhecer o erro de predição regressiva do instante anterior, é possível estabelecer um procedimento para atualizar essas grandezas, utilizando-se somente os correspondentes valores na ordem inferior. Esse procedimento define o filtro em treliça.

A partir do último elemento do vetor de erro da estimação definido em (3.10), e dos últimos elementos dos vetores dos erros da predição regressiva (3.16) e

da predição progressiva (3.22) representados respectivamente como  $b_i(n)$  e  $f_i(n)$  para ordem  $i$ , tem-se o seguinte conjunto de equações

$$e_i(n) = e_{i-1}(n) - b_{i-1}(n)k_i^d(n), \quad (3.25)$$

$$b_i(n) = b_{i-1}(n-1) - f_{i-1}(n)k_i^b(n) \quad (3.26)$$

e

$$f_i(n) = f_{i-1}(n) - b_{i-1}(n-1)k_i^f(n) \quad (3.27)$$

para  $1 \leq i \leq M$ . As equações (3.26) e (3.27) definem a estrutura em treliça baseada no erro da predição *a posteriori*

$$\begin{bmatrix} f_i(n) \\ b_i(n) \end{bmatrix} = \begin{bmatrix} 1 & -k_i^f(n) \\ -k_i^b(n) & 1 \end{bmatrix} \begin{bmatrix} f_{i-1}(n) \\ b_{i-1}(n-1) \end{bmatrix} \quad (3.28)$$

Através de (3.28) e (3.25) o problema de estimação dos mínimos quadrados de ordem  $M$  foi transformado em dois problemas de predição progressiva e regressiva de ordem 1 em vários estágios. Os parâmetros  $k_i^f(n)$  e os  $k_i^b(n)$  definem respectivamente os coeficientes de reflexão da predição progressiva e regressiva e os parâmetros  $k_i^d(n)$  definem os coeficientes de regressão do processo da estimação conjunta.

Os coeficientes  $k_i^d(n)$  apresentados em (3.12) definem o mesmo vetor dos coeficientes de regressão da equação (2.118). Considerando  $1 \leq i \leq M-1$ , os coeficientes  $k_i^f(n)$  apresentados em (3.23) definem o mesmo vetor apresentado em (2.136). A equação (2.144) relaciona o vetor rotacionado da resposta desejada no caso da predição progressiva  $\mathbf{dq1}^f(n)$  com o vetor dos coeficientes de reflexão  $k_{M-1}^f(n)$  da predição progressiva, através da matriz diagonal  $\mathbf{D}_{M-1}^{-1/2}(n-1)$ . Então, da mesma forma que o vetor  $\mathbf{dq1}^f(n)$ , o vetor  $k_{M-1}^f(n)$  pode ser interpretado com uma estimação de estados do sistema que resolve o problema dos mínimos quadrados de forma recorrente na ordem. Dividindo as equações (3.17) e (3.23) é possível relacionar os coeficientes de reflexão  $k_i^b(n)$  e  $k_i^f(n)$  através da seguinte expressão

$$\frac{k_i^b(n)}{k_i^f(n)} = \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^f(n)}. \quad (3.29)$$

As energias da predição regressiva e progressiva podem ser relacionadas dividindo ambos os lados da equação (3.19) e (3.24) respectivamente por  $\xi_{i-1}^b(n-1)$  e

$\xi_{i-1}^f(n)$  resultando na seguinte expressão

$$\frac{\xi_i^b(n)}{\xi_{i-1}^b(n-1)} = \frac{\xi_i^f(n)}{\xi_{i-1}^f(n)} = 1 - |\bar{\Delta}_{i-1}(n)|^2, \quad \text{para } 1 \leq i \leq M. \quad (3.30)$$

O termo

$$\bar{\Delta}_{i-1}(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^{f/2}(n)\xi_{i-1}^{b/2}(n-1)} = [0 \quad \bar{\mathbf{e}}_{i-1}^{bT}(n-1)] \bar{\mathbf{e}}_{i-1}^f(n), \quad (3.31)$$

representa a correlação cruzada entre os erros de predição progressiva e regressiva normalizados, respectivamente

$$\bar{\mathbf{e}}_{i-1}^f(n) = \frac{\mathbf{e}_{i-1}^f(n)}{\xi_{i-1}^{f/2}(n)} \quad (3.32)$$

e

$$\bar{\mathbf{e}}_{i-1}^b(n-1) = \frac{\mathbf{e}_{i-1}^b(n-1)}{\xi_{i-1}^{b/2}(n-1)}. \quad (3.33)$$

De (3.30) conclui-se que as energias dos erros da predição regressiva e progressiva podem ser atualizadas usando a correlação cruzada entre os erros da predição progressiva e regressiva normalizados, isto é

$$\xi_i^b(n) = \xi_{i-1}^b(n-1)(1 - |\bar{\Delta}_{i-1}(n)|^2) \quad (3.34)$$

e

$$\xi_i^f(n) = \xi_{i-1}^f(n)(1 - |\bar{\Delta}_{i-1}(n)|^2). \quad (3.35)$$

Para finalizar este item, relaciona-se as representações para as correlações cruzadas obtidas com os métodos recorrentes na ordem apresentados que resolvem o problema dos mínimos quadrados. As representações para as correlações cruzadas no caso de estimação podem ser relacionadas através de (2.139) e (3.12)

$$dq1_i(n) = \xi_{i-1}^{b/2}(n)k_i^d(n) = \frac{\Delta_{i-1}^d(n)}{\xi_{i-1}^{b/2}(n)}. \quad (3.36)$$

No caso da predição progressiva, as representações para a correlação cruzada podem ser relacionadas através de (2.144), (3.23) e (3.31)

$$dq1_i^f(n) = k_i^f(n)\xi_{i-1}^{b/2}(n-1) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^{b/2}(n-1)} = \xi_{i-1}^{f/2}(n)\bar{\Delta}_{i-1}(n) \quad (3.37)$$

e no caso da predição regressiva, através de (3.17) e (3.31)

$$\bar{\pi}_i^b(n) := k_i^b(n)\xi_{i-1}^{f/2}(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^{f/2}(n)} = \xi_{i-1}^{b/2}(n-1)\bar{\Delta}_{i-1}(n). \quad (3.38)$$

Cabe observar que o termo  $dq1_i^b(n)$  não tem uma relação direta com  $k_i^b(n)$  como ocorre com  $dq1_i^f(n)$  e  $k_i^f(n)$ .



### 3.2 Atualizações no tempo

São apresentadas a seguir as equações de atualização do operador de projeção no tempo e suas conseqüências nas demais variáveis envolvidas na resolução do problema dos mínimos quadrados.

Considerando a matriz de dados conforme (A.6) o operador de projeção definido na equação (2.18) pode ser expresso como

$$\mathbf{P}_M(n) = \begin{bmatrix} \lambda \mathbf{A}_M(n-1) \Phi_M^{-1}(n) \mathbf{A}_M^T(n-1) & \lambda^{1/2} \mathbf{A}_M(n-1) \Phi_M^{-1}(n) \mathbf{u}_M(n) \\ \lambda^{1/2} \mathbf{u}_M^T(n) \Phi_M^{-1}(n) \mathbf{A}_M^T(n-1) & \mathbf{u}_M^T(n) \Phi_M^{-1}(n) \mathbf{u}_M(n) \end{bmatrix},$$

ou ainda, considerando as equações (A.21) e (A.22), como

$$\mathbf{P}_M(n) = \begin{bmatrix} \lambda \mathbf{A}_M(n-1) \Phi_M^{-1}(n) \mathbf{A}_M^T(n-1) & \lambda^{1/2} \mathbf{A}_M(n-1) \mathbf{g}_M(n) \\ \lambda^{1/2} \mathbf{g}_M^T(n) \mathbf{A}_M^T(n-1) & 1 - \gamma_M(n) \end{bmatrix}.$$

Representando a inversa da matriz de autocorrelação conforme (A.23) após algumas manipulações algébricas obtém-se a seguinte expressão recorrente no tempo para atualizar o operador de projeção

$$\mathbf{P}_M(n) = \begin{bmatrix} \mathbf{P}_M(n-1) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} - \frac{\mathbf{X}_M(n) \mathbf{X}_M^T(n)}{\gamma_M(n)} \quad (3.39)$$

onde

$$\mathbf{X}_M^T(n) = \begin{bmatrix} -\lambda^{1/2} \mathbf{g}_M^T(n) \mathbf{A}_M^T(n-1) & \gamma_M(n) \end{bmatrix}.$$

A equação (3.39) será utilizada em seguida, para obter as atualizações no tempo do erro de estimação, dos erros das predições progressiva e regressiva e suas respectivas energias e das correlações cruzadas.

#### a) Atualização no tempo do vetor dos erros de estimação:

Substituindo (3.39) na equação do vetor do erro da estimação (2.16) resulta após algumas manipulações algébricas

$$\mathbf{e}_M(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{e}_M(n-1) \\ 0 \end{bmatrix} + \begin{bmatrix} -\lambda^{1/2} \mathbf{A}_M(n-1) \mathbf{g}_M(n) \\ \gamma_M(n) \end{bmatrix} \alpha_M(n), \quad (3.40)$$

onde

$$\gamma_M(n) \alpha_M(n) = \mathbf{X}_M^T(n) \begin{bmatrix} \lambda^{1/2} \mathbf{d}(n-1) \\ d(n) \end{bmatrix}. \quad (3.41)$$

Usando (A.20), é possível mostrar que  $\alpha_M(n)$  é o mesmo erro de estimação *a priori* definido na equação (2.54). Segundo (3.40), o último elemento do vetor  $\mathbf{e}_M(n)$  pode ser representado como  $e_M(n) = \gamma_M(n)\alpha_M(n)$ , resultado que foi obtido no apêndice A, (A.40), de outra forma.

**b) Atualização no tempo do vetor dos erros de predição regressiva:**

Considerando (3.39) para ordem  $M - 1$ , o vetor do erro de predição regressivo (2.31) pode ser atualizado no tempo como

$$\mathbf{e}_{M-1}^b(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{e}_{M-1}^b(n-1) \\ 0 \end{bmatrix} + \begin{bmatrix} -\lambda^{1/2} \mathbf{A}_{M-1}(n-1) \mathbf{g}_{M-1}(n) \\ \gamma_{M-1}(n) \end{bmatrix} \psi_{M-1}(n), \quad (3.42)$$

onde  $\psi_{M-1}(n) = \gamma_{M-1}^{-1}(n) \mathbf{X}_{M-1}^T(n) \mathbf{a}(n - M + 1)$ . De (3.42) decorre que  $b_{M-1}(n)$ , o último elemento do vetor  $\mathbf{e}_{M-1}^b(n)$  satisfaz:

$$b_{M-1}(n) = \gamma_{M-1}(n) \psi_{M-1}(n) = \mathbf{X}_{M-1}^T(n) \mathbf{a}(n - M + 1). \quad (3.43)$$

Usando (A.20) para ordem  $M - 1$ , é possível mostrar que  $\psi_{M-1}(n)$  é o mesmo erro de estimação regressiva *a priori* de ordem  $M - 1$  no instante  $n$  definido em (2.61). A relação  $b_{M-1}(n) = \gamma_{M-1}(n) \psi_{M-1}(n)$ , foi obtida de outra forma no Apêndice A, equação (A.43).

**c) Atualização no tempo do vetor dos erros de predição progressiva:**

Considerando (3.39) para ordem  $M - 1$ , o vetor do erro de predição progressiva (2.42), pode ser atualizado no tempo como

$$\mathbf{e}_{M-1}^f(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{e}_{M-1}^f(n-1) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -\lambda^{1/2} \mathbf{A}_{M-1}(n-2) \mathbf{g}_{M-1}(n-1) \\ \gamma_{M-1}(n-1) \end{bmatrix} \eta_{M-1}(n) \quad (3.44)$$

onde  $\eta_{M-1}(n) = \gamma_{M-1}^{-1}(n-1) \left[ 0 \ \mathbf{X}_{M-1}^T(n-1) \right] \mathbf{a}(n)$ . De (3.44) segue-se que  $f_{M-1}(n)$ , o último elemento do vetor  $\mathbf{e}_{M-1}^f(n)$  satisfaz:

$$f_{M-1}(n) = \gamma_{M-1}(n-1) \eta_{M-1}(n) = \left[ 0 \ \mathbf{X}_{M-1}^T(n-1) \right] \mathbf{a}(n). \quad (3.45)$$

Usando (A.20) para ordem  $M - 1$  e tempo  $n - 1$ , é possível mostrar que  $\eta_{M-1}(n)$  é o mesmo erro de estimação progressiva *a priori* de ordem  $M - 1$  no instante  $n$  definido em (2.64). A relação  $f_{M-1}(n) = \gamma_{M-1}(n-1) \eta_{M-1}(n)$ , foi obtida de outra forma no Apêndice A, equação (A.46).

**d) Atualização no tempo da energia de erro da predição regressiva:**

Considerando (3.39) para ordem  $M - 1$  e o último elemento do vetor  $\mathbf{e}_{M-1}^b(n)$  como em (3.43), obtém-se a partir da equação (2.32) a seguinte equação recorrente no tempo para atualização da energia do erro de predição regressivo

$$\xi_{M-1}^b(n) = \lambda \xi_{M-1}^b(n-1) + \frac{|b_{M-1}(n)|^2}{\gamma_{M-1}(n)}. \quad (3.46)$$

**e) Atualização no tempo da energia de erro da predição progressiva:**

Considerando (3.39) para ordem  $M - 1$  e tempo  $n - 1$  e o último elemento do vetor  $\mathbf{e}_{M-1}^f(n)$  como em (3.45), obtém-se a partir da equação (2.43) a seguinte equação recursiva no tempo e na ordem para atualização da energia do erro da predição progressiva:

$$\xi_{M-1}^f(n) = \lambda \xi_{M-1}^f(n-1) + \frac{|f_{M-1}(n)|^2}{\gamma_{M-1}(n-1)}. \quad (3.47)$$

**f) Atualização no tempo da correlação cruzada entre o sinal desejado e o erro da predição regressiva:**

Considerando a representação do vetor do erro da predição regressiva conforme (2.31), a equação da correlação cruzada (3.13) para  $i = M$  pode ser reescrita como

$$\Delta_{M-1}^d(n) = \mathbf{a}^T(n - M + 1)(\mathbf{I} - \mathbf{P}_{M-1}(n))\mathbf{d}(n).$$

Utilizando a expressão de atualização no tempo do operador de projeção (3.39) para ordem  $M - 1$ , resulta

$$\Delta_{M-1}^d(n) = \lambda \Delta_{M-1}^d(n-1) + \frac{b_{M-1}(n)e_{M-1}(n)}{\gamma_{M-1}(n)} \quad (3.48)$$

onde

$$\lambda \Delta_{M-1}^d(n-1) = \mathbf{a}^T(n - M + 1) \begin{bmatrix} \mathbf{I} - \mathbf{P}_{M-1}(n-1) \\ \mathbf{0}^T \end{bmatrix} \lambda^{1/2} \mathbf{d}(n-1),$$

sendo  $e_{M-1}(n)$  e  $b_{M-1}(n)$  respectivamente, os últimos elementos do vetor  $\mathbf{e}_{M-1}(n)$  definido em (3.41) e do vetor  $\mathbf{e}_{M-1}^b(n)$  definido em (3.43).

Considerando a representação do vetor do erro da predição regressiva conforme (2.31) e da predição progressiva conforme (2.42), a equação de correlação

cruzada (3.18) para  $i = M$ , pode ser reescrita como

$$\Delta_{M-1}(n) = \mathbf{a}^T(n) \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{I} - \mathbf{P}_{M-1}(n-1) \end{bmatrix} \mathbf{a}(n-M+1).$$

Utilizando a expressão de atualização no tempo do operador de projeção (3.39) para ordem  $M-1$  e tempo  $n-1$  resulta

$$\Delta_{M-1}(n) = \lambda \Delta_{M-1}(n-1) + \frac{b_{M-1}(n-1)f_{M-1}(n)}{\gamma_{M-1}(n-1)}, \quad (3.49)$$

onde

$$\Delta_{M-1}(n-1) = \mathbf{a}^T(n) \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \begin{bmatrix} \mathbf{I} - \mathbf{P}_{M-1}(n-2) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \end{bmatrix} \mathbf{a}(n-M+1),$$

$b_{M-1}(n-1)$  e  $f_{M-1}(n)$  são respectivamente os últimos elementos do vetor  $\mathbf{e}_{M-1}^b(n-1)$  definido na (3.43) e do vetor  $\mathbf{e}_{M-1}^f(n)$ , definido na equação (3.45).

**g) Atualização do fator de conversão:**

Dividindo ambos os lados das equações (3.46) e (A.52), respectivamente por  $\xi_{M-1}^b(n)$  e  $\gamma_{M-1}(n)$  é possível obter a seguinte igualdade:

$$\frac{\gamma_M(n)}{\gamma_{M-1}(n)} = \lambda \frac{\xi_{M-1}^b(n-1)}{\xi_{M-1}^b(n)} = 1 - \frac{|b_{M-1}(n)|^2}{\gamma_{M-1}(n)\xi_{M-1}^b(n)}. \quad (3.50)$$

Dividindo ambos os lados das equações (3.47) e (A.53), respectivamente por  $\xi_{M-1}^f(n)$  e por  $\gamma_{M-1}(n-1)$  é possível obter a seguinte igualdade:

$$\frac{\gamma_M(n)}{\gamma_{M-1}(n-1)} = \lambda \frac{\xi_{M-1}^f(n-1)}{\xi_{M-1}^f(n)} = 1 - \frac{|f_{M-1}(n)|^2}{\gamma_{M-1}(n-1)\xi_{M-1}^f(n)}. \quad (3.51)$$

As relações (3.50) e (3.51), bem como as equações de atualização no tempo das energias do erro da predição (3.46) e (3.47) e das correlações cruzadas (3.48) e (3.49), são válidas para qualquer ordem  $i$  com  $1 \leq i \leq M$ . As atualizações no tempo das energias  $\xi_{i-1}^b(n)$  e  $\xi_{i-1}^f(n)$  e das correlações cruzadas  $\Delta_{i-1}^d(n)$  e  $\Delta_{i-1}(n)$  aqui apresentadas, são usadas implicita ou explicitamente em todas as versões dos algoritmos RLS recursivos na ordem rápida com realimentação de erro. As várias formas possíveis de obter estas atualizações são apresentadas nas próximas seções, e na medida do possível, relacionadas às atualizações dos vetores de estados apresentados no capítulo 2.

### 3.3 Representações para a estrutura em treliça

Os algoritmos rápidos exploram a natureza sequencial dos dados de entrada, utilizando a predição progressiva e regressiva para calcular os erros de estimação de forma conveniente. A seção de predição dos algoritmos RLS rápidos recursivos na ordem pode ser representada através da estrutura em treliça baseada nos erros de predição *a posteriori*, conforme apresentado na equação (3.28). Entretanto, a representação da seção de predição através da estrutura em treliça, pode ser efetuada de várias outras formas, como por exemplo a baseada nos erros da predição *a posteriori* normalizados, a dos erros da predição *a priori*, normalizados ou não, e a dos erros da predição normalizados em ângulo. Essas quatro representações são apresentadas a seguir, sendo que inicialmente para facilitar comparações e consultas é repetida a estrutura em treliça baseada nos erros *a posteriori*.

#### 1) Estrutura em treliça com erros de predição *a posteriori*:

Conforme já apresentado em (3.28), os últimos elementos dos vetores dos erros de predição progressiva (3.22) e regressiva (3.16) *a posteriori*, respectivamente  $f_i(n)$  e  $b_i(n)$ , podem ser calculados da seguinte forma:

$$\begin{bmatrix} f_i(n) \\ b_i(n) \end{bmatrix} = \begin{bmatrix} 1 & -k_i^f(n) \\ -k_i^b(n) & 1 \end{bmatrix} \begin{bmatrix} f_{i-1}(n) \\ b_{i-1}(n-1) \end{bmatrix}. \quad (3.52)$$

Os coeficientes de reflexão  $k_i^f(n)$  e  $k_i^b(n)$  podem ser calculados através de (3.23) e (3.17), ou seja

$$k_i^f(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^b(n-1)} \quad \text{e} \quad k_i^b(n) = \frac{\Delta_{i-1}(n)}{\xi_{i-1}^f(n)},$$

com

$$\Delta_{i-1}(n) = \mathbf{e}_{i-1}^{fT}(n) \begin{bmatrix} 0 \\ \mathbf{e}_{i-1}^b(n-1) \end{bmatrix},$$

$$\xi_{i-1}^b(n-1) = \|\mathbf{e}_{i-1}^b(n-1)\|^2 \quad \text{e} \quad \xi_{i-1}^f(n) = \|\mathbf{e}_{i-1}^f(n)\|^2.$$

#### 2) Estrutura em treliça com erros de predição *a posteriori* normalizados:

Os últimos elementos dos vetores dos erros da predição progressiva e regressiva *a posteriori* normalizados, definidos respectivamente nas equações (3.32) e (3.33),

são representados para ordem  $i$  como

$$\bar{f}_i(n) = \frac{f_i(n)}{\xi_i^{f/2}(n)} \quad \text{e} \quad \bar{b}_i(n) = \frac{b_i(n)}{\xi_i^{b/2}(n)}. \quad (3.53)$$

Considerando então os valores normalizados de  $f_i(n)$  e  $b_i(n)$ , após algumas manipulações algébricas, a equação (3.52) pode ser transformada na seguinte estrutura em treliça, baseada nos erros normalizados da predição *a posteriori*,

$$\begin{bmatrix} \bar{f}_i(n) \\ \bar{b}_i(n) \end{bmatrix} = \frac{1}{a_i(n)} \begin{bmatrix} 1 & -\bar{\Delta}_{i-1}(n) \\ -\bar{\Delta}_{i-1}(n) & 1 \end{bmatrix} \begin{bmatrix} \bar{f}_{i-1}(n) \\ \bar{b}_{i-1}(n-1) \end{bmatrix}, \quad (3.54)$$

onde

$$a_i(n) = \sqrt{1 - |\bar{\Delta}_{i-1}(n)|^2}.$$

A equação (3.54) tem o efeito indesejável de amplificar os erros numéricos. Uma forma numericamente mais bem comportada é apresentada em [2], sendo dada por

$$\begin{bmatrix} \bar{f}_{i-1}(n) \\ \bar{b}_i(n) \end{bmatrix} = \begin{bmatrix} a_i(n) & \bar{\Delta}_{i-1}(n) \\ -\bar{\Delta}_{i-1}(n) & a_i(n) \end{bmatrix} \begin{bmatrix} \bar{f}_i(n) \\ \bar{b}_{i-1}(n-1) \end{bmatrix}. \quad (3.55)$$

O termo  $\bar{\Delta}_{i-1}(n)$  para  $1 \leq i \leq M$ , definido em (3.31), representa o coeficiente de reflexão da estrutura em treliça normalizada, isto é,

$$\bar{\Delta}_{i-1}(n) = \begin{bmatrix} 0 & \bar{\mathbf{e}}_{i-1}^b(n-1) \end{bmatrix} \bar{\mathbf{e}}_{i-1}^f(n)$$

com

$$\bar{\mathbf{e}}_{i-1}^b(n-1) = \mathbf{e}_{i-1}^b(n-1)/\xi_{i-1}^{b/2}(n-1) \quad \text{e} \quad \bar{\mathbf{e}}_{i-1}^f(n) = \mathbf{e}_{i-1}^f(n)/\xi_{i-1}^{f/2}(n).$$

Cabe observar que na equação (3.54), os cálculos são efetuados a partir da ordem 0, sendo necessário conhecer apenas  $\bar{\Delta}_{i-1}(n)$  para  $1 \leq i \leq M$  e

$$\bar{f}_0(n) = u(n)\xi_0^{-f/2}(n),$$

além dos erros de predição regressivos normalizados do instante anterior. Entretanto, em (3.55) os cálculos são efetuados a partir da ordem  $M$ , sendo necessário conhecer não só  $\bar{\Delta}_{i-1}(n)$  para  $1 \leq i \leq M$  e erros de predição regressivos normalizados do instante anterior, mas conhecer também  $\bar{f}_M(n)$ .

### 3) Estrutura em treliça com erros de predição a priori:

Considerando os erros *a priori* da predição progressiva e regressiva conforme definidos em (A.46) e (A.43) e as equações de atualização da seção 3.2, após algumas manipulações algébricas a equação (3.52) pode ser transformada na seguinte estrutura em treliça baseada nos erros da predição *a priori*:

$$\begin{bmatrix} \eta_i(n) \\ \psi_i(n) \end{bmatrix} = \begin{bmatrix} 1 & -k_i^f(n-1) \\ -k_i^b(n-1) & 1 \end{bmatrix} \begin{bmatrix} \eta_{i-1}(n) \\ \psi_{i-1}(n-1) \end{bmatrix}. \quad (3.56)$$

Considerando a relação  $e_M(n) = \gamma_M(n)\alpha_M(n)$ , a atualização na ordem do erro da estimação *a posteriori* (3.25) e as equações de atualização da seção 3.2, após algumas manipulações algébricas o erro da estimação *a priori*, pode ser expresso do seguinte modo:

$$\alpha_i(n) = \alpha_{i-1}(n) - \psi_{i-1}(n)k_i^d(n-1). \quad (3.57)$$

Observa-se que os coeficientes de reflexão e de regressão são os mesmo de (3.52), (3.28) e (3.25) porém, estão no instante de tempo  $n-1$ .

### 4) Estrutura em treliça com erros de predição a priori normalizados:

Por analogia à normalização do erro da predição regressiva *a priori* definido em (2.153), isto é

$$\bar{\psi}_i(n) = \frac{\psi_i(n)}{\lambda^{1/2}\xi_i^{b/2}(n-1)}, \quad (3.58)$$

define-se a normalização do erro da predição progressiva *a priori* como

$$\bar{\eta}_i(n) = \frac{\eta_i(n)}{\lambda^{1/2}\xi_i^{f/2}(n-1)}. \quad (3.59)$$

Substituindo (3.58) e (3.59) em (3.56), e reconhecendo (3.30) e (3.31), resulta após algumas manipulações algébricas:

$$\begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \frac{1}{a_i(n-1)} \begin{bmatrix} 1 & -\bar{\Delta}_{i-1}(n-1) \\ -\bar{\Delta}_{i-1}(n-1) & 1 \end{bmatrix} \begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_{i-1}(n-1) \end{bmatrix}.$$

Sendo que esta última equação, com em (3.54), tem o efeito indesejável de amplificar os erros numéricos. A forma estabilizada é dada por

$$\begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \begin{bmatrix} a_i(n-1) & \bar{\Delta}_{i-1}(n-1) \\ -\bar{\Delta}_{i-1}(n-1) & a_i(n-1) \end{bmatrix} \begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_{i-1}(n-1) \end{bmatrix}. \quad (3.60)$$

Neste caso, os coeficientes de reflexão da estrutura em treliça normalizada baseada no erro *a priori*, estão no instante de tempo  $n - 1$ . Cabe observar que em (3.60) os cálculos são efetuados a partir da ordem  $M$ . Então além de conhecer  $\bar{\Delta}_{i-1}(n-1)$  e  $\bar{\psi}_{i-1}(n-1)$  para  $1 \leq i \leq M$ , é necessário conhecer também  $\bar{\eta}_M(n)$ .

##### 5) Estrutura em treliça com erros de predição normalizados em ângulo:

Considerando os erros das predições progressiva e regressiva normalizados em ângulo, conforme definidos respectivamente nas equações (2.65) e (2.62), após algumas manipulações algébricas, é possível transformar a equação (3.52) em

$$\begin{bmatrix} \epsilon_i^f(n) \\ \epsilon_i^b(n) \end{bmatrix} = \begin{bmatrix} \frac{\gamma_i^{1/2}(n-1)}{\gamma_{i-1}^{1/2}(n-1)} & \frac{-k_i^f(n-1)\gamma_i^{1/2}(n-1)}{\gamma_{i-1}^{1/2}(n-1)} \\ \frac{-k_i^b(n-1)\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} & \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} \end{bmatrix} \begin{bmatrix} \epsilon_{i-1}^f(n) \\ \epsilon_{i-1}^b(n-1) \end{bmatrix}.$$

Considerando (3.37), (3.38), (3.50) e (3.51), resulta:

$$\begin{bmatrix} \epsilon_i^f(n) \\ \epsilon_i^b(n) \end{bmatrix} = \lambda^{1/2} \begin{bmatrix} \frac{\xi_{i-1}^{b/2}(n-2)}{\xi_{i-1}^{b/2}(n-1)} & -\frac{dq1_i^f(n-1)}{\xi_{i-1}^{b/2}(n-1)} \\ -\frac{\pi_i^b(n)}{\xi_{i-1}^{f/2}(n)} & \frac{\xi_{i-1}^{f/2}(n-1)}{\xi_{i-1}^{f/2}(n)} \end{bmatrix} \begin{bmatrix} \epsilon_{i-1}^f(n) \\ \epsilon_{i-1}^b(n-1) \end{bmatrix}. \quad (3.61)$$

As diferentes versões dos algoritmos RLS rápidos recorrentes na ordem utilizam diferentes representações para a estrutura em treliça. As representações apresentadas em (3.28) e (3.56) são as mais usuais e os algoritmos baseados nestas representações podem ser encontrados por exemplo em [38, cap. 17]. A representação baseada no erro normalizado da predição *a posteriori* (3.55) foi usada em [33] para desenvolver um algoritmo híbrido QR-LSL. A representação baseada no erro normalizado da predição *a priori* (3.60), que pretende ser uma contribuição original deste trabalho, será usada na próxima seção para desenvolver um outro algoritmo híbrido QR-LSL. A representação da estrutura em treliça baseada nos erros das predições normalizados em ângulo da equação (3.61) é usada de forma implícita no algoritmo QR-RLS apresentado em [38, cap. 18].

#### Relações entre os erros de predição normalizados

As relações entre os erros *a posteriori* e *a priori* nos casos de estimação, de predição regressiva e de predição progressiva não normalizados estão relacionados de



forma simples pelo fator de conversão, conforme as equações (A.40), (A.43) e (A.46) respectivamente. Como é de se esperar no caso dos erros de predição regressiva e progressiva normalizados, a relação entre os erros *a posteriori* e *a priori* ainda depende diretamente de fatores de conversão. Como todos esses erros aparecem nas estruturas em treliça apresentadas é conveniente encerrar esta seção apresentando as diferentes relações entre os erros de predição normalizados *a priori*, *a posteriori* e normalizados em ângulo.

A equação (2.62) relaciona os erros de predição regressiva *a priori*, *a posteriori* e o normalizado em ângulo, isto é

$$\epsilon_{i-1}^b(n) = \frac{b_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)} = \psi_{i-1}(n)\gamma_{i-1}^{1/2}(n). \quad (3.62)$$

Usando (3.62), (3.53) e (3.58) é possível relacionar os três tipos de erros de predição regressiva normalizados pela expressão

$$\epsilon_{i-1}^b(n) = \bar{b}_{i-1}(n) \frac{\xi_{i-1}^{b/2}(n)}{\gamma_{i-1}^{1/2}(n)} = \bar{\psi}_{i-1}(n)\lambda^{1/2}\xi_{i-1}^{b/2}(n-1)\gamma_{i-1}^{1/2}(n). \quad (3.63)$$

Desta última equação, juntamente com (3.50) para ordem  $i-1$ , resulta em

$$\bar{b}_{i-1}(n) = \gamma_i^{1/2}(n)\gamma_{i-1}^{1/2}(n)\bar{\psi}_{i-1}(n). \quad (3.64)$$

A equação (2.65) por sua vez relaciona os erros de predição progressiva *a priori*, *a posteriori* e normalizado em ângulo, isto é

$$\epsilon_{i-1}^f(n) = \frac{f_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)} = \eta_{i-1}(n)\gamma_{i-1}^{1/2}(n-1). \quad (3.65)$$

Usando (3.65), (3.53) e (3.59), é possível relacionar os três tipos de erros de predição progressiva normalizado da seguinte forma:

$$\epsilon_{i-1}^f(n) = \bar{f}_{i-1}(n) \frac{\xi_{i-1}^{f/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} = \bar{\eta}_{i-1}(n)\lambda^{1/2}\xi_{i-1}^{f/2}(n-1)\gamma_{i-1}^{1/2}(n-1). \quad (3.66)$$

Desta última equação juntamente com (3.51), para ordem  $i-1$ , resulta que

$$\bar{f}_{i-1}(n) = \gamma_{i-1}^{1/2}(n-1)\gamma_i^{1/2}(n)\bar{\eta}_{i-1}(n). \quad (3.67)$$

### 3.4 Atualização no tempo dos coeficientes da estrutura em treliça

Os coeficientes de reflexão e de regressão para cada uma das estruturas em treliça consideradas podem ser calculados de diferentes formas. Uma das formas, que apresenta vantagens do ponto de vista numérico, é a atualização no tempo a partir do valor anterior [14]. A seguir, são apresentadas expressões para a atualização no tempo dos coeficientes das estruturas em treliça não normalizadas e a normalizadas em ângulo.

#### Atualização dos coeficientes dos filtros em treliça não normalizados

Usando (3.48) para ordem  $i - 1$ , o coeficiente de regressão (3.12) pode ser expresso como

$$k_i^d(n) = \lambda \frac{\Delta_{i-1}^d(n-1)}{\xi_{i-1}^b(n)} + \frac{b_{i-1}(n)e_{i-1}(n)}{\xi_{i-1}^b(n)\gamma_{i-1}(n)}.$$

Esta última equação pode ser reescrita como

$$k_i^d(n) = k_i^d(n-1) \lambda \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} + \frac{b_{i-1}(n)e_{i-1}(n)}{\xi_{i-1}^b(n)\gamma_{i-1}(n)}. \quad (3.68)$$

A partir de (3.50) para ordem  $i - 1$  tem-se

$$\lambda \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} = 1 - \frac{|b_{i-1}(n)|^2}{\gamma_{i-1}(n)\xi_{i-1}^b(n)}. \quad (3.69)$$

Assim (3.68) pode ser reescrita como

$$k_i^d(n) = k_i^d(n-1) + \frac{b_{i-1}(n)}{\xi_{i-1}^b(n)\gamma_{i-1}(n)} (-b_{i-1}(n)k_i^d(n-1) + e_{i-1}(n)). \quad (3.70)$$

Com (3.57), o termo entre parênteses no lado direito de (3.70), pode ser expresso como

$$-b_{i-1}(n)k_i^d(n-1) + e_{i-1}(n) = \gamma_{i-1}(n)\alpha_i(n)$$

e finalmente  $k_i^d(n)$ , pode ser expresso como

$$k_i^d(n) = k_i^d(n-1) + \frac{\psi_{i-1}(n)\gamma_{i-1}(n)}{\xi_{i-1}^b(n)} \alpha_i(n). \quad (3.71)$$

Esta é a equação que atualiza o coeficiente de regressão de forma recorrente no tempo. De forma análoga, considerando porém (3.49), é possível obter as seguintes expressões de atualização no tempo para o coeficiente de reflexão definido em (3.23)

$$k_i^f(n) = k_i^f(n-1) + \frac{\psi_{i-1}(n-1)\gamma_{i-1}(n-1)}{\xi_{i-1}^b(n-1)}\eta_i(n) \quad (3.72)$$

e para o coeficiente de reflexão definido em (3.17)

$$k_i^b(n) = k_i^b(n-1) + \frac{\eta_{i-1}(n)\gamma_{i-1}(n-1)}{\xi_{i-1}^f(n)}\psi_i(n). \quad (3.73)$$

Na implementação dos algoritmos em treliça não normalizados, os coeficientes podem ser atualizados de duas formas. Por exemplo, no caso dos coeficientes de regressão  $k_i^d(n)$ , pode-se calculá-los através da correlação cruzada entre os erros de predição regressiva e os erros de estimação, como em (3.12) e (3.13) ou através da expressão (3.71). Quando implementados em precisão infinita estes dois métodos são equivalentes. Porém, como observado em [38, pág. 711] e em [14], quando implementados em precisão finita, podem existir grandes diferenças. A atualização direta dos coeficientes segundo (3.71), (3.73) e (3.72), apresenta vantagens do ponto de vista numérico sobre o cálculo dos coeficientes através do produtos internos de vetores de erro. A explicação para esse comportamento superior da atualização direta dos coeficientes, é a existência de um mecanismo de realimentação de erro que reduz a magnitude dos erros numéricos. A figura 3.1 ilustra o mecanismo de realimentação de erro para o caso da estimação conjunta, obtida a partir das equações (3.57) e (3.71). Deve-se

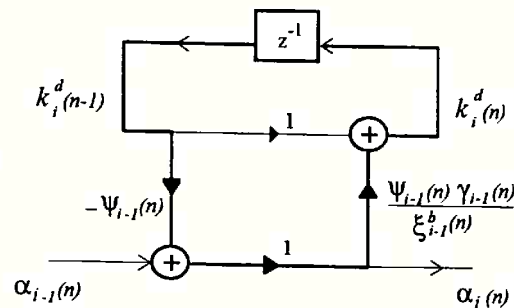


Figura 3.1: Mecanismo de realimentação do erro no cálculo do coeficiente de regressão.

observar que uma realimentação ‘negativa’ reduz o efeito das variações do parâmetro  $k_i^d(n)$ , pois o erro está sendo realimentado com sinal negativo. Para maiores detalhes ver [38, pág. 632 e pág. 708] e [14, 13]. Os mecanismos de realimentação no

cálculo dos coeficientes de reflexão da predição regressiva e progressiva são obtidos de forma análoga a partir das equações (3.73), (3.72) e (3.56). Devido a esta propriedade, fala-se de versões de algoritmos RLS rápidos com realimentação de erro. Estas versões utilizam as representações das estruturas em treliça baseadas nos erros *a posteriori*, nos erros *a priori* e nos erros normalizados; sendo caracterizadas pelo uso das expressões (3.71), (3.73) e (3.72) para atualizar os coeficientes.

### Atualização dos coeficientes dos filtros em treliça com erros normalizados em ângulo

De (3.48) para ordem  $i - 1$  e da equação (3.36), é possível obter a seguinte expressão de atualização no tempo para o coeficiente  $dq1_i(n)$ :

$$dq1_i(n) = (\lambda^{1/2} \frac{\xi_{i-1}^{b/2}(n-1)}{\xi_{i-1}^{b/2}(n)}) \lambda^{1/2} dq1_i(n-1) + (\frac{\bar{b}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)}) \epsilon_{i-1}(n). \quad (3.74)$$

Considerando a equação (3.50) para ordem  $i - 1$  reescrita como

$$\xi_{i-1}^b(n) \gamma_i(n) = \lambda \gamma_{i-1}(n) \xi_{i-1}^b(n-1)$$

e (3.57), resulta para a atualização na ordem do erro de estimação normalizado em ângulo  $\epsilon_i(n) = \gamma_i^{1/2}(n) \alpha_i(n)$  a seguinte expressão:

$$\epsilon_i(n) = -(\frac{\bar{b}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)}) \lambda^{1/2} dq1_i(n-1) + (\frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n)}) \epsilon_{i-1}(n). \quad (3.75)$$

Da mesma forma, de (3.49) reescrita para ordem  $i - 1$  e da equação (3.37), é possível obter a seguinte expressão para atualização no tempo do coeficiente  $dq1_i^f(n)$ :

$$dq1_i^f(n) = (\lambda^{1/2} \frac{\xi_{i-1}^{b/2}(n-2)}{\xi_{i-1}^{b/2}(n-1)}) \lambda^{1/2} dq1_i^f(n-1) + (\frac{\bar{b}_{i-1}(n-1)}{\gamma_{i-1}^{1/2}(n-1)}) \epsilon_{i-1}^f(n). \quad (3.76)$$

Considerando o erro da predição progressiva normalizado em ângulo

$$\epsilon_i^f(n) = \gamma_i^{1/2}(n-1) \eta_i(n)$$

e  $\eta_i(n)$  atualizado como em (3.56) resulta

$$\epsilon_i^f(n) = -(\frac{\bar{b}_{i-1}(n-1)}{\gamma_{i-1}^{1/2}(n-1)}) \lambda^{1/2} dq1_i^f(n-1) + (\frac{\gamma_i^{1/2}(n-1)}{\gamma_{i-1}^{1/2}(n-1)}) \epsilon_{i-1}^f(n). \quad (3.77)$$

Analogamente, de (3.49) para ordem  $i - 1$  e da equação (3.38), é possível obter a seguinte expressão para atualização no tempo do coeficiente  $\pi_i^b(n)$ :

$$\pi_i^b(n) = \left( \lambda^{1/2} \frac{\xi_{i-1}^{f/2}(n-1)}{\xi_{i-1}^{f/2}(n)} \right) \lambda^{1/2} \pi_i^b(n-1) + \left( \frac{\bar{f}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)} \right) \epsilon_{i-1}^b(n-1) \quad (3.78)$$

Considerando o erro da predição regressiva normalizado em ângulo

$$\epsilon_i^b(n) = \gamma_i^{1/2}(n) \psi_i(n)$$

e  $\psi_i(n)$  atualizado como em (3.56), resulta que

$$\epsilon_i^b(n) = - \left( \frac{\bar{f}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)} \right) \lambda^{1/2} \pi_i^b(n-1) + \left( \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} \right) \epsilon_{i-1}^b(n-1). \quad (3.79)$$

As equações (3.74) a (3.79) parecem a primeira vista um tanto complicadas. Entretanto, na seção 3.6 elas serão interpretadas como resultantes da aplicação de rotações sobre vetores de dois elementos, o que permitirá uma formulação muito mais compacta. Antes porém, é interessante completar as interpretações da seção 2.6 sobre as matrizes de sistema  $\mathbf{S}_M^k(n)$  e  $\mathbf{S}_M^g(n)$ .

### 3.5 Sobre as matrizes de sistema $\mathbf{S}_M^k(n)$ e $\mathbf{S}_M^g(n)$

Nesta seção, as equações de atualização no tempo são usadas para completar as interpretações dos elementos das matrizes de sistema  $\mathbf{S}_M^k(n)$  e  $\mathbf{S}_M^g(n)$ . A matriz  $\mathbf{S}_M^k(n)$  descreve as operações de adaptação e filtragem do algoritmo RLS que utilizam o procedimento de ortogonalização de Gram-Schmidt e  $\mathbf{S}_M^g(n)$  tem o mesmo papel no caso do algoritmo QR-RLS baseado em rotações de Givens.

Como  $\alpha_0(n) = d(n)$  a equação (3.57) pode ser expressa em termos de  $d(n)$ , isto é

$$\alpha_i(n) = d(n) - \sum_{j=1}^i \psi_{j-1}(n) k_j^d(n-1). \quad (3.80)$$

Consequentemente, para o erro normalizado em ângulo resulta que

$$\epsilon_i(n) = \gamma_i^{1/2}(n) (d(n) - \sum_{j=1}^i \psi_{j-1}(n) k_j^d(n-1)). \quad (3.81)$$

Substituindo (3.80) na equação (3.71), obtém-se a seguinte expressão:

$$k_i^d(n) = \sum_{j=1}^{i-1} s_{ij}^k(n) k_j^d(n-1) + s_{ii}^k(n) k_i^d(n-1) + \frac{b_{i-1}(n)}{\xi_{i-1}^b(n)} d(n), \quad (3.82)$$

sendo  $s_{ii}^k(n)$  e  $s_{ji}^k(n)$  definidos como

$$s_{ii}^k(n) = 1 - \frac{b_{i-1}(n)\psi_{i-1}(n)}{\xi_{i-1}^b(n)} = \lambda \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} \quad (3.83)$$

e

$$s_{ij}^k(n) = -\frac{b_{i-1}(n)}{\xi_{i-1}^b(n)}\psi_{j-1}(n). \quad (3.84)$$

Assim é possível expressar o vetor  $\mathbf{k}_M^d(n)$  e o erro de estimação normalizado em ângulo  $\epsilon_M(n)$  através da seguinte equação de atualização no tempo

$$\begin{bmatrix} k_1^d(n) \\ k_2^d(n) \\ \vdots \\ k_M^d(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^k(n) \begin{bmatrix} k_1^d(n-1) \\ k_2^d(n-1) \\ \vdots \\ k_M^d(n-1) \\ d(n) \end{bmatrix},$$

onde

$$\mathbf{S}_M^k(n) = \begin{bmatrix} s_{11}^k(n) & 0 & \cdots & 0 & b_0(n)\xi_0^{-b}(n) \\ s_{21}^k(n) & s_{22}^k(n) & \cdots & 0 & b_1(n)\xi_1^{-b}(n) \\ s_{31}^k(n) & s_{31}^k(n) & \cdots & 0 & b_1(n)\xi_1^{-b}(n) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ s_{M1}^k(n) & s_{M2}^k(n) & \cdots & s_{MM}^k(n) & b_M(n)\xi_M^{-b}(n) \\ -\gamma_M^{1/2}(n)\psi_0(n) & -\gamma_M^{1/2}(n)\psi_1(n) & \cdots & -\gamma_M^{1/2}(n)\psi_{M-1}(n) & \gamma_M^{1/2}(n) \end{bmatrix}$$

sendo  $\mathbf{S}_M^k(n)$  a mesma matriz de sistema apresentada na equação (2.161). Cabe observar que a matriz de transição de estados resulta triangular inferior, devido aos limites da somatória da equação (3.82).

Considerando (3.82), após algumas manipulações algébricas, conclui-se que o coeficiente  $dq1_i(n) = k_i^d(n)\xi_{i-1}^{b/2}(n)$  pode ser expresso como

$$dq1_i(n) = \sum_{j=1}^{i-1} s_{ij}^g(n)\lambda^{1/2}dq1_j(n-1) + s_{ii}^g(n)\lambda^{1/2}dq1_i(n-1) + \bar{b}_{i-1}(n)d(n), \quad (3.85)$$

sendo

$$s_{ij}^g(n) = -\bar{b}_{i-1}(n)\bar{\psi}_{j-1}(n) = -\left(\frac{\bar{b}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)}\right)\left(\frac{\bar{b}_{j-1}(n)}{\gamma_{j-1}^{1/2}(n)}\right)\left(\frac{\gamma_{i-1}^{1/2}(n)}{\gamma_j^{1/2}(n)}\right) \quad (3.86)$$

e

$$s_{ii}^g(n) = \frac{\xi_{i-1}^{b/2}(n)}{\lambda^{1/2}\xi_{i-1}^{b/2}(n-1)} - \bar{b}_{i-1}(n)\bar{\psi}_{i-1}(n).$$

Considerando (3.50) para ordem  $i - 1$ , a variável  $s_{ii}^g(n)$  pode ser expressa como

$$s_{ii}^g(n) = \lambda^{1/2} \frac{\xi_{i-1}^{b/2}(n-1)}{\xi_{i-1}^{b/2}(n)}. \quad (3.87)$$

De (3.81), o erro de estimação normalizado em ângulo  $\epsilon_i(n) = \gamma_i^{1/2}(n)\alpha_i(n)$  pode ser expresso como

$$\epsilon_i(n) = \gamma_i^{1/2}(n)(d(n) - \sum_{j=1}^i \bar{\psi}_{j-1}(n)\lambda^{1/2}dq1_j(n-1)), \quad (3.88)$$

sendo  $\bar{\psi}_{j-1}(n)$  o erro normalizado da predição regressiva *a priori*, conforme definido em (2.153). De (3.85) e de (3.88), o vetor  $\mathbf{dq1}(n)$  e o erro de estimação normalizado em ângulo  $\epsilon_M(n)$  podem ser atualizados no tempo através da seguinte equação

$$\begin{bmatrix} dq1_1(n) \\ dq1_2(n) \\ \vdots \\ dq1_M(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^g(n) \begin{bmatrix} \lambda^{1/2}dq1_1(n-1) \\ \lambda^{1/2}dq1_2(n-1) \\ \vdots \\ \lambda^{1/2}dq1_M(n-1) \\ d(n) \end{bmatrix},$$

onde

$$\mathbf{S}_M^g(n) = \begin{bmatrix} s_{11}^g(n) & 0 & \cdots & 0 & \bar{b}_0(n) \\ s_{21}^g(n) & s_{22}^g(n) & \cdots & 0 & \bar{b}_1(n) \\ s_{31}^g(n) & s_{32}^g(n) & \cdots & 0 & \bar{b}_2(n) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ s_{M1}^g(n) & s_{M2}^g(n) & \cdots & s_{MM}^g(n) & \bar{b}_M(n) \\ -\gamma_M^{1/2}(n)\bar{\psi}_0(n) & -\gamma_M^{1/2}(n)\bar{\psi}_1(n) & \cdots & -\gamma_M^{1/2}(n)\bar{\psi}_{M-1}(n) & \gamma_M^{1/2}(n) \end{bmatrix}$$

sendo  $\mathbf{S}_M^g(n)$  a mesma matriz de sistema apresentada na equação (2.146). De forma mais compacta pode-se escrever

$$\mathbf{S}_M^g(n) = \begin{bmatrix} \text{Diag} \left( \frac{\xi_{i-1}^{b/2}(n)}{\lambda^{1/2}\xi_{i-1}^{b/2}(n-1)} \right) - \text{Triang} \left( \bar{\mathbf{b}}_M(n)\bar{\boldsymbol{\psi}}_M^T(n) \right) & \bar{\mathbf{b}}_M(n) \\ -\gamma_M^{1/2}(n)\bar{\boldsymbol{\psi}}_M^T(n) & \gamma_M^{1/2}(n) \end{bmatrix},$$

a qual apresenta estrutura muito semelhante à da matriz de sistema do algoritmo RLS convencional descrita na equação (2.59), isto é

$$\mathbf{S}_M^\epsilon(n) = \begin{bmatrix} \mathbf{I} - \mathbf{g}_M(n)\mathbf{u}_M^T(n) & \mathbf{g}_M(n) \\ -\gamma_M^{1/2}(n)\mathbf{u}_M^T(n) & \gamma_M^{1/2}(n) \end{bmatrix}.$$

Por  $Triang(\cdot)$  indica-se uma matriz triangular inferior cujos elementos não nulos são idênticos aos da matriz argumento. Cabe lembrar que na tabela 2.2 é chamada a atenção que no domínio transformado, isto é, no domínio das variáveis do algoritmo QR-RLS, o papel do ganho de Kalman  $\mathbf{g}_M(n)$  é exercido pelo vetor  $\bar{\mathbf{b}}_M(n)$  dos erros de predição regressiva *a posteriori* normalizados e o papel do vetor do sinal de entrada  $\mathbf{u}_M(n)$  é exercido pelo vetor  $\bar{\boldsymbol{\psi}}_M(n)$  dos erros de predição regressiva *a priori* normalizados.

Como  $\eta_0(n) = u(n)$ , a equação (3.56) que calcula o erro *a priori* da predição progressiva  $\eta_i(n)$  pode ser expressa em termos de  $u(n)$ , isto é

$$\eta_i(n) = u(n) - \sum_{j=1}^i \psi_{j-1}(n-1)k_j^f(n-1).$$

Consequentemente, o erro normalizado em ângulo da predição progressiva

$$\epsilon_i^f(n) = \gamma_i^{1/2}(n-1)\eta_i(n)$$

pode ser expresso como

$$\epsilon_i^f(n) = \gamma_i^{1/2}(n-1)(u(n) - \sum_{j=1}^i \psi_{j-1}(n-1)k_j^f(n-1)). \quad (3.89)$$

Substituindo (3.89) em (3.72), resulta que

$$k_i^f(n) = \sum_{j=1}^{i-1} s_{ij}^k(n-1)k_j^f(n-1) + s_{ii}^k(n-1)k_i^f(n-1) + \frac{b_{i-1}(n-1)}{\xi_{i-1}^b(n-1)}u(n), \quad (3.90)$$

sendo que  $s_{ii}^k(n-1)$  e  $s_{ij}^k(n-1)$  são os mesmos definidos em (3.83) e (3.84), porém para o instante de tempo  $n-1$ . Assim, é possível concluir que a matriz que atualiza o vetor  $\mathbf{k}_{M-1}^f(n)$  e o erro  $\epsilon_{M-1}^f(n)$  é a mesma matriz  $\mathbf{S}_{M-1}^k(n-1)$  que atualizou no tempo o vetor  $\mathbf{k}_{M-1}^d(n-1)$  e o erro  $\epsilon_{M-1}(n-1)$ .

Considerando (3.90), após algumas manipulações algébricas conclui-se que o coeficiente  $dq1_i^f(n) = k_i^f(n)\xi_{i-1}^{b/2}(n-1)$  pode ser expresso como

$$\begin{aligned} dq1_i^f(n) &= \sum_{j=1}^{i-1} s_{ij}^g(n-1)\lambda^{1/2}dq1_j^f(n-1) + \\ &+ s_{ii}^g(n-1)\lambda^{1/2}dq1_i^f(n-1) + \bar{b}_{i-1}(n-1)u(n). \end{aligned} \quad (3.91)$$

De acordo com (3.89), o erro da predição progressiva normalizado em ângulo

$$\epsilon_i^f(n) = \gamma_i^{1/2}(n-1)\eta_i(n)$$



pode ser expresso como

$$\epsilon_i^f(n) = \gamma_i^{1/2}(n-1)(u(n) - \sum_{j=1}^i \bar{\psi}_{j-1}(n-1)\lambda^{1/2}dq1_j^f(n-1)). \quad (3.92)$$

Cabe observar que  $s_{ii}^g(n-1)$  e  $s_{ij}^g(n-1)$  são os mesmos elementos definidos em (3.87) e (3.86) porém para o instante de tempo  $n-1$ . Consequentemente, a matriz de sistema que atualiza no tempo o vetor  $\mathbf{dq1}^f(n)$  e o erro  $\epsilon_{M-1}^f(n)$  é a mesma matriz de sistema  $\mathbf{S}_{M-1}^g(n-1)$  que atualizou no tempo o vetor  $\mathbf{dq1}_{M-1}(n-1)$  e o erro  $\epsilon_{M-1}(n-1)$ . O conjunto de equações (3.83) e (3.84) e o conjunto de equações (3.87) e (3.86) complementam respectivamente, as interpretações dos elementos das matrizes de sistema  $\mathbf{S}_M^k(n)$  e  $\mathbf{S}_M^g(n)$  efetuadas no capítulo 2. A interpretação dos elementos da matriz de transição de estados em função dos erros de predição regressiva é aparentemente inédita. O usual é interpretar os elementos de  $\mathbf{S}_M^g(n)$  em termos dos ângulos das rotações de Givens que definem a matriz unitária  $\mathbf{Q}_M(n)$  da seção 2.6 do capítulo 2. Cabe lembrar que os ângulos das rotações de Givens definem a estrutura de McWhirter apresentada no apêndice B. Na próxima seção será demonstrado como cada um dos elementos  $s_{ii}^g(n)$  e  $s_{ij}^g(n)$  estão relacionados com os ângulos das rotações de Givens.

### 3.6 Atualizações com rotações de Givens

Dado um vetor de dois elementos, é possível calcular a sua representação polar, isto é, a sua norma e o ângulo correspondente através de uma redução de Givens ('Givens annihilation'). Este fato é usado a seguir, considerando os vetores de dois elementos e norma unitária, definidos a partir das equações (3.30), (3.50) e (3.51). No caso da equação (3.30), é possível obter a energia do erro de predição e o ângulo associado a correlação cruzada  $\bar{\Delta}_{i-1}(\cdot)$ . Os ângulos obtidos são usados nas estruturas em treliça normalizadas. No caso das equações (3.50) e (3.51), é possível obter a energia do erro de predição ou o fator de conversão e o ângulo associado ao erro da predição normalizado. Os ângulos obtidos são usados nas estruturas de McWhirter que aparecem nos algoritmos QR-RLS.

### 1) Coeficientes de reflexão da estrutura em treliça normalizada:

A correlação cruzada  $\bar{\Delta}_{i-1}(\cdot)$  representa o coeficiente de reflexão das estruturas em treliça com erros de predição normalizados definidas em (3.55) e (3.60). Associa-se a seguir  $\bar{\Delta}_{i-1}(\cdot)$  ao ângulo de rotação e a norma do vetor que representa a atualização na ordem da energia de predição. A equação (3.30), reescrita como

$$\frac{\xi_i^f(n)}{\xi_{i-1}^f(n)} + |\bar{\Delta}_{i-1}(n)|^2 = 1, \quad (3.93)$$

define o vetor  $[\xi_i^{f/2}(n) \xi_{i-1}^{-f/2}(n) \bar{\Delta}_{i-1}(n)]$  de norma unitária e ângulo

$$\cos \phi_i(n) = \frac{\xi_i^{f/2}(n)}{\xi_{i-1}^{f/2}(n)} = \sqrt{1 - |\bar{\Delta}_{i-1}(n)|^2} \quad (3.94)$$

e

$$\sin \phi_i(n) = \bar{\Delta}_{i-1}(n).$$

De (3.37) tem-se

$$\sin \phi_i(n) = \frac{dq1_i^f(n)}{\xi_{i-1}^{f/2}(n)}. \quad (3.95)$$

Assim, de (3.93) a (3.95) resulta

$$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \cdot \begin{bmatrix} \xi_i^{f/2}(n) \\ dq1_i^f(n) \end{bmatrix}, \quad (3.96)$$

onde  $\xi_{i-1}^{f/2}(n)$  representa a norma do vetor  $[\xi_i^{f/2}(n) \ dqq1_i^f(n)]$ . Usando as representações para  $\bar{\Delta}_{i-1}(n)$  em função de  $\phi_i(n)$ , as equações (3.55) e (3.60) podem ser reescritas como

$$\begin{bmatrix} \bar{f}_{i-1}(n) \\ \bar{b}_i(n) \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \cdot \begin{bmatrix} \bar{f}_i(n) \\ \bar{b}_{i-1}(n-1) \end{bmatrix} \quad (3.97)$$

e

$$\begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n-1) & \sin \phi_i(n-1) \\ -\sin \phi_i(n-1) & \cos \phi_i(n-1) \end{bmatrix} \cdot \begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_{i-1}(n-1) \end{bmatrix}. \quad (3.98)$$

Conhecendo então  $dq1_i^f(n)$  para  $1 \leq i \leq M$  e  $\xi_M^{f/2}(n)$  e usando (3.96), é possível calcular as energias de erro da predição progressiva de ordem  $i-1$  para  $1 \leq i \leq M$  e os coeficientes de reflexão das estruturas em treliça com erros de predição normalizados.

De (3.30) tem-se ainda

$$\cos \phi_i(n) = \frac{\xi_i^{b/2}(n)}{\xi_{i-1}^{b/2}(n-1)} = \sqrt{1 - |\bar{\Delta}_{i-1}(n)|^2} \quad (3.99)$$

e considerando (3.38)

$$\sin \phi_i(n) = \frac{\pi_i^b(n)}{\xi_{i-1}^{b/2}(n-1)} = \bar{\Delta}_{i-1}(n). \quad (3.100)$$

De (3.99) e (3.100) resulta a seguinte expressão

$$\begin{bmatrix} \xi_{i-1}^{b/2}(n-1) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \xi_i^{b/2}(n) \\ \pi_i^b(n) \end{bmatrix}, \quad (3.101)$$

a qual representa uma outra forma de obter os ângulos  $\phi_i(n)$ . Entretanto, esta expressão não é conveniente devido ao deslocamento de tempo envolvido na atualização da energia do erro da predição regressiva.

## 2) Coeficientes da estrutura de McWhirter para os casos de estimação e predição progressiva:

A equação (3.50) reescrita para a ordem  $i$  como

$$\frac{\gamma_i(n)}{\gamma_{i-1}(n)} + \frac{|b_{i-1}(n)|^2}{\gamma_{i-1}(n)\xi_{i-1}^b(n)} = 1,$$

define o vetor

$$\begin{bmatrix} \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n)} & \frac{b_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)\xi_{i-1}^{b/2}(n)} \end{bmatrix}$$

de norma unitária e ângulo  $\theta_i^b(n)$  especificado por

$$\cos \theta_i^b(n) = \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n)} \quad (3.102)$$

e

$$\sin \theta_i^b(n) = \frac{b_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)\xi_{i-1}^{b/2}(n)}.$$

De (3.50) observa-se que  $\cos \theta_i^b(n)$  pode também ser obtido através das energias de erro da predição regressiva, isto é

$$\cos \theta_i^b(n) = \frac{\lambda^{1/2}\xi_{i-1}^{b/2}(n-1)}{\xi_{i-1}^{b/2}(n)}. \quad (3.103)$$

Usando as relações de erros da predição regressiva conforme (3.63) e lembrando que  $b_{i-1}(n) = \bar{b}_{i-1}(n)\xi_{i-1}^{b/2}(n)$ , é possível representar  $\sin \theta_i^b(n)$  como

$$\sin \theta_i^b(n) = \frac{\bar{b}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n)}, \quad (3.104)$$

ou

$$\sin \theta_i^b(n) = \gamma_i^{1/2}(n)\bar{\psi}_{i-1}(n) \quad (3.105)$$

ou ainda, como

$$\sin \theta_i^b(n) = \frac{\epsilon_{i-1}^b(n)}{\xi_{i-1}^{b/2}(n)}. \quad (3.106)$$

Comparando (3.102) e (B.15) conclui-se que os ângulos  $\theta_i^b(n)$ , para  $1 \leq i \leq M$ , são exatamente os ângulos que atualizam a matriz triangular  $\mathbf{R}_M(n)$  de forma recorrente no tempo, sendo também os ângulos que definem a matriz de sistema  $\mathbf{S}_M^g(n)$ . De fato, comparando (3.87) e (3.103) conclui-se que

$$s_{ii}^g(n) = \cos \theta_i^b(n). \quad (3.107)$$

Usando  $\bar{b}_{i-1}(n) = \sin \theta_i^b(n)\gamma_{i-1}^{1/2}(n)$  e  $\bar{\psi}_{j-1}(n) = \sin \theta_j^b(n)\gamma_j^{-1/2}(n)$  em (3.86), resulta

$$s_{ij}^g(n) = -\sin \theta_i^b(n) \sin \theta_j^b(n) \frac{\gamma_{i-1}^{1/2}(n)}{\gamma_j^{1/2}(n)}. \quad (3.108)$$

Usando (3.104), (3.103) e (3.102), as equações (3.74) e (3.75) podem ser expressas em termos dos ângulos  $\theta_i^b(n)$  como

$$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \cdot \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}. \quad (3.109)$$

Esta equação é a mesma apresentada no apêndice B, equação (B.13) e define a estrutura de McWhirter no caso da estimação. De forma análoga, usando (3.104), (3.103) e (3.102), porém para o instante  $n-1$ , as equações (3.76) e (3.77) podem ser expressas em termos dos ângulos  $\theta_i^b(n-1)$  como

$$\begin{bmatrix} dq1_i^f(n) \\ \epsilon_i^f(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n-1) & \sin \theta_i^b(n-1) \\ -\sin \theta_i^b(n-1) & \cos \theta_i^b(n-1) \end{bmatrix} \cdot \begin{bmatrix} \lambda^{1/2} dq1_i^f(n-1) \\ \epsilon_{i-1}^f(n) \end{bmatrix}, \quad (3.110)$$

caracterizando a estrutura de McWhirter no caso da predição progressiva. Cabe notar que esta equação calcula os  $dq1_i^f(n)$  e que estes podem ser usados em (3.96) para obter os coeficientes de reflexão da estrutura em treliça normalizada.

As equações (3.104) a (3.106) representam três formas diferentes de calcular os  $\sin \theta_i^b(n)$ . Assim, os ângulos  $\theta_i^b(n)$  podem ser obtidos reescrevendo (3.104) e (3.102) como

$$\begin{bmatrix} \gamma_{i-1}^{1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \cdot \begin{bmatrix} \gamma_i^{1/2}(n) \\ \bar{b}_{i-1}(n) \end{bmatrix}, \quad (3.111)$$

onde  $\gamma_{i-1}^{1/2}(n)$  representa a norma do vetor  $[\gamma_i^{1/2}(n) \ \bar{b}_{i-1}(n)]$ . Conhecendo então  $\bar{b}_{i-1}(n)$  para  $1 \leq i \leq M$ , e  $\gamma_M^{1/2}(n)$ , é possível calcular os parâmetros  $\gamma_i^{1/2}(n)$  e os ângulos  $\theta_i^b(n)$ , para  $1 \leq i \leq M$  em ordem decrescente, isto é, de  $i = M$  até  $i = 1$ .

De (3.105) e (3.102) resulta

$$\begin{bmatrix} \gamma_i^{-1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \cdot \begin{bmatrix} \gamma_{i-1}^{-1/2}(n) \\ \bar{\psi}_{i-1}(n) \end{bmatrix}, \quad (3.112)$$

onde  $\gamma_i^{-1/2}(n)$  representa a norma do vetor  $[\gamma_{i-1}^{-1/2}(n) \ \bar{\psi}_{i-1}(n)]$ . Conhecendo  $\bar{\psi}_{i-1}(n)$  para  $1 \leq i \leq M$  e  $\gamma_0^{-1/2}(n) = 1$ , é possível calcular  $\gamma_i^{-1/2}(n)$  e os ângulos  $\theta_i^b(n)$  para  $1 \leq i \leq M$ , em ordem crescente, isto é, de  $i = 1$  até  $i = M$ . Neste caso porém, é conveniente usar as expressões

$$\cos \theta_i^b(n) = \frac{\gamma_{i-1}^{-1/2}(n)}{\gamma_i^{-1/2}(n)} \quad (3.113)$$

e

$$\sin \theta_i^b(n) = \frac{\bar{\psi}_{i-1}(n)}{\gamma_i^{-1/2}(n)}. \quad (3.114)$$

A terceira forma de obter o ângulo  $\theta_i^b(n)$ , resulta de (3.106) e (3.103), reecritas como

$$\begin{bmatrix} \xi_{i-1}^{b/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \cdot \begin{bmatrix} \lambda^{1/2} \xi_{i-1}^{b/2}(n-1) \\ \epsilon_{i-1}^b(n) \end{bmatrix}, \quad (3.115)$$

onde  $\xi_{i-1}^{b/2}(n)$  representa a norma do vetor  $[\lambda^{1/2} \xi_{i-1}^{b/2}(n-1) \ \epsilon_{i-1}^b(n)]$ . Conhecendo os erros da predição regressiva normalizados em ângulo  $\epsilon_{i-1}^b(n)$  e  $\xi_{i-1}^{b/2}(n-1)$  para  $1 \leq i \leq M$ , é possível atualizar no tempo  $\xi_{i-1}^{b/2}(n)$  e calcular os ângulos  $\theta_i^b(n)$ , para  $1 \leq i \leq M$ .

### 3) Coeficientes da estrutura de McWhirter no caso da predição regressiva:

De forma análoga, assim como foram obtidos os ângulos da estrutura de McWhirter para os casos de estimação e predição progressiva, é possível obter os

ângulos da estrutura de McWhirter para o caso da predição regressiva. A equação (3.51) reescrita como

$$\frac{\gamma_i(n)}{\gamma_{i-1}(n-1)} + \frac{|f_{i-1}(n)|^2}{\gamma_{i-1}(n-1)\xi_{i-1}^f(n)} = 1,$$

define o vetor

$$\begin{bmatrix} \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} & \frac{f_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)\xi_{i-1}^{f/2}(n)} \end{bmatrix}$$

de norma unitária e ângulo  $\theta_i^f(n)$  especificado por

$$\cos \theta_i^f(n) = \frac{\gamma_i^{1/2}(n)}{\gamma_{i-1}^{1/2}(n-1)} \quad (3.116)$$

e

$$\sin \theta_i^f(n) = \frac{f_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)\xi_{i-1}^{f/2}(n)}.$$

De (3.51) observa-se que  $\cos \theta_i^f(n)$  pode também ser obtido através das energias de erro da predição progressiva, isto é

$$\cos \theta_i^f(n) = \frac{\lambda^{1/2}\xi_{i-1}^{f/2}(n-1)}{\xi_{i-1}^{f/2}(n)}. \quad (3.117)$$

Usando as relações de erros da predição progressiva conforme (3.66), e lembrando que  $f_{i-1}(n) = \bar{f}_{i-1}(n)\xi_{i-1}^{f/2}(n)$ , é possível representar  $\sin \theta_i^f(n)$  como

$$\sin \theta_i^f(n) = \frac{\bar{f}_{i-1}(n)}{\gamma_{i-1}^{1/2}(n-1)}, \quad (3.118)$$

ou

$$\sin \theta_i^f(n) = \frac{\bar{\eta}_{i-1}(n)}{\gamma_i^{-1/2}(n)}, \quad (3.119)$$

ou ainda, como

$$\sin \theta_i^f(n) = \frac{\epsilon_{i-1}^f(n)}{\xi_{i-1}^{f/2}(n)}. \quad (3.120)$$

Reconhecendo (3.116), (3.117) e (3.118) em (3.78) e (3.79) resulta

$$\begin{bmatrix} \pi_i^b(n) \\ \epsilon_i^b(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \cdot \begin{bmatrix} \lambda^{1/2}\pi_i^b(n-1) \\ \epsilon_{i-1}^b(n-1) \end{bmatrix}, \quad (3.121)$$

isto é, os ângulos  $\theta_i^f(n)$  fazem a atualização no tempo de  $\pi_i^b(n)$  e a atualização no tempo e na ordem de  $\epsilon_i^b(n)$ . Estas equações caracterizam uma estrutura muito semelhante a estrutura de McWhirter (ver figura B.2) para o caso da predição regressiva.

A diferença consiste simplesmente na existência de um atraso extra entre cada estágio da estrutura de McWhirter.

As equações (3.118) a (3.120) representam três formas diferentes de calcular  $\sin \theta_i^f(n)$ . Assim, uma forma possível para obter o ângulo  $\theta_i^f(n)$  resulta de (3.117) e (3.120) reescritas como

$$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \cdot \begin{bmatrix} \lambda^{1/2} \xi_{i-1}^{f/2}(n-1) \\ \epsilon_{i-1}^f(n) \end{bmatrix}, \quad (3.122)$$

onde  $\xi_{i-1}^{f/2}(n)$  representa a norma do vetor  $[\lambda^{1/2} \xi_{i-1}^{f/2}(n-1) \quad \epsilon_{i-1}^f(n)]$ . Conhecendo os erros normalizados em ângulo da predição progressiva  $\epsilon_{i-1}^f(n)$  e  $\lambda^{1/2} \xi_{i-1}^{f/2}(n-1)$ , para  $1 \leq i \leq M$ , é possível obter  $\xi_{i-1}^{f/2}(n)$  e os ângulos  $\theta_i^f(n)$  em ordem crescente.

De (3.116) e (3.119) resulta

$$\begin{bmatrix} \gamma_i^{-1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \cdot \begin{bmatrix} \gamma_{i-1}^{-1/2}(n-1) \\ \bar{\eta}_{i-1}(n) \end{bmatrix}, \quad (3.123)$$

onde  $\gamma_i^{-1/2}(n)$  representa a norma do vetor  $[\gamma_{i-1}^{-1/2}(n-1) \quad \bar{\eta}_{i-1}(n)]$ . Neste caso é conveniente calcular

$$\cos \theta_i^f(n) = \frac{\gamma_{i-1}^{-1/2}(n-1)}{\gamma_i^{-1/2}(n)}$$

e

$$\sin \theta_i^f(n) = \frac{\bar{\eta}_{i-1}(n)}{\gamma_i^{-1/2}(n)}.$$

Conhecendo então  $\gamma_{i-1}^{-1/2}(n-1)$  e  $\bar{\eta}_{i-1}(n)$ , é possível calcular os parâmetros  $\gamma_i^{-1/2}(n)$  e os ângulos  $\theta_i^f(n)$  em ordem crescente para  $1 \leq i \leq M$ .

Uma terceira forma de obter os ângulos  $\theta_i^f(n)$  resulta de (3.116) e (3.118), que podem ser reescritas como

$$\begin{bmatrix} \gamma_{i-1}^{1/2}(n-1) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \cdot \begin{bmatrix} \gamma_i^{1/2}(n) \\ \bar{f}_{i-1}(n) \end{bmatrix}. \quad (3.124)$$

Entretanto, esta forma não é muito conveniente devido ao deslocamento de tempo envolvido para atualizar a raiz quadrada do fator de conversão.

### 3.7 Algoritmos recorrentes no tempo e na ordem

Nos algoritmos RLS rápidos recursivos na ordem não normalizados, os erros de estimação são atualizados através da equação (3.25), isto é

$$e_i(n) = e_{i-1}(n) - b_{i-1}(n)k_i^b(n),$$

ou então através de (3.57), isto é

$$\alpha_i(n) = \alpha_{i-1}(n) - \psi_{i-1}(n)k_i^b(n-1).$$

Em ambos os casos, além de conhecer o erro de estimação da ordem inferior, é necessário conhecer o coeficiente de regressão  $k_i^d(\cdot)$  e o erro da predição regressiva. Os coeficientes de regressão  $k_i^d(\cdot)$ , são obtidos a partir das correlações cruzadas dos erros da predição regressiva e estimação e das energias da predição regressiva. Por sua vez, os erros de predição regressiva são obtidos através de uma estrutura em treliça. Os coeficientes da estrutura em treliça  $k_i^f(\cdot)$  e  $k_i^b(\cdot)$ , são obtidos a partir das correlações cruzadas dos erros da predição e das energias da predição.

#### Algoritmo RLS rápido recursivo na ordem com erros a priori

Um exemplo de algoritmo RLS rápido recursivo na ordem não normalizado, é a versão apresentada por Ling em [12], que é baseada nos erros de estimação e predição *a priori*. Neste algoritmo, apresentado na tabela 3.1, o erro de estimação *a priori* é obtido de (3.57), e os coeficientes de regressão são calculados segundo (3.71). Os erros de predição regressiva, necessários para este cálculo, são obtidos com auxílio do filtro em treliça definido pela equação (3.56), sendo que os correspondentes coeficientes da estrutura em treliça são calculados segundo (3.72) e (3.73). As energias dos erros de predição progressiva e regressiva, necessárias para o cálculo dos coeficientes de regressão e de reflexão, são obtidas de (3.46) e (3.47) reescritas para ordem  $i-1$  e em termos dos erros da predição *a priori*. Finalmente, o fator de conversão, necessário para o cálculo das energias de erro de predição, dos coeficientes do filtro em treliça e dos coeficientes de regressão, é obtido de (A.52). Como no cálculo dos coeficientes de regressão e reflexão existe um o mecanismo de realimentação de erro, esta versão é conhecida como algoritmo LSL ('Least Square Lattice') com realimentação de erro. Considerações sobre a inicialização deste algoritmo, assim como três outras versões



de algoritmos LSL, podem ser encontradas em [14] e em [38, cap. 17]. De qualquer forma, dentre todas as versões de algoritmos LSL, o da tabela 3.1 parece ser o mais favorável do ponto de vista numérico, razão pela qual ele foi aqui apresentado.

Tabela 3.1

Algoritmo LSL com erros <i>a priori</i>	
<b>Seção de predição:</b> ( $\psi_0(n) = \eta_0(n) = u(n)$ )	
For $i = 1, 2, \dots, M$ do	
Estrutura em treliça:	
$\begin{bmatrix} \eta_i(n) \\ \psi_i(n) \end{bmatrix} = \begin{bmatrix} 1 & -k_i^f(n-1) \\ -k_i^b(n-1) & 1 \end{bmatrix} \begin{bmatrix} \eta_{i-1}(n) \\ \psi_{i-1}(n-1) \end{bmatrix}$	
Energias de erro de predição	
$\xi_{i-1}^f(n) = \lambda \xi_{i-1}^f(n-1) + \gamma_{i-1}(n-1)  \eta_{i-1}(n) ^2$	
$\xi_{i-1}^b(n) = \lambda \xi_{i-1}^b(n-1) + \gamma_{i-1}(n)  \psi_{i-1}(n) ^2$	
Coeficientes de reflexão	
$k_i^b(n) = k_i^b(n-1) + \frac{\eta_{i-1}(n) \gamma_{i-1}(n-1)}{\xi_{i-1}^f(n)} \psi_i(n)$	
$k_i^f(n) = k_i^f(n-1) + \frac{\psi_{i-1}(n-1) \gamma_{i-1}(n-1)}{\xi_{i-1}^b(n-1)} \eta_i(n)$	
Fator de conversão	
$\gamma_i(n) = \gamma_{i-1}(n) - \gamma_{i-1}^2(n) \frac{ \psi_{i-1}(n) ^2}{\xi_{i-1}^b(n)}$	
<b>Estimação conjunta</b> ( $\alpha_0(n) = d(n)$ )	
For $i = 1, 2, \dots, M$ do	
$\alpha_i(n) = \alpha_{i-1}(n) - \psi_{i-1}(n) k_i^d(n-1)$	
$k_i^d(n) = k_i^d(n-1) + \frac{\psi_{i-1}(n) \gamma_{i-1}(n)}{\xi_{i-1}^b(n)} \alpha_i(n)$	
<b>Erro de estimação <i>a posteriori</i></b>	
$e_M(n) = \gamma_M(n) \alpha_M(n)$	

Nos algoritmos QR-RLS rápidos recursivos na ordem baseados em rotações, os erros de estimação e os coeficientes de regressão normalizados são atualizados na ordem através da equação (3.109), isto é

$$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}.$$

Neste caso, além de conhecer os erros de estimação normalizados em ângulo das ordens inferiores e os coeficientes de regressão normalizados do tempo anterior, é necessário

conhecer também os ângulos  $\theta_i^b(\cdot)$ . A atualização dos ângulos  $\theta_i^b(\cdot)$  é feita através dos erros de predição regressiva normalizados. As três diferentes versões de algoritmos QR-RLS rápidos apresentados a seguir, se diferenciam pela forma de obter os erros da predição regressiva normalizados.

### Algoritmo QR-RLS rápido recursivo na ordem com erros normalizados em ângulo

O algoritmo QR-RLS rápido com erro normalizado em ângulo é apresentado em [7] e [38, cap. 18]. Esta versão está representada através dos passos 1 a 7 na tabela 3.2. No passo 6 a equação (3.109) faz a atualização na ordem do erro de estimação normalizado em ângulo. Os ângulos  $\theta_i^b(n)$  são calculados no passo 4 usando (3.115).

Tabela 3.2

Algoritmo QR-RSL com erros normalizados em ângulo	
Predição progressiva ( $\epsilon_0^f(n) = u(n)$ ) For $i = 1, 2, \dots, M$ do	
<b>Passo 1:</b>	$\begin{bmatrix} dq1_i^f(n) \\ \epsilon_i^f(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n-1) & \sin \theta_i^b(n-1) \\ -\sin \theta_i^b(n-1) & \cos \theta_i^b(n-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i^f(n-1) \\ \epsilon_{i-1}^f(n) \end{bmatrix}$
<b>Passo 2:</b>	$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_{i-1}^{f/2}(n-1) \\ \epsilon_{i-1}^f(n) \end{bmatrix}$
Predição regressiva ( $\epsilon_0^b(n) = u(n)$ ) For $i = 1, 2, \dots, M$ do	
<b>Passo 3:</b>	$\begin{bmatrix} \pi_i^b(n) \\ \epsilon_i^b(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^f(n) & \sin \theta_i^f(n) \\ -\sin \theta_i^f(n) & \cos \theta_i^f(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \pi_i^b(n-1) \\ \epsilon_{i-1}^b(n-1) \end{bmatrix}$
<b>Passo 4:</b>	$\begin{bmatrix} \xi_{i-1}^{b/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_{i-1}^{b/2}(n-1) \\ \epsilon_{i-1}^b(n) \end{bmatrix}$
Fator de conversão	
<b>Passo 5:</b>	$\gamma_i^{1/2}(n) = \gamma_{i-1}^{1/2}(n) \cos \theta_i^b(n)$
Estimação conjunta ( $\epsilon_0(n) = d(n)$ ) For $i = 1, 2, \dots, M$ do	
<b>Passo 6:</b>	$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}$
Erro de estimação <i>a posteriori</i>	
<b>Passo 7:</b>	$e_M(n) = \gamma_M^{1/2}(n) \epsilon_M(n)$

No passo 3 os erros da predição regressiva normalizados em ângulo necessários para obter  $\theta_i^b(n)$  são calculados a partir os ângulos  $\theta_i^f(\cdot)$  usando (3.121); tais ângulos são calculados no passo 2 usando (3.122). No passo 1 os erros da predição progressiva normalizados em ângulo necessários para obter os ângulos  $\theta_i^f(\cdot)$ , são calculados a partir de  $\theta_i^b(n-1)$  usando (3.110). A estrutura em treliça é obtida indiretamente através dos erros de predição normalizados em ângulo calculados nos passos 1 e 3. Nos passos 1, 3 e 6 é utilizada a estrutura de McWhirter respectivamente, para os casos da predição progressiva e regressiva e estimação. Nos passos 2 e 4 é utilizada a redução de Givens ('Givens annihilation'), processo que é chamado de redução circular no caso de processamento utilizando CORDIC.

### Algoritmo QR-LSL com erros a posteriori

O algoritmo QR-LSL *a posteriori* foi introduzido por Regalia em [33]. Esta versão está representada através dos passos 1 a 7 na tabela 3.3. No passo 6 a equação (3.109) faz a atualização na ordem do erro de estimação normalizado em ângulo como no algoritmo da tabela 3.2. Neste caso, porém o ângulo  $\theta_i^b(n)$  é obtido no passo 5b usando a equação (3.111), e  $\gamma_M^{1/2}(n)$  é calculado no passo 5a usando (2.155). Por sua vez, os erros da predição regressiva são obtidos no passo 4b através da estrutura em treliça definida pela equação (3.97), sendo que  $\bar{f}_M(n)$  é calculada no passo 4a usando (3.118) para ordem  $M$ . Os coeficientes de reflexão normalizados são calculados no passo 3 usando (3.96), sendo que  $\xi_M^{f/2}(n)$  é obtida no passo 2 usando (3.122) para ordem  $M$ . Os  $dq1_i^f(n)$  necessários para calcular os ângulos  $\phi_i(n)$  no passo 3 e o erro da predição progressiva normalizado em ângulo de ordem  $M$ , são obtidos no passo 1 usando (3.110). Cabe observar, que nos passos 1 e 6 é utilizada a estrutura de McWhirter, respectivamente para os casos de predição progressiva e de estimação. No passo 4b é utilizada a estrutura em treliça com erros *a posteriori* normalizados. Nos passos 2, 3 e 5b é utilizada a redução de Givens para o cálculo de ângulos. Considerações sobre a estabilidade numérica e a inicialização deste algoritmo são apresentadas na seção 4.2 do próximo capítulo.

Tabela 3.3

Algoritmo QR-LSL com erros <i>a posteriori</i>	
<b>Passo 1:</b> Predição progressiva ( $\varepsilon_0^f(n) = u(n)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i^f(n) \\ \varepsilon_i^f(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n-1) & \sin \theta_i^b(n-1) \\ -\sin \theta_i^b(n-1) & \cos \theta_i^b(n-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i^f(n-1) \\ \varepsilon_{i-1}^f(n) \end{bmatrix}$	
<b>Passo 2:</b> Energia da predição progressiva	
$\begin{bmatrix} \xi_M^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_{M+1}^f(n) & \sin \theta_{M+1}^f(n) \\ -\sin \theta_{M+1}^f(n) & \cos \theta_{M+1}^f(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_M^{f/2}(n-1) \\ \varepsilon_M^f(n) \end{bmatrix}$	
<b>Passo 3:</b> Coeficiente de reflexão normalizado For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \xi_i^{f/2}(n) \\ dq1_i^f(n) \end{bmatrix}$	
<b>Passo 4:</b> Filtro em treliça normalizado	
a) $\bar{f}_M(n) = \sin \theta_{M+1}^f(n) \gamma_M^{1/2}(n-1)$	
b) For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \bar{f}_{i-1}(n) \\ \bar{b}_i(n) \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \bar{f}_i(n) \\ \bar{b}_{i-1}(n-1) \end{bmatrix}$	
com $\bar{b}_0(n) = f_0(n)$ .	
<b>Passo 5:</b> Ângulos para a estimação conjunta	
a) $\gamma_M^{1/2}(n) = \sqrt{1 - \sum_{i=1}^M  \bar{b}_{i-1}(n) ^2}$	
b) For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \gamma_{i-1}^{1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \gamma_i^{1/2}(n) \\ \bar{b}_{i-1}(n) \end{bmatrix}$	
<b>Passo 6:</b> Estimação conjunta ( $\varepsilon_0(n) = d(n)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i(n) \\ \varepsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \varepsilon_{i-1}(n) \end{bmatrix}$	
<b>Passo 7:</b> Erro de estimação <i>a posteriori</i> $e_M(n) = \varepsilon_M(n) \gamma_M^{1/2}(n)$	

### 3.8 Um novo algoritmo QR-LSL

Uma outra versão de algoritmo QR-LSL pode ser obtida calculando o ângulo  $\theta_i^b(n)$  através dos erros da predição regressiva *a priori* normalizados  $\bar{\psi}_{i-1}(n)$ , sendo que neste caso, é conveniente calcular  $\bar{\psi}_{i-1}(n)$  através da estrutura em treliça definida pelos erros da predição *a priori* normalizados. O algoritmo resultante está represen-

tado através dos passos 1 a 7 na tabela 3.4. Os passos 1, 2, 3 e 6 desta tabela são idênticos aos passos 1, 2, 3 e 6 da tabela 3.3. No passo 5 o ângulo  $\theta_i^b(n)$  é obtido em ordem crescente usando (3.112). No passo 4b os erros da predição regressiva *a priori* normalizados são calculados usando (3.98), sendo que  $\bar{\eta}_M(n)$  é calculada no passo 4a usando (3.119) para ordem  $M$ .

Tabela 3.4

Algoritmo QR-LSL com erros <i>a priori</i>	
<b>Passo 1:</b> Predição progressiva ( $\varepsilon_0^f(n) = u(n)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i^f(n) \\ \varepsilon_i^f(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n-1) & \sin \theta_i^b(n-1) \\ -\sin \theta_i^b(n-1) & \cos \theta_i^b(n-1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i^f(n-1) \\ \varepsilon_{i-1}^f(n) \end{bmatrix}$	
<b>Passo 2:</b> Energia do erro da predição progressiva	
$\begin{bmatrix} \xi_M^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_{M+1}^f(n) & \sin \theta_{M+1}^f(n) \\ -\sin \theta_{M+1}^f(n) & \cos \theta_{M+1}^f(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_M^{f/2}(n-1) \\ \varepsilon_M^f(n) \end{bmatrix}$	
<b>Passo 3:</b> Coeficiente de reflexão normalizado For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \xi_i^{f/2}(n) \\ dq1_i^f(n) \end{bmatrix}$	
<b>Passo 4:</b> Filtro em treliça normalizado	
a) $\bar{\eta}_M(n) = \gamma_M^{-1/2}(n-1) \frac{\sin \theta_{M+1}^f(n)}{\cos \theta_{M+1}^f(n)}$	
b) For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \bar{\eta}_{i-1}(n) \\ \bar{\psi}_i(n) \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n-1) & \sin \phi_i(n-1) \\ -\sin \phi_i(n-1) & \cos \phi_i(n-1) \end{bmatrix} \begin{bmatrix} \bar{\eta}_i(n) \\ \bar{\psi}_{i-1}(n-1) \end{bmatrix}$	
com $\bar{\psi}_0(n) = \bar{\eta}_0(n)$ .	
<b>Passo 5:</b> Ângulos para a estimação conjunta ( $\gamma_0^{-1/2}(n) = 1$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} \gamma_i^{-1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \gamma_{i-1}^{-1/2}(n) \\ \bar{\psi}_{i-1}(n) \end{bmatrix}$	
<b>Passo 6:</b> Estimação conjunta ( $\varepsilon_0(n) = d(n)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i(n) \\ \varepsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \varepsilon_{i-1}(n) \end{bmatrix}$	
<b>Passo 7:</b> Erro de estimação <i>a posteriori</i> $e_M(n) = \varepsilon_M(n) / \gamma_M^{-1/2}(n)$	

Cabe observar que  $\gamma_{i-1}^{-1/2}(n)$ , isto é, o inverso da raiz quadrada do fator de conversão, é atualizado e não  $\gamma_{i-1}^{1/2}(n)$  como acontece no algoritmo da tabela 3.3. A atualização de

$\gamma_{i-1}^{-1/2}(n)$  é feita simultaneamente com o cálculo do ângulo  $\theta_i^b(n)$ , o que é feito em ordem ascendente e não em ordem descendente como o algoritmo da tabela 3.3. No algoritmo da tabela 3.3 o fator  $\gamma_{i-1}^{1/2}(n)$  é obtido com uma expressão extra. Considerações sobre a estabilidade numérica e a inicialização deste algoritmo são apresentadas na seção 4.3 do próximo capítulo.

A tabela 3.5 mostra a complexidade computacional dos algoritmos apresentados nas tabelas 3.1 a 3.4. A versão da tabela 3.1 tem o menor número de operações, entretanto, a versão da tabela 3.4 tem um paralelismo inerente, que pode ser explorado em implementações rápidas, como será mostrado a seguir. Cabe ainda observar que o algoritmo da tabela 3.1 é o único dos quatro algoritmos considerados que não necessita da operação raiz quadrada, o que do ponto de vista de implementação em computador de uso geral, é sem dúvida uma grande vantagem. Entretanto, como será visto no capítulo 4, os dois algoritmos QR-LSL, tabelas 3.3 e 3.4, apresentam vantagem quanto a estabilidade numérica.

Tabela 3.5

Complexidade computacional				
Tabela	×	÷	√	+ e -
3.1	$17M + 1$	$4M$	-	$9M$
3.2	$22M + 1$	$4M$	$2M$	$8M$
3.3	$19M + 5$	$4M + 2$	$2M + 2$	$9M + 2$
3.4	$18M + 4$	$4M + 4$	$2M + 1$	$8M + 1$

### Considerações sobre implementação do novo algoritmo

Para finalizar este capítulo mostra-se que é possível implementar o algoritmo da tabela 3.4 de forma paralela. No passo 4b os erros de predição *a priori* no instante  $n$  usam os ângulos  $\phi_i$  do instante  $n - 1$ . Assim, os passos 2 e 3 podem ser executados em paralelo como os passos 4, 5, 6 e 7. Da mesma forma, o passo 1 do instante  $n$  pode ser calculado em paralelo com os passos 6 e 7 do instante  $n - 1$ . Reordenando-se então os passos do algoritmo, chega-se a figura 3.2, que mostra o diagrama do algoritmo com os níveis de precedência e o paralelismo inerente.

Todos os passos do novo algoritmo, exceto os passos 4a e 7, podem ser implementados usando unidades CORDIC operando em modo circular de rotação ('circular vector rotation mode') para executar as rotações planares e operando no modo

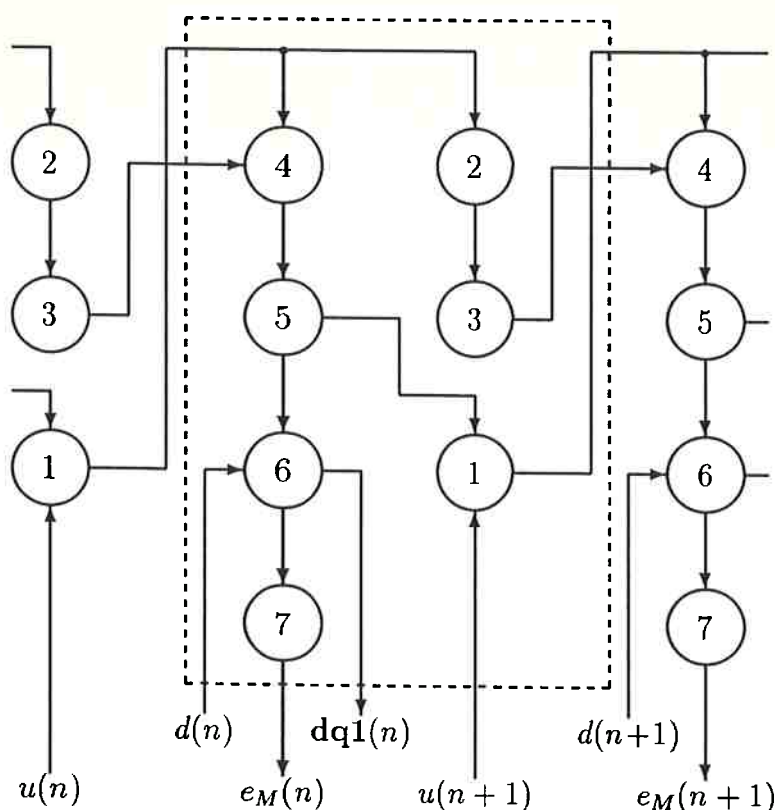


Figura 3.2: Fluxograma da implementação paralela. Os passos da tabela 3.4 estão indicados pelos números correspondentes.

de redução circular ('circular Y-reduction mode') para calcular os ângulos de rotação [41]. As tabelas 3.6 e 3.7 resumem a complexidade aritmética, considerando respectivamente, um processador de propósito geral e um processador CORDIC. Estas tabelas mostram a complexidade aritmética dos conjuntos de passos  $\{4, 5, 6, 7\}$  e  $\{2, 3, 1\}$ , os quais de acordo com a figura 3.2, podem ser executados em paralelo. Comparando na tabela 3.6 a complexidade aritmética, observa-se que existem  $4M$  multiplicações a mais para o primeiro conjunto de passos. As operações de divisão e norma estão equilibradas. Comparando na tabela 3.7 a complexidade aritmética dos dois conjuntos de passos, observa-se que existem  $M$  rotações circulares a mais no primeiro conjunto de passos. As rotações circulares são efetuadas nos passos 4b, 6 e 1, sendo que os passos 6 e 1 definem respectivamente estruturas de McWhirter para o caso da estimação e da predição progressiva. Cada conjunto contém  $M$  reduções circulares ('Givens annihi-

lations') que são feitas nos passos 3 e 5, e  $M$  multiplicações por  $\lambda^{1/2}$  nos passos 1 e 6. Com esse desequilíbrio, surge naturalmente o desejo de equilibrar a complexidade dos dois conjuntos de passos que são executados em paralelo. Entretanto, devido a precedência entre diferentes passos e devido ao fato que as operações aritméticas aparecem em blocos, o equilíbrio entre os conjuntos de operações não parece possível. Uma modificação simples da implementação paralela da figura 3.2, consiste em retirar o passo 4 do lado direito da caixa pontilhada e colocá-lo no lado esquerdo de modo a executar o passo 4 depois do passo 1 e não antes do passo 5 (isto muda a posição do passo 4 da primeira para a segunda coluna na figura 3.2). Nas tabelas 3.6 e 3.7 isso corresponde a uma troca de lugar do passo 4, e das correspondentes operações, da primeira para a segunda coluna.

Tabela 3.6

Complexidade computacional para um processador de propósito geral			
Passos	4, 5, 6, 7	2, 3, 1	Total
$\times$	$9M + 1$	$5M + 1$	$14M + 2$
$\div$	$2M + 2$	$2M + 2$	$4M + 4$
Normas (vetor $2 \times 1$ )	$M$	$M + 1$	$2M + 1$

Tabela 3.7

Complexidade computacional para CORDIC			
Passos	4, 5, 6, 7	2, 3, 1	Total
$\times$	$M + 1$	$M + 1$	$2M + 2$
$\div$	2	—	2
Redução circular	$M$	$M + 1$	$2M + 1$
Rotação circular	$2M$	$M$	$3M$
Total	$4M + 3$	$3M + 2$	$7M + 5$



### 3.9 Comentários

Neste capítulo mostrou-se vários conjuntos de equações que resolvem o problema dos mínimos quadrados de forma recorrente na ordem e no tempo. Utilizando as interpretações efetuadas no capítulo 2, mostrou-se que tais equações podem ser obtidas de forma comum, tanto no método de ortogonalização baseado no procedimento de ortogonalização de Gram-Schmidt, quanto no método baseado em rotações de Givens. Deste modo chegou-se a uma interpretação, em termos dos erros de predição *a priori* e *a posteriori*, e fator de conversão, para cada um dos elementos da matriz de sistema que descreve as operações de adaptação e filtragem dos algoritmos QR-RLS. Tal interpretação para todos os elementos da matriz de sistema é aparentemente inédita. Com isto complementa-se as relações entre os métodos de resolução do problema dos mínimos quadrados de forma recorrente na ordem e no tempo apresentadas no capítulo 2. Usando a definição de erro da predição regressiva *a priori* normalizado, introduzida no capítulo 2, apresentou-se uma nova estrutura em treliça, baseada nos erros de predição *a priori* normalizados. A partir desta estrutura em treliça foi desenvolvido um novo algoritmo QR-LSL. Uma das vantagens deste novo algoritmo sobre os demais é que apresenta um paralelismo inerente que pode ser explorado em implementações rápidas. Apesar de necessitar no total de  $7M + 5$  ciclos de CORDIC, tal paralelismo torna possível a implementação em um intervalo de tempo correspondente a  $4M + 3$  ciclos de CORDIC.

Outros algoritmos podem ser obtidos combinando de forma conveniente as equações apresentadas. Entretanto, mais importante que o desenvolvimento de outros algoritmos é obter um entendimento dos mecanismos de propagação de erro, que levam muitas vezes estes algoritmos a operarem de forma pouco adequada. No próximo capítulo são apresentadas condições para garantir propagação estável de erros numéricos nos algoritmos QR-LSL.

## Capítulo 4

### Estabilidade numérica dos algoritmos QR-LSL

Neste capítulo é analisado o comportamento numérico dos algoritmos QR-LSL apresentados no capítulo 3. Para isto, consideram-se os conceitos de sistema mínimo, consistência retrógrada ('backward consistency') e estabilidade retrógrada ('backward stability'). A aplicação dos conceitos de consistência e estabilidade retrógradas, originários da área de análise numérica, na investigação da estabilidade de algoritmos RLS rápidos leva a uma análise conceitual muito clara que não necessita de hipóteses sobre a estacionaridade dos sinais. A introdução destes conceitos na análise de filtros adaptativos se deve a Slock [10], Slock e Kailath [11] e Regalia [31, 32]. Os trabalhos dos dois primeiros autores são voltados para investigação de filtros adaptativos transversais rápidos, enquanto que Regalia se concentra nos algoritmos QR-RLS recorrentes na ordem. Nas referências [31, 32] foram estabelecidas as condições para que o algoritmo QR-LSL *a posteriori* apresente a propriedade de estabilidade retrógrada. Uma das condições fundamentais para manter esta propriedade é a minimalidade, no sentido da teoria de sistemas. Até então, este algoritmo era o único RLS rápido com a propriedade de estabilidade retrógrada. Um dos objetivos deste capítulo é mostrar que o algoritmo QR-LSL *a priori*, introduzido no capítulo 3, também possui esta propriedade.

A utilização dos conceitos de minimalidade, consistência e estabilidade retrógradas na análise da estabilidade de algoritmos adaptativos é recente e pouco difundida na literatura. Por este motivo, na seção 4.1 do texto a seguir, é feita uma revisão destes conceitos e dos resultados mais importantes de [32]. Na seção 4.2 é apresentada uma análise da estabilidade do algoritmo QR-LSL *a posteriori*. O objetivo é facilitar a análise do algoritmo QR-LSL *a priori*, que é assunto da seção 4.3. Na seção 4.4 são feitas algumas considerações sobre a representação finita das

variáveis internas dos algoritmos QR-LSL analisados. Finalmente, na seção 4.5 são apresentados resultados de simulação para verificar o desempenho do novo algoritmo QR-LSL.

## 4.1 Propagação de erro e consistência retrógrada

A seção de estimação conjunta dos algoritmos RLS rápidos está subordinada à seção de predição, que é a parte crítica na análise de estabilidade numérica [32]. Esta seção pode ser descrita por um sistema dinâmico de tempo discreto e não linear do tipo

$$\zeta(n+1) = T\{\zeta(n), u(n+1)\}, \quad (4.1)$$

onde  $\zeta(n)$  é o vetor de estados que contém todas as variáveis necessárias para propagar a solução do instante  $n$  para o instante  $n+1$ . Elas são tais que, todas as outras variáveis do algoritmo podem ser obtidas a partir delas. Quando a informação  $u(n+1)$  está disponível, a função  $T\{\cdot\}$  informa como é feita a atualização do vetor de estados  $\zeta(n)$ . Tanto a função  $T\{\cdot\}$  como o vetor de estados  $\zeta(n)$  variam de um algoritmo para outro. Por exemplo, as seções de predição dos algoritmos das tabelas 3.1 e 3.2 propagam  $5M$  variáveis, definindo vetores  $\zeta(n)$  formados respectivamente pelas seguintes variáveis

$$\{\psi_{i-1}(n), \xi_{i-1}^f(n), \xi_{i-1}^b(n), k_i^b(n), k_i^f(n), \quad 1 \leq i \leq M\} \quad (4.2)$$

e

$$\{dq1_i^f(n), \xi_{i-1}^{f/2}(n), \pi_i^b(n), \epsilon_{i-1}^b(n), \xi_{i-1}^{b/2}(n), \quad 1 \leq i \leq M\}. \quad (4.3)$$

Para facilitar a identificação das variáveis do vetor de estados dos algoritmos das tabelas 3.3 e 3.4, estes foram reescritos respectivamente nas tabelas 4.1 e 4.2. Da tabela 4.1 é imediato que a seção de predição para o algoritmo QR-LSL *a priori* propaga  $2M+1$  variáveis, que podem ser agrupadas no seguinte vetor de estados:

$$\zeta(n) = [dq1_1^f(n), \dots, dq1_M^f(n), \xi_M^{f/2}(n), \bar{b}_0(n), \dots, \bar{b}_{M-1}(n)]^T. \quad (4.4)$$

Tabela 4.1

Algoritmo QR-LSL com erros <i>a posteriori</i>	
<b>Passo 5:</b> Ângulos para a estimação conjunta	
a)	$\gamma_M^{1/2}(n) = \sqrt{1 - \sum_{i=1}^M  \bar{b}_{i-1}(n) ^2}$
b)	For $i = M, M-1, \dots, 1$ do
	$\begin{bmatrix} \gamma_{i-1}^{1/2}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \gamma_i^{1/2}(n) \\ \bar{b}_{i-1}(n) \end{bmatrix}$
<b>Passo 1:</b> Predição progressiva ( $\varepsilon_0^f(n+1) = u(n+1)$ )	
	For $i = 1, 2, \dots, M$ do
	$\begin{bmatrix} dq1_i^f(n+1) \\ \varepsilon_i^f(n+1) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i^f(n) \\ \varepsilon_{i-1}^f(n+1) \end{bmatrix}$
<b>Passo 2:</b> Energia da predição progressiva	
	$\begin{bmatrix} \xi_M^{f/2}(n+1) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta_{M+1}^f(n+1) & \sin \theta_{M+1}^f(n+1) \\ -\sin \theta_{M+1}^f(n+1) & \cos \theta_{M+1}^f(n+1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_M^{f/2}(n) \\ \varepsilon_M^f(n+1) \end{bmatrix}$
<b>Passo 3:</b> Coeficiente de reflexão normalizado	
	For $i = M, M-1, \dots, 1$ do
	$\begin{bmatrix} \xi_{i-1}^{f/2}(n+1) \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n+1) & \sin \phi_i(n+1) \\ -\sin \phi_i(n+1) & \cos \phi_i(n+1) \end{bmatrix} \begin{bmatrix} \xi_i^{f/2}(n+1) \\ dq1_i^f(n+1) \end{bmatrix}$
<b>Passo 4:</b> Filtro em treliça normalizado	
a)	$\bar{f}_M(n+1) = \sin \theta_{M+1}^f(n+1) \gamma_M^{1/2}(n)$
b)	For $i = M, M-1, \dots, 1$ do
	$\begin{bmatrix} \bar{f}_{i-1}(n+1) \\ \bar{b}_i(n+1) \end{bmatrix} = \begin{bmatrix} \cos \phi_i(n+1) & \sin \phi_i(n+1) \\ -\sin \phi_i(n+1) & \cos \phi_i(n+1) \end{bmatrix} \begin{bmatrix} \bar{f}_i(n+1) \\ \bar{b}_{i-1}(n) \end{bmatrix}$
	com $\bar{b}_0(n+1) = f_0(n+1)$ .
<b>Passo 6:</b> Estimação conjunta ( $\varepsilon_0(n) = d(n)$ )	
	For $i = 1, 2, \dots, M$ do
	$\begin{bmatrix} dq1_i(n) \\ \varepsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \varepsilon_{i-1}(n) \end{bmatrix}$
<b>Passo 7:</b> Erro de estimação <i>a posteriori</i>	
	$e_M(n) = \varepsilon_M(n) \gamma_M^{1/2}(n)$

Da mesma forma, da tabela 4.2 verifica-se facilmente que o vetor de estados do algoritmo QR-LSL *a priori* tem também dimensão  $2M + 1$  e é dado por:

$$\zeta(n) = [dq1_1^f(n), \dots, dq1_M^f(n), \xi_M^{f/2}(n), \bar{\psi}_0(n), \dots, \bar{\psi}_{M-1}(n)]^T. \quad (4.5)$$

Cabe notar que, estritamente falando, os algoritmos das tabelas 3.3 e 3.4 propagam mais que  $2M + 1$  variáveis no tempo, ou seja os vetores de estados tem a rigor dimensão maior que  $2M + 1$ . Entretanto, como as tabelas 4.1 e 4.2 mostram, uma

simples reordenação dos passos deixa claro que somente é necessário propagar  $2M + 1$  variáveis no tempo. Além disso, como nas tabelas 3.3 e 4.1 são realizadas exatamente os mesmos cálculos esses ‘dois’ algoritmos são numericamente equivalentes. A mesma observação se aplica aos algoritmos das tabelas 3.4 e 4.2.

Tabela 4.2

Algoritmo QR-LSL com erros <i>a priori</i>	
<b>Passo 5:</b> Ângulos para a estimação conjunta ( $\gamma_0^{-1/2}(n) = 1$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} \gamma_i^{-1/2}(n) \\ 0 \end{bmatrix}$	$= \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \gamma_{i-1}^{-1/2}(n) \\ \psi_{i-1}(n) \end{bmatrix}$
<b>Passo 1:</b> Predição progressiva ( $\varepsilon_0^f(n+1) = u(n+1)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i^f(n+1) \\ \varepsilon_i^f(n+1) \end{bmatrix}$	$= \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i^f(n) \\ \varepsilon_{i-1}^f(n+1) \end{bmatrix}$
<b>Passo 2:</b> Energia do erro da predição progressiva	
$\begin{bmatrix} \xi_M^{f/2}(n+1) \\ 0 \end{bmatrix}$	$= \begin{bmatrix} \cos \theta_{M+1}^f(n+1) & \sin \theta_{M+1}^f(n+1) \\ -\sin \theta_{M+1}^f(n+1) & \cos \theta_{M+1}^f(n+1) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \xi_M^{f/2}(n) \\ \varepsilon_M^f(n+1) \end{bmatrix}$
<b>Passo 3:</b> Coeficiente de reflexão normalizado For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \xi_{i-1}^{f/2}(n) \\ 0 \end{bmatrix}$	$= \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \xi_i^{f/2}(n) \\ dq1_i^f(n) \end{bmatrix}$
<b>Passo 4:</b> Filtro em treliça normalizado	
a) $\bar{\eta}_M(n+1) = \gamma_M^{-1/2}(n) \frac{\sin \theta_{M+1}^f(n+1)}{\cos \theta_{M+1}^f(n+1)}$	
b) For $i = M, M-1, \dots, 1$ do	
$\begin{bmatrix} \bar{\eta}_{i-1}(n+\bar{r}) \\ \bar{\psi}_i(n+1) \end{bmatrix}$	$= \begin{bmatrix} \cos \phi_i(n) & \sin \phi_i(n) \\ -\sin \phi_i(n) & \cos \phi_i(n) \end{bmatrix} \begin{bmatrix} \bar{\eta}_i(n+1) \\ \bar{\psi}_{i-1}(n) \end{bmatrix}$
com $\bar{\psi}_0(n+1) = \bar{\eta}_0(n+1)$ .	
<b>Passo 6:</b> Estimação conjunta ( $\varepsilon_0(n) = d(n)$ ) For $i = 1, 2, \dots, M$ do	
$\begin{bmatrix} dq1_i(n) \\ \varepsilon_i(n) \end{bmatrix}$	$= \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ -\sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_{i(n-1)} \\ \varepsilon_{i-1}(n) \end{bmatrix}$
<b>Passo 7:</b> Erro de estimação <i>a posteriori</i> $e_M(n) = \varepsilon_M(n) / \gamma_M^{-1/2}(n)$	

Analisar a estabilidade numérica diretamente a partir de (4.1) para um dado algoritmo é a princípio difícil, devido à característica não linear da função  $T\{\cdot\}$ . Metodologias de análise tradicionais associam uma função de Lyapunov a (4.1) [39]

ou usam linearização local mais a teoria de médias para descrever o sistema (4.1) por meio de um sistema linear variante no tempo [11]. Os resultados apresentados em [32] por sua vez, enfatizam a relação entre propagação de erro e consistência retrógrada, que é um conceito originário da análise numérica. Esta relação permite investigar a questão da estabilidade sem fazer considerações sobre a estacionaridade dos sinais e, apesar de utilizar também resultados clássicos da teoria de sistemas adaptativos robustos (exponencialmente estáveis), evita o uso explícito de métodos de linearização. Tal metodologia parte do fato de que os algoritmos RLS convergem exponencialmente rápido em aritmética exata e estabelece condições para que esta propriedade, se traduza em estabilidade exponencial. O conceito de consistência retrógrada é usado como um elo de ligação entre convergência e estabilidade exponenciais. Além disso, utilizando o conceito de minimalidade, originário da teoria de sistemas, são estabelecidas as condições para que a atualização (no tempo) do vetor de estados tenha a propriedade de estabilidade retrógrada, o que garante uma propagação estável dos erros numéricos (em precisão finita).

O texto a seguir esta organizado da seguinte forma: inicialmente são apresentadas as definições necessárias, em seguida são relacionados dois resultados clássicos sobre robustez numérica de algoritmos RLS e para finalizar, são apresentados resultados que usam os conceitos e definições apresentados.

**Definição 1:** *Conjunto de estados alcançáveis*

Supondo a existência de um completo controle sobre a sequência de observação  $\{u(\ell), 0 \leq \ell \leq n\}$  e supondo também uma variação da sequência de observação sobre todas as possíveis realizações; então, por construção, o vetor de estados (4.1) pode alcançar somente aqueles estados consistentes com a solução exata de algum problema de predição pré-janelado. Representa-se por  $\mathcal{S}_e(n)$  o conjunto de todos os estados alcançáveis  $\zeta(n)$  no instante  $n$  usando aritmética de precisão exata e por  $\mathcal{S}_f(n)$  o conjunto de todos os estados alcançáveis  $\zeta(n)$  no instante  $n$  usando aritmética de precisão finita. Este último conjunto,  $\mathcal{S}_f(n)$ , vai depender fortemente das convenções de programação, tamanho da palavra binária e outras considerações numéricas.

### Regiões de $\mathcal{S}_e(\cdot)$ :

$\alpha$ -interior de  $\mathcal{S}_e(\cdot)$ : um estado  $\zeta(n)$  pertence a um  $\alpha$ -interior de  $\mathcal{S}_e(\cdot)$  se para todo  $\delta(n)$  satisfazendo  $\|\delta(n)\| < \alpha$  vale

$$\zeta(n) + \delta(n) \in \mathcal{S}_e(n),$$

sendo  $\alpha$  uma constante positiva. No caso de conjuntos que são delimitados por uma hipersfera de raio finito, um  $\alpha$ -interior pode ser entendido como um interior fechado e limitado de  $\mathcal{S}_e(n)$ . Convém observar que, dado um  $\alpha$ -interior, sempre existem estados interiores a  $\mathcal{S}_e(n)$  que não pertencem ao  $\alpha$ -interior.

Borda de  $\mathcal{S}_e(n)$ : um estado  $\zeta(n) \in \mathcal{S}_e(n)$  pertence a borda de  $\mathcal{S}_e(n)$  se para  $\forall \varepsilon > 0$ , existe uma perturbação  $\delta(n)$ , satisfazendo  $\|\delta(n)\| < \varepsilon$  tal que

$$\zeta(n) + \delta(n) \notin \mathcal{S}_e(n).$$

O conjunto de todos os pontos de borda é representado por  $\partial\mathcal{S}_e(n)$ .

### Definição 2: *Consistência retrógrada ('backward consistency')*

A seção de predição de um algoritmo apresenta a propriedade de **consistência retrógrada** quando a condição

$$\mathcal{S}_f(n) \subset \mathcal{S}_e(n), \quad (4.6)$$

para todo  $n$ , pode ser estruturalmente induzida. Em outras palavras, a solução calculada, portanto afetada de erros numéricos, deve corresponder em cada instante de tempo a uma solução exata de um problema de predição linear em que se considera um sinal pré-janelado perturbado.

Um exemplo envolvendo a alcançabilidade da solução exata e alcançabilidade da solução calculada usa o fator de conversão  $\gamma_i(n)$ . A variável  $\gamma_i(n)$ , apresenta em aritmética exata uma faixa de alcançabilidade  $0 \leq \gamma_i(n) \leq 1$  (ver equação (A.19) e (A.54) do apêndice A). Supondo que esta variável tenha sido calculada através da expressão (A.22) e que  $\Phi_M^{-1}(n)$  apresente um autovalor negativo em precisão finita, então a faixa de alcançabilidade de  $\gamma_i(n)$  excede o limite superior de 1, podendo levar a uma divergência numérica. Entretanto, se a variável  $\gamma_i(n)$  for calculada através de (B.14), e considerando que a condição  $|\cos \theta_i^b(\mathbf{n})| \leq 1$  possa ser assegurada numericamente, a faixa de alcançabilidade em precisão finita de (B.14) vai garantir  $\gamma_i(n) \in [0, 1]$ .

**Definição 3:** *Estabilidade retrógrada*

Um algoritmo apresenta a propriedade de **estabilidade retrógrada**, se além de apresentar a propriedade da consistência retrógrada, isto é satisfazer (4.6), a solução calculada permanece com uma variância limitada em torno da solução exata. Em outras palavras, uma perturbação uniforme em qualquer ponto do algoritmo implica em uma saída equivalente a uma outra entrada, sendo que a resposta perturbada na saída tem variância limitada em torno da resposta não perturbada.

**Definição 4:** *Sistema mínimo*

Um algoritmo é considerado **mínimo**, no sentido de teoria de sistemas, quando não é possível descrever o sistema (4.1) utilizando um vetor de estados  $\zeta(n)$  de dimensão menor e todas as outras grandezas podem ser calculadas a partir de  $\zeta(n)$  e  $u(n+1)$ . Um algoritmo é considerado **não mínimo** se o vetor de estados  $\zeta(\cdot)$  pode ser representado com dimensão reduzida sem nenhuma perda de informação, independentemente da seqüência de observação e do instante de tempo. Intuitivamente, isto significa que os parâmetros propagados em questão, contêm uma redundância; pois pode-se propagar a mesma informação com menos parâmetros. Neste caso, o vetor de estados (4.1) apresenta uma dimensão maior do que a estritamente necessária.

**Definição 5:** *Estabilidade exponencial*

Seja o sistema (4.1). No instante  $n_0$  é introduzida uma perturbação  $\delta(n_0)$  no vetor de estados  $\zeta(n_0)$  produzindo o vetor de estados perturbado

$$\tilde{\zeta}(n_0) = \zeta(n_0) + \delta(n_0).$$

Usando a mesma seqüência de dados  $\{u(\cdot)\}$ , a mesma função  $T$  do sistema (4.1) e supondo que não existam erros aritméticos subsequentes, observa-se a evolução do sistema perturbado

$$\tilde{\zeta}(n+1) = T\{\tilde{\zeta}(n), u(n+1)\}, \quad (4.7)$$

para  $n \geq n_0$ . Utilizando-se o sistema (4.1) como uma referência, idealmente após a perturbação o sistema (4.7) deve retornar a trajetória de (4.1), isto é,

$$\lim_{n \rightarrow \infty} |\zeta(n) - \tilde{\zeta}(n)| = 0. \quad (4.8)$$



O sistema é dito **exponencialmente estável** em relação à perturbação  $\delta(n_0)$ , se após a perturbação a trajetória do sistema (4.7) retorna a trajetória do sistema (4.1) de forma exponencialmente rápida [6, 5]. Isto é equivalente a considerar que o sistema variante no tempo que atualiza a diferença dos estados

$$\Delta\zeta(n) = \tilde{\zeta}(n) - \zeta(n),$$

ou seja,

$$\Delta\zeta(n+1) = F_{\Delta}[n, u(n+1)]\Delta\zeta(n), \quad n > n_0, \quad (4.9)$$

é exponencialmente estável. Observando que podem existir perturbações para as quais o sistema (4.1) não é exponencialmente estável, denomina-se de  $\mathcal{C}$  a classe das perturbações  $\delta(n)$  para as quais pode-se garantir que (4.1) é exponencialmente estável.

### Resultado Clássico 1: *Estabilidade exponencial e robustez numérica*

Considerando que a perturbação permanece na classe  $\mathcal{C}$  e que ela permanece ‘suficientemente pequena’, então a seqüência de estados perturbados

$$\tilde{\zeta}(n+1) = T\{\tilde{\zeta}(n), u(n+1)\} + \delta(n+1) \quad (4.10)$$

para todo  $n$ , **permanece com uma variância limitada em torno do sistema de referência** (4.1). Este fato é usado em [31] invocando resultados de robustez para sistemas exponencialmente estáveis [5]. Sem pretender apresentar uma demonstração rigorosa deste resultado, mas somente procurando esclarecer esta afirmação e suas possíveis interpretações, cabe fazer as seguintes observações:

- a) A metodologia clássica para a análise da estabilidade de sistemas perturbados está baseada na linearização do sistema (4.10) em torno da trajetória seguida pelo sistema não perturbado (4.1). Pressupõe-se que a perturbação é suficientemente pequena para que o sistema linearizado represente bem a dinâmica do sistema em torno da trajetória original. Deste procedimento de linearização, resulta um sistema que, como o sistema (4.9), descreve a evolução no tempo da diferença entre os estados dos sistemas perturbado e não perturbado:

$$\begin{aligned} \Delta\zeta(n+1) &= T\{\zeta(n) + \Delta\zeta(n), u(n+1)\} - \zeta(n+1) + \delta(n+1) \\ &\approx F[n+1]\Delta\zeta(n) + \delta(n+1), \end{aligned} \quad (4.11)$$

com  $F[n+1] = \partial T / \partial \zeta$  para  $\zeta = \zeta(n)$ . Para perturbações suficientemente pequenas a hipótese feita de que o sistema (4.1) é exponencialmente estável para perturbações na classe  $\mathcal{C}$  pode ser aplicada ao sistema 'livre'

$$\Delta \zeta(n+1) = F[n+1] \Delta \zeta(n),$$

o qual deve ser exponencialmente estável [5]. Com isso, a conclusão de que a seqüência de estados do sistema perturbado permanece com uma variância limitada em torno da seqüência dos estados do sistema não perturbado decorre imediatamente do Teorema da Estabilidade Total para Tempo Discreto [5, pág. 26].

- b) No contexto, a perturbação  $\delta(n+1)$  representa a diferença entre o vetor de estados  $T\{\tilde{\zeta}(n), u(n+1)\}$  e  $\tilde{\zeta}(n+1)$ . A perturbação  $\delta(\cdot)$  vai variar com  $\tilde{\zeta}(\cdot)$ , modificando a estrutura de realimentação do sistema. Resultados de robustez numérica para sistemas exponencialmente estáveis, como os invocados no item a, permitem a dependência entre vetor de estados e perturbação, desde que a perturbação permaneça suficientemente pequena. Isto requer uma resolução numérica suficientemente fina para estabilidade permanecer intacta, dependendo da característica da realimentação induzida pela perturbação  $\delta(\cdot)$ .
- c) Se o sistema (4.1) não é exponencialmente estável sob a perturbação (o que pode acontecer se  $\delta(\cdot)$  sair da classe  $\mathcal{C}$ ), a influência composta da perturbação  $\delta(\cdot)$  em (4.10) pode forçar  $\tilde{\zeta}(\cdot)$  a desviar-se arbitrariamente longe de  $\zeta(\cdot)$ , o que no contexto considerado significa divergência numérica.

Quando a seqüência de estados perturbados  $\tilde{\zeta}(\cdot)$  permanece com uma variância limitada em torno do sistema de referência (4.1), a atualização do vetor de estados apresenta robustez em relação às imprecisões aritméticas. Cabe observar ainda, que a perturbação  $\delta(\cdot)$  representa o erro numérico o qual está tipicamente limitado em função do tamanho da palavra binária usada na representação.

$\overline{\Delta\Delta}$

O resultado clássico 1 é fundamental para garantir a estabilidade retrógrada dos algoritmos RLS, sendo utilizado na demonstração das propriedades 3 e 4 do texto a seguir. Porém, para abordar o problema de estabilidade retrógrada é necessário

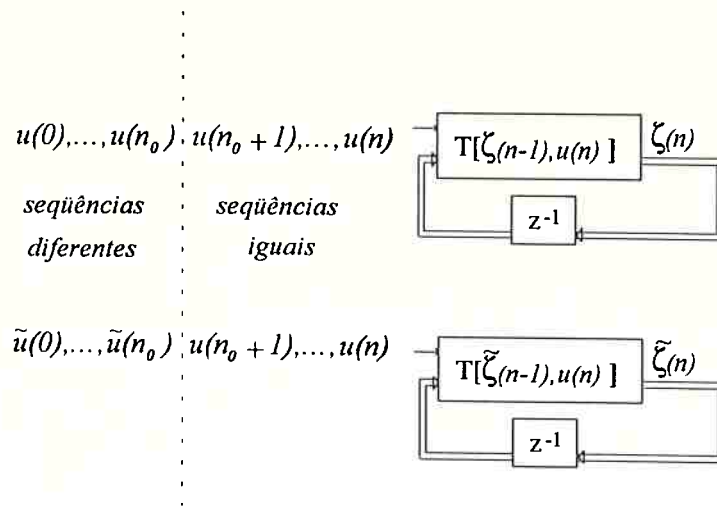


Figura 4.1: Modelo conceitual para estabelecer a estabilidade do sistema (4.1) em relação as perturbações de estado que não violam a propriedade de consistência retrógrada.

ainda um outro resultado que será utilizado para caracterizar o conjunto de estados alcançáveis  $\mathcal{S}_e(n)$ .

**Resultado clássico 2:** *A convergência exponencial dos algoritmos RLS em precisão exata e com excitação persistente implica em estabilidade exponencial para perturbações isoladas que não retirem o estado  $\zeta(n)$  do conjunto  $\mathcal{S}_e(n)$ .*

*Prova* [31]: Dado um estado  $\zeta(n_0)$  ele representa a solução exata obtida em (4.1) para uma seqüência de observação  $\{u(\ell), 0 \leq \ell \leq n_0\}$ . Seja  $\tilde{\zeta}(\cdot)$  qualquer estado perturbado, tal que

$$\tilde{\zeta}(n_0) \in \mathcal{S}_e(n_0)$$

de forma arbitrária. Pela definição de  $\mathcal{S}_e(n)$ , o estado  $\tilde{\zeta}(n_0)$  pode ser entendido como a solução exata de alguma outra seqüência de entrada  $\{\tilde{u}(\ell), 0 \leq \ell \leq n_0\}$ . As características exatas de  $\{u(\ell)\}$  e  $\{\tilde{u}(\ell)\}$  não são relevantes. A partir do instante de tempo  $n_0+1$  aplica-se a mesma seqüência futura  $\{u(n_0+1), u(n_0+2), \dots, u(n)\}$  nos sistemas (4.1) e (4.10), conforme apresentado na figura 4.1. O que mantém as trajetórias  $\zeta(\cdot)$  e  $\tilde{\zeta}(\cdot)$  separadas são as seqüências de entrada diferentes para  $\ell \leq n_0$ . Resultados de robustez numérica para os algoritmos dos mínimos quadrados, e.g. [38, 6], mostram que a observação passada tem influência exponencialmente decrescente na solução futura,

quando a seqüência de observação  $\{u(\cdot)\}$  é ponderada por  $\lambda < 1$  e satisfaz a uma condição de excitação persistente. Consequentemente, com  $\lambda < 1$ , sabe-se que as seqüências de observações passadas  $\{u(\ell)\}$  e  $\{\tilde{u}(\ell)\}$  serão excluídas exponencialmente rápido com o avanço do tempo, o que mostra que o sistema perturbado converge exponencialmente para a solução obtida com aritmética de precisão exata.

$\overline{\Delta\Delta}$

Do resultado clássico 2 segue de imediato que, para excitações persistentes, pode-se caracterizar a classe  $\mathcal{C}$  como constituindo de todas as perturbações que não colocam o estado  $\zeta(\cdot)$  fora de  $\mathcal{S}_e(n)$ . Além disso, o conjunto  $\mathcal{S}_e(n)$  pode ser entendido como um domínio de estabilidade do algoritmo. Desta forma chegou-se a:

**Propriedade 1:** *O conjunto  $\mathcal{S}_e(\cdot)$  de estados alcançáveis representa um domínio de estabilidade do algoritmo.*

Após ter identificado o conjunto  $\mathcal{S}_e(\cdot)$  como um domínio de estabilidade do algoritmo RLS é conveniente detalhar um pouco melhor a possível evolução dos estados no conjunto  $\mathcal{S}_e(\cdot)$  em função das características do algoritmo e do sinal de entrada.

**Propriedade 2:** *Características decorrentes da minimalidade.*

- a) Supondo que o sistema (4.1) é mínimo e que a seqüência de observação  $\{u(\cdot)\}$  é limitada, então vale:
- a.i) a evolução do vetor de estados está restrita a borda  $\partial\mathcal{S}_e(n)$  se, e somente se,  $\{u(\cdot)\}$  é perfeitamente preditível;
  - a.ii) a evolução do vetor de estados está restrita a um  $\alpha$ -interior de  $\mathcal{S}_e(n)$  se, e somente se,  $\{u(\cdot)\}$  é uma excitação persistente;
- b) Se o sistema (4.1) não é mínimo, todos os pontos do conjunto de estados alcançáveis  $\mathcal{S}_e(n)$  são pontos da borda deste conjunto.

*Prova* [31]: A demonstração da parte a é dependente do particular algoritmo e será demonstrado nas seções 4.2 e 4.3 para os algoritmos QR-LSL *a posteriori* e *a priori*. De qualquer forma, vale a pena adiantar que de forma geral, a constante  $\alpha$ , característica do  $\alpha$ -interior, depende diretamente da persistência da excitação; isto é, quanto 'menos persistente' for o sinal  $u(n)$ , menor será a constante  $\alpha$  e mais próximo da borda o

algoritmo irá operar.

A demonstração da parte **b** é uma consequência quase direta da definição de sistemas mínimos. Se o algoritmo é não mínimo pode-se expressar um subconjunto de variáveis de estado como funções implícitas ou explícitas das restantes, levando a restrições do tipo

$$f[\zeta(n)] = \mathbf{0}, \quad (4.12)$$

na especificação do conjunto  $\mathcal{S}_e(n)$ , sendo essas restrições válidas para todo  $n$ , para todas seqüências de observações  $\{u(\cdot)\}$  e para qualquer elemento de  $\mathcal{S}_e(n)$ . Em [10] e em [32] são apresentadas as restrições para os algoritmos baseados no filtro transversal e para os algoritmos baseados nos filtros em treliça. Como existem mais variáveis em  $\zeta(n)$  que graus de liberdade no conjunto solução  $\mathcal{S}_e(n)$ , então vai sempre existir uma perturbação local que coloca  $\zeta(n)$  fora do conjunto de soluções estabelecidos por (4.12), e portanto, fora do conjunto  $\mathcal{S}_e(n)$ . Desta forma, sempre poderão existir erros numéricos que ‘jogam’ os estados para fora do conjunto  $\mathcal{S}_e(n)$ .

$\overline{\Delta\Delta}$

Da propriedade 2 observa-se que, satisfazer o critério da consistência retrógrada, isto é

$$\mathcal{S}_f(n) \subset \mathcal{S}_e(n),$$

para todo  $n$ , é pouco viável para um algoritmo não mínimo, o que leva a identificar a não minimalidade como a propriedade chave para explicar a instabilidade de algoritmos rápidos não mínimos [32]. Em [10] mostra-se que para a seção de predição de algoritmos RLS rápidos, a dimensão mínima de  $\zeta(n)$  é de  $2M + 1$ , para todo  $n \geq 2M$ , onde  $M$  é a ordem do problema de predição. Isto se deve ao fato que toda a informação necessária para atualizar a matriz de autocorrelação ou a sua inversa está contida nos vetores dos coeficientes dos filtros da predição progressiva e regressiva e do vetor do ganho de Kalman, conforme equação (A.29) do apêndice A. Como os três vetores estão intimamente relacionados, bastam dois deles mais um escalar (devido a normalização de dois dos vetores) para fazer a atualização. Este assunto será tratado em maior detalhe no lema 1 da seção 4.2.

Do item **a.ii** da propriedade 2, e dos resultados clássicos 1 e 2 conclui-se a seguinte propriedade [32]:

**Propriedade 3:** *O  $\alpha$ -interior de  $\mathcal{S}_e(n)$  é um domínio de estabilidade exponencial para um algoritmo RLS que satisfaz a propriedade 2a.ii. Caso as perturbações sejam suficientemente pequenas, isto é, a quantização numérica seja suficientemente fina, a diferença entre as saídas dos sistemas implementados em precisão exata e em precisão finita tem variância limitada.*

*Prova [31]:* Se os estados evoluem em um  $\alpha$ -interior então a seqüência de entrada é necessariamente persistente. Além disso, se as perturbações são suficientemente pequenas elas não retiram o vetor de estados do conjunto  $\mathcal{S}_e(n)$ . Conseqüentemente, pela propriedade clássica 2 a convergência exponencial para perturbações isoladas é garantida. Desta forma, da propriedade clássica 1 conclui-se que, se as perturbações são suficientemente pequenas, então a diferença entre seqüências de estados uniformemente perturbados e a solução exata tem variância limitada.

$\overline{\Delta\Delta}$

Finalmente da propriedade 3 e das definições apresentadas decorre a seguinte propriedade:

**Propriedade 4:** *Caso um algoritmo RLS seja mínimo e tenha consistência retrógrada, então no caso de excitação persistente, ele apresenta estabilidade retrógrada para uma quantização numérica suficientemente fina.*

*Prova [31]:* A consistência retrógrada implica que a solução calculada é sempre a solução exata de algum problema de predição pré-janelado (nas seções 4.2 e 4.3 demonstra-se esta afirmação para os algoritmos das tabelas 4.1 e 4.2), sendo necessário mostrar apenas que a solução calculada tem uma variância limitada em torno da solução exata. Da propriedade 2a.ii, o valor exato está restrito a um  $\alpha$ -interior de  $\mathcal{S}_e(n)$ , o qual é domínio de estabilidade exponencial. Para um outro valor de  $\alpha$  apropriado e para perturbações suficientemente pequenas, qualquer estado perturbado está também contido neste outro  $\alpha$ -interior, de forma que o sistema uniformemente perturbado (4.10) está também limitado em um domínio de estabilidade exponencial. Isso porque a estabilidade exponencial, como mencionado na propriedade 3, implica em erro acumulado limitado, desde que  $\delta(n)$  permaneça uniformemente pequeno. Além disso, os estados do sistema de precisão finita (4.10) apresentam uma variância limitada em torno dos estados do sistema exato (4.1). Convém observar que, a es-

colha da constante  $\alpha$  para representar o  $\alpha$ -interior de  $\mathcal{S}_e(n)$ , está relacionada com o condicionamento da matriz de dados  $\mathbf{A}_M(n)$  (nas seções 4.2 e 4.3 demonstra-se esta afirmação para os algoritmos das tabelas 4.1 e 4.2).

$\overline{\Delta\Delta}$

São poucos os algoritmos dos mínimos quadrados rápidos que apresentam a propriedade de estabilidade retrógrada. Até o presente momento são conhecidos apenas dois algoritmos. O primeiro algoritmo, que é o da tabela 4.1, foi apresentado por Regalia e Bellanger em [33]. O segundo está apresentado na tabela 4.2, sendo uma contribuição deste trabalho. Na seção 4.3 faz-se uma análise do comportamento numérico do algoritmo QR-LSL *a priori* da tabela 4.2. Os resultados são obtidos por analogia com os apresentados em [32]. Por este motivo, apresenta-se na próxima seção a análise do algoritmo QR-LSL *a posteriori*.

## 4.2 Algoritmo QR-LSL com erros a posteriori

Nesta seção é considerado o algoritmo QR-LSL *a posteriori* apresentado na tabela 4.1. Os resultados aqui apresentados podem ser encontrados na referência [31]. São reproduzidos com o objetivo de possibilitar uma melhor compreensão dos conceitos envolvidos e esclarecer eventuais dúvidas que podem surgir durante a leitura de [31]. Por um lado, são demonstradas as afirmações apresentadas em [31] sem demonstração, por outro lado, algumas demonstrações apresentadas em detalhe em [31] são apenas delineadas, chamando-se somente a atenção aos aspectos de maior interesse. Os correspondentes resultados para o algoritmo QR-LSL *a priori* podem ser obtidos em grande parte estabelecendo-se relações entre as variáveis dos dois algoritmos QR-LSL considerados. Daí o interesse nas demonstrações apresentadas nesta seção. No texto a seguir, mostra-se inicialmente que o algoritmo é mínimo e é estabelecido qual o conjunto dos estados alcançáveis  $\mathcal{S}_e(n)$  associado. Em seguida, mostra-se que o algoritmo satisfaz a propriedade **2a**, isto é, que com excitação persistente o algoritmo opera em um  $\alpha$ -interior do domínio de estabilidade  $\mathcal{S}_e(n)$  e no caso de excitações preditíveis o algoritmo opera na borda  $\delta\mathcal{S}_e(n)$  do domínio de estabilidade. Depois são consideradas as convenções numéricas necessárias para garantir a consistência retrógrada e conseqüentemente a estabilidade retrógrada do algoritmo. Finalmente, para encerrar, são feitos alguns comentários a respeito da inicialização do algoritmo.

### 4.2.1 Minimalidade

Na seção 4.1 foi mostrado que o vetor de estados do algoritmo QR-LSL *a posteriori* é dado por

$$\zeta(n) = \begin{bmatrix} \text{dq}1_M^f(n) \\ \xi_M^{f/2}(n) \\ \bar{\mathbf{b}}_M(n) \end{bmatrix}. \quad (4.13)$$

Como a dimensão de  $\zeta(n)$  é de  $2M + 1$ , sendo  $M$  a ordem do problema de predição, o algoritmo é mínimo, conforme resultados descritos na seção 4.1. Estritamente falando, o algoritmo da tabela 4.1 é mínimo, sendo que o da tabela 3.3 não o é. Entretanto, como os cálculos realizados em ambos são exatamente os mesmos eles são numericamente equivalentes.

### 4.2.2 Caracterização do domínio de estabilidade $\mathcal{S}_e(n)$

**Propriedade 5:** *Características do domínio de estabilidade  $\mathcal{S}_e(n)$*

**5a)** Considerando  $u(0) \neq 0$  e passado o período de inicialização <sup>1</sup>, isto é,  $n > 2M$ , o conjunto de estados alcançáveis  $\mathcal{S}_e(n)$  consiste de todos os estados  $\zeta(n)$  que satisfazem:

$$\xi_M^{f/2}(n) > 0 \quad \text{e} \quad \|\bar{\mathbf{b}}_M(n)\| < 1. \quad (4.14)$$

**5b)** As condições (4.14) são equivalentes a

$$|\phi_i(n)| < \pi/2 \quad \text{e} \quad |\theta_i^b(n)| < \pi/2, \quad \text{para} \quad 1 \leq i \leq M. \quad (4.15)$$

**5c)** Um  $\alpha$ -interior de  $\mathcal{S}_e(n)$  é o conjunto de todos os estados que, além de satisfazer (4.14) e (4.15), ainda satisfazem

$$|\phi_i(n)| \leq \alpha_1 < \pi/2 \quad \text{e} \quad |\theta_i^b(n)| \leq \alpha_1 < \pi/2 \quad (4.16)$$

para  $1 \leq i \leq M$  e

$$\xi_M^{f/2}(n) \geq \alpha_2 > 0, \quad (4.17)$$

sendo  $\alpha_1$  e  $\alpha_2$  constantes positivas.

**5d)** A borda do conjunto de estados alcançáveis  $\delta\mathcal{S}_e(n)$  é formada pelos estados que

<sup>1</sup>Para o intervalo de tempo  $0 \leq n \leq 2M$ , a seqüência de entrada fornece menos do que  $2M + 1$  graus de liberdade para controlar a evolução do vetor de estados.



satisfazem

$$\xi_M^{f/2}(n) = 0 \quad \text{e/ou} \quad \|\bar{\mathbf{b}}_M(n)\| = 1.$$

A demonstração completa das propriedades 5a a 5d pode ser encontrada em [31]. Como estas propriedades serão utilizadas na seção 4.3 para mostrar resultados equivalentes para o algoritmo QR-LSL *a priori*, serão reproduzidas aqui, com algumas modificações e adaptações, aquelas partes que se mostram importantes para um melhor entendimento do assunto.

### Demonstração de 5a

A prova desta propriedade consiste na demonstração dos seguintes itens:

5a.i) Qualquer solução dos mínimos quadrados satisfaz (4.14);

5a.ii) Qualquer estado satisfazendo (4.14) pode ser de fato alcançável para uma seqüência de entrada apropriada.

### Demonstração de 5a.i

Na demonstração de 5a.i, considera-se a restrição  $u(0) \neq 0$ . De (2.98), tem-se para a ordem  $M$

$$\xi_M^f(n) = \lambda^n |u(0)|^2 + \|\mathbf{dq}2_{M+1}^f(n)\|^2.$$

Como  $\xi_M^f(n)$  está representada como a soma de dois elementos positivos então

$$\xi_M^{f/2}(n) \geq 0.$$

Entretanto como  $u(0) \neq 0$  a possibilidade de  $\xi_M^{f/2}(n) = 0$  fica excluída.

A restrição  $u(0) \neq 0$  assegura teoricamente que a matriz de autocorrelação é não singular. Desta forma, a partir de (A.19) é possível estabelecer a seguinte faixa de valores para o fator de conversão:  $0 < \gamma_M(n) \leq 1$ . Assim, de (2.155) conclui-se que  $0 \leq \|\bar{\mathbf{b}}_M(n)\| < 1$ .

### Demonstração de 5a.ii

A demonstração de 5a.ii será apenas delineada. Ela pode ser feita em duas partes. Inicialmente, considera-se os vetores normalizados dos coeficientes dos filtros

transversais da predição progressiva  $\mathbf{F}_M(n)$ , da predição regressiva  $\mathbf{B}_M(n)$  e do ganho de Kalman  $\mathbf{G}_M(n)$ , definidos respectivamente em (A.27), (A.25) e (A.30). Mostra-se então que, dados os vetores  $\mathbf{F}_M(n)$ ,  $\mathbf{B}_M(n)$  e  $\mathbf{G}_M(n)$ , necessariamente existe uma inversa da matriz de autocorrelação  $\Phi_{M-1}^{-1}(n)$  positiva definida e com posto do deslocamento três. Desta forma, necessariamente existe uma matriz de dados  $\mathbf{A}_{M-1}(n)$  com estrutura de Toeplitz e pré-janelada satisfazendo  $\Phi_{M-1}^{-1}(n) = (\mathbf{A}_{M-1}^T(n) \mathbf{A}_{M-1}(n))^{-1}$ . Como existe tal matriz de dados, existe uma seqüência de entrada  $\{u(\ell), 0 \leq \ell \leq n\}$  que no instante  $n$  faz os vetores de coeficientes dos filtros transversais da predição progressiva, da predição regressiva e do ganho de Kalman assumirem os valores previamente especificados. Em seguida, mostra-se que a partir dos elementos de um vetor de estados

$$\zeta(n) = \begin{bmatrix} \mathbf{dq}\mathbf{1}_M^f(n) \\ \xi_M^{f/2}(n) \\ \bar{\mathbf{b}}_M(n) \end{bmatrix},$$

satisfazendo (4.14) é possível determinar os correspondentes vetores de coeficientes dos filtros transversais de predição progressiva, de predição regressiva e do ganho de Kalman; existindo então uma seqüência de entrada que faz com que o vetor de estados pré-fixado seja realmente alcançado.

A primeira parte é conclusão imediata do seguinte lema [10, 31]:

**Lema 1:**

Existe uma matriz  $\Phi_{M-1}^{-1}(n) = \Phi_{M-1}^{-T}(n) > \circ$  satisfazendo

$$\begin{aligned} & \begin{bmatrix} \Phi_{M-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} - \lambda \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \Phi_{M-1}^{-1}(n) \end{bmatrix} = \\ & = \mathbf{G}_M(n)\mathbf{G}_M^T(n) + \mathbf{F}_M(n)\mathbf{F}_M^T(n) - \mathbf{B}_M(n)\mathbf{B}_M^T(n) \end{aligned} \quad (4.18)$$

se, e somente se, as três condições seguintes forem satisfeitas:

- l.1)  $0 < \gamma_{M-1}(n) \leq 1$
- l.2)  $B_M(\frac{z}{\lambda^{1/2}})B_M(\frac{1}{z\lambda^{1/2}}) = F_M(\frac{z}{\lambda^{1/2}})F_M(\frac{1}{z\lambda^{1/2}}) + G_M(\frac{z}{\lambda^{1/2}})G_M(\frac{1}{z\lambda^{1/2}})$
- l.3)  $B_M(\frac{z}{\lambda^{1/2}})$  é um polinômio de fase mínima.

Os polinômios em  $z$  satisfazem

$$F_M(\frac{z}{\lambda^{1/2}}) = \mathbf{Z}_M^T \mathbf{F}_M(n), \quad (4.19)$$

$$B_M\left(\frac{z}{\lambda^{1/2}}\right) = \mathcal{Z}_M^T \mathbf{B}_M(n), \quad (4.20)$$

$$G_M\left(\frac{z}{\lambda^{1/2}}\right) = \mathcal{Z}_M^T \mathbf{G}_M(n), \quad (4.21)$$

sendo

$$\mathcal{Z}_M = \left[ 1 \quad \frac{z}{\lambda^{1/2}} \quad \frac{z^2}{\lambda^{2/2}} \quad \cdots \quad \frac{z^{M-1}}{\lambda^{\frac{M-1}{2}}} \right]^T.$$

Por simplicidade a dependência no tempo dos polinômios foi omitida. Embora este lema não seja demonstrado aqui, vale a pena fazer algumas observações a respeito:

- ◊ A condição *l.2*) é equivalente a existência de  $\Phi_{M-1}^{-1}(n)$  e se satisfeita, as condições *l.1*) e *l.3*) são equivalentes a matriz  $\Phi_{M-1}^{-1}(n)$  ser positiva definida, conforme apresentado em [10] e [31].
- ◊ A igualdade (4.18) é equivalente a afirmar que o posto de deslocamento da matriz de autocorrelação é três. Tal fato está mostrado no apêndice A, equação (A.31).
- ◊ A igualdade de polinômios da condição *l.2*) é facilmente obtida a partir de (4.18) multiplicando esta igualdade pela esquerda por  $\mathcal{Z}_M^T$  e pela direita por  $\mathcal{W}_M$

$$\mathcal{W}_M = \left[ 1 \quad \frac{w}{\lambda^{1/2}} \quad \frac{w^2}{\lambda^{2/2}} \quad \cdots \quad \frac{w^{M-1}}{\lambda^{\frac{M-1}{2}}} \right]^T,$$

resultando em

$$\begin{aligned} (1 - zw) \mathcal{Z}_{M-1}^T \Phi_{M-1}^{-1}(n) \mathcal{W}_{M-1} &= \\ &= \mathcal{Z}_M^T (\mathbf{G}_M(n) \mathbf{G}_M^T(n) + \mathbf{F}_M(n) \mathbf{F}_M^T(n) - \mathbf{B}_M(n) \mathbf{B}_M^T(n)) \mathcal{W}_M. \end{aligned}$$

Para obter *l.2* basta escolher  $zw = 1$ .

- ◊ Pode-se dizer também que as condições *l.1*, *l.2* e *l.3* garantem simplesmente que os coeficientes dos filtros transversais da predição progressiva, da predição regressiva e do ganho de Kalman tem propriedades que os caracterizam como solução de um problema dos mínimos quadrados com pré-janelamento.

Se o lema 1 é satisfeito, então é possível obter de (4.18) uma matriz  $\Phi_{M-1}^{-1}(n)$  que permite a fatoração na forma  $\Phi_{M-1}^{-1}(n) = (\mathbf{A}_{M-1}^T(n) \mathbf{A}_{M-1}(n))^{-1}$ . Obtendo-se assim, uma matriz de dados sequenciais pré-janelada para todo  $n \geq 2M$ . Nestas condições pode-se associar ao conjunto de parâmetros a solução exata de um problema de predição pré-janelado.

Feito isso, é necessário mostrar que com os elementos do vetor de estados

$$\zeta(n) = \begin{bmatrix} \mathbf{dq}\mathbf{1}_M^f(n) \\ \xi_M^{f/2}(n) \\ \bar{\mathbf{b}}_M(n) \end{bmatrix}$$

é possível obter vetores candidatos aos vetores de coeficientes dos filtros das predições progressiva e regressiva e ao vetor do ganho de Kalman para todas as ordens satisfazendo o lema 1. Isto pode ser mostrado considerando a equação (C.23) do apêndice C para a atualização na ordem dos polinômios  $F_i(\frac{z}{\lambda^{1/2}})$ ,  $B_i(\frac{z}{\lambda^{1/2}})$  e  $G_i(\frac{z}{\lambda^{1/2}})$ :

$$\begin{bmatrix} F_{i+1}(\frac{z}{\lambda^{1/2}}) \\ B_{i+1}(\frac{z}{\lambda^{1/2}}) \\ G_{i+1}(\frac{z}{\lambda^{1/2}}) \end{bmatrix} = \Sigma_i(n, z) \begin{bmatrix} F_i(\frac{z}{\lambda^{1/2}}) \\ B_i(\frac{z}{\lambda^{1/2}}) \\ G_i(\frac{z}{\lambda^{1/2}}) \end{bmatrix} \quad (4.22)$$

onde

$$\Sigma_i(n, z) = \begin{bmatrix} 1 & -\frac{\sin \phi_i(n)}{\cos \phi_i(n)} & 0 \\ \frac{\sin \phi_i(n)}{\cos \phi_i(n)} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{z}{\cos \theta_i^b(n)} & \frac{\sin \theta_i^b(n)}{\cos \theta_i^b(n)} \\ 0 & \frac{z \sin \theta_i^b(n)}{\cos \theta_i^b(n)} & \frac{1}{\cos \theta_i^b(n)} \end{bmatrix}.$$

Como observado no apêndice C, embora estes polinômios dependam de  $n$ , tal fato é omitido em (4.22) para facilitar a notação. Além disso, como também observado no apêndice C, para ordem zero vale

$$F_1(\frac{z}{\lambda^{1/2}}) = B_1(\frac{z}{\lambda^{1/2}}) = \xi_0^{-f/2}(n) \quad \text{e} \quad G_1(\frac{z}{\lambda^{1/2}}) = 0,$$

e no instante zero o polinômio  $G_i(\frac{z}{\lambda^{1/2}})$  é nulo para  $1 \leq i \leq M$ . Dados  $\mathbf{dq}\mathbf{1}_M^f(n)$  e  $\xi_M^{f/2}(n)$ , a expressão (3.96), permite calcular os coeficientes de reflexão da estrutura em treliça normalizada, isto é, os coeficientes  $\sin \phi_i(n)$ . Como  $|\phi_i(n)| < \pi/2$  os ângulos  $\phi_i(n)$  ficam especificados. Analogamente os ângulos  $\theta_i^b(n)$  são obtidos a partir do vetor  $\bar{\mathbf{b}}_M(n)$  através de (3.111). Desta forma, especificados  $\phi_i(n)$  e  $\theta_i^b(n)$  a expressão (4.22) permite calcular os correspondentes polinômios  $F_M(\frac{z}{\lambda^{1/2}})$ ,  $B_M(\frac{z}{\lambda^{1/2}})$  e  $G_M(\frac{z}{\lambda^{1/2}})$ . Em [31] está demonstrado formalmente que a partir de um vetor de estados  $\zeta(n)$  qualquer, pertencente a  $\mathcal{S}_e(n)$ , é possível calcular os polinômios  $F_M(\frac{z}{\lambda^{1/2}})$ ,  $B_M(\frac{z}{\lambda^{1/2}})$  e  $G_M(\frac{z}{\lambda^{1/2}})$  satisfazendo o lema 1. Estes estão portanto associados a uma solução exata de um problema de predição. Levando a concluir que qualquer estado  $\zeta(n)$  que satisfaz (4.14) está associado a uma solução exata de um problema de predição, sendo então alcançável.

### Demonstração de 5b

A seguir, demonstra-se a equivalência entre (4.14) e (4.15). A equação (3.94) pode ser reescrita como

$$\xi_i^{f/2}(n) = \xi_{i-1}^{f/2}(n) \cos \phi_i(n),$$

o que permite relacionar convenientemente a energia da predição progressiva de ordem  $M$  e a energia de ordem zero através da seguinte expressão:

$$\xi_M^{f/2}(n) = \xi_0^{f/2}(n) \prod_{i=1}^M \cos \phi_i(n). \quad (4.23)$$

As energias de erro de predição satisfazem  $\xi_i^{f/2}(n) \geq 0$ ,  $0 \leq i \leq M$ . Como  $u(0) \neq 0$ , então  $\xi_0^f(n) = \sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell)|^2 > 0$ . Desta forma, como  $\xi_M^{f/2}(n) > 0$ , necessariamente  $\xi_{i-1}^{f/2}(n) > 0$  e  $\cos \phi_i(n) > 0$  para todo  $1 \leq i \leq M$ . Por outro lado, se  $\cos \phi_i(n) > 0$  para todo  $1 \leq i \leq M$  e  $u(n) \neq 0$  é imediato de (4.23) que  $\xi_M^{f/2}(n) > 0$ . Desta forma conclui-se que

$$\xi_M^{f/2}(n) > 0 \Leftrightarrow |\phi_i(n)| < \pi/2, \quad \text{para } 1 \leq i \leq M.$$

De (3.102) obtém-se

$$\gamma_i^{1/2}(n) = \gamma_{i-1}^{1/2}(n) \cos \theta_i^b(n).$$

Como  $0 < \gamma_i^{1/2}(n) \leq 1$  para todo  $0 \leq i \leq M$ , então  $|\theta_i^b(n)| < \pi/2$  para  $1 \leq i \leq M$ . Lembrando que  $\gamma_0^{1/2}(n) = 1$  pode-se relacionar  $\gamma_M^{1/2}(n)$  com os ângulos  $\theta_i^b(n)$  através da seguinte expressão

$$\gamma_M^{1/2}(n) = \prod_{i=1}^M \cos \theta_i^b(n). \quad (4.24)$$

Usando (4.24) a equação (2.155) pode ser reescrita como

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \prod_{i=1}^M \cos \theta_i^b(n);$$

de onde conclui-se que

$$\|\bar{\mathbf{b}}_M(n)\| < 1 \Leftrightarrow |\theta_i^b(n)| < \pi/2, \quad \text{para } 1 \leq i \leq M.$$

### Demonstração de 5c

Da definição, um  $\alpha$ -interior de  $\mathcal{S}_e(n)$  é constituído pelos estados  $\zeta(n)$  que satisfazem

$$\xi_M^{f/2}(n) \geq \alpha > 0 \quad \text{e} \quad \|\bar{\mathbf{b}}_M(n)\| \leq 1 - \alpha < 1,$$

para  $0 < \alpha < 1$ . De (4.23) resulta que

$$\xi_M^{f/2}(n) = \xi_0^{f/2}(n) \prod_{i=1}^M \cos \phi_i(n) \geq \alpha,$$

de onde é possível concluir que

$$\cos \phi_i(n) \geq \frac{\alpha}{\xi_0^{f/2}(n)}.$$

Portanto  $|\phi_i(n)| \leq \alpha' < \pi/2$ , onde  $\alpha' = \arccos(\alpha \xi_0^{-f/2}(n))$ .

De (2.155) resulta que

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \prod_{i=1}^M \cos \theta_i^b(n) \leq (1 - \alpha)^2,$$

ou ainda,

$$\prod_{i=1}^M \cos \theta_i^b(n) \geq 1 - (1 - \alpha)^2 = \alpha(2 - \alpha).$$

Desta forma resulta que

$$\cos \theta_i^b(n) \geq \alpha(2 - \alpha), \quad \text{para} \quad 1 \leq i \leq M.$$

Portanto  $|\theta_i^b(n)| \leq \alpha'' < \pi/2$ , onde  $\alpha'' = \arccos(\alpha(2 - \alpha))$ .

Escolhendo  $\alpha_1 = \max\{\alpha', \alpha''\}$  e  $\alpha_2 = \alpha$  tem-se as condições (4.16) e (4.17), isto é

$$|\phi_i(n)| \leq \alpha_1 < \pi/2 \quad \text{e} \quad |\theta_i^b(n)| \leq \alpha_1 < \pi/2 \quad (4.25)$$

e

$$\xi_M^{f/2}(n) \geq \alpha_2 > 0. \quad (4.26)$$

Estas condições também definem um interior do conjunto  $\mathcal{S}_e(n)$ .

A propriedade 5d é consequência direta da definição de borda de  $\mathcal{S}_e(n)$  e da propriedade 5a.

$\triangle\triangle$

Estabelecidos a minimalidade do algoritmo e o domínio de estabilidade é necessário demonstrar a propriedade 2 que relaciona propriedades da excitação com propriedades do conjunto dos estados alcançáveis.

### 4.2.3 Demonstração da Propriedade 2

2a.i) A evolução do vetor de estados está restrita a borda  $\delta S_e(n)$  se, e somente se, a excitação  $\{u(\cdot)\}$  for perfeitamente preditível.

**Demonstração:** Para demonstrar 2a.i associa-se  $\xi_M^{f/2}(n) = 0$  e/ou  $\|\bar{\mathbf{b}}_M(n)\| = 1$  a sinais perfeitamente preditíveis.

Como  $\xi_M^{f/2}(n)$  é a energia do erro da predição progressiva, então  $\xi_M^{f/2}(n) = 0$  se, e somente se, a seqüência de entrada é perfeitamente preditível. Cabe notar que, assumindo uma seqüência de observação  $\{u(\cdot)\}$  não nula, segue da equação (4.23) que  $\xi_M^{f/2}(n) = 0$  se, e somente se,  $|\phi_i(n)| = \pi/2$  para algum  $i \leq M$ .

Falta então associar  $\|\bar{\mathbf{b}}_M(n)\| = 1$  a sinais perfeitamente preditíveis. Inicialmente, é necessário observar que se for considerado  $M = n + 1$ , então vale  $\|\bar{\mathbf{b}}_M(n)\| = 1$  para qualquer que seja o sinal de observação. Para se chegar a esta conclusão basta considerar a decomposição QR da matriz de dados  $\mathbf{A}_M(n)$  que, neste caso se reduz a

$$\mathbf{Q}_M(n)\mathbf{A}_M(n) = \mathbf{R}_M(n)$$

com  $\mathbf{Q}\mathbf{1}_M(n) = \mathbf{Q}_M(n)$  (ver equações (2.66) a (2.70)), pois  $M = n + 1$ . Como  $\mathbf{Q}\mathbf{1}_M^T(n)$  é uma matriz unitária e sua última linha é exatamente o vetor  $\bar{\mathbf{b}}_M^T(n)$  (ver seção 2.5, equação (2.138)), então vale  $\|\bar{\mathbf{b}}_M(n)\| = 1$ , para  $M = n + 1$ . No caso de um sinal perfeitamente preditível existe  $i_0$  tal que  $\bar{b}_i(n) = 0$ , para  $i \geq i_0$ . Resulta então, para qualquer  $M \geq i_0$  que

$$\|\bar{\mathbf{b}}_M(n)\|^2 = \sum_{i=1}^M |\bar{b}_{i-1}(n)|^2 = \underbrace{\sum_{i=1}^{i_0} |\bar{b}_{i-1}(n)|^2}_{\|\bar{\mathbf{b}}_{i_0}(n)\|^2} + \underbrace{\sum_{i=i_0+1}^{n+1} |\bar{b}_{i-1}(n)|^2}_0 = \|\bar{\mathbf{b}}_{i_0}(n)\|^2 = 1.$$

De (2.155) segue que para qualquer  $M \geq i_0$

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \gamma_M(n) = \|\bar{\mathbf{b}}_{i_0}(n)\|^2 = 1 - \gamma_{i_0}(n) = 1,$$

ou seja,

$$\gamma_i(n) = \gamma_{i_0}(n) = 0, \quad \text{para } i \geq i_0.$$

Consequentemente, para um sinal perfeitamente preditível, considerando um problema de ordem  $M$ , vale

$$\|\bar{\mathbf{b}}_M(n)\| = 1 \quad \text{e} \quad \gamma_M(n) = 0.$$

Para completar, é necessário demonstrar que se  $\|\bar{\mathbf{b}}_M(n)\| = 1$ , então o sinal é preditível. Da equação (2.155) segue que  $\|\bar{\mathbf{b}}_M(n)\| = 1$  se, e somente se,  $\gamma_M(n) = 0$ , o que por sua vez acontece se, e somente se,  $|\theta_i^b(n)| = \pi/2$  para algum  $i \leq M$ . Considerando  $i_0$  o índice para o qual isto acontece, então  $\cos \theta_{i_0}^b(n) = 0$ . Como

$$\|\bar{\mathbf{b}}_M(n)\|^2 = \|\bar{\mathbf{b}}_{i_0}(n)\|^2 + \sum_{i=i_0+1}^M |\bar{b}_{i-1}(n)|^2 = 1$$

e

$$\|\bar{\mathbf{b}}_{i_0}(n)\|^2 = 1 - \gamma_{i_0}(n) = 1 - \underbrace{\prod_{i=1}^{i_0} \cos \theta_i^b(n)}_0 = 1,$$

então  $\sum_{i=i_0+1}^M |\bar{b}_{i-1}(n)|^2 = 0$  para  $\forall M > i_0$ . Desta forma, como o erro de predição é nulo, o sinal é perfeitamente preditível.

$\overline{\Delta\Delta}$

Cabe notar que devido ao pré-janelamento as colunas da matriz de dados são sempre linearmente independentes para qualquer seqüência de entrada  $\{u(\cdot)\}$  não nula, excluindo assim teoricamente a existência de sinais perfeitamente preditíveis. Entretanto, pode-se escolher seqüências de entrada de tal forma que a matriz de dados fique, com o passar do tempo, arbitrariamente próxima de uma situação de posto deficiente. Isto faz com que o algoritmo passe a operar arbitrariamente próximo da borda do domínio de estabilidade  $\mathcal{S}_e(\cdot)$ .

A 'distância' da matriz  $\mathbf{A}_M(n)$  de uma situação de posto deficiente pode ser medida utilizando-se a dispersão  $\chi(n)$  dos seus valores singulares definida como

$$\chi(n) = \frac{s_{\max}(n)}{s_{\min}(n)},$$

onde  $s_{\max}(n)$  e  $s_{\min}(n)$  são respectivamente os valores singulares máximo e mínimo de  $\mathbf{A}_M(n)$ . Antes de demonstrar a propriedade 2a.ii será demonstrada a seguinte propriedade para um sinal de energia  $\xi_0^f(n)$  finita:

**Propriedade 6:**  $\zeta(n)$  pertence a um  $\alpha$ -interior de  $\mathcal{S}_e(n) \Leftrightarrow$  existe uma constante  $r_0 > 0$  tal que  $\chi(n) < r_0 < \infty$ .

( $\Leftarrow$ ) Considerando que  $\zeta(n)$  não evolui em um  $\alpha$ -interior <sup>2</sup>, então  $\zeta(n)$  se aproxima arbitrariamente próximo da borda. Ora, da propriedade 2a.i, em que se mostrou que

<sup>2</sup>Prova-se que  $A \Leftarrow B$ , provando que  $(\text{não } A) \Rightarrow (\text{não } B)$ .



uma evolução na borda do conjunto de estabilidade acontece se, e somente se, o sinal é perfeitamente preditível, conclui-se que o sinal deve se aproximar arbitrariamente de uma situação em que se torna perfeitamente preditível. Isto significa por sua vez que,  $\chi(n)$  cresce indefinidamente e não pode ser limitada.

( $\Rightarrow$ ) Basta demonstrar que  $s_{\max}^2(n)$  e  $s_{\min}^{-2}(n)$  são limitados superiormente.

O limite superior para  $s_{\max}^2(n)$  pode ser obtido considerando as relações:

$$s_{\max}^2(n) \leq \sum_{i=1}^M s_i^2(n) = \text{traço}(\Phi_M(n)) = \sum_{i=0}^{M-1} \xi_o^f(n-i) \leq \frac{M}{\lambda^{M-1}} \xi_o^f(n).$$

Para obter o limite superior de  $s_{\min}^{-2}(n)$ , considera-se

$$\begin{aligned} s_{\min}^{-2}(n) &\leq \sum_{i=1}^M s_i^{-2}(n) = \text{traço}(\Phi_M^{-1}(n)) = \text{traço}(\mathbf{R}_M^{-1}(n)\mathbf{R}_M^{-T}(n)) = \\ &= \|\mathbf{R}_M^{-1}(n)\|_F^2 = \sum_{i=1}^M \|\mathbf{B}_i(n)\|^2; \end{aligned}$$

onde  $\mathbf{B}_i(n)$  são os vetores de coeficientes normalizados do filtro de erro de predição regressiva, conforme definidos em (A.25) e  $\|\cdot\|_F$  indica a norma de Frobenius. Considere-se (C.16)

$$\underbrace{\begin{bmatrix} \mathbf{F}_{i+1}^T(n) \\ \mathbf{B}_{i+1}^T(n) \\ \mathbf{G}_{i+1}^T(n) \end{bmatrix}}_{\mathbf{M}_1} = \Sigma_i(n) \underbrace{\begin{bmatrix} [\mathbf{F}_i^T(n) \ 0] \\ [0 \ \mathbf{B}_i^T(n)] \\ [\mathbf{G}_i^T(n) \ 0] \end{bmatrix}}_{\mathbf{M}_4}$$

com a matriz  $\Sigma_i(n)$  representada como em (C.19), isto é,

$$\Sigma_i(n) = \frac{1}{\cos \phi_i(n) \cos \theta_i^b(n)} \cdot \underbrace{\begin{bmatrix} 1 & -\sin \phi_i(n) & 0 \\ -\sin \phi_i(n) & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{M}_2} \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \lambda^{1/2} & \sin \theta_i^b(n) \\ 0 & \lambda^{1/2} \sin \theta_i^b(n) & 1 \end{bmatrix}}_{\mathbf{M}_3}.$$

Verifica-se facilmente que se  $\|\mathbf{B}_i(n)\|$ ,  $\|\mathbf{F}_i(n)\|$  e  $\|\mathbf{G}_i(n)\|$  são limitadas então, com  $|\phi_i(n)| \leq \alpha_1 < \pi/2$  e  $|\theta_i^b(n)| \leq \alpha_1 < \pi/2$ , as normas de  $\|\mathbf{B}_{i+1}(n)\|$ ,  $\|\mathbf{F}_{i+1}(n)\|$  e

$\|\mathbf{G}_{i+1}(n)\|$  também são limitadas. Para isto, basta considerar a norma de Frobenius das matrizes  $\mathbf{M}_2$  e  $\mathbf{M}_3$  e as colunas das matrizes  $\mathbf{M}_1$  e  $\mathbf{M}_4$ . Representando as colunas de  $\mathbf{M}_1$  e  $\mathbf{M}_4$  por  $\mathbf{m}_{1,j}$  e  $\mathbf{m}_{4,j}$ , com  $j = 1, 2, \dots, M$ , obtém-se

$$\|\mathbf{m}_{1,j}\|^2 \leq \frac{1}{\cos^2 \phi_i(n) \cos^2 \theta_i^b(n)} \|\mathbf{M}_2\|_F^2 \|\mathbf{M}_3\|_F^2 \|\mathbf{m}_{4,j}\|^2.$$

Somando para todos os valores de  $j$  resulta

$$\begin{aligned} & \|\mathbf{B}_{i+1}(n)\|^2 + \|\mathbf{F}_{i+1}(n)\|^2 + \|\mathbf{G}_{i+1}(n)\|^2 \leq \\ & \leq \frac{1}{\cos^2 \phi_i(n) \cos^2 \theta_i^b(n)} \|\mathbf{M}_2\|_F^2 \|\mathbf{M}_3\|_F^2 \cdot \\ & \cdot (\|\mathbf{B}_i(n)\|^2 + \|\mathbf{F}_i(n)\|^2 + \|\mathbf{G}_i(n)\|^2). \end{aligned}$$

Como

$$\|\mathbf{M}_2\|_F^2 = 3 + 2 \sin^2 \phi_i(n),$$

$$\|\mathbf{M}_3\|_F^2 = 2 + \lambda + (1 + \lambda) \sin^2 \theta_i^b(n)$$

e

$$\frac{1}{\cos^2 \phi_i(n) \cos^2 \theta_i^b(n)} \leq \frac{1}{\cos^2 \alpha_1} < \infty,$$

então as normas de  $\|\mathbf{B}_{i+1}(n)\|$ ,  $\|\mathbf{F}_{i+1}(n)\|$  e  $\|\mathbf{G}_{i+1}(n)\|$  são limitadas superiormente caso as normas de  $\|\mathbf{B}_i(n)\|$ ,  $\|\mathbf{F}_i(n)\|$  e  $\|\mathbf{G}_i(n)\|$  também o sejam. Como  $\mathbf{B}_1(n) = 1$ ,  $\mathbf{F}_1(n) = 1$  e  $\mathbf{G}_1(n) = 0$ , então por indução as normas de  $\|\mathbf{B}_i(n)\|$ ,  $\|\mathbf{F}_i(n)\|$  e  $\|\mathbf{G}_i(n)\|$  também são limitadas para todo  $i$  entre 1 e  $M$ . Desta forma existe necessariamente uma constante  $\alpha_3$  que limita superiormente  $\|\mathbf{R}_M^{-1}(n)\|_F^2$ . Com isso

$$s_{\min}^{-2}(n) \leq \alpha_3;$$

resultando, portanto

$$\chi(n) = \frac{s_{\max}(n)}{s_{\min}(n)} \leq \sqrt{\frac{M}{\lambda^{M-1}} \alpha_3 \xi_o^f(n)}.$$

Consequentemente para um sinal de potência finita, se o vetor de estados evolui em um  $\alpha$ -interior de  $S_e(n)$ , então necessariamente a matriz de dados tem um espalhamento de valores singulares limitado.

$\overline{\Delta\Delta}$

2a.ii) A evolução do vetor de estados está restrita a um  $\alpha$ -interior de  $\mathcal{S}_e(n)$  se, e somente se,  $\{u(\cdot)\}$  é uma excitação persistente.

**Demonstração [31]:** Uma excitação é dita persistente quando existem duas constantes positivas  $b_1$  e  $b_2$  tais que

$$0 < b_1 \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq b_2 \mathbf{I} < \infty. \quad (4.27)$$

No apêndice D está demonstrado que a excitação é persistente se, e somente se, a energia do sinal pré-janelado é uniformemente limitada superiormente e inferiormente e o espalhamento dos valores singulares é uniformemente limitado superiormente. Isto é equivalente a afirmar que, se existem as constantes positivas  $b_1$  e  $b_2$  que satisfazem (4.27) então, existem constantes positivas  $\beta_1$ ,  $\beta_2$  e  $r_0$  tais que

$$0 < \beta_1 \leq \xi_0^{f/2}(n) \leq \beta_2 < \infty$$

e

$$\chi(n) < r_0 < \infty.$$

Como já foi visto pela propriedade 6 estas últimas condições são equivalentes a garantir que o estado  $\zeta(n)$  pertence a um  $\alpha$ -interior de  $\mathcal{S}_e(n)$ .

$\overline{\Delta\Delta}$

#### 4.2.4 Consistência retrógrada

Para assegurar que o algoritmo da tabela 3.3 possui a propriedade de consistência retrógrada, isto é  $\mathcal{S}_f(n) \subset \mathcal{S}_e(n)$ , é necessário garantir que (4.14) e (4.15) sejam satisfeitas mesmo na presença de erros numéricos. Quanto a  $\|\bar{\mathbf{b}}_M(n)\| < 1$ , mostrou-se em [31] que é possível garantir tal condição quando as rotações  $\{\phi_i(n)\}$  que propagam as variáveis  $b_{i-1}(n)$  são implementadas de forma passiva. Isto significa que a norma do vetor rotacionado deve ser menor ou igual a norma do vetor antes da rotação, o que pode ser assegurado tanto em uma implementação em CORDIC [31], como em um computador de uso geral ou processador de sinais. Nesses dois últimos casos é necessário utilizar o truncamento do módulo como regra de quantização, o que no caso de aritmética de ponto flutuante corresponde normalmente ao simples truncamento da mantissa. Cabe observar que este tipo de rotação não fornece média

zero para as perturbações levando a um vício (acúmulo de 'bias'), particularmente quando se usa poucos bits.

A outra condição a ser assegurada é  $\xi_M^f(n) > 0$ . Como a variável é calculada como a norma do vetor

$$\left[ \lambda^{1/2} \xi_M^f(n-1) \quad \epsilon_M^f(n) \right],$$

a condição  $\xi_M^f(n) > 0$  pode ser facilmente satisfeita por qualquer método de cálculo de norma que não forneça resultados inconsistentes.

### Seqüências de dados mal condicionadas

Se o sinal de entrada for tal que, apesar do pré-janelamento, com o passar do tempo ele se aproxime arbitrariamente próximo da situação de sinal preditível então pode-se chegar a situação em que  $\xi_M^f(n) = 0$ . Supondo é claro, que o método de cálculo da norma permite obter um resultado nulo para vetores

$$\left[ \lambda^{1/2} \xi_M^f(n-1) \quad \epsilon_M^f(n) \right]$$

muitos pequenos. Nessa situação, para a precisão numérica utilizada, o sinal é preditível e o algoritmo passa a operar na borda  $\delta S_e(n)$  do conjunto de estados alcançáveis e não mais em um  $\alpha$ -interior. Ou ainda, para a precisão numérica utilizada, a excitação não é persistente. Nestas condições, uma outra convenção numérica é importante para assegurar o bom comportamento do algoritmo.

Nos passos 2, 3 e 5 do algoritmo da tabela 4.1 é possível acontecer que os dois elementos a serem apresentados para calcular o ângulo sejam nulos. Neste caso considera-se a convenção numérica sugerida em [31], a qual consiste em escolher os ângulos de rotação igual a zero quando o vetor do lado direito das equações (3.122), (3.96) e (3.111) são vetores nulos. Esta convenção assegura que os ângulos de rotação sejam funções contínuas do elemento a ser eliminado, sendo além disto, facilmente implementada tanto em CORDIC como em um computador de uso geral ou processador de sinais.

A partir da expressão (4.23), conclui-se que se  $\xi_M^f(n) = 0$  então existe um índice  $i_0 \leq M$  para o qual vale  $|\phi_{i_0}(n)| = \pi/2$ , o que por sua vez implica em

$$\xi_{i_0}^f(n) = \xi_{i_0+1}^f(n) = \dots = \xi_M^f(n) = 0.$$

Como necessariamente

$$\epsilon_{i_0}^f(n) = \epsilon_{i_0+1}^f(n) = \dots = \epsilon_M^f(n) = 0,$$

então com a convenção numérica adotada resulta

$$\phi_{i_0+1}(n) = \phi_{i_0+2}(n) = \dots = \phi_M(n) = 0.$$

Desta forma, como  $|\phi_i(n)| < \pi/2$ , para  $1 \leq i < i_0$  e  $|\phi_{i_0}(n)| = \pi/2$ , somente um dos ângulos  $\phi_i(n)$  atinge a borda do hipercubo  $|\phi_i(n)| < \pi/2$ ,  $1 \leq i \leq M$ . Nesta situação, o filtro em treliça degenera. A figura 4.2 ilustra, neste caso, a configuração da estrutura em treliça para uma ordem  $M = 3$  e  $i_0 = 2$ . A parte correspondente aos índices inferiores a  $i_0$  formam um circuito marginalmente estável. A parte corres-

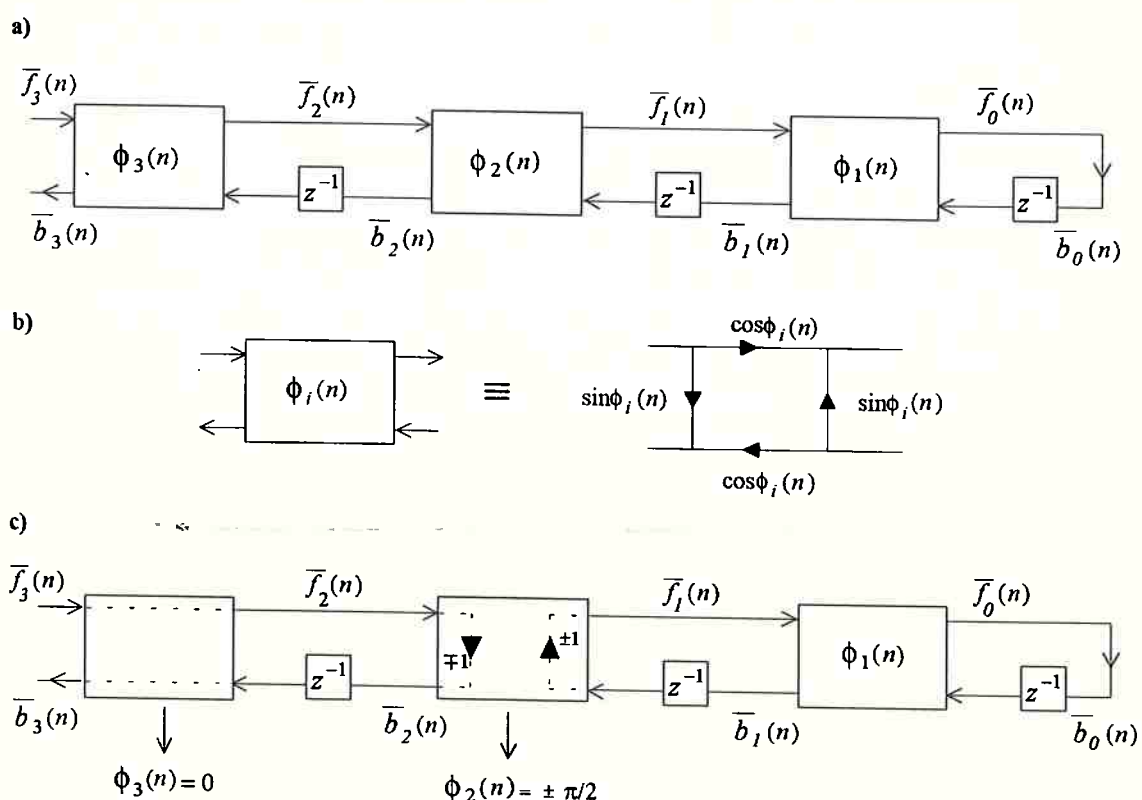


Figura 4.2: a) Filtro em treliça com erro *a posteriori* normalizado com  $M = 3$ ; b) Detalhe de cada bloco; c) Situação em que o posto da matriz de dados é deficiente com  $i_0 = 2$ .

pondente aos índices superiores a  $i_0$  degeneram para uma linha de atrasos, de modo que os erros de predição regressivos  $\bar{b}_i(n)$  são nulos, já que são versões atrasadas do

erro de predição progressiva  $f_M(n)$  que também é nulo. Como  $\bar{b}_i(n) = 0$  e  $\gamma_i(n) = 0$  para  $i \geq i_0$  resulta do passo 5 da tabela 4.1 e da convenção numérica adotada, que os ângulos  $\theta_i^b(n)$  são nulos para  $i_0 < i \leq M$ , sendo  $|\theta_{i_0}^b(n)| = \pi/2$ . Desta forma, as correspondentes seções da estrutura de McWhirter (passos 1 e 6) desaparecem e a ordem efetiva do algoritmo se reduz ao posto da matriz  $\mathbf{A}_M(n)$ .

### Inicialização

Considerações mais detalhadas sobre o algoritmo da tabela 4.1 são apresentadas em [31]. Cabe entretanto, chamar a atenção que o algoritmo permite a utilização de uma inicialização exata. Esta é feita quando se supõe uma seqüência de observação nula anterior ao instante  $n = 0$ , a qual leva o vetor de estados para zero. A seqüência de observação nula é perfeitamente preditível e nestas circunstâncias o vetor de estados  $\zeta(n)$  permanece na borda do conjunto de estados alcançáveis. Segundo [31] esta forma de inicialização é conhecida por apresentar problemas numéricos. Isto pode ser explicado no caso do algoritmo em questão, pelo fato de que no período de inicialização os estados evoluem na borda da região de estabilidade  $\delta\mathcal{S}_e(n)$  ou arbitrariamente próxima a ela [31]. Como estes estados não estão em um  $\alpha$ -interior de  $\mathcal{S}_e(n)$ , a estabilidade exponencial não é garantida. Desta forma, os erros numéricos podem se acumular de forma significativa neste período. Além disso, caso não sejam respeitadas as convenções numéricas já mencionadas, pequenas perturbações podem fazer com que os estados saiam do domínio de estabilidade resultando em estados inconsistentes com um problema dos mínimos quadrados. Caso isto ocorra, tais erros podem levar a instabilidade do algoritmo. No caso do algoritmo QR-LSL *a posteriori* é necessário utilizar as convenções numéricas mencionadas anteriormente para que os problemas numéricos associados a uma inicialização exata não instabilizem o algoritmo. Por outro lado, inicializações não exatas equivalem a uma escolha do vetor de estados  $\zeta(0) \neq 0$ , pertencentes a um  $\alpha$ -interior do domínio de estabilidade  $\mathcal{S}_e(n)$ , o que explica o melhor comportamento numérico de tal tipo de inicialização.

### 4.3 Algoritmo QR-LSL com erros a priori

Nesta seção é considerado o algoritmo QR-LSL *a priori* apresentado nas tabelas 3.4 e 4.2. Mostra-se inicialmente que este algoritmo é mínimo e é estabelecido o conjunto dos estados alcançáveis  $\mathcal{S}_e(n)$  associado. Por analogia, ao apresentado na seção anterior, mostra-se que com excitação persistente o algoritmo opera em um  $\alpha$ -interior do domínio de estabilidade  $\mathcal{S}_e(n)$  e no caso de excitações preditíveis, opera em uma borda generalizada do domínio de estabilidade. São também consideradas as convenções numéricas necessárias para garantir a consistência retrógrada e conseqüentemente, a estabilidade retrógrada do algoritmo. Para encerrar são feitos comentários sobre a inicialização do algoritmo.

#### 4.3.1 Minimalidade

Na seção 4.1 foi mostrado que o vetor de estados usado para descrever o algoritmo QR-LSL *a priori* é dado por

$$\zeta(n) = \begin{bmatrix} dq1_M^f(n) \\ \xi_M^{f/2}(n) \\ \bar{\psi}_M(n) \end{bmatrix}. \quad (4.28)$$

Como a dimensão deste vetor é  $2M + 1$  este algoritmo também é mínimo. Analogamente ao que acontece no algoritmo QR-LSL *a posteriori*, a rigor o algoritmo da tabela 3.4, não é mínimo, sendo que somente o da tabela 4.2 o é. Entretanto, com os cálculos realizados em ambas as versões são exatamente os mesmos, eles são numericamente equivalentes.

#### 4.3.2 Caracterização do domínio de estabilidade

**Propriedade 5':** *Características do domínio de estabilidade  $\mathcal{S}_e(n)$*

5'a) Considerando  $u(0) \neq 0$  e  $n > 2M$ , o conjunto de estados alcançáveis  $\mathcal{S}_e(n)$  consiste de todos os estados  $\zeta(n)$  que satisfazem:

$$\xi_M^{f/2}(n) > 0 \quad \text{e} \quad \|\bar{\psi}_M(n)\| < \infty. \quad (4.29)$$

5'b) As condições (4.29) são equivalentes a

$$|\phi_i(n)| < \pi/2 \quad \text{e} \quad |\theta_i^b(n)| < \pi/2, \quad \text{para} \quad 1 \leq i \leq M. \quad (4.30)$$

5'c) Um  $\alpha$ -interior de  $\mathcal{S}_e(n)$  é o conjunto de estados que, além de satisfazer (4.29) e (4.30), ainda satisfazem

$$|\phi_i(n)| \leq \alpha_1 < \pi/2, \quad \text{para } 1 \leq i \leq M. \quad (4.31)$$

e

$$\xi_M^{f/2}(n) > \alpha_2 > 0, \quad (4.32)$$

sendo  $\alpha_1$  e  $\alpha_2$  constantes positivas.

5'd) A borda do conjunto de estados alcançáveis  $\delta\mathcal{S}_e(n)$  é formada pelos estados que satisfazem

$$\xi_M^{f/2}(n) = 0. \quad (4.33)$$

### Demonstração da Propriedade 5'

Comparando (4.28) e (4.13) observa-se que a diferença entre os vetores de estados consiste apenas na forma como o erro de predição regressiva está representado, em um vetor de estados se usa o erro de predição regressiva *a posteriori*  $\bar{\mathbf{b}}_M(n)$  e no outro o erro de predição regressiva *a priori*  $\bar{\boldsymbol{\psi}}_M(n)$ . De (3.64) tem-se a seguinte relação entre os elementos destes vetores

$$\bar{b}_{i-1}(n) = \gamma_i^{1/2}(n) \gamma_{i-1}^{1/2}(n) \bar{\psi}_{i-1}(n), \quad 1 \leq i \leq M. \quad (4.34)$$

Esta expressão mostra que existe uma correspondência biunívoca entre  $\bar{b}_{i-1}(n)$  e  $\bar{\psi}_{i-1}(n)$ ,  $1 \leq i \leq M$ , quando  $0 < \gamma_i(n) \leq 1$ ,  $0 \leq i \leq M$ . Como vale

$$\gamma_i^{1/2}(n) = \gamma_{i-1}^{1/2}(n) \cos \theta_i^b(n), \quad 1 \leq i \leq M,$$

(ver (3.102)) basta a condição  $0 < \gamma_M(n) \leq 1$ , o que é equivalente a  $|\theta_i^b(n)| < \pi/2$ ,  $1 \leq i \leq M$ , para garantir a correspondência biunívoca entre  $\bar{b}_{i-1}(n)$  e  $\bar{\psi}_{i-1}(n)$ . Desta forma, a cada vetor de estados (4.13) do algoritmo QR-LSL *a posteriori* corresponde um único vetor de estados (4.28) do algoritmo QR-LSL *a priori*, quando o fator de conversão  $\gamma_M(n)$  associado satisfaz  $0 < \gamma_M(n) \leq 1$ .

### Demonstração de 5'a

A prova da propriedade 5'a é análoga a demonstração de 5a e consiste na demonstração dos seguintes itens:



5'a.i) Qualquer solução dos mínimos quadrados satisfaz (4.29).

5'a.ii) Qualquer solução satisfazendo (4.29) pode ser de fato alcançável para uma seqüência de entrada apropriada.

#### Demonstração de 5'a.i

Os argumentos para mostrar que  $\xi_M^{f/2}(n) > 0$  para  $u(0) \neq 0$  são os mesmos usados para o algoritmo QR-LSL *a posteriori*. Desta forma, basta mostrar apenas que qualquer solução dos mínimos quadrados satisfaz  $\|\bar{\psi}_M(n)\| < \infty$ . A equação (2.154) implica que  $0 \leq \|\bar{\psi}_M(n)\| < \infty$  se, e somente se,  $1 \leq \gamma_M^{-1}(n) < \infty$ , o que é equivalente a  $0 < \gamma_M(n) \leq 1$ , condição que é obtida em qualquer problema dos mínimos quadrados, desde que a restrição  $u(0) \neq 0$  seja satisfeita.

#### Demonstração de 5'a.ii

Neste item basta mostrar que com os elementos do vetor de estados (4.28), constrõe-se vetores candidatos aos vetores dos coeficientes dos filtros de predição progressiva e regressiva e ao vetor do ganho de Kalman para todas as ordens satisfazendo o lema 1. Os coeficientes de reflexão da estrutura em treliça normalizada, isto é os ângulos  $\phi_i(n)$  são obtidos de forma recorrente a partir de  $\mathbf{dq}1^f(n)$  e  $\xi_M^{f/2}(n)$  através da equação (3.96). Os ângulos  $\theta_i^b(n)$ , que definem a estrutura de McWhirter, são obtidos através da equação (3.112) a partir do vetor  $\bar{\psi}_M(n)$ . Desta forma, utilizando (4.22) pode-se sempre obter os vetores de coeficientes dos filtros da predição progressiva e regressiva e o vetor do ganho de Kalman satisfazendo o lema 1 da seção 4.2. Consequentemente, qualquer estado  $\zeta(n)$  que satisfaz (4.29) está associado a uma solução exata do problema de predição sendo então alcançável.

#### Demonstração de 5'b

Da demonstração 5b da seção 4.2 sabe-se que

$$\xi_M^{f/2}(n) > 0 \Leftrightarrow |\phi_i(n)| < \pi/2$$

e que

$$\|\bar{\mathbf{b}}_M(n)\| < 1 \Leftrightarrow |\theta_i^b(n)| < \pi/2,$$

para  $1 \leq i \leq M$ . Usando (2.154) e (2.155), isto é

$$\|\bar{\psi}_M(n)\|^2 = \gamma_M^{-1}(n) - 1 \text{ e } \|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \gamma_M(n),$$

observa-se que

$$\|\bar{\mathbf{b}}_M(n)\| < 1 \Leftrightarrow 1 \leq \gamma_M^{-1}(n) < \infty \Leftrightarrow \|\bar{\psi}_M(n)\| < \infty.$$

Desta forma conclui-se que

$$\|\bar{\psi}_M(n)\| < \infty \Leftrightarrow |\theta_i^b(n)| < \pi/2, \text{ para } 1 \leq i \leq M.$$

### Demonstração de 5'c

No caso do algoritmo QR-LSL *a priori* um  $\alpha$ -interior é caracterizado por

$$\xi_M^{f/2}(n) \geq \alpha > 0.$$

Não existe uma restrição deste tipo sobre  $\bar{\psi}_M(n)$  pois a única restrição com relação  $\bar{\psi}_M(n)$  é  $\|\bar{\psi}_M(n)\| < \infty$ . Esta restrição é atendida por qualquer vetor  $\bar{\psi}_M(n)$  afetado por uma perturbação  $\delta(n)$  finita. De acordo com a demonstração de 5c, seção 4.2, a condição  $\xi_M^{f/2}(n) \geq \alpha$  é equivalente a

$$|\phi_i(n)| \leq \alpha_1 < \pi/2, \tag{4.35}$$

onde

$$\alpha_1 = \arccos \frac{\alpha}{\xi_0^{f/2}(n)}.$$

A demonstração de 5'd é consequência direta da definição de borda e da propriedade 5'a.

$\overline{\Delta\Delta}$

Estabelecidos a minimalidade e o domínio de estabilidade do algoritmo é necessário demonstrar que o algoritmo satisfaz a propriedade 2.

### 4.3.3 Propriedade 2 para o algoritmo QR-LSL *a priori*

No caso do algoritmo QR-LSL *a priori* é conveniente reformular a propriedade 2a.i da seguinte forma:

**2a.i)** A evolução do vetor de estados está restrita a borda generalizada do conjunto  $\mathcal{S}_e(n)$  se, e somente se,  $\{u(\cdot)\}$  é perfeitamente preditível. Por borda generalizada entende-se aqui como o conjunto

$$\{\zeta(n) \mid \xi_M^{f/2}(n) = 0\} \cup \{\zeta(n) \mid \|\bar{\psi}_M(n)\| = \infty\}.$$

*Prova:* Da demonstração da propriedade **2a.i** para o algoritmo QR-LSL *a posteriori* sabe-se que

$$\xi_M^{f/2}(n) = 0 \quad \text{e} \quad \|\bar{\mathbf{b}}_M(n)\| = 1$$

se, e somente se,  $\{u(\cdot)\}$  é perfeitamente preditível. Como

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \gamma_M(n) \quad \text{e} \quad \|\bar{\psi}_M(n)\|^2 = \gamma_M^{-1}(n) - 1$$

vale

$$\|\bar{\mathbf{b}}_M(n)\| = 1 \quad \Leftrightarrow \quad \gamma_M^{-1}(n) = \infty \quad \Leftrightarrow \quad \|\bar{\psi}_M(n)\| = \infty.$$

$\overline{\Delta\Delta}$

Como no caso da propriedade **2a.i** é conveniente reformular a proposição da propriedade **2a.ii**:

**2a.ii** A evolução do vetor de estados está restrita a um  $\alpha$ -interior de  $\mathcal{S}_e(n)$ , fechado e limitado no que concerne as variáveis  $\xi_M^{f/2}(n)$  e  $\psi_{i-1}$ , para  $1 \leq i \leq M$ , se, e somente se,  $\{u(\cdot)\}$  é uma excitação persistente.

*Prova:* Considerando a propriedade **2a.ii** para o algoritmo QR-LSL *a posteriori*, basta provar que o  $\alpha$ -interior associado ao algoritmo QR-LSL *a posteriori* corresponde a um  $\alpha$ -interior, com as propriedades do enunciado, associado ao algoritmo QR-LSL *a priori*. Uma das características do  $\alpha$ -interior associado ao algoritmo QR-LSL *a posteriori* é

$$\|\bar{\mathbf{b}}_M(n)\| = 1 - \alpha,$$

onde  $0 < \alpha < 1$ . Como

$$\|\bar{\mathbf{b}}_M(n)\|^2 = 1 - \gamma_M(n) \quad \text{e} \quad \|\bar{\psi}_M(n)\|^2 = \gamma_M^{-1}(n) - 1,$$

resulta

$$\gamma_M(n) \geq 1 - (1 - \alpha)^2$$

e

$$\|\bar{\psi}_M(n)\| \leq \frac{1 - \alpha}{\sqrt{1 - (1 - \alpha)^2}} < \frac{1}{\sqrt{\alpha}}.$$

Consequentemente, como em um  $\alpha$ -interior existe uma relação biunívoca entre  $\mathbf{b}_M(n)$  e  $\bar{\boldsymbol{\psi}}_M(n)$  (ver propriedade 5', seção 4.3.2), então existe uma relação biunívoca entre os elementos dos conjuntos

$$\{\zeta_1(n) \mid \|\bar{\mathbf{b}}_M(n)\| \leq 1 - \alpha, \xi_M^{f/2}(n) \geq \alpha\} = \mathcal{S}_{e1}^\alpha(n)$$

e

$$\{\zeta_2(n) \mid \|\bar{\boldsymbol{\psi}}_M(n)\| \leq \frac{1 - \alpha}{\sqrt{1 - (1 - \alpha)^2}}, \xi_M^{f/2}(n) \geq \alpha\} = \mathcal{S}_{e2}^\alpha(n),$$

onde:  $1 > \alpha > 0$ ;  $\zeta_1(n)$  é definido conforme (4.13);  $\zeta_2(n)$  é definido conforme (4.28);  $\mathcal{S}_{e1}^\alpha(n)$  é um  $\alpha$ -interior associado ao algoritmo QR-LSL *a posteriori*;  $\mathcal{S}_{e2}^\alpha(n)$  é um  $\alpha$ -interior, fechado e limitado no que tange as variáveis  $\bar{\boldsymbol{\psi}}_M(n)$  e  $\xi_M^{f/2}(n)$ , associado ao algoritmo QR-LSL *a priori*.

$\overline{\Delta\Delta}$

Para finalizar, cabe ainda mencionar que conforme a demonstração de 5c um  $\alpha$ -interior associado ao algoritmo QR-LSL *a posteriori* pode ser caracterizado pelas condições

$$|\phi_i(n)| \leq \alpha' < \pi/2, \quad |\theta_i^b(n)| \leq \alpha'' < \pi/2$$

e

$$\xi_M^{f/2}(n) > \alpha,$$

onde  $\alpha'$ ,  $\alpha''$  e  $\alpha$  são constantes positivas. Desta forma, também um  $\alpha$ -interior, limitado e fechado no que tange as variáveis  $\xi_M^{f/2}(n)$  e  $\bar{\boldsymbol{\psi}}_M(n)$ , associado ao algoritmo QR-LSL *a priori* pode ser caracterizado pelas mesmas condições. Consequentemente, no caso de excitação persistente, também no algoritmo QR-LSL *a priori* os ângulos  $\phi_i(n)$  e  $\theta_i^b(n)$ , para  $1 \leq i \leq M$ , devem satisfazer a propriedade 5b para uma constante  $\alpha_1 = \max\{\alpha', \alpha''\}$ .

#### 4.3.4 Consistência retrógrada

Para assegurar que o algoritmo da tabela 4.2 possui a propriedade da consistência retrógrada é necessário garantir que (4.29) e (4.30) sejam satisfeitas mesmo na presença de erros numéricos. Uma condição a ser assegurada é  $\xi_M^{f/2}(n) > 0$ . Como o cálculo é feito da mesma forma que no algoritmos QR-LSL *a posteriori*, as mesmas observações da seção 4.2 se aplicam, sendo possível utilizar qualquer método de cálculo de norma que não forneça resultados inconsistentes.

Como nenhuma restrição é imposta aos vetores  $dq1_M^f(n)$  e  $\bar{\psi}_M(n)$ , não é possível que os erros numéricos ‘puxem’ o vetor  $\bar{\psi}_M(n)$  para fora do conjunto de estabilidade exponencial. Desta forma as rotações  $\theta_i^b(n)$  e  $\phi_i(n)$  que propagam estas variáveis não precisam ser passivas em precisão finita. Sendo assim, é possível garantir a consistência retrógrada e usar arredondamento em todas as operações do algoritmo proposto o que não é o caso do algoritmo QR-LSL *a posteriori*. Este fato simplifica a implementação e é bastante conveniente quando usa-se palavra binária de tamanho reduzido como é mostrado na próxima seção. Existem, entretanto, alguns cuidados que devem ser tomados. Em situações em que  $\|\bar{\psi}_M(n)\|$  e  $\gamma_M^{-1}(n)$  assumem valores muito elevados, podem surgir problemas, caso estes valores não forem limitados no maior valor representável ou em outro limite pré-fixado. Por exemplo, no caso do pacote de ‘software’ MatLab, pode acontecer de variáveis internas assumirem valores elevados, fazendo com que cálculos posteriores forneçam como resultado Inf (‘Infinity’). Com um resultado intermediário deste tipo o algoritmo fornecerá resultados sem significado. Uma solução simples para este problema é a introdução de um limite superior para a variável  $\gamma_M^{-1}(n)$ .

### Seqüências de dados mal condicionadas

A seguir é examinado o comportamento do algoritmo QR-LSL *a priori* quando  $\zeta(n)$  está na borda de  $S_e(n)$ . Quando  $\zeta(n)$  se aproxima da borda o mapeamento de um para um entre os vetores de estados (4.28) e (4.13), usado anteriormente para obter os resultados de estabilidade do algoritmo proposto, se aproxima da singularidade. Conseqüentemente, os resultados de estabilidade de [31] não podem ser transpostos diretamente para garantir a propagação estável do erro.

Supondo que o método de cálculo da norma permita obter um resultado nulo para vetores

$$\left[ \lambda^{1/2} \xi_M^{f/2}(n-1) \quad \epsilon_M^f(n) \right]$$

com valores muito pequenos, então para uma determinado sinal de entrada é possível obter  $\xi_M^{f/2}(n) = 0$ . Nestas condições é necessário utilizar a mesma convenção numérica considerada na seção anterior para este caso. Tal convenção deve ser considerada nos passos 2, 3 e 5 do algoritmo da tabela 4.2, onde existe a possibilidade de que os vetores de dois elementos usados para cálculo do ângulo tenham elementos nulos.

Neste caso escolhe-se o valor zero para o ângulo de rotação. Cabe lembrar que esta convenção assegura que os ângulos de rotação sejam funções contínuas do elemento a ser eliminado e, além disso, é facilmente implementada em processador CORDIC.

A partir da expressão (4.23) conclui-se que se  $\xi_M^{f/2}(n) = 0$  então existe um índice  $i_0 \leq M$  para o qual vale  $\phi_{i_0}(n) = \pi/2$ , o que por sua vez implica em

$$\xi_{i_0}^{f/2}(n) = \xi_{i_0+1}^{f/2}(n) = \dots = \xi_M^{f/2}(n) = 0.$$

Como necessariamente

$$\epsilon_{i_0}^f(n) = \epsilon_{i_0+1}^f(n) = \dots = \epsilon_M^f(n) = 0,$$

então da convenção numérica usada resulta que  $\theta_{M+1}^f(n) = 0$  e  $\phi_i(n) = 0$ , para  $i_0 < i \leq M$ . Como consequência obtém-se para  $i = i_0$

$$\bar{\psi}_{i_0}(n) = \mp \bar{\eta}_{i_0}(n) \text{ e } \bar{\eta}_{i_0-1}(n) = \pm \bar{\psi}_{i_0-1}(n-1)$$

e para  $i_0 < i \leq M$

$$\bar{\eta}_{i-1}(n) = \bar{\eta}_i(n) = 0 \text{ e } \bar{\psi}_i(n) = \bar{\psi}_{i-1}(n-1) = 0.$$

Como  $\gamma_0^{-1/2}(n) = 1$  e  $\gamma_i^{-1/2}(n) \geq \gamma_{i-1}^{-1/2}(n)$ , no passo 5 da tabela 4.2 obtém-se  $\theta_i^b(n) = 0$ , para  $i_0 < i \leq M$ . No caso limite as seções correspondentes a  $\theta_i^b(n)$ ,  $i_0 < i \leq M$ , da estrutura de McWhirter do passo 1 efetivamente desaparecem. Como  $|\phi_i(n)| < \pi/2$ ,  $1 \leq i < i_0$  e  $|\phi_{i_0}(n)| = \pi/2$ , somente um dos ângulos  $\phi_i(n)$  atinge a borda do hiberno  $|\phi_i(n)| < \pi/2$ ,  $1 \leq i \leq M$ . Desta forma, os estágios inferiores do filtro em treliça do passo 4b são conectadas em malha fechada na seção correspondente de  $\phi_{i_0}(n) = \pi/2$  e as seções em cascata com índices superiores se degeneram para uma linha de atrasos. Tal situação é ilustrada na figura 4.3 para uma estrutura em treliça com  $M = 3$  e  $i_0 = 2$ . Para  $i > i_0$  os valores assumidos para as variáveis internas não estão mais representando significativamente o sinal de entrada. A informação significativa é fornecida pelas variáveis com índice inferior ou igual a  $i_0$ . Esta situação é análoga àquela descrita para o algoritmo QR-LSL *a posteriori*.

Para ambos os algoritmos QR-LSL pode ser conveniente usar a seguinte condição

$$\frac{\pi}{2} - \alpha \leq \phi_{i_0}(n) \leq \frac{\pi}{2}, \quad (4.36)$$

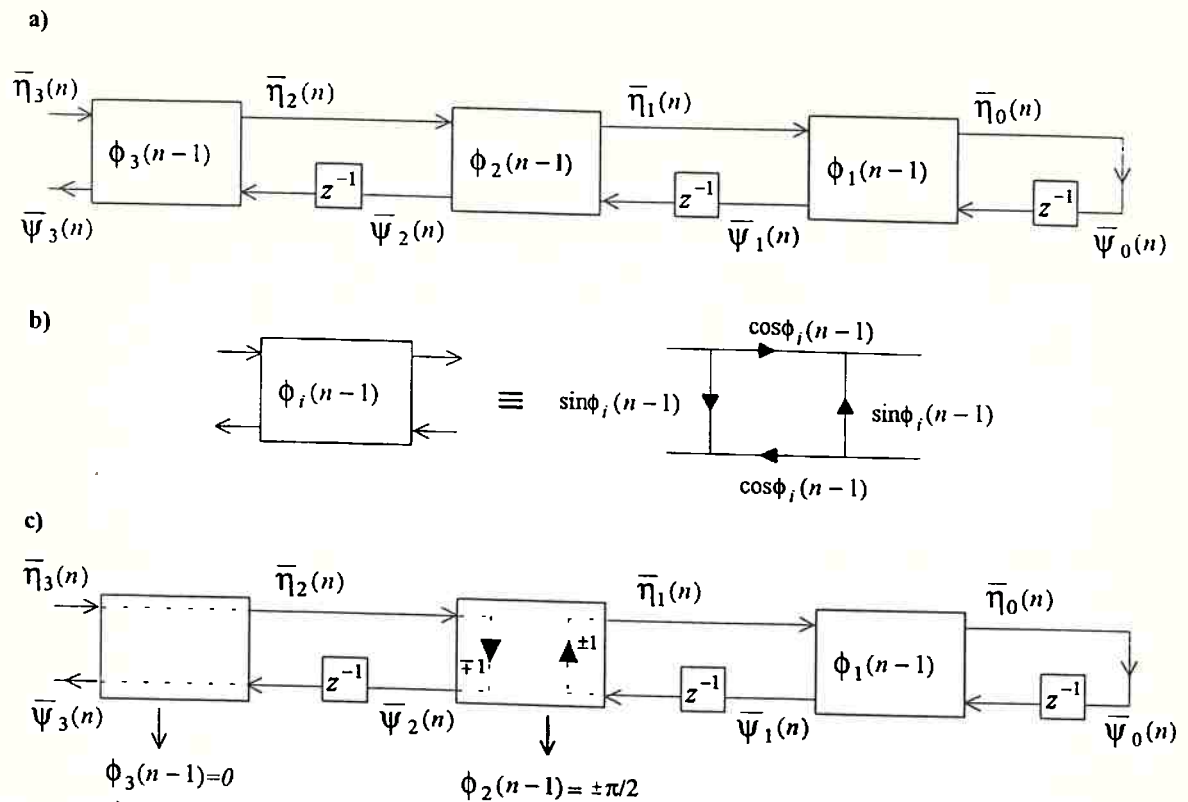


Figura 4.3: a) Filtro em treliça com erro *a priori* normalizado com  $M = 3$ ; b) detalhe de cada bloco; c) situação em que o posto da matriz de dados é deficiente com  $i_0 = 2$ .

com um  $\alpha$  apropriadamente pequeno, para identificar  $i_0$  e interpretar corretamente as variáveis internas. Esta condição pode ser usada para reduzir a ordem do algoritmo fazendo todos os ângulos com índices maiores que  $i_0$  assumirem o valor zero ou então diminuir a ordem de  $M$  para  $i_0$ . Tais resultados mostram que estes algoritmos também revelam que o posto do sinal de entrada é  $i_0$  e que bastaria utilizar  $M = i_0$ .

Na verdade, o comportamento das variáveis do algoritmo QR-LSL *a priori* que acaba de ser descrito, não leva em conta erros de quantização. Na prática é utilizado o arredondamento das variáveis do algoritmo e as rotações não são passivas. Isto faz com que, para  $i > i_0$ , a situação que leva a indeterminação do cálculo dos ângulos  $\phi_i(n)$ ,  $i_0 < i \leq M$ , normalmente não chegue rigorosamente a ocorrer. Isto porque normalmente os valores assumidos por  $\xi_i^{f/2}(n)$  e  $dq1_i^f(n)$ ,  $i_0 < i \leq M$ , são pequenos, não exatamente nulos, e determinados pelos erros de quantização. Nesse caso deve-se utilizar a condição (4.36) com  $\alpha$  suficientemente pequeno, para não interpretar erroneamente os ângulos  $\phi_i(n)$ ,  $i_0 < i \leq M$ , pois estes não representam mais nenhuma característica significativa do sinal de entrada.

## Inicialização

No algoritmo QR-LSL *a priori* não é possível supor uma seqüência de observação nula para  $n < 0$ . Isto é decorrência do fato de que, por exemplo, para o instante  $n = 0$  o cálculo do passo 4a da tabela 4.2 pode ser reescrito como

$$\bar{\eta}_M(1) = \gamma_M^{-1/2}(0) \frac{\sin \theta_{M+1}^f(1)}{\cos \theta_{M+1}^f(1)} = \frac{\gamma_M^{-1/2}(0)}{\lambda^{1/2} \xi_M^{f/2}(0)} \epsilon_M^f(1).$$

Então deve-se inicializar  $\xi_M^{f/2}(0)$  com um valor positivo 'pequeno'. Isto corresponde a uma inicialização não exata, que corresponde a escolher um vetor de estados  $\zeta(0)$  pertencente a uma  $\alpha$ -interior de  $\mathcal{S}_e(n)$ . Uma inicialização conveniente para o algoritmo em questão, com a seqüência de passos conforme tabela 4.2, é a seguinte:

$$\begin{aligned} \bar{\psi}_{i-1}(0) = dq1_i^f(0) = 0 \quad \text{para } 1 \leq i \leq M, \\ \epsilon_0^f(1) = u(0) \neq 0 \quad \text{e } \xi_M^{f/2}(0) \neq 0. \end{aligned}$$

Para uma seqüência de passos conforme tabela 3.4, é conveniente inicializar da seguinte forma:

$$\begin{aligned} \bar{\psi}_{i-1}(-1) = dq1_i^f(-1) = \theta_i^b(-1) = \phi_i(-1) = 0 \quad \text{para } 1 \leq i \leq M, \\ \epsilon_0^f(0) = u(0) \neq 0, \quad \xi_M^{f/2}(-1) \neq 0 \quad \text{e } \gamma_M^{-1/2}(-1) = 1. \end{aligned}$$

No caso de se implementar o algoritmo executando primeiramente os passos 4 e 2 (ver figura 3.2) a variável  $\epsilon_M^f(0)$  deve também ser inicializada. Neste caso é conveniente usar  $\epsilon_M^f(0) = u(0)$ , escolha que é coerente com  $\theta_i^b(n) = 0$ , para  $1 \leq i \leq M$ .

Para finalizar esta seção convém observar que os demais algoritmos que utilizam estruturas em treliça não são mínimos. O vetor de estados pode ser decomposto em uma componente mínima e outra redundante que na situação ideal é sempre nula [32]. No caso destes algoritmos a componente mínima sozinha é a princípio estável não saindo do domínio de estabilidade. Porém, a propagação da componente redundante pode apresentar problemas. Os erros de quantização fazem com que esta componente redundante não seja nula, de modo que o vetor de estados sai do domínio de estabilidade  $\mathcal{S}_e(n)$  associado ao algoritmo. Além disso no caso de seqüências mal condicionadas, mesmo a componente mínima pode sair do domínio de estabilidade [32].



## 4.4 Considerações sobre representação finita

A seguir são determinados valores médios quadráticos de algumas variáveis de interesse, usadas nos algoritmos QR-LSL. Para isso é suposto que a seqüência de observação é estacionária. Os objetivos são: obter uma orientação sobre a precisão com que se deve representar os erros de predição regressiva e outras variáveis relacionadas; e, explicar teoricamente o comportamento, observado por meio de simulações de algumas variáveis dos algoritmos QR-LSL.

Inicialmente serão considerados os erros de predição regressiva *a posteriori*. No caso de seqüências estacionárias bem condicionadas os erros de predição regressiva *a posteriori* normalizados se aproximam estocasticamente da ortogonalidade. De fato, como será mostrado a seguir, a aplicação do operador esperança em  $\bar{\mathbf{b}}_M(n)\bar{\mathbf{b}}_M^T(n)$  resulta em

$$E \left[ \bar{\mathbf{b}}_M(n)\bar{\mathbf{b}}_M^T(n) \right] \approx \text{diag} \left\{ \lambda^{i-1}(1-\lambda) \right\}, \quad (4.37)$$

onde  $1 \leq i \leq M$ . Em [31] é usado  $(1-\lambda)$  em vez de  $\lambda^{i-1}(1-\lambda)$ , contudo resultados de simulações confirmam que (4.37) é uma aproximação melhor. A aproximação para os elementos da diagonal é obtida a partir da equação (A.52) reescrita para ordem  $i-1$  como

$$\left| \bar{b}_{i-1}(n) \right|^2 = \gamma_{i-1}(n) - \gamma_i(n). \quad (4.38)$$

Aplicando o operador esperança em (4.38) e expressando o fator de conversão em função das energias de predição progressiva e regressiva conforme (A.51) resulta:

$$E \left[ \left| \bar{b}_{i-1}(n) \right|^2 \right] = \lambda^{i-1} E \left[ \frac{\xi_{i-1}^b(n)}{\xi_{i-1}^f(n)} \right] - \lambda^i E \left[ \frac{\xi_i^b(n)}{\xi_i^f(n)} \right]. \quad (4.39)$$

Considerando a ordem  $k$  (com  $k = i$  ou  $k = i-1$ ) e  $\xi_k^b(n) = \sum_{\ell=0}^n \lambda^{n-\ell} \left| e_k^b(\ell) \right|^2$ , cada um dos termos do lado direito da equação anterior pode ser expresso como

$$E \left[ \frac{\xi_k^b(n)}{\xi_k^f(n)} \right] = E \left[ \frac{\sum_{\ell=0}^n \lambda^{n-\ell} \left| e_k^b(\ell) \right|^2}{\xi_k^f(n)} \right] = \sum_{\ell=0}^n \lambda^{n-\ell} E \left[ \frac{\left| e_k^b(\ell) \right|^2}{\xi_k^f(n)} \right]. \quad (4.40)$$

Aplicando o princípio das médias [9], que neste caso é válido para  $n \rightarrow \infty$  e  $\lambda$  próximo de 1, tem-se

$$E \left[ \frac{\left| e_k^b(\ell) \right|^2}{\xi_k^f(n)} \right] \approx \frac{E \left[ \left| e_k^b(\ell) \right|^2 \right]}{E \left[ \xi_k^f(n) \right]}.$$

Desta forma obtém-se

$$E \left[ \frac{\xi_k^b(n)}{\xi_k^f(n)} \right] \approx \sum_{\ell=0}^n \lambda^{n-\ell} \frac{E \left[ |e_k^b(\ell)|^2 \right]}{E \left[ \xi_k^f(n) \right]},$$

ou ainda, considerando  $\xi_k^f(n) = \sum_{t=0}^n \lambda^{n-t} |e_k^f(t)|^2$ ,

$$E \left[ \frac{\xi_k^b(n)}{\xi_k^f(n)} \right] \approx \sum_{\ell=0}^n \lambda^{n-\ell} \frac{E \left[ |e_k^b(\ell)|^2 \right]}{\sum_{t=0}^n \lambda^{n-t} E \left[ |e_k^f(t)|^2 \right]}.$$

No caso de seqüências estacionárias bem condicionadas pode se fazer a seguinte aproximação

$$E \left[ |e_k^b(\ell)|^2 \right] \approx E \left[ |e_k^f(t)|^2 \right] = P_k,$$

obtendo-se assim

$$E \left[ \frac{\xi_k^b(n)}{\xi_k^f(n)} \right] \approx \sum_{\ell=0}^n \lambda^{n-\ell} \frac{P_k}{\sum_{t=0}^n \lambda^{n-t} P_k} = 1. \quad (4.41)$$

Usando (4.41) a equação (4.39), que representa os elementos da diagonal principal da matriz  $E \left[ \bar{\mathbf{b}}_M(n) \bar{\mathbf{b}}_M^T(n) \right]$ , pode ser expressa como

$$E \left[ |\bar{b}_{i-1}(n)|^2 \right] \approx \lambda^{i-1} (1 - \lambda), \quad \text{para } 1 \leq i \leq M. \quad (4.42)$$

Cabe observar ainda que para  $\lambda$  próximo de 1 existe um risco de 'underflow' a medida que a norma de  $\bar{\mathbf{b}}_M(n)$  contrai. Neste caso o uso de rotações passivas torna esta arriscada ocorrência ainda mais provável. Entretanto, caso não se use rotações passivas a propriedade de consistência retrógrada não fica assegurada podendo levar a instabilidade.

De forma análoga ao caso onde se usa os erros da predição regressiva *a posteriori* normalizados, no caso onde são usados os erros de predição regressiva *a priori* normalizados a matriz  $\bar{\boldsymbol{\psi}}_M(n) \bar{\boldsymbol{\psi}}_M^T(n)$  também se aproxima estocasticamente da ortogonalidade. De fato, a aplicação do operador esperança em  $\bar{\boldsymbol{\psi}}_M(n) \bar{\boldsymbol{\psi}}_M^T(n)$  resulta em

$$E \left[ \bar{\boldsymbol{\psi}}_M(n) \bar{\boldsymbol{\psi}}_M^T(n) \right] \approx \text{diag} \left\{ \lambda^{-i} (1 - \lambda) \right\}.$$

A aproximação para os elementos da diagonal principal é obtida a partir de (3.112)

$$|\bar{\psi}_{i-1}(n)|^2 = \gamma_i^{-1}(n) - \gamma_{i-1}^{-1}(n). \quad (4.43)$$

Aplicando o operador esperança em (4.43) e expressando o fator de conversão em função das energias de predição progressiva e regressiva conforme (A.51) resulta

$$E \left[ \left| \bar{\psi}_{i-1}(n) \right|^2 \right] = \lambda^{-i} E \left[ \frac{\xi_i^f(n)}{\xi_i^b(n)} \right] - \lambda^{-i+1} E \left[ \frac{\xi_{i-1}^f(n)}{\xi_{i-1}^b(n)} \right].$$

De forma completamente análoga ao cálculo de (4.41) pode-se mostrar que

$$E \left[ \frac{\xi_k^f(n)}{\xi_k^b(n)} \right] \approx 1$$

de modo que

$$E \left[ \left| \bar{\psi}_{i-1}(n) \right|^2 \right] \approx \lambda^{-i}(1 - \lambda), \quad \text{para } 1 \leq i \leq M. \quad (4.44)$$

Neste caso também existe um risco de 'underflow' a medida que a norma de  $\bar{\psi}_M(n)$  se contrai e  $\lambda$  se aproxima de 1. Entretanto a possibilidade de usar arredondamento para representar  $\bar{\psi}_{i-1}(n)$  torna menos provável a ocorrência de 'underflow'.

Uma expressão aproximada para a variância assintótica dos ângulos  $\{\theta_i^b(n)\}$  pode ser obtida a partir de (3.103)

$$E \left[ \cos^2 \theta_i^b(n) \right] = \lambda E \left[ \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} \right], \quad (4.45)$$

sendo que

$$E \left[ \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} \right] = E \left[ \frac{\sum_{\ell=0}^{n-1} \lambda^{n-1-\ell} |e_k^b(\ell)|^2}{\xi_{i-1}^b(n)} \right] = \sum_{\ell=0}^{n-1} \lambda^{n-1-\ell} E \left[ \frac{|e_k^b(\ell)|^2}{\xi_{i-1}^b(n)} \right].$$

Aplicando novamente o princípio das médias [9], que é válido para  $n \rightarrow \infty$  e  $\lambda$  próximo de 1, e considerando o caso de seqüências estacionárias bem condicionadas onde  $E \left[ |e_k^b(\cdot)|^2 \right] \approx P_k$ , tem-se

$$E \left[ \frac{|e_k^b(\ell)|^2}{\xi_{i-1}^b(n)} \right] \approx \frac{E \left[ |e_k^b(\ell)|^2 \right]}{E \left[ \sum_{t=0}^n \lambda^{n-t} |e_k^b(t)|^2 \right]} = \frac{1}{\sum_{t=0}^n \lambda^{n-t}}.$$

Desta última expressão decorre que

$$E \left[ \frac{\xi_{i-1}^b(n-1)}{\xi_{i-1}^b(n)} \right] \approx \sum_{\ell=0}^{n-1} \lambda^{n-1-\ell} \frac{1}{\sum_{t=0}^n \lambda^{n-t}} = \frac{1 - \lambda^n}{1 - \lambda^{n+1}} \approx 1 \quad (4.46)$$

e assim

$$E[\cos^2 \theta_i^b(n)] \approx \lambda \quad \text{e} \quad E[\sin^2 \theta_i^b(n)] \approx 1 - \lambda. \quad (4.47)$$

Estas expressões confirmam resultados apresentados em [31, 36]. Quando se usa representação em ponto fixo a equação (4.47) é útil para estimar a precisão dos registros que armazenam  $\sin \theta_i^b(n)$ . Como  $\lambda \approx 1$  a primeira equação de (4.47) serve também para estimar um limite inferior para o comprimento da mantissa em uma representação em ponto flutuante. Considerando  $\sin \theta_i^b(n) \approx \theta_i^b(n)$  para  $\theta_i^b(n)$  pequeno tem-se

$$E[(\theta_i^b(n))^2] \approx 1 - \lambda. \quad (4.48)$$

Esta expressão é útil quando os algoritmos são implementados em CORDIC onde os ângulos  $\theta_i^b(n)$  são armazenados diretamente.

Cabe observar que o fator de conversão  $\gamma_i(n)$  e o seu inverso  $\gamma_i^{-1}(n)$  assumem valores em torno de  $\lambda^i$  e  $\lambda^{-i}$ . Para  $0 \ll \lambda < 1$  estes valores são próximos de 1. Então, considerando uma representação em ponto flutuante a precisão com que  $\gamma_i(n)$  e  $\gamma_i^{-1}(n)$  devem ser armazenados dá uma indicação sobre o número necessário de bits da mantissa. Como as equações (4.38) e (4.43) representam a diferença entre dois números próximos de 1, elas podem fornecer uma orientação de como ocorre o compromisso entre o número de bits usado para a representação conveniente das variáveis e o fator de esquecimento nos algoritmos QR-LSL. Para obter esta orientação no algoritmo QR-LSL com *a posteriori*, que usa os erros  $\bar{b}_{i-1}(n)$ , considera-se que o menor valor de (4.42) obedece a relação

$$\frac{\alpha}{2^b} = \lambda^M (\lambda^{-1} - 1),$$

onde  $\alpha$  é uma constante positiva proporcional a precisão que se deseja obter no algoritmo e  $b$  é o número de bits em que se representa a mantissa. Aplicando  $\log_2$  resulta

$$\Delta \text{bits } \bar{b} := b - \log_2(\alpha) = -M \log_2(\lambda) - \log_2(\lambda^{-1} - 1). \quad (4.49)$$

Para obter a orientação de como ocorre o compromisso entre o número de bits usado para a representação conveniente das variáveis e o fator de esquecimento no algoritmo QR-LSL *a priori*, que usa os erros  $\bar{\psi}_{i-1}(n)$ , considera-se que o menor valor de (4.44) obedece a relação

$$\frac{\alpha}{2^b} = (\lambda^{-1} - 1).$$

Aplicando  $\log_2$  resulta

$$\Delta bits_{\bar{\psi}} := b - \log_2(\alpha) = -\log_2(\lambda^{-1} - 1). \quad (4.50)$$

A diferença entre as relações (4.49) e (4.50) é do fator  $-M \log_2(\lambda)$  e que significa um incremento do número de bits para representar as variáveis com a mesma precisão no algoritmo que usa os erros de predição *a posteriori* normalizados. A figura 4.4a mostra o efeito do fator  $-M \log_2(\lambda)$  em termos de número de bits quando o fator de esquecimento está na faixa de  $0.9 \leq \lambda \leq 0.999$  e para as ordens  $M$  de 2, 11, 22, 35 e 48. Observa-se nitidamente a importância deste fator em termos de número de bits quando a ordem  $M$  é elevada e o fator de esquecimento pequeno. Por outro lado, para valores de  $\lambda$  próximos de um este termo é insignificante. Na figura 4.4b

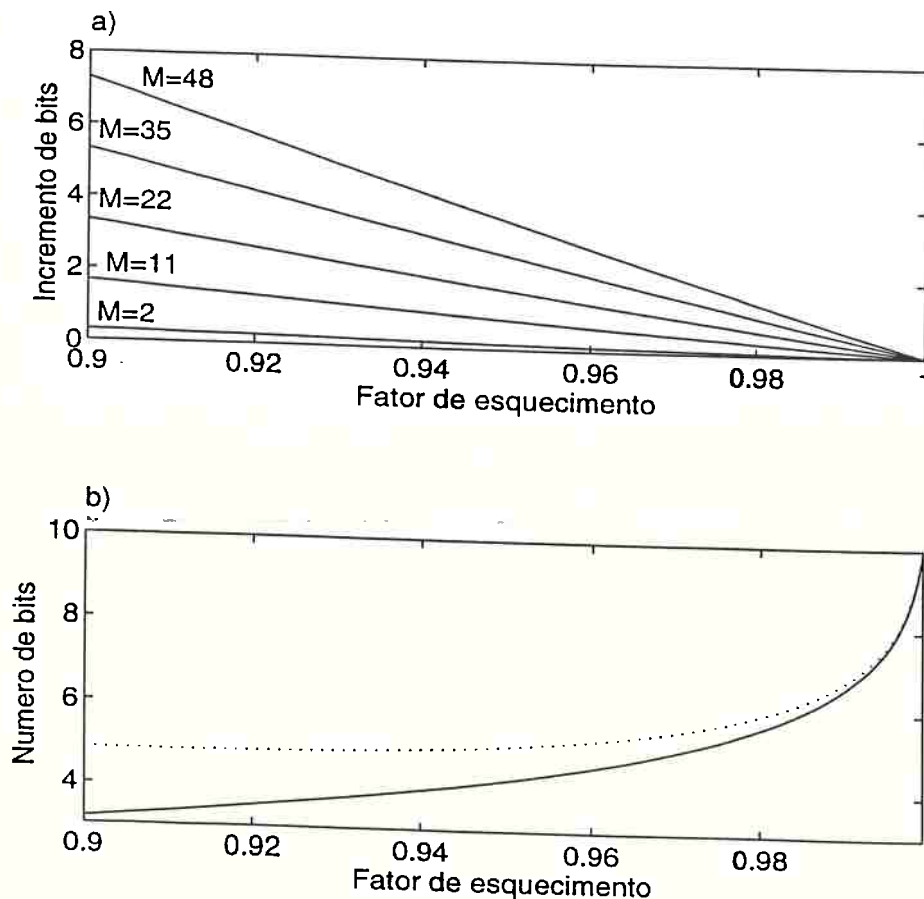


Figura 4.4: a) Incremento do número de bits devido ao fator  $-M \log_2(\lambda)$ ; b)  $\Delta bits \times$  fator de esquecimento para ordem  $M = 11$ , linha cheia refere-se ao QR-LSL *a priori* e linha pontilhada refere-se ao QR-LSL *a posteriori*.

estão representadas as curvas de  $\Delta bits_{\bar{b}}$  em linha pontilhada e  $\Delta bits_{\bar{\psi}}$  em linha cheia quando o fator de esquecimento esta na faixa de  $0.9 \leq \lambda \leq 0.999$  e a ordem  $M = 11$ . Verifica-se que para  $0.9 \leq \lambda \leq 0.98$  existe diferença em torno de um bit entre as curvas  $\Delta bits_{\bar{b}}$  e  $\Delta bits_{\bar{\psi}}$ .

Na próxima seção apresenta-se um exemplo de simulação para verificar a utilidade das curvas da figura 4.4b. Cabe observar ainda que para seqüências não estacionárias as aproximações efetuadas nesta seção (que resultaram nas expressões (4.42), (4.44) e (4.47)) são bastante grosseiras.

## 4.5 Resultados de simulação

Para testar o algoritmo QR-LSL *a priori* foram efetuados vários experimentos. Por exemplo, as simulações apresentadas em [7, 12, 19, 33, 38] foram realizadas tendo sido obtido os mesmos resultados. Apresenta-se a seguir resultados obtidos usando o algoritmo em uma configuração de equalizador adaptativo e em uma configuração de identificação de parâmetros. É também apresentado o desempenho na presença de seqüências mal condicionadas. As simulações foram feitas supondo um processador dotado de um multiplicador-acumulador, como os processadores digitais de sinais comerciais. Foi usada aritmética de ponto flutuante com 8 bits no expoente (incluindo o bit de sinal) e variando o número de bits da mantissa.

### Configuração equalizador adaptativo

O diagrama de blocos da figura 4.5 representa o caso usado nas referências [12, 19, 38]. O bloco canal é usado para modelar a interferência intersímbolos no canal de comunicação digital. Adotou-se a resposta impulsiva do canal como

$$h(n) = \begin{cases} \frac{1}{2}[1 + \cos(\frac{2\pi}{W}(n-2))], & n = 1, 2, 3 \\ 0, & n > 3 \end{cases}$$

O parâmetro  $W$  controla a dispersão da amplitude do canal, isto é o espalhamento dos autovalores do sinal de entrada do filtro adaptativo. O sinal  $\{v(n)\}$  representa uma seqüência de ruído branco gaussiano de média nula. A mensagem digital, aqui representada como  $\{a(n)\}$ , é uma seqüência randômica binária,  $a(n) = \pm 1$ . Os sinais  $a(n)$  e  $v(n)$  são não correlacionados. O filtro adaptativo faz uma estimação linear

de cada símbolo  $d(n)$ . Usou-se como sinal de referência a seqüência binária  $\{a(n)\}$  atrasada de 7 amostras e um filtro adaptativo com 11 coeficientes. Considerou-se  $\{v(n)\}$  de forma a ter-se na entrada do filtro adaptativo uma relação sinal ruído de  $30dB$ . O desempenho do algoritmo QR-LSL *a priori* foi avaliado através de médias de conjunto do erro quadrático de estimação *a priori* (média em 30 realizações).

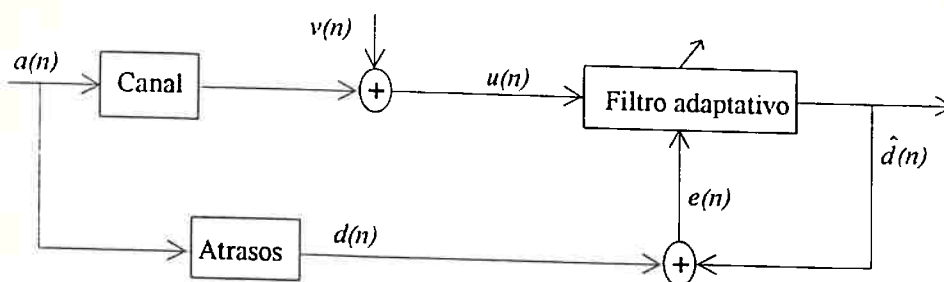


Figura 4.5: Equalizador adaptativo

Para o número de bits da mantissa entre 16 e 24 com um fator de esquecimento igual a 0.99, as médias de conjunto do erro quadrático de estimação *a priori* (EMQ) são praticamente as mesmas. A figura 4.6 mostra o EMQ em função do

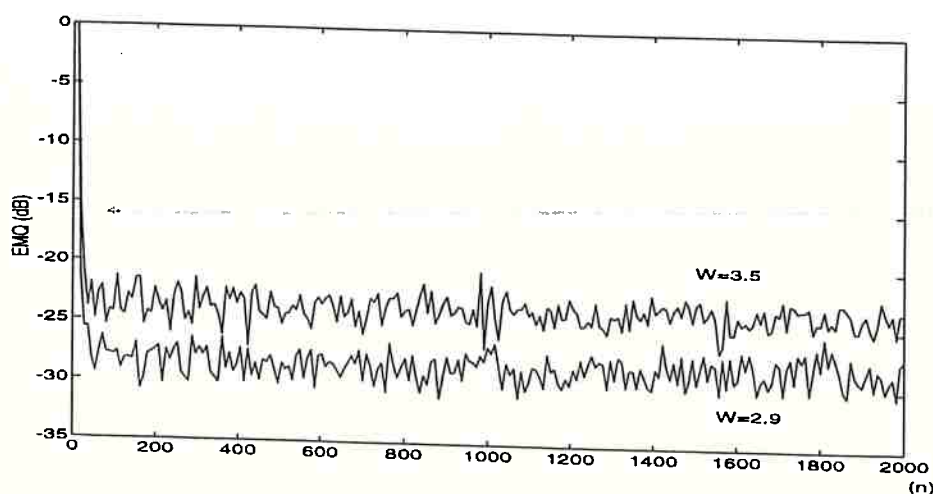


Figura 4.6: Erro médio quadrático de estimação *a priori* do algoritmo QR-LSL com erro *a priori* mudando o espalhamento dos autovalores com  $W = 2.9$  e  $W = 3.5$ . Fator de esquecimento  $\lambda = 0.99$  e 16 bits de mantissa.

número de iterações ( $n$ ) para  $W = 2.9$  e  $W = 3.5$ . Os valores de  $W$  usados representam respectivamente um espalhamento dos autovalores de 4 e 49. Depois de atingir a

convergência a média de conjunto do erro quadrático de estimação *a priori* resultou aproximadamente em  $-29dB$  para  $W = 2.9$  e  $-24dB$  para  $W = 3.5$ . Cabe notar que o aumento do erro quadrático com o aumento do espalhamento dos autovalores é uma característica dos filtros de Wiener [38]. Como esperado a taxa de convergência independe do espalhamento dos autovalores da matriz de autocorrelação, sendo assim a convergência acontece em torno de  $2M$  iterações. Realizando este mesmo experimento com os demais algoritmos considerados no capítulo 3, seção 3.7, observou-se que para um número de bits de mantissa entre 16 e 24 os resultados obtidos com o quatro algoritmos são praticamente idênticos.

O algoritmo LSL *a priori* (tabela 3.1) é conhecido como sendo numericamente robusto [21, 38], apresentando um bom comportamento com um número reduzido de bits. Entretanto, este algoritmo não satisfaz a propriedade da estabilidade retrógrada, podendo divergir na presença de sinais mal condicionados. Por exemplo, na presença de perturbações abruptas no sinal de entrada ele pode ter dificuldades para voltar a funcionar corretamente. Para ilustrar este comportamento mudou-se bruscamente no instante  $n = 500$  a relação sinal ruído (SNR) na entrada do filtro adaptativo durante um intervalo de 25 amostras. A figura 4.7 mostra o EMQ em função do número de iterações para o algoritmo QR-LSL *a priori* (linha cheia) e o algoritmo LSL *a priori* (linha pontilhada) para  $\lambda = 0.90$  e para mantissa representada com 9 bits. Verifica-se um comportamento robusto do algoritmo QR-LSL *a priori* em relação as perturbações. Quando a mantissa é representada com 23 bits os dois algoritmos apresentam o mesmo comportamento, isto é, o efeito da perturbação desaparece em torno de 400 iterações.

### Comparação entre os algoritmos QR-LSL

Para um número de bits da mantissa inferior a 16, com  $\lambda = 0.99$  o EMQ não assume mais os valores esperados. Porém, diminuindo o valor de  $\lambda$  é possível obter ainda um EMQ esperado. A figura 4.8 mostra as médias temporais do erro quadrático de estimação *a priori* (EMQ) em função do número de bits da mantissa para  $\lambda = 0.92$  e  $W = 2.9$ . As médias temporais foram efetuadas em  $10^3$  pontos após passadas  $10^4$  iterações. A curva cheia é do EMQ do algoritmo QR-LSL *a priori* e a curva pontilhada é do algoritmo QR-LSL *a posteriori*. Verifica-se que para garantir



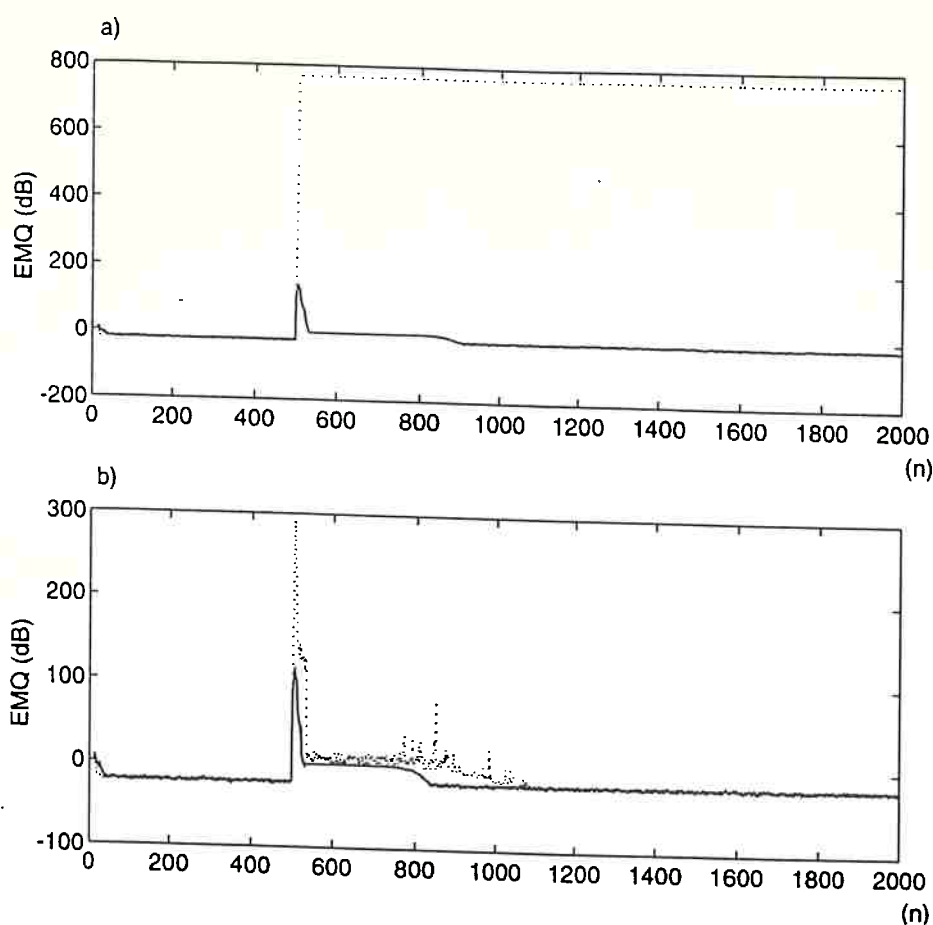


Figura 4.7: Erro médio quadrático dos algoritmos LSL *a priori* (linha pontilhada) e QR-LSL *a priori* (linha cheia). a)  $SNR = -154\text{dB}$ ; b)  $SNR = -143\text{dB}$ .

uma diferença de até  $0.5\text{dB}$  entre o valor do EMQ representado com precisão finita e o valor do EMQ esperado, deve-se usar pelo menos 10 bits para o algoritmo QR-LSL *a posteriori* e pelo menos 9 bits para o algoritmo QR-LSL *a priori*.

Foram calculadas também as médias temporais do erro quadrático de estimação *a priori* em função do número de bits da mantissa para valores de  $\lambda$  entre 0.9 e 0.999. Para cada valor de  $\lambda$  considerado seleccionou-se o número de bits em que o EMQ é representado com uma diferença de até  $0.5\text{dB}$  entre o valor obtido com precisão finita e o valor esperado. A figura 4.9 mostra a curva do número de bits da mantissa em função do  $\lambda$ . Verifica-se que para valores de  $\lambda$  menores do que 0.97 existe uma vantagem em torno de 1 bit para o algoritmo QR-LSL *a priori*. Esta vantagem desaparece para  $\lambda$  maior que 0.97. Quanto menor o valor de  $\lambda$ , menor é o número de

bits que é possível utilizar. A figura 4.9 fornece uma indicação aproximada do incremento do número de bits com o aumento do fator de esquecimento. Esta curva segue de forma aproximada a mesma 'lei' que determinou a curva da figura 4.4b, mantendo inclusive a vantagem em torno de 1 bit para o algoritmo QR-LSL *a priori*.

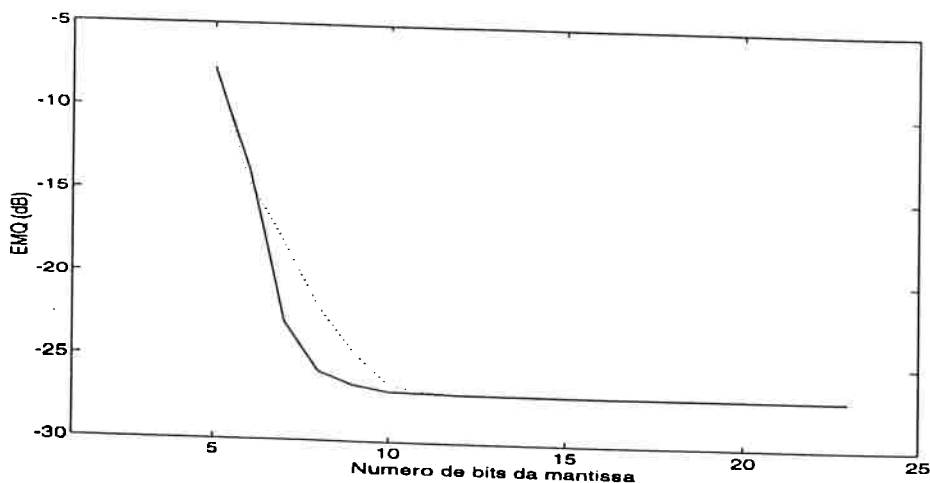


Figura 4.8: Erro quadrático médio em função do número de bits da mantissa com  $\lambda = 0.92$ . Algoritmos QR-LSL *a priori* (linha cheia) e *a posteriori* (linha pontilha).

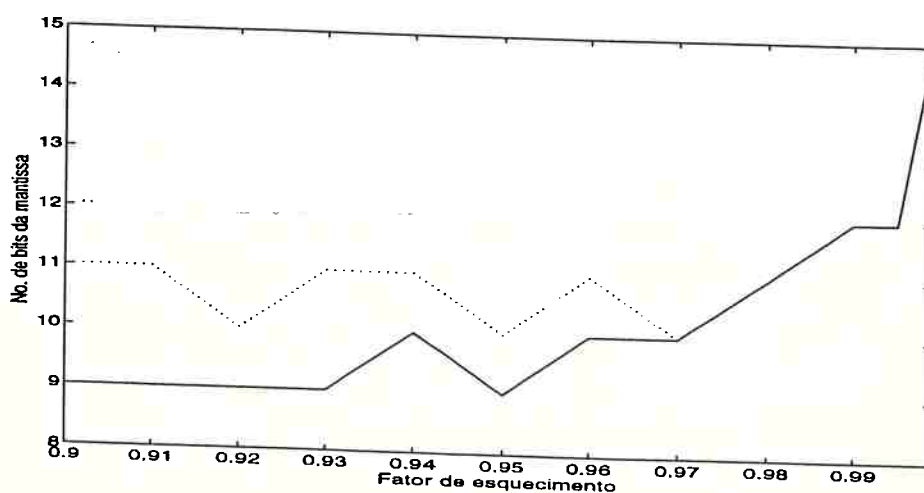


Figura 4.9: Número de bits da mantissa em função do fator de esquecimento. Algoritmos QR-LSL *a priori* (linha cheia) e *a posteriori* (linha pontilha).

### Configuração para identificação de parâmetros

No experimento de identificação de parâmetros considerou-se o diagrama de blocos da figura 4.10. O sinal de entrada do filtro adaptativo  $u(n)$  caracteriza um processo estacionário do tipo autoregressivo de ordem 5, isto é

$$u(n) = \sum_{i=1}^5 h1(i)u(n-i) + v_1(n).$$

O sinal de referência foi gerado supondo

$$d(n) = \sum_{i=1}^5 h2(i)u(n-i) + v_2(n),$$

sendo  $v_2(n)$  um ruído branco aditivo. Os ruídos  $v_2(n)$  e  $v_1(n)$  são não correlacionados. O sistema a ser identificado  $H2$  é um filtro FIR com 5 coeficientes e para isso considerou-se um filtro adaptativo com  $M = 6$ . As variâncias de  $u(n)$  e  $v_2(n)$  são respectivamente  $-4dB$  e  $-80dB$ . A figura 4.11a mostra a trajetória dos coeficientes de reflexão  $-\sin \phi_i(n)$  e a figura 4.11b mostra os coeficientes escalonados  $dq1_i(n)/\xi_0^{b/2}(n)$ . As linhas pontilhadas indicam os valores teóricos (coeficientes ótimos). Fez-se média de conjunto em 30 pontos e o fator de esquecimento usado foi  $\lambda = 0.995$ . Os valores simulados convergem em média para os valores teóricos.

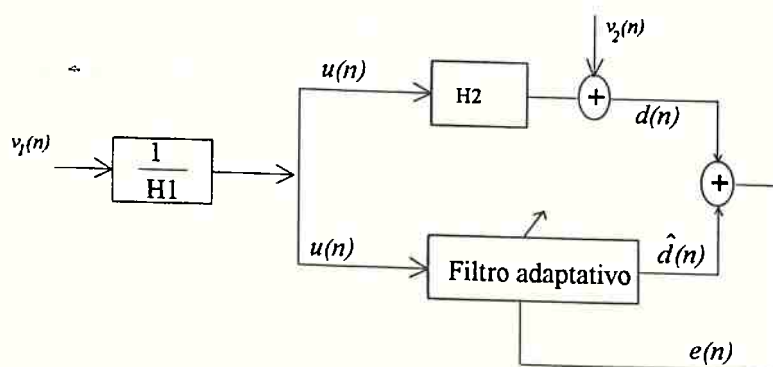


Figura 4.10: Diagrama de blocos para identificação de sistemas.

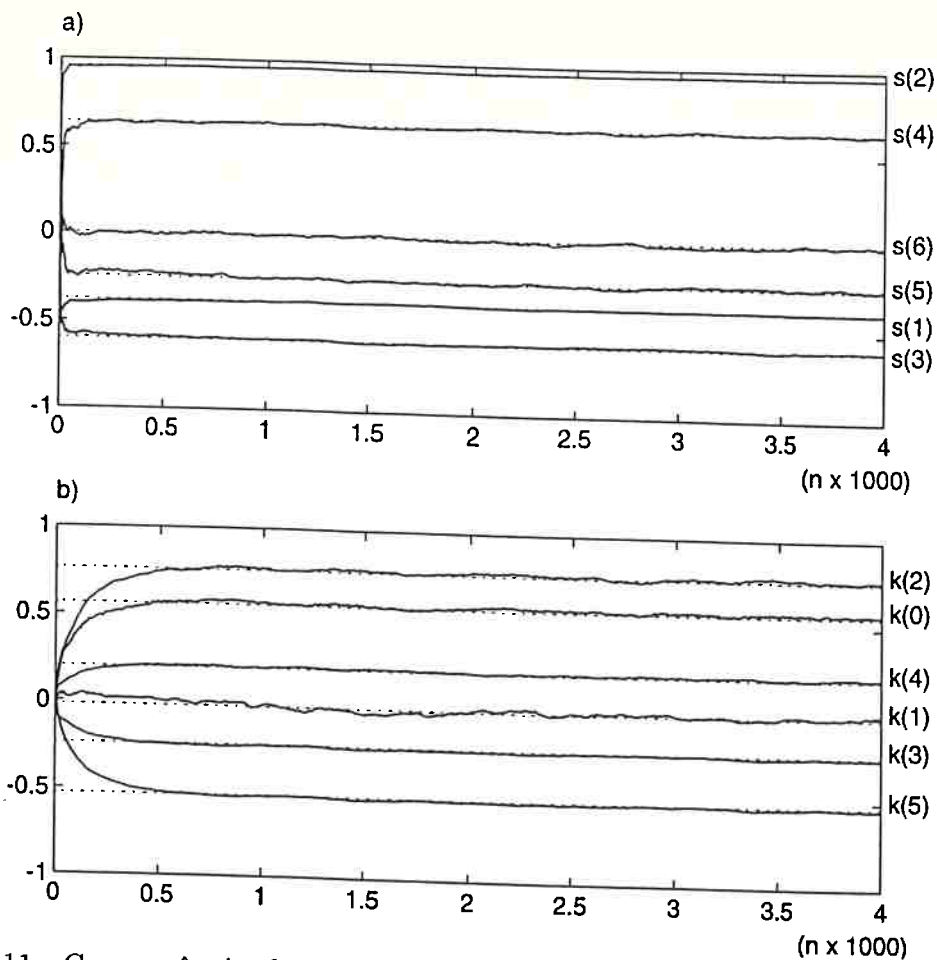


Figura 4.11: Convergência dos coeficientes  $s(i) = -\sin \phi_i(n)$  (a) e dos coeficientes escalonados  $k(i) = dq1_i(n)/\xi_0^{b/2}(n)$  (b). Fator de esquecimento 0.995, 8 bits de expoente e 16 de mantissa. Algoritmo QR-LSL *a priori*, valores teóricos estão indicados pelas linhas pontilhadas.

### Seqüências mal condicionadas

Para observar o comportamento do algoritmo próximo da borda do domínio de estabilidade  $\mathcal{S}_e(n)$  utilizou-se como entrada seqüências periódicas simples. Seqüências deste tipo chegam arbitrariamente próximo (dentro da precisão utilizada) de uma situação de sinal perfeitamente preditível e podem ser consideradas como casos limites de seqüências mal condicionadas. Usou-se uma ordem  $M$  maior que a ordem  $i_0$  necessária (a partir da qual o filtro em treliça degenera para linhas de atrasos, ver seção 4.3.4). A seguir são feitas algumas observações sobre os resultados de inúmeros experimentos em que, além de se considerar tais situações, foram variados o número

de bits da mantissa e o fator de esquecimento.

- ◊ Em todos os experimentos com sinais periódicos o algoritmo QR-LSL *a priori* se mostrou estável, mesmo com um número de bits de mantissa muito pequeno. por exemplo 4 bits de mantissa. Esse compromisso é também apresentado pelo algoritmo QR-LSL *a posteriori*, desde que programado utilizando-se rotações passivas. Tipicamente foram realizadas experimentos até 150000 iterações.
- ◊ Para  $i < i_0$  as variáveis do algoritmo QR-LSL *a priori* convergem em todos os exemplos simulados para os valores esperados após  $2M$  iterações com uma precisão que depende do tamanho da palavra binária e do fator de esquecimento. Em particular a condição  $|\phi_{i_0}(n)| \approx \pi/2$  é satisfeita para o índice  $i_0$  esperado com boa precisão.
- ◊ Depois de ocorrer o desacoplamento aproximado, para  $i > i_0$ , passadas algumas iterações  $n \gg 2M$ , as variáveis do algoritmo se acomodam em valores que são função de  $M$ , do  $\lambda$  e do número de bits usado na representação destas variáveis. Sendo que o número de iterações necessárias para as variáveis se acomodarem depende essencialmente do valor de  $\lambda$ .

Como exemplo considera-se a seguinte seqüência de observação

$$\{u(\cdot)\} = \{0 \ -1 \ 0 \ 1 \ 0 \ -1 \ 0 \ 1 \ 0 \ -1 \ 0 \ 1 \ \dots\}.$$

A figura 4.12 mostra o comportamento das variáveis  $\xi_i^{f/2}(n)$ ,  $-\sin \phi_i(n)$ ,  $\bar{\eta}_{M+1}(n)$  e  $\gamma_i^{1/2}(n)$  para uma ordem  $M = 7$ ,  $\lambda = 0.92$ , 16 bits de mantissa e 8 bits de expoente.

Esse comportamento pode ser resumidamente descrito da seguinte forma:

- ◊ A medida que a ordem aumenta os valores de  $20 * \log_{10}(\xi_i^{f/2}(n))$  (figura 4.12a) se acomodam em valores menores, sendo limitado pelo menor valor representável com 8 bits de expoente ( $-854dB$ );
- ◊ A figura 4.12b mostra a envoltória da curva do erro da predição progressiva *a priori* normalizado  $\bar{\eta}_7(n)$ . Quanto mais próximo de 1 for o valor de  $\lambda$ , menor é o valor em que  $|\bar{\eta}_7(n)|$  se acomoda. No caso  $|\bar{\eta}_7(n)|$  se acomodou em 0.5470. O número de 'bumps' depende da ordem do filtro;

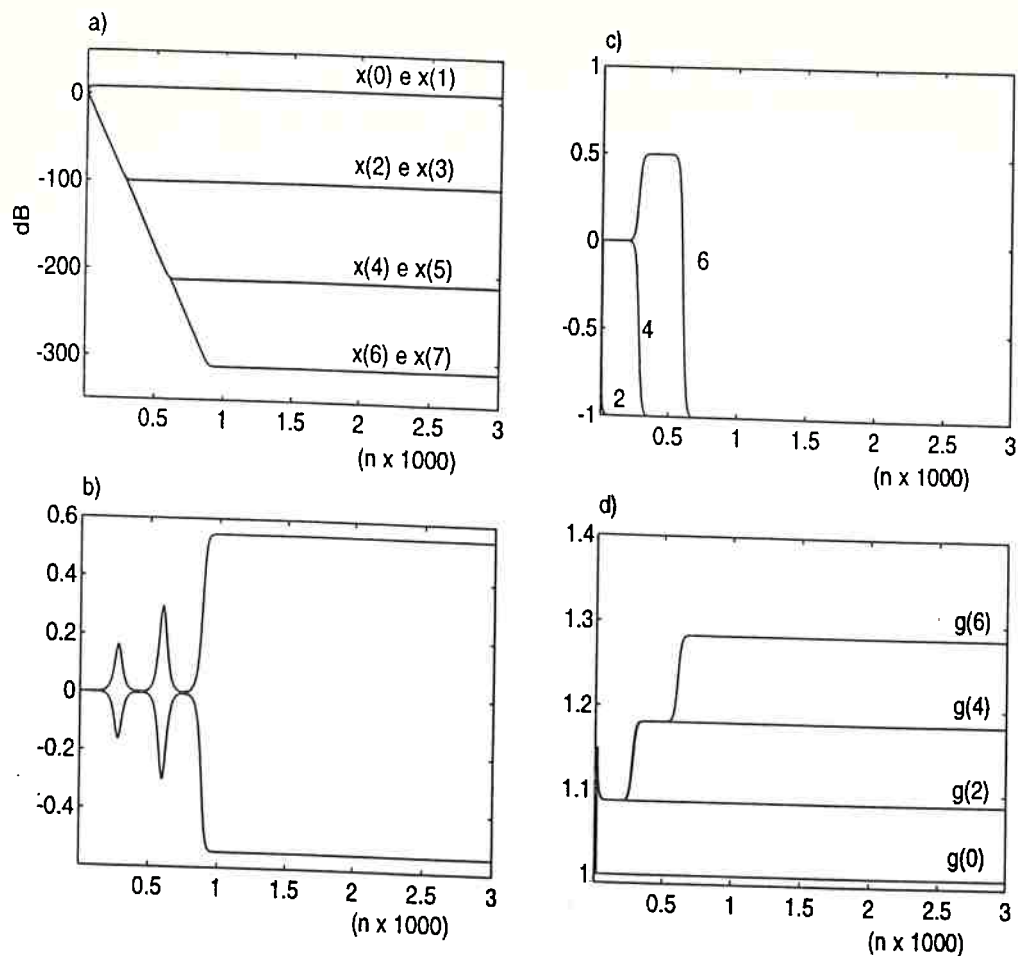


Figura 4.12: a) Energia do erro da predição progressiva  $x(i) = 20 \log_{10} \xi_i^{f/2}(n)$ ; b) envoltória do erro da predição regressiva *a priori* normalizado  $\bar{\eta}_7(n)$ ; c) coeficiente de reflexão  $-\sin \phi_i(n)$ , a correspondente curva está indicada pelo índice  $i$ ; d) fator de conversão  $g(i) = \gamma_i^{-1/2}(n)$ .

- ◊ A variável  $-\sin \phi_i(n)$  converge para 0 ou  $-1$  alternando entre estes valores com o incremento de  $i$ . A figura 4.12c mostra  $-\sin \phi_i(n)$  para  $i = 2, 4$  e  $6$ . Para  $i = 1, 3, 5$  e  $7$  os coeficientes  $-\sin \phi_i(n)$  assumem o valor zero;
- ◊ A variável  $\gamma_i^{-1/2}(n)$  converge para  $\lambda^{-i/2}$  ou para uma situação em que varia entre  $\lambda^{-(i-1)/2}$  e  $\lambda^{-(i+1)/2}$ . Com o incremento de  $i$  os valores se alternam entre estas duas situações. A figura 4.12d mostra o comportamento de  $\gamma_i^{-1/2}(n)$  para  $i = 0, 2, 4$  e  $6$ , sendo que estes valores convergem para 1.0000, 1.0870, 1.1815 e 1.2843 o que, como esperado são aproximadamente iguais a  $\lambda^{-0/2}$ ,  $\lambda^{-2/2}$ ,  $\lambda^{-4/2}$  e  $\lambda^{-6/2}$ . As variáveis  $\gamma_i^{-1/2}(n)$ ,  $i = 1, 3, 5$  e  $7$ , se alteram entre os valores  $\gamma_{i-1}^{-1/2}(n)$

e  $\gamma_{i+1}^{-1/2}(n)$  assumindo em um instante de tempo um valor e no seguinte o outro.

Para um sinal DC o comportamento das variáveis é semelhante sendo que  $\gamma_i^{-1/2}(n)$  não oscila assumindo os valores  $\lambda^{-i/2}$  para toda ordem  $i$ .

É importante notar que quando se reduz o número de bits da mantissa deve-se tomar o cuidado de utilizar valores de  $\lambda$  que ainda possam ser representados corretamente. Por exemplo, caso não se tome este cuidado o comportamento das variáveis não mais assume a forma descrita na figura 4.12.

Para finalizar, cabe observar que para  $i > i_0$  o comportamento descrito se deve aparentemente aos erros numéricos, que no caso de um sinal periódico simples como o utilizado, podem ter forte correlação com o sinal de entrada.

## 4.6 Comentários

Neste capítulo enfatizou-se que é possível obter algoritmos RLS rápidos com a propriedade da estabilidade retrógrada. Obter algoritmos RLS rápidos com tal propriedade, é de fato bastante importante, pois mostra que estes algoritmos não necessariamente são instáveis quando implementados em precisão finita, como era considerado até recentemente na literatura.

A maior parte das definições e propriedades utilizadas neste capítulo foram introduzidas em [31] para mostrar que o algoritmo QR-LSL *a posteriori* tem estabilidade retrógrada. Utilizando as mesmas definições e propriedades mostrou-se que o algoritmo QR-LSL *a priori*, introduzido no capítulo 3, apresenta as seguintes características: é mínimo, isto é, o vetor de estados é formado por apenas  $2M + 1$  variáveis, e além disso, satisfaz as propriedades de consistência e estabilidade retrógradas. Em comparação com o algoritmo QR-LSL *a posteriori*, o algoritmo proposto não necessita de rotações passivas para garantir a estabilidade retrógrada. Tal fato decorre de que o domínio de estabilidade do algoritmo QR-LSL *a priori* é aberto. Isto é, não existe um limite superior para variáveis que definem o vetor de estados. No caso do algoritmo QR-LSL *a posteriori* o domínio de estabilidade é fechado no que concerne aos erros de predição. Por este motivo há necessidade de rotações passivas para garantir a atuação do algoritmo dentro do domínio de estabilidade.

Da mesma forma que o algoritmo QR-LSL *a posteriori*, o algoritmo QR-LSL

*a priori* é estável mesmo para seqüências mal condicionadas, como sinais preditíveis. Nestas circunstâncias outros algoritmos RLS rápidos, que não possuem a propriedade da estabilidade retrógrada, apresentam problemas de instabilidade.

Considerando sinais estacionários, foi feita uma análise dos valores médios quadráticos que algumas variáveis dos algoritmos QR-LSL podem assumir. Tal análise permitiu obter informações sobre a faixa dinâmica, e conseqüentemente sobre a precisão necessária para armazenamento das variáveis consideradas. No caso do valor médio quadrático dos erros normalizados da predição regressiva, relacionou-se precisão necessária para armazenamento da variável, com o fator de conversão  $\lambda$  e a ordem  $M$  do problema de predição. Observou-se que existe um incremento do número de bits, dado por  $-M \log_2 \lambda$ , para representar o erro normalizado de predição regressiva *a posteriori*, com a mesma predição do erro normalizado da predição regressiva *a priori*. Entretanto, este incremento tem importância significativa apenas para ordens elevadas e fator de esquecimento pequeno. Convém notar que se as variáveis dos algoritmos QR-LSL não forem representadas com a precisão necessária, apesar da consistência retrógrada ser mantida devido as convenções numéricas adotadas, a resposta fornecida pode não ter significado.

Existem outras versões de algoritmos QR-RLS rápidos, além das três apresentadas neste trabalho. Como exemplo pode-se citar [1, 19, 22, 34, 40]. Em particular, a referência [34], publicada em novembro de 1995, apresenta um algoritmo que é semelhante ao QR-LSL *a priori*, utilizando também a atualização do inverso do fator de conversão e erros de predição *a priori* normalizados. Entretanto este algoritmo foi obtido de forma totalmente diferente da apresentada neste trabalho. Além disso, não foram feitas considerações sobre a estabilidade retrógrada. A existência de outros algoritmos que apresentam a estabilidade retrógrada, com uma complexidade computacional reduzida, é uma questão em aberto.



## Referências bibliográficas

- [1] A.A. Rontogiannis, S. Theodoridis: *New fast inverse QR least squares adaptive algorithms*, ICASSP 1995 - Detroit, Proc.IEEE Int. Conf. Acoustics, Speech and Signal Processing, pp.1412-1415.
- [2] A.H. Gray, Jr., J.D. Markel: *A normalized digital filter structure*, IEEE Trans. on Acoustics, Speech and Signal Processing, vol.23, no.3, pp.268-277, June 1975.
- [3] A.H. Sayed: *Displacement structure in signal processing and mathematics*, PhD thesis, Stanford University, Stanford, CA, August 1992.
- [4] A.H. Sayed, T. Kailath: *A state-space approach to adaptive RLS filtering*, IEEE Signal Processing Magazine, pp.18-60, July 1994.
- [5] B.D.O. Anderson, R.R. Bitmead, C.R. Johnson, Jr., P.V. Kokotovic, R.L. Kosut, I.M.Y. Mareels, L. Praly, B.D. Riedle: *Stability of adaptive systems: Passivity and averaging analysis*, Cambridge, MA: M.I.T. Press, 1986.
- [6] B.D.O. Anderson, C.R. Johnson Jr: *Exponential convergence of adaptive identification and control algorithms*, Automática, vol. 18, pp.1-13, January 1982.
- [7] B. Yang, J.F. Böhme: *Rotation-based RLS algorithms: unified derivations, numerical properties, and parallel implementations*, IEEE Trans. on Signal Processing, vol.40, no.5, pp.1151-1167, May 1992.
- [8] C.P. Rialan, L.L. Scharf: *Fast algorithms for computing QR and Cholesky factors of Toeplitz operators*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.36, no.11, pp.1740-1748, November 1988.
- [9] C. Samson, V.U. Reddy: *Fixed point error analysis of the normalized ladder algorithm*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.31, no.5, pp.1177-1191, October 1983.

- [10] D.T.M. Slock: *Backward consistency concept and round-off error propagation dynamics in recursive least-squares algorithms*, Optical Engineering, vol.31, no.6, pp.1153-1169, June 1992.
- [11] D.T.M. Slock, T. Kailath: *Numerically stable fast transversal filters for recursive least squares adaptive filtering*, IEEE Trans. on Signal Processing, vol.39, no.1, pp.92-114, January 1991.
- [12] F. Ling: *Givens rotation based least squares lattice and related algorithms*, IEEE Trans. on Signal Processing, vol.39, no.7, pp.1541-1551, July 1991.
- [13] F. Ling, D. Manolakis, J.G. Proakis: *A recursive modified Gram-Schmidt algorithm for least-squares estimation*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.34, no.4, pp.829-836, August 1986.
- [14] F. Ling, D. Manolakis, J.G. Proakis: *Numerically robust least-squares lattice-ladder algorithms with direct updating of the reflection coefficients*, IEEE Trans. on Acoustics, Speech and Signal Processing, vol.34, no.4, pp.837-845, August 1986.
- [15] G. Carayannis, D.G. Manolakis, N. Kalouptidis: *A fast sequential algorithm for least-squares filtering an prediction*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.31, no.6, pp.1394-1402, December 1983.
- [16] G.W. Stewart: *Introduction to matrix computations*, Academic Press, 1973.
- [17] H. Leung, S. Haykin: *Stability of recursive QRD-LS algorithms using finite-precision systolic arrays implementation*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.37, no.5, pp.760-763, May 1989.
- [18] H. Lev-Ari, T. Kailath, J. Cioffi: *Least squares adaptive lattice and transversal filters: A unified geometric theory*, IEEE Trans. on Information Theory, vol.30, no.2, pp.222-236, March 1984.
- [19] I.K. Proudler, J.G. McWhirter, T.J. Sheperd: *Computationally efficient QR decomposition approach to least squares adaptive filtering*, IEE Proceedings - F, vol.138, no.4, pp.341-353, August 1991.

- [20] J.G. McWhirter: *Recursive least squares minimization using a systolic array*, Proc. SPIE Int. Soc. Opt. Eng., vol. 431, pp.105-112, 1983.
- [21] J.G. Proakis, C.M. Rader, F. Ling, C.L. Nikias: *Advanced digital signal processing*, Maxwell Macmillan International Editions, 1992.
- [22] J.M. Cioffi: *The fast adaptive ROTOR's RLS algorithm*, IEEE Trans. on Acoustics, Speech and Signal Processing, vol.38, no.4, pp.631-653, April 1990.
- [23] J.M. Cioffi, T. Kailath: *Fast, recursive least-squares transversal filters for adaptive filtering*, IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.32, pp.304-337, April 1984.
- [24] J.R.B. Gimenez: *Sobre a estabilidade numérica dos algoritmos de mínimos quadrados rápidos*, Tese de Doutorado, UNICAMP, SP, janeiro de 1995.
- [25] J.R.Bunch, R.C. LeBorne: *Errors accumulation effects for a posteriori RLSL prediction filter*, IEEE Trans. on Signal Processing, vol.43, no.1, pp.150-159, January 1995.
- [26] L. Ljung, M. Morf, D. Falconer: *Fast calculation of gain matrices for recursive estimation schemes*, Int. J. Control, vol.27, pp. 1-19.
- [27] M. Bellanger: *Analyse des signaux et filtrage numérique adaptatif*, Masson et CNET-ENST, Paris, 1989.
- [28] M.D. Miranda, M. Gerken: *A hybrid least squares QR-lattice algorithm using a priori errors*. Submetido ao IEEE Trans. on Signal Processing, em revisão
- [29] M.D. Miranda, M. Gerken: *A hybrid QR-lattice least squares algorithm using a priori errors*. Proceedings 38<sup>TH</sup> MWSCAS, Rio de Janeiro, August 1995.
- [30] M.D. Miranda, M. Gerken: *Sobre algoritmos RLS que utilizam triangularização ortogonal*. Anais do 13<sup>o</sup> SBT, Águas de Lindóia, SP, setembro de 1995.
- [31] P.A. Regalia: *Numerical stability properties of a QR-based fast least squares algorithm*, IEEE Trans. on Signal Processing, vol.41, no.6, pp.2096-2109, June 1993.

- [32] P.A. Regalia: *Numerical stability issues in fast least-squares adaptation algorithms*, Optical Engineering, vol.31, no.6, pp.1144-1152, June 1992.
- [33] P.A. Regalia, M.G. Bellanger: *On the duality between fast QR methods and lattice methods in least squares adaptive filtering*, IEEE Trans. on Signal Processing, vol.39, no.4, pp.879-891, April 1991.
- [34] P.G. Park, T. Kailath: *A lattice algorithm dual to the extended inverse QR algorithm*, Signal Processing, vol. 47, no.2, pp. 115-133, November 1995.
- [35] P. Lancaster, M. Tismenetsky: *Theory of matrices*, Academic Press, 2nd. Ed., 1985.
- [36] P.S.R. Diniz, M.G. Siqueira: *Fixed-point error analysis of the QR-recursive least square algorithm*, IEEE Trans. on Circuits and Systems, vol.42, no.5, pp.334-348, May 1995.
- [37] R.C. North, J.R. Zeidler, W.H.Ku, T.R. Albert: *A floating point arithmetic error analysis of direct and indirect coefficient updating techniques for adaptive lattice filters*, IEEE Trans. on Signal Processing, vol.34, no.5, pp.1809-1823, May 93.
- [38] S. Haykin: *Adaptive filter theory*, Prentice Hall Int., 2nd. Ed., 1991.
- [39] S. Ljung, L. Ljung: *Error propagation properties of recursive least-squares adaptation algorithms*, Automática, vol.21, no.2, pp.157-167, 1985.
- [40] S.T. Alexander, A.L. Ghirnikar: *A method for recursive least squares filtering based upon an inverse QR decomposition*, IEEE Trans. on Signal Processing, vol.41, no.1, pp.20-30 January 1993.
- [41] Y.H. Hu: *CORDIC based VLSI architectures for digital signal processing*, IEEE Signal Processing Magazine, pp.17-34, July 1992.

## Apêndice A

### Propriedades estruturais resultantes da utilização de dados seqüenciais

São apresentadas neste apêndice as várias representações da matriz de dados, da matriz de autocorrelação e da sua inversa, das correlações cruzadas e do fator de conversão, considerando que a seqüência de observação é formada por dados seqüenciais deslocados no tempo (série temporal). Muitas das equações aqui apresentadas são bem conhecidas na literatura especializada, porém para facilitar a apresentação dos capítulos 2 e 3 e tornar a notação mais familiar elas serão rerepresentadas. Além disso, demonstra-se as relações entre as várias formas de representação da inversa da matriz de autocorrelação, das correlações cruzadas e do fator de conversão.

#### A.1 Representações para a matriz de dados

A matriz de dados  $\mathbf{A}_M(n)$  de dimensão  $(n+1) \times M$ , com  $\{u(\ell)\}$  representando uma seqüência pré-janelada de dados para  $0 \leq \ell \leq n$ , apresenta a seguinte forma

$$\mathbf{A}_M(n) = \Lambda^{1/2}(n) \begin{bmatrix} u(0) & 0 & \dots & 0 \\ u(1) & u(0) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ u(M-1) & u(M-2) & \dots & u(0) \\ \vdots & \vdots & \ddots & \vdots \\ u(n-1) & u(n-2) & \dots & u(n-M) \\ u(n) & u(n-1) & \dots & u(n-M+1) \end{bmatrix}, \quad (\text{A.1})$$

onde  $\Lambda(n) = \text{diag}\{\lambda^n, \lambda^{n-1}, \dots, 1\}$  é uma matriz de ponderação sendo  $\lambda$  o fator de esquecimento satisfazendo  $0 << \lambda < 1$ . A matriz  $\Lambda^{1/2}(n)$  pondera cada linha de

índice  $\ell + 1$  da matriz de dados por  $\lambda^{\frac{n-\ell}{2}}$ . Aproveitando o fato da matriz  $\mathbf{A}_M(n)$  ser formada por dados sequenciais deslocados no tempo, são apresentadas a seguir algumas formas de representá-la.

a) Representação em termos das  $n + 1$  linhas,

$$\mathbf{A}_M(n) = \Lambda^{1/2}(n) \left[ \mathbf{u}_M(0) \quad \mathbf{u}_M(1) \quad \dots \quad \mathbf{u}_M(n) \right]^T, \quad (\text{A.2})$$

onde

$$\mathbf{u}_M^T(\ell) = \left[ u(\ell) \quad u(\ell - 1) \quad \dots \quad u(\ell - M + 1) \right] \quad (\text{A.3})$$

é o vetor dos dados, de ordem  $M \times 1$ , no instante  $\ell$ ,  $0 \leq \ell \leq n$ .

b) Representação em termos das  $M$  colunas,

$$\mathbf{A}_M(n) = \left[ \mathbf{a}(n) \quad \mathbf{a}(n - 1) \quad \dots \quad \mathbf{a}(n - M + 1) \right], \quad (\text{A.4})$$

onde

$$\mathbf{a}(n - i + 1) = \Lambda^{1/2}(n) \left[ u(1 - i) \quad u(2 - i) \quad \dots \quad u(n - i + 1) \right]^T, \quad (\text{A.5})$$

para  $1 \leq i \leq M$ , são vetores de ordem  $(n + 1) \times 1$ , linearmente independentes entre si.

c) No caso de atualização no tempo, a matriz de dados pode ser representada em termos dos seguintes blocos:

$$\mathbf{A}_M(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{A}_M(n - 1) \\ \mathbf{u}_M^T(n) \end{bmatrix}. \quad (\text{A.6})$$

A matriz  $\mathbf{A}_M(n - 1)$  de ordem  $n \times M$ , representa a janela de dados no instante  $n - 1$ . No instante  $n$  um novo dado deve ser considerado  $u(n)$ . A inclusão do novo dado aumenta o número de linhas da matriz de dados, sendo  $\mathbf{u}_M^T(n)$  o vetor de entrada definido em (A.3) para  $\ell = n$ .

d) No caso da atualização na ordem do vetor passado  $\mathbf{a}(n - M + 1)$ , a representação pode ser feita como

$$\mathbf{A}_M(n) = \left[ \mathbf{A}_{M-1}(n) \quad \mathbf{a}(n - M + 1) \right]. \quad (\text{A.7})$$

e) No caso da atualização na ordem do vetor futuro  $\mathbf{a}(n)$  a representação pode ser feita como

$$\mathbf{A}_M(n) = \begin{bmatrix} \mathbf{a}(n) & \mathbf{0}^T \\ & \mathbf{A}_{M-1}(n-1) \end{bmatrix}. \quad (\text{A.8})$$

## A.2 Representações para matriz de autocorrelação

A matriz de autocorrelação dos dados de entrada, de ordem  $M \times M$ , definida como

$$\Phi_M(n) = \mathbf{A}_M^T(n) \mathbf{A}_M(n) = \sum_{\ell=0}^n \lambda^{n-\ell} \mathbf{u}_M(\ell) \mathbf{u}_M^T(\ell), \quad (\text{A.9})$$

pode ser atualizada no tempo e/ou na ordem, conforme se apresenta a seguir.

a) A atualização no tempo de  $\Phi_M(n)$  pode ser obtida representando-se  $\mathbf{A}_M(n)$  conforme definida em (A.6), resultando em

$$\Phi_M(n) = \lambda \Phi_M(n-1) + \mathbf{u}_M(n) \mathbf{u}_M^T(n) \quad (\text{A.10})$$

b) A atualização na ordem de  $\Phi_M(n)$  pode ser obtida representando-se  $\mathbf{A}_M(n)$ , conforme definida em (A.7), resultando em

$$\Phi_M(n) = \begin{bmatrix} \Phi_{M-1}(n) & \Omega_{M-1}^b(n) \\ \Omega_{M-1}^{bT}(n) & E_0^b(n) \end{bmatrix}, \quad n \geq M \geq 2, \quad (\text{A.11})$$

onde

$$\tilde{E}_0^b(n) := \|\mathbf{a}(n-M+1)\|^2 = \sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell-M+1)|^2 \quad (\text{A.12})$$

e

$$\Omega_{M-1}^b(n) = \mathbf{A}_{M-1}^T(n) \mathbf{a}(n-M+1) = \sum_{\ell=0}^n \lambda^{n-\ell} \mathbf{u}_{M-1}(\ell) u(\ell-M+1). \quad (\text{A.13})$$

c) A atualização na ordem e no tempo de  $\Phi_M(n)$  pode ser obtida representando-se  $\mathbf{A}_M(n)$ , conforme definida em (A.8), resultando em

$$\Phi_M(n) = \begin{bmatrix} \xi_0^f(n) & \Omega_{M-1}^{fT}(n) \\ \Omega_{M-1}^f(n) & \Phi_{M-1}(n-1) \end{bmatrix}, \quad (\text{A.14})$$

onde

$$\xi_0^f(n) := \|\mathbf{a}(n)\|^2 = \sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell)|^2 \quad (\text{A.15})$$

e

$$\Omega_{M-1}^f(n) = \begin{bmatrix} \mathbf{0} & \mathbf{A}_{M-1}^T(n-1) \end{bmatrix} \mathbf{a}(n) = \sum_{\ell=0}^n \lambda^{n-\ell} \mathbf{u}_{M-1}(\ell-1)u(\ell). \quad (\text{A.16})$$

Os elementos da diagonal principal de  $\Phi_M(n)$  são definidos como

$$\mathbf{a}^T(n-i+1)\mathbf{a}(n-i+1) = \sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell-i)|^2$$

para  $1 \leq i \leq M$ . Devido ao pré-janelamento tem-se a seguinte igualdade

$$\sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell-i)|^2 = \sum_{\ell=0}^{n-i} \lambda^{n-i-\ell} |u(\ell)|^2 = \xi_0^f(n-i),$$

assim é possível expressar os elementos da diagonal principal de  $\Phi_M(n)$  como

$$\mathbf{a}^T(n-i+1)\mathbf{a}(n-i+1) = \xi_0^f(n-i)$$

para  $1 \leq i \leq M$ . Cabe observar que  $E_0^b(n) = \xi_0^f(n-M)$ .

### A.3 Representações para a inversa da matriz de autocorrelação

A seguir são apresentadas três formas de atualizar no tempo e/ou na ordem a inversa da matriz de autocorrelação:

a) Aplicando o lema da inversão matricial na equação (A.10) [38, cap. 13], tem-se

$$\Phi_M^{-1}(n) = \lambda^{-1} \left[ \mathbf{I} - \mathbf{g}(n)\mathbf{u}_M^T(n) \right] \Phi_M^{-1}(n-1) \quad (\text{A.17})$$

onde

$$\mathbf{g}_M(n) = \frac{\lambda^{-1}\Phi_M^{-1}(n-1)\mathbf{u}_M(n)}{1 + \lambda^{-1}\mathbf{u}_M^T(n)\Phi_M^{-1}(n-1)\mathbf{u}_M(n)}. \quad (\text{A.18})$$

Definindo

$$\gamma_M(n) = \frac{\lambda}{\lambda + \mathbf{u}_M^T(n)\Phi_M^{-1}(n-1)\mathbf{u}_M(n)}, \quad (\text{A.19})$$

o vetor  $\mathbf{g}_M(n)$  pode ser expresso como [38, 27]

$$\mathbf{g}_M(n) = \lambda^{-1}\gamma_M(n)\Phi_M^{-1}(n-1)\mathbf{u}_M(n), \quad (\text{A.20})$$



o que é equivalente a

$$\mathbf{g}_M(n) = \Phi_M^{-1}(n) \mathbf{u}_M(n). \quad (\text{A.21})$$

Isolando  $\gamma_M(n)$  em (A.20), e utilizando (A.17), tem-se que

$$\gamma_M(n) = 1 - \mathbf{g}_M^T(n) \mathbf{u}_M(n) = 1 - \mathbf{u}_M^T(n) \Phi_M^{-1}(n) \mathbf{u}_M(n). \quad (\text{A.22})$$

Com auxílio das equações (A.17) e (A.20), pode-se estabelecer a seguinte expressão para atualização no tempo da inversa da matriz de autocorrelação  $\Phi_M^{-1}(n)$ :

$$\Phi_M^{-1}(n) = \lambda^{-1} \Phi_M^{-1}(n-1) - \frac{\mathbf{g}_M(n) \mathbf{g}_M^T(n)}{\gamma_M(n)}. \quad (\text{A.23})$$

b) Considerando o caso da predição regressiva, com auxílio das equações (A.11) e (2.33), resulta a seguinte equação de atualização recursiva na ordem:

$$\Phi_M^{-1}(n) = \begin{bmatrix} \Phi_{M-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \mathbf{B}_M(n) \mathbf{B}_M^T(n), \quad (\text{A.24})$$

onde

$$\mathbf{B}_M^T(n) = \begin{bmatrix} -\mathbf{w}_{M-1}^{bT}(n) & 1 \end{bmatrix} \xi_{M-1}^{-b/2}(n). \quad (\text{A.25})$$

c) Considerando o caso da predição progressiva, com auxílio das equações (A.14) e (2.44), resulta a seguinte equação de atualização recursiva no tempo e na ordem:

$$\Phi_M^{-1}(n) = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \Phi_{M-1}^{-1}(n-1) \end{bmatrix} + \mathbf{F}_M(n) \mathbf{F}_M^T(n), \quad (\text{A.26})$$

onde

$$\mathbf{F}_M^T(n) = \begin{bmatrix} 1 & -\mathbf{w}_{M-1}^{fT}(n) \end{bmatrix} \xi_{M-1}^{-f/2}(n). \quad (\text{A.27})$$

As equações (A.23), (A.24) e (A.26) representam três formas equivalentes de obter a inversa da matriz de autocorrelação. Em aritmética de precisão infinita, a diferença entre as equações (A.24) e (A.26) é nula, resultando em

$$\begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \Phi_{M-1}^{-1}(n-1) \end{bmatrix} - \begin{bmatrix} \Phi_{M-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \mathbf{F}_M(n) \mathbf{F}_M^T(n) - \mathbf{B}_M(n) \mathbf{B}_M^T(n) = \mathbf{0}. \quad (\text{A.28})$$

Utilizando a equação (A.23), na ordem  $M - 1$  para representar  $\Phi_{M-1}^{-1}(n - 1)$  em (A.28), tem-se

$$\begin{aligned} & \begin{bmatrix} \Phi_{M-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} - \lambda \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \Phi_{M-1}^{-1}(n) \end{bmatrix} = \\ & = \mathbf{G}_M(n)\mathbf{G}_M^T(n) + \mathbf{F}_M(n)\mathbf{F}_M^T(n) - \mathbf{B}_M(n)\mathbf{B}_M^T(n) \end{aligned} \quad (\text{A.29})$$

onde

$$\mathbf{G}_M^T(n) = \lambda^{1/2} \gamma_{M-1}^{-1/2}(n) \begin{bmatrix} 0 & \mathbf{g}_{M-1}^T(n) \end{bmatrix} \quad (\text{A.30})$$

A diferença (A.29) pode ser reescrita como

$$\mathbf{P}(n) - \lambda \mathbf{L} \mathbf{P}(n) \mathbf{L}^T = \begin{bmatrix} \mathbf{G}_M(n) & \mathbf{B}_M(n) & \mathbf{F}_M(n) \end{bmatrix} \mathbf{J} \begin{bmatrix} \mathbf{G}_M^T(n) \\ \mathbf{B}_M^T(n) \\ \mathbf{F}_M^T(n) \end{bmatrix}, \quad (\text{A.31})$$

$$\text{onde } \mathbf{P}(n) = \begin{bmatrix} \Phi_{M-1}^{-1}(n) & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}, \mathbf{J} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ e } \mathbf{L} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}.$$

A equação (A.31) relaciona as três formas de representar a inversa da matriz de autocorrelação e revela que o 'resíduo do deslocamento' tem posto 3 [10].

## A.4 Correlações cruzadas

1) O vetor de correlação cruzada entre as seqüências de observações definidas pela matriz  $\mathbf{A}_M(n)$  e o vetor de respostas desejadas  $\mathbf{d}(n)$ , definido na equação (2.7)

$$\boldsymbol{\Omega}_M(n) = \mathbf{A}_M^T(n) \mathbf{d}(n) = \sum_{\ell=0}^n \lambda^{n-\ell} \mathbf{u}_M(\ell) d(\ell),$$

pode ser representado como

$$\boldsymbol{\Omega}_M(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{A}_M^T(n-1) & \mathbf{u}_M(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \mathbf{d}(n-1) \\ d(n) \end{bmatrix}. \quad (\text{A.32})$$

A equação (A.32) torna evidente a seguinte atualização no tempo:

$$\boldsymbol{\Omega}_M(n) = \lambda \boldsymbol{\Omega}_M(n-1) + \mathbf{u}_M(n) d(n). \quad (\text{A.33})$$

Da mesma forma, os vetores  $\Omega_{M-1}^b(n)$  e  $\Omega_{M-1}^f(n)$ , definidos respectivamente em (A.13) e (A.16), podem ser obtidos de forma recorrente no tempo como

$$\Omega_{M-1}^b(n) = \lambda \Omega_{M-1}^b(n-1) + \mathbf{u}_{M-1}(n)u(n-M+1)$$

e

$$\Omega_{M-1}^f(n) = \lambda \Omega_{M-1}^f(n-1) + \mathbf{u}_{M-1}(n-1)u(n).$$

2) Considerando a representação da matriz de dados conforme (A.7), o vetor do erro da predição regressiva definido em (2.25), pode ser expresso como

$$\mathbf{e}_{M-1}^b(n) = \mathbf{A}_M(n) \begin{bmatrix} -\mathbf{w}_{M-1}^b(n) \\ 1 \end{bmatrix}. \quad (\text{A.34})$$

A correlação cruzada entre o vetor do erro da predição regressiva  $\mathbf{e}_{M-1}^b(n)$  e o vetor da resposta desejada  $\mathbf{d}(n)$

$$\Delta_{M-1}^d(n) = \mathbf{e}_{M-1}^{bT}(n)\mathbf{d}(n),$$

pode ser expressa como

$$\Delta_{M-1}^d(n) = \begin{bmatrix} -\mathbf{w}_{M-1}^{bT}(n) & 1 \end{bmatrix} \mathbf{A}_M^T(n)\mathbf{d}(n).$$

Usando (2.7) tem-se

$$\Delta_{M-1}^d(n) = \begin{bmatrix} -\mathbf{w}_{M-1}^{bT}(n) & 1 \end{bmatrix} \Omega_M(n). \quad (\text{A.35})$$

3) Considerando a representação da matriz de dados conforme (A.8), o vetor do erro da predição regressiva definido em (2.36) pode ser reescrito como

$$\mathbf{e}_{M-2}^f(n) = \mathbf{A}_{M-1}(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \end{bmatrix}.$$

A correlação cruzada entre o vetor do erro da predição regressiva  $\mathbf{e}_{M-2}^f(n)$  e o vetor de respostas desejadas no caso da predição regressiva  $\mathbf{a}(n-M+1)$

$$\Delta_{M-2}(n) = \mathbf{e}_{M-2}^{fT}(n)\mathbf{a}(n-M+1),$$

pode ser reescrita como

$$\Delta_{M-2}(n) = \begin{bmatrix} 1 & -\mathbf{w}_{M-2}^{fT}(n) \end{bmatrix} \mathbf{A}_{M-1}^T(n)\mathbf{a}(n-M+1).$$

Usando (A.13) tem-se

$$\Delta_{M-2}(n) = \begin{bmatrix} 1 & -\mathbf{w}_{M-2}^{fT}(n) \end{bmatrix} \boldsymbol{\Omega}_{M-1}^b(n). \quad (\text{A.36})$$

- 4) Considerando o vetor do erro da predição regressiva conforme apresentado em (A.34), porém para  $M - 2$  e  $n - 1$ , a correlação cruzada entre o vetor do erro da predição regressiva  $\mathbf{e}_{M-2}^b(n - 1)$  e o vetor de respostas desejadas no caso da predição progressiva  $\mathbf{a}(n)$

$$\Delta_{M-2}(n) = \begin{bmatrix} 0 & \mathbf{e}_{M-2}^{bT}(n - 1) \end{bmatrix} \mathbf{a}(n),$$

pode ser reescrita como

$$\Delta_{M-2}(n) = \begin{bmatrix} -\mathbf{w}_{M-2}^{bT}(n - 1) & 1 \end{bmatrix} \begin{bmatrix} 0 & \mathbf{A}_{M-1}^T(n - 1) \end{bmatrix} \mathbf{a}(n).$$

Usando (A.16) tem-se

$$\Delta_{M-2}(n) = \begin{bmatrix} -\mathbf{w}_{M-2}^{bT}(n - 1) & 1 \end{bmatrix} \boldsymbol{\Omega}_{M-1}^f(n). \quad (\text{A.37})$$

Como (A.37) e (A.36) são iguais resulta na seguinte igualdade de produto interno de vetores:

$$\begin{bmatrix} 1 & -\mathbf{w}_{M-2}^{fT}(n) \end{bmatrix} \boldsymbol{\Omega}_{M-1}^b(n) = \begin{bmatrix} -\mathbf{w}_{M-2}^{bT}(n - 1) & 1 \end{bmatrix} \boldsymbol{\Omega}_{M-1}^f(n). \quad (\text{A.38})$$

## A.5 Fator de conversão

A seguir são apresentadas as interpretações usuais para o fator de conversão:

- a) Utilizando a equação (2.3), resulta a seguinte expressão para o erro de estimação *a posteriori*:

$$e_M(n) = d(n) - \mathbf{u}_M^T(n) \mathbf{w}_M(n). \quad (\text{A.39})$$

Representando a operação de atualização da filtragem conforme (2.53) em (A.39), e reconhecendo na expressão resultante o erro de estimação *a priori* conforme definido em (2.54), tem-se que

$$e_M(n) = (1 - \mathbf{u}_M^T(n) \mathbf{g}_M(n)) \alpha_M(n).$$

Com auxílio de (A.21) é possível reconhecer  $\gamma_M(n)$  conforme definido em (A.22) resultando em

$$e_M(n) = \alpha_M(n)\gamma_M(n). \quad (\text{A.40})$$

Da última linha da equação (2.25), tem-se a seguinte expressão para o erro da predição regressiva *a posteriori*:

$$b_{M-1}(n) = u(n - M + 1) - \mathbf{u}_{M-1}^T(n)\mathbf{w}_{M-1}^b(n) = \mathbf{u}_M^T(n) \begin{bmatrix} -\mathbf{w}_{M-1}^b(n) \\ 1 \end{bmatrix} \quad (\text{A.41})$$

e de (2.60), (2.59) e (2.61) tem-se que a operação de adaptação do filtro de erro da predição regressiva é definida como

$$\mathbf{w}_{M-1}^b(n) = \mathbf{w}_{M-1}^b(n-1) + \mathbf{g}_{M-1}(n)\psi_{M-1}(n). \quad (\text{A.42})$$

Substituindo (A.42) em (A.41) resulta

$$b_{M-1}(n) = \psi_{M-1}(n)\gamma_{M-1}(n). \quad (\text{A.43})$$

Da última linha da equação (2.36), tem-se a seguinte expressão para o erro da predição progressiva *a posteriori*:

$$f_{M-1}(n) = u(n) - \mathbf{u}_{M-1}^T(n-1)\mathbf{w}_{M-1}^f(n) = \mathbf{u}_M^T(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-1}^f(n) \end{bmatrix} \quad (\text{A.44})$$

e de (2.63), (2.59) e (2.64) tem-se que a operação de adaptação do filtro de erro da predição-progressiva é definida como

$$\mathbf{w}_{M-1}^f(n) = \mathbf{w}_{M-1}^f(n-1) + \mathbf{g}_{M-1}(n-1)\eta_{M-1}(n). \quad (\text{A.45})$$

Substituindo (A.45) em (A.44) resulta

$$f_{M-1}(n) = \eta_{M-1}(n)\gamma_{M-1}(n-1). \quad (\text{A.46})$$

Desta forma,  $\gamma_M(n)$  pode ser interpretado como fator de conversão, transformando os erros *a priori* nos erro *a posteriori*, isto ocorre tanto no caso da estimação, como nos casos das predições progressiva e regressiva. O que já era esperado, porque os casos de predição são problemas de estimação para um sinal desejado particular.

b) A equação (A.23) pode ser reescrita como

$$\lambda \Phi_M^{-1}(n-1) \Phi_M(n) = \mathbf{I} - \lambda \Phi_M(n-1) \frac{\mathbf{g}_M(n) \mathbf{g}_M^T(n)}{\gamma_M(n)}, \quad (\text{A.47})$$

usando  $\mathbf{g}_M(n)$  como definido em (A.20) tem-se

$$\lambda \Phi_M^{-1}(n-1) \Phi_M(n) = \mathbf{I} - \mathbf{u}_M(n) \mathbf{g}_M^T(n).$$

Aplicando o determinante em ambos os lados da equação anterior tem-se

$$\det(\lambda \Phi_M^{-1}(n-1) \Phi_M(n)) = \det(\mathbf{I} - \mathbf{u}_M(n) \mathbf{g}_M^T(n)).$$

Como  $\det(\mathbf{I} - \mathbf{u}_M(n) \mathbf{g}_M^T(n)) = \det(1 - \mathbf{u}_M^T(n) \mathbf{g}_M(n))$ , com auxílio de (A.22) resulta a seguinte expressão:

$$\lambda^M \frac{\det(\Phi_M(n-1))}{\det(\Phi_M(n))} = \det(1 - \mathbf{u}_M^T(n) \mathbf{g}_M(n)) = \gamma_M(n). \quad (\text{A.48})$$

Desta forma, fica demonstrado que  $\gamma_M(n)$  é proporcional à relação entre os determinantes da matriz de autocorrelação dos dados de entrada em instantes de tempo sucessivos.

c) Aplicando a seguinte regra de determinante [35]

$$\det \begin{bmatrix} \mathbf{a} & \mathbf{b} \\ \mathbf{c} & \mathbf{d} \end{bmatrix} = \det(\mathbf{d} - \mathbf{c} \mathbf{a}^{-1} \mathbf{b}) \det(\mathbf{a})$$

nas equações (A.11) e (A.14), e reconhecendo respectivamente (2.33) e (2.44), resulta

$$\xi_{M-1}^b(n) = \frac{\det |\Phi_M(n)|}{\det |\Phi_{M-1}(n)|}, \quad (\text{A.49})$$

e

$$\xi_{M-1}^f(n) = \frac{\det |\Phi_M(n)|}{\det |\Phi_{M-1}(n-1)|}. \quad (\text{A.50})$$

Destas duas últimas expressões conclui-se que

$$\gamma_{M-1}(n) = \lambda^{M-1} \frac{\xi_{M-1}^b(n)}{\xi_{M-1}^f(n)}. \quad (\text{A.51})$$

A variável  $\gamma(\cdot)$  é proporcional também à relação entre as energias de predição progressiva e regressiva.

A partir da representação (A.22) para  $\gamma(\cdot)$ , e considerando as equações de atualização da inversa da matriz de autocorrelação (A.24) e (A.26), é possível obter expressões de atualizações no tempo e/ou na ordem para  $\gamma(\cdot)$  como demonstrado a seguir.

- a) Considerando (A.24) é possível reescrever (A.22) em termos do erro da predição regressiva *a posteriori*

$$\gamma_M(n) = \gamma_{M-1}(n) - \frac{|b_{M-1}(n)|^2}{\xi_{M-1}^b(n)}. \quad (\text{A.52})$$

- b) Considerando (A.26) é possível reescrever (A.22) em termos do erro de predição progressiva *a posteriori*

$$\gamma_M(n) = \gamma_{M-1}(n-1) - \frac{|f_{M-1}(n)|^2}{\xi_{M-1}^f(n)}. \quad (\text{A.53})$$

Para finalizar esta seção, cabe observar que

$$0 \leq \gamma_M(n) \leq 1, \quad (\text{A.54})$$

o que decorre diretamente de (A.19), pois o termo  $\mathbf{u}_M^T(n) \Phi_M^{-1}(n-1) \mathbf{u}_M(n)$  é sempre maior ou igual a zero, podendo ser infinito quando  $\Phi_M(n-1)$  é singular [38].

## Apêndice B

### Erro normalizado em ângulo e estrutura de McWhirter

Neste apêndice é demonstrado que o último elemento do vetor de respostas desejadas  $\mathbf{d}(n)$  rotacionado por  $\mathbf{Q}_M(n)$  é o erro de estimação normalizado em ângulo  $\epsilon_M(n)$ , resultado que pode ser encontrado na referência [20]. Decorrente desta demonstração faz-se uma interpretação geométrica para a atualização no tempo dos vetores  $\mathbf{dq1}(n)$  e  $\mathbf{dq2}(n)$ . É apresentada também a estrutura de McWhirter, a qual permite o cálculo do erro de estimação sem efetuar o cálculo dos coeficientes do filtro transversal.

#### B.1 Erro normalizado em ângulo

Na seção 2.3 do capítulo 2, definiu-se  $\mathbf{Q}_M(n)$  como a matriz que ortogonaliza a matriz de dados e rotaciona o vetor de respostas desejadas, resultando na matriz triangular superior  $\mathbf{R}_M(n)$  e no vetor  $\mathbf{dq}(n)$ :

$$\mathbf{Q}_M(n) \begin{bmatrix} \mathbf{A}_M(n) & \mathbf{d}(n) \end{bmatrix} = \begin{bmatrix} \mathbf{R}_M(n) & \mathbf{dq1}(n) \\ \mathbf{0} & \mathbf{dq2}(n) \end{bmatrix}. \quad (\text{B.1})$$

Considerando a atualização no tempo da matriz unitária  $\mathbf{Q}_M(n)$  conforme equação (2.84), a equação (B.1) pode ser reescrita como

$$\begin{aligned} \mathbf{T}_M(n) \begin{bmatrix} \mathbf{Q}_M(n-1) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \mathbf{A}_M(n-1) & \lambda^{1/2} \mathbf{d}(n-1) \\ \mathbf{u}_M^T(n) & d(n) \end{bmatrix} = \\ = \mathbf{T}_M(n) \begin{bmatrix} \lambda^{1/2} \mathbf{R}_M(n-1) & \lambda^{1/2} \mathbf{dq1}(n-1) \\ \mathbf{0} & \lambda^{1/2} \mathbf{dq2}(n-1) \\ \mathbf{u}_M^T(n) & d(n) \end{bmatrix}. \quad (\text{B.2}) \end{aligned}$$



A matriz  $\mathbf{T}_M(n)$  é o produto de  $M$  rotações de Givens, utilizadas para anular os elementos do vetor  $\mathbf{u}_M^T(n)$  na equação (B.2). Conseqüentemente,  $\mathbf{T}_M(n)$  apresenta a seguinte estrutura:

$$\mathbf{T}_M(n) = \begin{bmatrix} \mathbf{V}(n) & \mathbf{0} & \mathbf{y}(n) \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{x}^T(n) & \mathbf{0}^T & c(n) \end{bmatrix}. \quad (\text{B.3})$$

O elemento  $c(n)$  representa

$$c(n) = \prod_{i=1}^M \cos \theta_i^b(n), \quad (\text{B.4})$$

sendo que os ângulos  $\theta_i^b(n)$  são calculados de forma a eliminar os elementos do vetor  $\mathbf{u}_M(n)$  em (B.2).

A equação (B.2) pode ser reescrita como

$$\begin{bmatrix} \mathbf{V}(n) & \mathbf{0} & \mathbf{y}(n) \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{x}^T(n) & \mathbf{0}^T & c(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} \mathbf{R}_M(n-1) & \lambda^{1/2} \mathbf{dq1}(n-1) \\ \mathbf{0} & \lambda^{1/2} \mathbf{dq2}(n-1) \\ \mathbf{u}_M^T(n) & d(n) \end{bmatrix} = \\ = \begin{bmatrix} \mathbf{R}_M(n) & \mathbf{dq1}(n) \\ \mathbf{0} & \mathbf{dq2}(n) \end{bmatrix}.$$

Desta última equação matricial é possível obter as seguintes expressões de recorrência no tempo:

$$\mathbf{R}_M(n) = \lambda^{1/2} \mathbf{V}(n) \mathbf{R}_M(n-1) + \mathbf{y}(n) \mathbf{u}_M^T(n), \quad (\text{B.5})$$

$$\mathbf{dq1}(n) = \lambda^{1/2} \mathbf{V}(n) \mathbf{dq1}(n-1) + \mathbf{y}(n) d(n), \quad (\text{B.6})$$

$$\mathbf{dq2}(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{dq2}(n-1) \\ c(n) d(n) - \lambda^{1/2} \mathbf{x}^T(n) \mathbf{dq1}(n-1) \end{bmatrix} \quad (\text{B.7})$$

e

$$-\lambda^{1/2} \mathbf{x}^T(n) \mathbf{R}_M(n-1) + c(n) \mathbf{u}_M^T(n) = 0. \quad (\text{B.8})$$

A equação (B.8) pode ser reescrita como

$$\lambda^{1/2} \mathbf{x}^T(n) \mathbf{R}_M(n-1) = \mathbf{u}_M^T(n) c(n). \quad (\text{B.9})$$

Define-se o último elemento do vetor  $\mathbf{dq2}(n)$ , equação (B.7), como

$$v(n) = c(n)d(n) - \lambda^{1/2}\mathbf{x}^T(n)\mathbf{dq1}(n-1).$$

Considerando (2.74) para  $n-1$  e a equação (B.9), tem-se que

$$\lambda^{1/2}\mathbf{x}^T(n)\mathbf{dq1}(n-1) = c(n)\mathbf{u}_M^T(n)\mathbf{w}_M(n-1),$$

assim  $v(n)$  pode ser reescrito como

$$v(n) = c(n)(d(n) - \mathbf{u}_M^T(n)\mathbf{w}_M(n-1)).$$

Reconhecendo o erro de predição *a priori*, conforme definido na equação (2.54), tem-se  $v(n) = c(n)\alpha_M(n)$ .

Da equação (2.78), conclui-se que o último elemento do vetor

$$\mathbf{e}_M(n) = \mathbf{Q2}^T(n)\mathbf{Q2}(n)\mathbf{d}(n)$$

é o erro de estimação *a posteriori* de ordem  $M$  no instante  $n$ . Com auxílio das equações (2.84) e (B.3) é possível obter a seguinte equação recorrente no tempo para atualização da matriz  $\mathbf{Q2}(n)$ :

$$\begin{aligned} \mathbf{Q2}(n) &= \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{x}^T(n) & \mathbf{0}^T & c(n) \end{bmatrix} \begin{bmatrix} \mathbf{Q1}(n-1) & \mathbf{0} \\ \mathbf{Q2}(n-1) & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{Q2}(n-1) & \mathbf{0} \\ -\mathbf{x}^T(n)\mathbf{Q1}(n-1) & c(n) \end{bmatrix}. \end{aligned} \quad (\text{B.10})$$

Desta última equação é possível expressar (2.78) como

$$\mathbf{Q2}^T(n)\mathbf{dq2}(n) = \begin{bmatrix} \mathbf{Q2}^T(n-1) & -\mathbf{Q1}^T(n-1)\mathbf{x}(n) \\ \mathbf{0}^T & c(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2}\mathbf{dq2}(n-1) \\ c(n)\alpha_M(n) \end{bmatrix}.$$

De onde conclui-se que o último elemento do vetor  $\mathbf{Q2}^T(n)\mathbf{Q2}(n)\mathbf{d}(n)$  pode ser expresso como

$$e_M(n) = c^2(n)\alpha_M(n).$$

Então,  $c^2(n)$  é a variável que transforma o erro de estimação *a priori* no erro de estimação *a posteriori*, representando o fator de conversão  $\gamma_M(n)$ , conforme definido em (A.40). Assim, o último elemento do vetor  $\mathbf{dq2}(n)$

$$v(n) = \gamma_M^{1/2}(n)\alpha_M(n) = \epsilon_M(n),$$

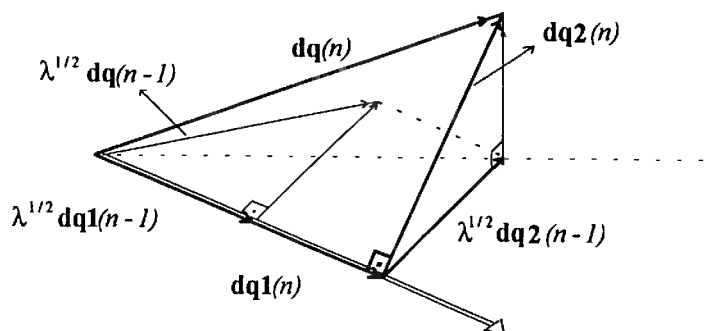


Figura B.1: Atualização no tempo do vetor da resposta desejada rotacionado.

representa o erro normalizado em ângulo definido em (2.57) e este vetor pode ser expresso como

$$\mathbf{dq2}(n) = \begin{bmatrix} \lambda^{1/2} \mathbf{dq2}(n-1) \\ \epsilon_M(n) \end{bmatrix}. \quad (\text{B.11})$$

Cabe observar que a matriz  $\mathbf{V}(n)$  é a mesma matriz  $\mathbf{\Gamma}(n)$  na seção 2.6 do capítulo 2 e os vetores  $\mathbf{y}(n)$  e  $\mathbf{x}(n)$  são respectivamente os vetores  $\bar{\mathbf{b}}(n)$  e  $\gamma_M^{1/2}(n) \bar{\boldsymbol{\psi}}(n)$ .

### Interpretação geométrica

Da equação (B.6) é possível observar que o vetor  $\mathbf{dq1}(n)$  fica no mesmo espaço  $M$  dimensional de  $\mathbf{dq1}(n-1)$ , isto é, a sua dimensão não aumenta. Da equação (B.11) é possível observar e o vetor  $\mathbf{dq2}(n)$  tem duas componentes ortogonais, o vetor  $\mathbf{dq2}(n-1)$  e uma nova componente na direção que foi acrescido o avanço do tempo  $\epsilon_M(n)$  que é o erro de estimação normalizado em ângulo. A figura B.1 ilustra esta interpretação.

## B.2 Estrutura de McWhirter

A equação (2.87) mostra que as operações de adaptação e de filtragem dos algoritmos QR-RLS podem ser descritas pelo seguinte sistema variante no tempo

$$\begin{bmatrix} \mathbf{dq1}(n) \\ \epsilon_M(n) \end{bmatrix} = \mathbf{S}_M^g(n) \begin{bmatrix} \lambda^{1/2} \mathbf{dq1}(n-1) \\ d(n) \end{bmatrix}, \quad (\text{B.12})$$

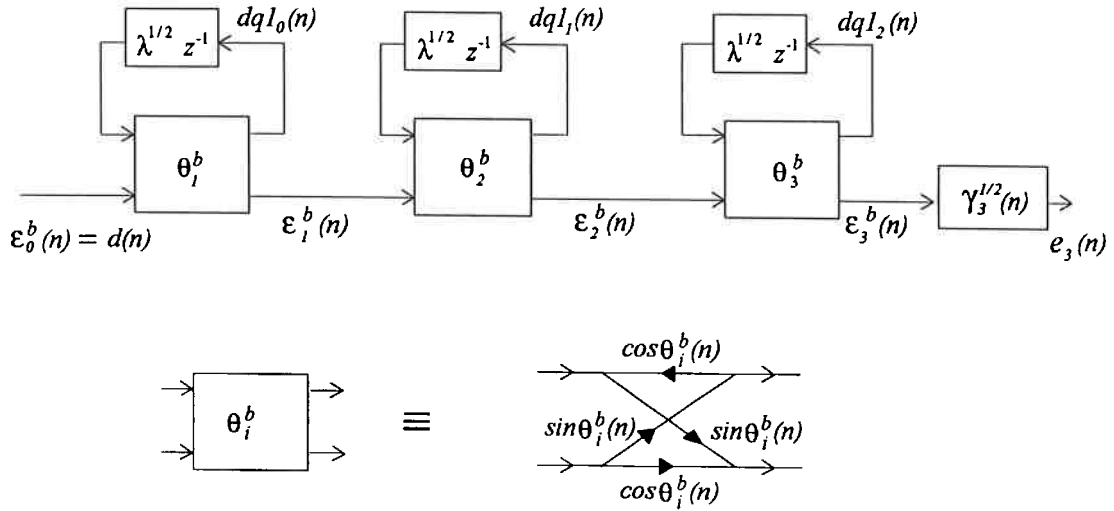


Figura B.2: Estrutura de McWhirter para  $M = 3$ .

onde a matriz unitária  $\mathbf{S}_M^g(n)$  é a matriz de sistema. Os erros de estimação *a posteriori* podem ser obtidos diretamente, a partir de (B.12), sem efetuar a operação de substituições sucessivas necessárias no cálculo do vetor de coeficientes do filtro transversal  $\mathbf{w}_M(n)$  [20]. A equação (B.12) caracteriza um sistema ortogonal variante no tempo devido as características da própria matriz  $\mathbf{S}_M^g(n)$ , a qual representa o produto de  $M$  rotações de Givens. Este sistema pode ser representado como uma cascata de sistemas ortogonais de primeira ordem, conhecida como estrutura de McWhirter [33], conforme representado na figura B.2 para ordem  $M = 3$ . Nesta estrutura cada bloco é caracterizado por uma rotação de Givens definida por  $\theta_i^b(n)$  na seguinte forma:

$$\begin{bmatrix} dq1_i(n) \\ \epsilon_i(n) \end{bmatrix} = \begin{bmatrix} \cos \theta_i^b(n) & \sin \theta_i^b(n) \\ \sin \theta_i^b(n) & \cos \theta_i^b(n) \end{bmatrix} \begin{bmatrix} \lambda^{1/2} dq1_i(n-1) \\ \epsilon_{i-1}(n) \end{bmatrix}. \quad (\text{B.13})$$

Cabe observar que neste caso a raiz quadrada de  $\gamma_M^{1/2}(n)$  pode ser calculada através da seguinte expressão [38]:

$$\gamma_M^{1/2}(n) = \prod_{i=1}^M \cos \theta_i^b(n). \quad (\text{B.14})$$

A equação (B.14) pode ser reescrita como

$$\gamma_i^{1/2}(n) = \gamma_{i-1}^{1/2}(n) \cos \theta_i^b(n) \quad (\text{B.15})$$

para qualquer ordem  $i$ , com  $1 \leq i \leq M$ . Esta última, expressão permite realizar a atualização na ordem do fator de conversão.

## Apêndice C

### Sobre a atualização na ordem dos coeficientes do filtro de erro de predição

Neste apêndice são apresentadas as equações de atualização na ordem dos coeficientes dos filtros de erro de predição da estrutura transversal. Demonstra-se que a atualização na ordem destes coeficientes normalizados e do vetor normalizado do ganho de Kalman pode ser feita a partir dos ângulos que definem a estrutura em treliça normalizada  $\phi_i(n)$  e dos ângulos que definem a estrutura de McWhirter  $\theta_i^b(n)$ .

#### C.1 Atualização dos coeficientes do filtro transversal

Considerando a representação da matriz de autocorrelação conforme (A.14) para ordem  $M - 1$ , as equações (2.44) e (2.39) reescritas para ordem  $M - 1$ , podem ser agrupadas como

$$\Phi_{M-1}(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \end{bmatrix} = \begin{bmatrix} \xi_{M-2}^f(n) \\ \mathbf{0} \end{bmatrix}. \quad (\text{C.1})$$

Usando a representação de  $\Phi_M(n)$ , conforme (A.11), e reconhecendo (A.36) e (C.1) tem-se que:

$$\Phi_M(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \xi_{M-2}^f(n) \\ \mathbf{0} \\ \Delta_{M-2}(n) \end{bmatrix}. \quad (\text{C.2})$$

Considerando a representação da matriz de autocorrelação, conforme (A.11) para ordem  $M - 1$ , as equações (2.33) e (2.28) reescritas para ordem  $M - 1$  podem

ser agrupadas como

$$\Phi_{M-1}(n) \begin{bmatrix} -\mathbf{w}_{M-2}^b(n) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \xi_{M-2}^b(n) \end{bmatrix} \quad (\text{C.3})$$

Usando a representação de  $\Phi_M(n)$ , conforme (A.14), e reconhecendo (A.37) e (C.3), tem-se que:

$$\Phi_M(n) \begin{bmatrix} 0 \\ -\mathbf{w}_{M-2}^b(n-1) \\ 1 \end{bmatrix} = \begin{bmatrix} \Delta_{M-2}(n) \\ 0 \\ \xi_{M-2}^b(n-1) \end{bmatrix}. \quad (\text{C.4})$$

Subtraindo (C.2) da equação (C.4) multiplicada em ambos os lados por  $k_{M-1}^f(n)$ , resulta em

$$\begin{aligned} \Phi_M(n) \left( \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \\ 0 \end{bmatrix} - k_{M-1}^f(n) \begin{bmatrix} 0 \\ -\mathbf{w}_{M-2}^b(n-1) \\ 1 \end{bmatrix} \right) &= \\ &= \begin{bmatrix} \xi_{M-2}^f(n) \\ \mathbf{0} \\ \Delta_{M-2}(n) \end{bmatrix} - k_{M-1}^f(n) \begin{bmatrix} \Delta_{M-2}(n) \\ 0 \\ \xi_{M-2}^b(n-1) \end{bmatrix}. \end{aligned} \quad (\text{C.5})$$

De (3.23) tem-se que

$$\Delta_{M-2}(n) = \xi_{M-2}^f(n-1)k_{M-1}^f(n)$$

e de (3.24), é possível concluir que

$$\xi_{M-1}^f(n) = \xi_{M-2}^f(n) - k_{M-1}^f(n)\Delta_{M-2}(n),$$

assim, a equação (C.5) pode ser reescrita como

$$\Phi_M(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-1}^f(n) \end{bmatrix} = \begin{bmatrix} \xi_{M-1}^f(n) \\ 0 \end{bmatrix}. \quad (\text{C.6})$$

Subtraindo (C.4) da equação (C.2) multiplicada em ambos os lados por  $k_{M-1}^b(n)$

resulta em

$$\begin{aligned} \Phi_M(n) \left( \begin{bmatrix} 0 \\ -\mathbf{w}_{M-2}^b(n-1) \\ 1 \end{bmatrix} - k_{M-1}^b(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \\ 0 \end{bmatrix} \right) &= \\ &= \begin{bmatrix} \Delta_{M-2}(n) \\ 0 \\ \xi_{M-2}^b(n-1) \end{bmatrix} - k_{M-1}^b(n) \begin{bmatrix} \xi_{M-2}^f(n) \\ 0 \\ \Delta_{M-2}(n) \end{bmatrix}. \end{aligned} \quad (\text{C.7})$$

De (3.17) tem-se que

$$\Delta_{M-2}(n) = k_{M-1}^b(n) \xi_{M-2}^f(n)$$

e de (3.19) é possível concluir que

$$\xi_{M-1}^b(n) = \xi_{M-2}^b(n-1) - k_{M-1}^b(n) \Delta_{M-2}(n),$$

assim (C.7) pode ser reescrita como:

$$\Phi_M(n) \begin{bmatrix} -\mathbf{w}_{M-1}^b(n) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \xi_{M-1}^b(n) \end{bmatrix}. \quad (\text{C.8})$$

De (C.5) e (C.6) decorre a seguinte equação recorrente na ordem para cálculo do vetor de coeficientes do filtro de predição progressiva:

$$\begin{bmatrix} 1 \\ -\mathbf{w}_{M-1}^f(n) \end{bmatrix} = \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \\ 0 \end{bmatrix} - k_{M-1}^f(n) \begin{bmatrix} 0 \\ -\mathbf{w}_{M-2}^b(n-1) \\ 1 \end{bmatrix}. \quad (\text{C.9})$$

De (C.7) e (C.8) decorre a seguinte equação recorrente na ordem e no tempo para cálculo do vetor de coeficientes do filtro de predição regressiva:

$$\begin{bmatrix} -\mathbf{w}_{M-1}^b(n) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\mathbf{w}_{M-2}^b(n-1) \\ 1 \end{bmatrix} - k_{M-1}^b(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{M-2}^f(n) \\ 0 \end{bmatrix}. \quad (\text{C.10})$$

As equações (C.9) e (C.10) são válidas para qualquer ordem  $i$  com  $1 \leq i \leq M$ , sendo usadas para atualização na ordem dos coeficientes dos filtros de predição progressiva e regressiva a partir dos coeficientes de reflexão da predição progressiva e regressiva.

As equações (C.9) e (C.10) são recorrentes na ordem e no tempo, entretanto é possível eliminar a recorrência no tempo destas equações. Considere-se a equação (A.42) para ordem  $i - 1$  reescrita como

$$-\mathbf{w}_{i-1}^b(n-1) = -\mathbf{w}_{i-1}^b(n) + \mathbf{g}_{i-1}(n)\psi_{i-1}(n). \quad (\text{C.11})$$

Substituindo (C.11) nas equações (C.9) e (C.10) reescritas para ordem  $i$ , resulta em

$$\begin{bmatrix} 1 \\ -\mathbf{w}_i^f(n) \end{bmatrix} = \begin{bmatrix} 1 \\ -\mathbf{w}_{i-1}^f(n) \\ 0 \end{bmatrix} + \quad (\text{C.12})$$

$$-k_i^f(n) \begin{bmatrix} 0 \\ -\mathbf{w}_{i-1}^b(n) \\ 1 \end{bmatrix} - k_i^f(n)\psi_{i-1}(n) \begin{bmatrix} 0 \\ \mathbf{g}_{i-1}(n) \\ 0 \end{bmatrix}$$

e

$$\begin{bmatrix} -\mathbf{w}_i^b(n) \\ 1 \end{bmatrix} = -k_i^b(n) \begin{bmatrix} 1 \\ -\mathbf{w}_{i-1}^f(n) \\ 0 \end{bmatrix} + \quad (\text{C.13})$$

$$+ \begin{bmatrix} 0 \\ -\mathbf{w}_{i-1}^b(n) \\ 1 \end{bmatrix} + \psi_{i-1}(n) \begin{bmatrix} 0 \\ \mathbf{g}_{i-1}(n) \\ 0 \end{bmatrix}.$$

Com isto foi introduzido na atualização dos coeficientes do filtro transversal de erro de predição o vetor do ganho do Kalman  $\mathbf{g}_{i-1}(n)$ . A atualização na ordem do vetor  $\mathbf{g}_i(n)$  pode ser obtida substituindo (A.24) em (A.21) e reconhecendo a equação (A.41) para ordem  $i - 1$  resultando em

$$\mathbf{g}_i(n) = \begin{bmatrix} \mathbf{g}_{i-1}(n) \\ 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{w}_{i-1}^b(n) \\ 1 \end{bmatrix} \frac{b_{i-1}(n)}{\xi_{i-1}^b(n)}. \quad (\text{C.14})$$

A atualização na ordem do vetor  $\mathbf{g}_i(n)$  pode também ser obtida substituindo (A.26) em (A.21) e reconhecendo a equação (A.44) para ordem  $i - 1$  resultando em

$$\mathbf{g}_i(n) = \begin{bmatrix} 0 \\ \mathbf{g}_{i-1}(n-1) \end{bmatrix} + \begin{bmatrix} 1 \\ -\mathbf{w}_{i-1}^f(n) \end{bmatrix} \frac{f_{i-1}(n)}{\xi_{i-1}^f(n)}. \quad (\text{C.15})$$



Com (C.15) a atualização do vetor  $\mathbf{g}_i(n)$  é feita de forma recorrente na ordem e no tempo. Usando (C.14) a atualização do vetor  $\mathbf{g}_i(n)$  é feita de forma recorrente apenas na ordem.

## C.2 Atualização dos coeficientes do filtro transversal normalizado

As equações (C.12) e (C.13) juntamente com (C.14) podem ser reescritas em função dos vetores normalizados  $\mathbf{F}_{i+1}(n)$ ,  $\mathbf{B}_{i+1}(n)$  e  $\mathbf{G}_{i+1}(n)$  (definidos para uma ordem  $M$  em (A.25), (A.27) e (A.30) respectivamente), resultando em

$$\begin{bmatrix} \mathbf{F}_{i+1}^T(n) \\ \mathbf{B}_{i+1}^T(n) \\ \mathbf{G}_{i+1}^T(n) \end{bmatrix} = \Sigma_i(n) \begin{bmatrix} [\mathbf{F}_i^T(n) \ 0] \\ [0 \ \mathbf{B}_i^T(n)] \\ [\mathbf{G}_i^T(n) \ 0] \end{bmatrix}, \quad \text{para } 1 \leq i \leq M, \quad (\text{C.16})$$

onde  $\Sigma_i(n)$  é definida como a seguinte matriz:

$$\Sigma_i(n) = \begin{bmatrix} \frac{\xi_{i-1}^{f/2}(n)}{\xi_{i-1}^{f/2}(n)} & \frac{-k_i^f(n)\xi_{i-1}^{b/2}(n)}{\xi_{i-1}^{f/2}(n)} & \frac{-\gamma_{i-1}^{1/2}(n)k_i^f(n)\psi_{i-1}(n)}{\lambda^{1/2}\xi_{i-1}^{f/2}(n)} \\ \frac{-k_i^b(n)\xi_{i-1}^{f/2}(n)}{\xi_{i-1}^{b/2}(n)} & \frac{\xi_{i-1}^{b/2}(n)}{\xi_{i-1}^{b/2}(n)} & \frac{\gamma_{i-1}^{1/2}(n)\psi_{i-1}(n)}{\lambda^{1/2}\xi_{i-1}^{b/2}(n)} \\ 0 & \frac{\lambda^{1/2}b_{i-1}(n)}{\gamma_i^{1/2}(n)\xi_{i-1}^{b/2}(n)} & \frac{\gamma_{i-1}^{1/2}(n)}{\gamma_i^{1/2}(n)} \end{bmatrix}. \quad (\text{C.17})$$

Utilizando as equações (3.37) e (3.38) juntamente com as equações do item 3.6 a matriz  $\Sigma_i(n)$  pode ser expressa em função dos ângulos  $\phi_i(n)$  e  $\theta_i^b(n)$ , isto é

$$\Sigma_i(n) = \begin{bmatrix} \frac{1}{\cos \phi_i} & \frac{-\lambda^{1/2} \sin \phi_i}{\cos \phi_i \cos \theta_i^b} & \frac{-\sin \phi_i \sin \theta_i^b}{\cos \phi_i \cos \theta_i^b} \\ \frac{\sin \phi_i}{\cos \phi_i} & \frac{\lambda^{1/2}}{\cos \phi_i \cos \theta_i^b} & \frac{\sin \theta_i^b}{\cos \phi_i \cos \theta_i^b} \\ 0 & \frac{\lambda^{1/2} \sin \theta_i^b}{\cos \theta_i^b} & \frac{1}{\cos \theta_i^b} \end{bmatrix}. \quad (\text{C.18})$$

Os índices  $(n)$  dos ângulos  $\phi_i(n)$  e  $\theta_i^b(n)$  foram suprimidos para simplificar a notação. A matriz  $\Sigma_i(n)$  pode ser representada como o produto de duas matrizes uma em

função dos ângulos  $\phi_i(n)$  e outra em função dos ângulos  $\theta_i^b(n)$

$$\Sigma_i(n) = \begin{bmatrix} 1 & -\frac{\sin \phi_i}{\cos \phi_i} & 0 \\ \frac{\sin \phi_i}{\cos \phi_i} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{\lambda^{1/2}}{\cos \theta_i^b} & \frac{\sin \theta_i^b}{\cos \theta_i^b} \\ 0 & \frac{\lambda^{1/2} \sin \theta_i^b}{\cos \theta_i^b} & 1 \end{bmatrix}. \quad (\text{C.19})$$

As equações (C.16) fazem a atualização na ordem dos vetores de coeficientes de predição normalizados  $\mathbf{F}_{i+1}(n)$  e  $\mathbf{B}_{i+1}(n)$  e do vetor do ganho de Kalman normalizado  $\mathbf{G}_{i+1}(n)$  a partir dos ângulos que definem a estrutura em treliça normalizada  $\phi_i(n)$  e dos ângulos que definem a estrutura de McWhirter  $\theta_i^b(n)$ . Esta atualização pode também ser representada através de polinômios em  $z$ . Considere-se o vetor

$$\mathbf{Z}_{i+1} = \left[ 1 \quad \frac{z}{\lambda^{1/2}} \quad \frac{z^2}{\lambda^{2/2}} \quad \dots \quad \frac{z^i}{\lambda^{i/2}} \right]^T$$

e os polinômios em  $z$

$$F_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) = \frac{\mathbf{Z}_{i+1}^T}{\xi_i^{f/2}(n)} \begin{bmatrix} 1 \\ -\mathbf{w}_i^f(n) \end{bmatrix}, \quad (\text{C.20})$$

$$B_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) = \frac{\mathbf{Z}_{i+1}^T}{\xi_i^{b/2}(n)} \begin{bmatrix} -\mathbf{w}_i^b(n) \\ 1 \end{bmatrix} \quad (\text{C.21})$$

e

$$G_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) = \frac{\mathbf{Z}_{i+1}^T}{\gamma_i^{1/2}(n)} \begin{bmatrix} 0 \\ \mathbf{g}_i(n) \end{bmatrix}. \quad (\text{C.22})$$

As equações de atualização na ordem dos vetores dos coeficientes (C.16) podem ser expressas em termos de atualização na ordem dos polinômios em  $z$

$$\begin{bmatrix} F_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) \\ B_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) \\ G_{i+1}\left(\frac{z}{\lambda^{1/2}}\right) \end{bmatrix} = \Sigma_i(n, z) \begin{bmatrix} F_i\left(\frac{z}{\lambda^{1/2}}\right) \\ B_i\left(\frac{z}{\lambda^{1/2}}\right) \\ G_i\left(\frac{z}{\lambda^{1/2}}\right) \end{bmatrix}, \quad \text{para } 1 \leq i \leq M, \quad (\text{C.23})$$

onde

$$\Sigma_i(n, z) = \begin{bmatrix} 1 & -\frac{\sin \phi_i}{\cos \phi_i} & 0 \\ \frac{\sin \phi_i}{\cos \phi_i} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{z}{\cos \theta_i^b} & \frac{\sin \theta_i^b}{\cos \theta_i^b} \\ 0 & \frac{z \sin \theta_i^b}{\cos \theta_i^b} & 1 \end{bmatrix}.$$

Cabe observar que nas expressões (C.20) a (C.22) a dependência com o tempo  $n$  foi omitida para facilitar a notação. Além disso, para ordem 0 vale  $F_1\left(\frac{z}{\lambda^{1/2}}\right) = \xi_0^{-f/2}(n)$ ,  $B_1\left(\frac{z}{\lambda^{1/2}}\right) = \xi_0^{-b/2}(n)$ ,  $G_1\left(\frac{z}{\lambda^{1/2}}\right) = 0$  e no instante zero  $G_i\left(\frac{z}{\lambda^{1/2}}\right) = 0$  para  $1 \leq i \leq M$ .

## Apêndice D

### Condição de excitação persistente e condicionamento da matriz de dados

Para sinais perfeitamente preditíveis e  $\lambda < 1$  a matriz de dados com o passar do tempo se aproxima arbitrariamente de uma situação de posto nulo. Para sinais não perfeitamente preditíveis a matriz de dados  $\mathbf{A}_M(n)$  pode não estar próxima de tal situação. A ‘distância’ de  $\mathbf{A}_M(n)$  da ‘deficiência do posto’ pode ser expressa utilizando-se a dispersão dos seus valores singulares, definida como

$$\chi(n) = \frac{s_{\max}(n)}{s_{\min}(n)},$$

onde  $s_{\max}(n)$  e  $s_{\min}(n)$  são respectivamente o maior e o menor entre os  $M$  valores singulares  $s_i(n)$ , com  $1 \leq i \leq M$ , de  $\mathbf{A}_M(n)$ . Quando  $\chi(n) \rightarrow \infty$ , isto é, quando existe uma grande dispersão de valores singulares, a matriz  $\mathbf{A}_M(n)$  se aproxima da ‘deficiência do posto’, e vice-versa. É interessante notar que os valores singulares de  $\mathbf{A}_M(n)$ , e conseqüentemente  $\chi(n)$ , estão diretamente relacionados à condição de persistência da excitação que exige, para que a excitação seja persistente, a existência de duas constantes positivas  $b_1$  e  $b_2$  tais que <sup>1</sup>

$$0 < b_1 \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq b_2 \mathbf{I} < \infty. \quad (\text{D.1})$$

Para se chegar a esta conclusão basta considerar a decomposição em valores singulares de  $\mathbf{A}_M(n)$ :

$$\mathbf{A}_M(n) = \mathbf{U}_M(n) \mathbf{S}_M(n) \mathbf{V}_M(n), \quad (\text{D.2})$$

onde  $\mathbf{S}_M(n) = \text{Diag}\{s_i(n), 1 \leq i \leq M\}$  e  $\mathbf{U}_M(n)$  e  $\mathbf{V}_M(n)$  são matrizes unitárias, respectivamente de dimensão  $(n+1) \times M$  e  $M \times M$ . Com (D.2) resulta de imediato

$$\Phi_M(n) = \mathbf{A}_M^T(n) \mathbf{A}_M(n) = \mathbf{V}_M^T(n) \mathbf{S}_M^2(n) \mathbf{V}_M(n) \quad (\text{D.3})$$

<sup>1</sup>A notação  $\mathbf{B} < \mathbf{C}$  indica que a matriz  $\mathbf{C} - \mathbf{B}$  é positiva definida, e  $\mathbf{B} \leq \mathbf{C}$  indica que a matriz  $\mathbf{C} - \mathbf{B}$  é positiva semidefinida.

e conseqüentemente

$$b_2 \mathbf{I} - \mathbf{A}_M^T(n) \mathbf{A}_M(n) \geq 0 \Leftrightarrow b_2 \mathbf{I} - \mathbf{S}_M^2(n) \geq 0 \Leftrightarrow b_2 \geq s_{\max}^2(n) \quad (\text{D.4})$$

e

$$\mathbf{A}_M^T(n) \mathbf{A}_M(n) - b_1 \mathbf{I} \geq 0 \Leftrightarrow \mathbf{S}_M^2(n) - b_1 \mathbf{I} \geq 0 \Leftrightarrow b_1 \leq s_{\min}^2(n). \quad (\text{D.5})$$

Conclui-se portanto que se existem constantes  $0 < b_1 < b_2 < \infty$  tais que para qualquer  $n \geq M - 1 \geq 0$ <sup>2</sup> vale  $0 < b_1 \leq s_{\min}^2(n) \leq s_{\max}^2(n) \leq b_2$ , então o espalhamento dos valores singulares  $\chi(n)$  é limitado superiormente, isto é,

$$\chi(n) < \sqrt{\frac{b_2}{b_1}} \quad (\text{D.6})$$

e a condição de persistência de excitação

$$0 < b_1 \mathbf{I} \leq s_{\min}^2(n) \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq s_{\max}^2(n) \mathbf{I} \leq b_2 \mathbf{I} < \infty \quad (\text{D.7})$$

é atendida. As relações (D.4) e (D.5) mostram que se a condição (D.7) de persistência de excitação é atendida então o espalhamento dos valores singulares  $\chi(n)$  é limitado superiormente, pois  $s_{\min}(n)$  é limitado inferiormente e  $s_{\max}(n)$  é limitado superiormente. Por outro lado, como  $\chi(n)$  pode ser limitado mesmo em situações em que  $s_{\max}(n)$  e  $s_{\min}(n)$  são arbitrariamente grandes ou arbitrariamente pequenos é necessário introduzir uma condição a mais para fazer com que (D.1) seja equivalente a existência de um limite superior para  $\chi(n)$ . Uma possibilidade é exigir que a potência do sinal  $\xi_0^f(n) = \sum_{\ell=0}^n \lambda^{n-\ell} |u(\ell)|^2$  seja limitada inferiormente e superiormente para  $\forall n \geq 0$  por duas constantes positivas  $\beta_1$  e  $\beta_2$ :

$$0 < \beta_1 \leq \xi_0^f(n) \leq \beta_2 < \infty. \quad (\text{D.8})$$

Supondo então que  $\chi(n)$  é limitado superiormente, isto é,

$$\chi(n) < r_o, \quad \forall n \geq 0,$$

é relativamente simples obter um limite inferior para  $s_{\min}^2(n)$  e superior para  $s_{\max}^2(n)$ :

<sup>2</sup>Com  $u(0) \neq 0$ , deve-se considerar  $n$  tal que  $n \geq M - 1 > 0$ , pois caso contrário  $\mathbf{A}_M(n)$  terá pelo menos uma coluna nula e portanto  $s_{\min}(n)$  será necessariamente nulo.

a) Limite inferior para  $s_{\min}^2(n)$ :

A norma de Frobenius de  $\Phi_M(n)$  satisfaz

$$\begin{aligned}\xi_0^f(n) &< \left( \sum_{i,j=1}^M |\Phi_{i,j}(n)|^2 \right)^{1/2} = \|\Phi_M(n)\|_F = \\ &= \sqrt{\text{traço}(\Phi_M^T(n)\Phi_M(n))} = \sqrt{\text{traço}(\mathbf{V}_M^T(n)\mathbf{S}_M^4(n)\mathbf{V}_M(n))} = \\ &= \sqrt{\text{traço}(\mathbf{S}_M^4(n))} = \sqrt{\sum_{i=1}^M s_i^4(n)} \leq \sqrt{M} s_{\max}^2(n).\end{aligned}$$

Desta forma

$$\xi_0^f(n) < \sqrt{M} s_{\max}^2(n). \quad (\text{D.9})$$

Como

$$\frac{s_{\max}^2(n)}{s_{\min}^2(n)} < r_o^2, \quad (\text{D.10})$$

resulta então que

$$\xi_0^f(n) < \sqrt{M} r_o^2 s_{\min}^2(n)$$

e portanto

$$\frac{\beta_1}{\sqrt{M} r_o^2} \leq \frac{\xi_0^f(n)}{\sqrt{M} r_o^2} < s_{\min}^2(n).$$

b) Limite superior para  $s_{\max}^2(n)$ :

Considerando que

$$\sum_{i=1}^M s_i^2(n) = \text{traço}(\Phi_M(n)) = \sum_{i=0}^{M-1} \xi_0^f(n-i)$$

e que  $\xi_0^f(n) \geq \lambda^i \xi_0^f(n-i)$ , resultam as seguintes desigualdades:

$$M s_{\min}^2(n) \leq \sum_{i=1}^M s_i^2(n) = \sum_{i=0}^{M-1} \xi_0^f(n-i) \leq \sum_{i=0}^{M-1} \frac{\xi_0^f(n)}{\lambda^{M-1}} = M \frac{\xi_0^f(n)}{\lambda^{M-1}}.$$

Desta forma

$$s_{\min}^2(n) \leq \frac{\xi_0^f(n)}{\lambda^{M-1}}. \quad (\text{D.11})$$

Utilizando (D.10) resulta então

$$s_{\max}^2(n) < r_o^2 \frac{\xi_0^f(n)}{\lambda^{M-1}}.$$

Considerando (D.8) obtém-se o resultado desejado

$$s_{\max}^2(n) < \frac{r_o^2 \beta_2}{\lambda^{M-1}}.$$

Com isto a persistência da excitação é garantida pois

$$0 < \frac{\beta_1}{\sqrt{M}r_o^2} \mathbf{I} < s_{\min}^2(n) \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq s_{\max}^2(n) \mathbf{I} < \frac{r_o^2 \beta_2}{\lambda^{M-1}} \mathbf{I} < \infty.$$

Finalmente, cabe notar que no caso de excitação persistente a potência  $\xi_0^f(n)$  do sinal necessariamente é limitada inferiormente e superiormente por duas constantes positivas. Esta é uma conclusão direta das expressões (D.11), (D.9), (D.4) e (D.5):

$$\lambda^{M-1} b_1 \leq \lambda^{M-1} s_{\min}^2(n) \leq \xi_0^f(n) \leq \sqrt{M} s_{\max}^2(n) \leq \sqrt{M} b_2.$$

Estes resultados podem ser resumidos no seguinte lema:

**Lema:** As seguintes afirmações são equivalentes

1) Para qualquer instante  $n$  a energia do sinal pré-janelado é uniformemente limitada superiormente e inferiormente, e o espalhamento dos valores singulares da matriz de dados  $\mathbf{A}_M(n)$  é uniformemente limitado superiormente, isto é, existem constantes positivas  $\beta_1$ ,  $\beta_2$  e  $r_o$  tais que

$$0 < \beta_1 \leq \xi_0^f(n) \leq \beta_2 < \infty$$

e

$$\chi(n) < r_o < \infty.$$

2) A excitação é persistente, isto é, existem constantes positivas  $b_1$  e  $b_2$  de tal forma que

$$0 < b_1 \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq b_2 \mathbf{I} < \infty.$$

Além disso, dadas as condições de 1, valem:

a)

$$b_1 = \frac{\beta_1}{r_o^2 \sqrt{M}} \quad \text{e} \quad b_2 = \frac{r_o^2 \beta_2}{\lambda^{M-1}}$$

b) e para qualquer instante  $n \geq M - 1 > 0$ ,

$$0 < \frac{\xi_0^f(n)}{\sqrt{M}r_o^2} \mathbf{I} \leq s_{\min}^2(n) \mathbf{I} \leq \mathbf{A}_M^T(n) \mathbf{A}_M(n) \leq s_{\max}^2(n) \mathbf{I} \leq \frac{r_o^2 \xi_0^f(n)}{\lambda^{M-1}} \mathbf{I} < \infty,$$

onde  $s_{\min}^2(n)$  e  $s_{\max}^2(n)$  são respectivamente o menor e o maior valor singular da matriz de dados  $\mathbf{A}_M(n)$ .

## Apêndice E

### Lista dos principais símbolos

$\mathbf{a}(n - i + 1) = [\lambda^{\frac{n}{2}} u(0 - i + 1), \lambda^{\frac{n-1}{2}} u(1 - i + 1), \dots, \lambda^{\frac{1}{2}} u(n - 1 - i + 1), u(n - i + 1)]^T$   
 - vetor de dados com  $n + 1$  linhas, sendo o elemento da linha  $\ell + 1$  igual a  $\lambda^{\frac{n-\ell}{2}} u(\ell - i + 1)$  para  $0 \leq \ell \leq n$  e  $1 \leq i \leq M$ ;

$\mathbf{A}_M(n) = [A(i, j)] = \lambda^{\frac{n-i+1}{2}} [u(i - j)]$  - matriz de dados pré-janelada de dimensão  $(n + 1) \times M$ ;

$b_{i-1}(\ell)$  - resposta de um filtro de erro da predição regressiva de ordem  $i - 1$  no instante  $\ell$  com coeficientes ótimos do instante  $\ell$ ;

$\mathbf{b}_i(\ell) = [b_0(\ell), b_1(\ell), \dots, b_{i-1}(\ell)]^T$  - vetor de erro de predição regressiva com  $i$  linhas;

$b'_{i-1}(\ell)$  - resposta de um filtro de erro da predição regressiva de ordem  $i - 1$  no instante  $\ell$  com coeficientes ótimos do instante  $n$ ;

$\mathbf{b}'_i(\ell) = [b'_0(\ell), b'_1(\ell), \dots, b'_{i-1}(\ell)]^T$  - vetor de erro de predição regressiva com  $i$  linhas;

$d(n)$  - resposta desejada no instante  $n$ ;

$\mathbf{d}(n) = [\lambda^{\frac{n}{2}} d(0), \lambda^{\frac{n-1}{2}} d(1), \dots, \lambda^{\frac{1}{2}} d(n - 1), d(n)]^T$  - vetor de resposta desejada com  $n + 1$  linhas, sendo o elemento da linha  $\ell + 1$  igual a  $\lambda^{\frac{n-\ell}{2}} d(\ell)$ , para  $0 \leq \ell \leq n$ ;

$\mathbf{D}_M(n) = \text{diag}\{\xi_0^b(n), \dots, \xi_{M-1}^b(n)\}$  - matriz diagonal de dimensão  $M \times M$ , com o  $i$ -ésimo elemento da diagonal definido pela energia da resposta de um filtro de erro da predição regressiva de ordem  $i - 1$  com  $1 \leq i \leq M$ ;

$e_{i-1}(\ell)$  - erro de estimação de ordem  $i - 1$  no instante  $\ell$  obtido com coeficientes ótimos do instante  $n$ ;

$\mathbf{e}_{i-1}(n) = [\lambda^{\frac{n}{2}} e_{i-1}(0), \lambda^{\frac{n-1}{2}} e_{i-1}(1), \dots, \lambda^{\frac{1}{2}} e_{i-1}(n-1), e_{i-1}(n)]^T$  - vetor com  $n+1$  elementos, com o elemento da linha  $\ell+1$  definido como  $\lambda^{\frac{n-\ell}{2}} e_{i-1}(\ell)$  para  $0 \leq \ell \leq n$ ;

$\mathbf{e}_{i-1}^b(n) = [\lambda^{\frac{n}{2}} b'_{i-1}(0), \lambda^{\frac{n-1}{2}} b'_{i-1}(1), \dots, \lambda^{\frac{1}{2}} b'_{i-1}(n-1), b'_{i-1}(n)]^T$  - vetor com  $n+1$  linhas, com o elemento da linha  $\ell+1$  definido como  $\lambda^{\frac{n-\ell}{2}} b'_{i-1}(\ell)$  para  $0 \leq \ell \leq n$ ;

$\mathbf{e}_{i-1}^f(n) = [\lambda^{\frac{n}{2}} f'_{i-1}(0), \lambda^{\frac{n-1}{2}} f'_{i-1}(1), \dots, \lambda^{\frac{1}{2}} f'_{i-1}(n-1), f'_{i-1}(n)]^T$  - vetor com  $n+1$  linhas, com o elemento da linha  $\ell+1$  definido como  $\lambda^{\frac{n-\ell}{2}} f'_{i-1}(\ell)$  para  $0 \leq \ell \leq n$ ;

$f_{i-1}(\ell)$  - resposta de um filtro de erro da predição progressiva de ordem  $i-1$  no instante  $\ell$  com coeficientes ótimos do instante  $\ell$ ;

$f'_{i-1}(\ell)$  - resposta de um filtro de erro da predição progressiva de ordem  $i-1$  no instante  $\ell$  com coeficientes ótimos do instante  $n$ ;

$\mathbf{g}_i(n)$  - vetor com  $i$  elementos definido como  $\Phi_i^{-1}(n) \mathbf{u}_i(n)$ ;

$\mathbf{K}_i(n)$  - matriz triangular superior de dimensão  $i \times i$  cujos elementos da diagonal principal são unitários.

$k_i^b(n)$  - coeficiente de reflexão da predição regressiva com  $1 \leq i \leq M$ ;

$\mathbf{k}_M^b(n) = [k_1^b(n), \dots, k_M^b(n)]$  - vetor com  $M$  linhas, com os elementos definidos pelos coeficientes de reflexão da predição regressiva  $k_i^b(n)$ ;

$k_i^d(n)$  - coeficiente de regressão com  $1 \leq i \leq M$ ;

$\mathbf{k}_M^d(n) = [k_1^d(n), \dots, k_M^d(n)]$  - vetor com  $M$  linhas, com os elementos definidos pelos coeficientes regressão  $k_i^d(n)$ ;

$k_i^f(n)$  - coeficiente de reflexão da predição progressiva com  $1 \leq i \leq M$ ;

$\mathbf{k}_M^f(n) = [k_1^f(n), \dots, k_M^f(n)]$  - vetor com  $M$  linhas, com os elementos definidos pelos coeficientes de reflexão da predição progressiva  $k_i^f(n)$ ;

$\mathbf{P}_M(n) = \mathbf{A}_M(n) (\mathbf{A}_M^T(n) \mathbf{A}_M(n))^{-1} \mathbf{A}_M^T(n)$  - matriz de dimensão  $(n+1) \times (n+1)$  que representa o operador de projeção no espaço definido pelas colunas da matriz  $\mathbf{A}_M(n)$ ;



$Q_i(n)$  - matriz unitária de dimensão  $(n + 1) \times (n + 1)$  que triangulariza uma matriz de dados com  $i$  colunas e  $n + 1$  linhas;

$R_i(n)$  - matriz triangular superior de dimensão  $i \times i$ ;

$S_i^c(n)$  - matriz de dimensão  $(i + 1) \times (i + 1)$ , que descreve as operações de adaptação e filtragem do algoritmo RLS convencional;

$S_i^g(n)$  - matriz de dimensão  $(i + 1) \times (i + 1)$ , que descreve as operações de adaptação e filtragem dos algoritmos QR-RLS que usam rotações de Givens;

$S_i^k(n)$  - matriz de dimensão  $(i + 1) \times (i + 1)$ , que descreve as operações de adaptação e filtragem dos algoritmos que utilizam o procedimento de ortogonalização de Gram-Schmidt;

$u(n)$  - dado de entrada no instante  $n$ ;

$\mathbf{u}_i(n-t) = [u(n-t), u(n-t-1), \dots, u(n-t-i+1)]^T$  - vetor de dados com  $i$  linhas e  $0 \leq t \leq \ell$ ;

$w_{i-1}(n)$  - coeficiente do filtro transversal de um problema de estimação de ordem  $M$  no tempo  $n$  com  $1 \leq i \leq M$ ;

$\mathbf{w}_M(n) = [w_0(n), \dots, w_{M-1}(n)]^T$  - vetor com  $M$  linhas, com elementos definidos pelos coeficientes do filtro transversal de estimação no tempo  $n$ ;

$w_{i-1}^b(n)$  - coeficiente do filtro transversal de um problema de predição regressiva de ordem  $M$  no tempo  $n$  com  $1 \leq i \leq M$ ;

$\mathbf{w}_M^b(n) = [w_0^b(n), \dots, w_{M-1}^b(n)]^T$  - vetor com  $M$  linhas, com elementos definidos pelos coeficientes do filtro transversal de predição regressiva no tempo  $n$ ;

$w_{i-1}^f(n)$  - coeficiente do filtro transversal de um problema de predição progressiva de ordem  $M$  no tempo  $n$  com  $1 \leq i \leq M$ ;

$\mathbf{w}_M^f(n) = [w_0^f(n), \dots, w_{M-1}^f(n)]^T$  - vetor com  $M$  linhas, com elementos definidos pelos coeficientes do filtro transversal de predição progressiva no tempo  $n$ ;

$\alpha_{i-1}(n)$  - erro de estimação *a priori* de ordem  $i - 1$  no instante  $n$ ;

$\gamma_{i-1}(n)$  - fator de conversão;

$\epsilon_{i-1}(n)$  - erro de estimação normalizado em ângulo de ordem  $i - 1$  no instante  $n$ ;

$\epsilon_{i-1}^b(n)$  - erro da predição regressiva normalizado em ângulo de ordem  $i - 1$  no instante  $n$ ;

$\epsilon_{i-1}^f(n)$  - erro da predição progressiva normalizado em ângulo de ordem  $i - 1$  no instante  $n$ ;

$\eta_{i-1}(n)$  - erro de predição progressiva de ordem  $i - 1$  no instante  $n$ ;

$\theta_i^b(n)$  - ângulos que definem a estrutura de McWhirter nos casos de estimação e predição regressiva de ordem  $M$  com  $1 \leq i \leq M$ ;

$\theta_i^f(n)$  - ângulos que definem a estrutura de McWhirter no caso de predição progressiva de ordem  $M$  com  $1 \leq i \leq M$ ;

$\lambda$  - fator de esquecimento;

$\Lambda(n) = \text{diag}(\lambda^n, \lambda^{n-1}, \dots, \lambda^n, 1)$  - matriz diagonal de dimensão  $(n + 1) \times (n + 1)$ , com o elemento da linha  $\ell + 1$  igual a  $\lambda^{n-\ell}$ , para  $0 \leq \ell \leq n$ ;

$\xi_{i-1}(n) = \mathbf{e}_{i-1}^T(n) \mathbf{e}_{i-1}(n)$  - energia do erro de estimação;

$\xi_{i-1}^b(n) = \mathbf{e}_{i-1}^{bT}(n) \mathbf{e}_{i-1}^b(n)$  - energia do erro da predição regressiva;

$\xi_{i-1}^f(n) = \mathbf{e}_{i-1}^{fT}(n) \mathbf{e}_{i-1}^f(n)$  - energia do erro de predição progressiva;

$\phi_i(n)$  - ângulos que definem os coeficientes de reflexão de uma estrutura em treliça normalizada para um problema de predição de ordem  $M$  com  $1 \leq i \leq M$ ;

$\Phi_M(n) = \mathbf{A}_M^T(n) \mathbf{A}_M(n)$  - matriz de autocorrelação;

$\psi_{i-1}(n)$  - erro de predição regressiva de ordem  $i - 1$  no instante  $n$ ;

$\Omega_M(n) = \mathbf{A}_M^T(n) \mathbf{d}_M(n)$  - vetor de correlação cruzada;