INFINITE PRECISION ANALYSIS OF THE FAST QR ALGORITHMS BASED ON BACKWARD PREDICTION ERRORS

José A. Apolinário Jr.*, **César A. Medina S.****, **and Paulo S. R. Diniz**

Abstract - The conventional QR Decomposition Recursive Least Squares (QRD-RLS) method requires the order of N^2 multiplications—O[N^2]—per output sample. Nevertheless, a number of Fast QRD-RLS algorithms have been proposed with O[N] of complexity. Particularly the Fast QRD-RLS algorithms based on backward prediction errors are well known for their good numerical behaviors and low complexities. In such a scenario, considering a case where fixed-point arithmetic is employed, an infinite precision analysis offering the mean square values of the internal variables becomes very attractive for a practical implementation. In addition to this, a finite-precision analysis requires the estimates of these mean square values. In this work, we first present an overview of the main Fast QRD-RLS algorithms, followed by an infinite precision analysis concerning the steady state mean square values of the internal variables of four FQR-RLS algorithms. We stress the fact that the goal of this paper is the presentation of the infinite precision analysis results, the expressions for the mean square values of the internal variables, for all FQR algorithms based on backward prediction errors. The validity of these analytical expressions is verified through computer simulations, carried out in a system identification setup. In the appendixes, the pseudo-code detailed implementations of each algorithm are listed.

Keywords: Adaptive systems, Fast Algorithms, QR decomposition.

Resumo - O método convencional dos Mínimos Quadrados Recursivos usando Decomposição QR requer da ordem de N^2 multiplicações—O[N^2]—por amostra de saída. Contudo, vários algoritmos QRD-RLS tem sido propostos com O[N] de complexidade. Particularmente os chamados algoritmos QRD Rápidos baseados em erros de predição retrógrada são bem conhecidos por seus bons comportamentos numéricos e por suas baixas complexidades. Em tal cenário, considerando o caso onde a aritmética em ponto fixo é empregada, uma análise em precisão infinita oferecendo os valores médios quadráticos das variáveis internas torna-se bastante atraente para uma implementação prática. Além disto, uma análise em precisão finita requer as estimativas destes valores médios quadráticos. Neste trabalho, apresentamos inicialmente uma recapitulação dos principais algoritmos QR Rápidos, seguida por uma análise em precisão infinita relativa aos valores médios quadráticos em regime estacionário das variáveis internas de quatro destes algoritmos. Ressaltamos que o objetivo deste artigo é apresentar os resultados da análise em precisão infinita, as expressões para os valores médios quadráticos das variáveis internas, para todos os algoritmos FQR baseados em erros de predição retrógrada. A validação destas expressões analíticas é obtida por meio de simulações em computador conduzidas num ambiente de identificação de sistema. Nos apêndices, as implementações detalhadas em pseudo-código de cada algoritmo são listadas.

Palavras-chave: Sistemas Adaptativos, Algorítmos Rápidos, Decomposição QR.

1. INTRODUCTION

Since the first QR Decomposition (QRD) based Fast RLS algorithm introduced by John Cioffi in 1990 [1], many other Fast QRD-based RLS algorithms were developed [2, 3, 4, 5, 6]. It can be seen on [5] that Fast QRD-RLS algorithms can be classified in terms of the type of triangularization applied to the input data matrix (upper or lower triangular) and the type of error vector (*a posteriori* or *a priori*) involved in the updating process. It can be seen from the Gram-Schmidt orthogonalization procedure that an upper triangularization (notation being the same as in [5]) involves the updating of forward prediction errors while a lower triangularization involves the updating of backward prediction errors. Table 1 [1] presents this classification as well as points out how these algorithms will be designated hereafter. Also note that only for the algorithms [2] and [3], a formal demonstration of the numerical stability is known; these algorithms are backward stable and minimal in the sense of system theory [2, 7].

Table 1. Classification of the Fast QR algorithms.

Error Type	Prediction	
	Forward	**Backward**
A Posteriori	FQR_POS_F [1]	**FQR_POS_B** [2, 6]
A Priori	FQR_PRI_F [5]	**FQR_PRI_B** [3, 4]

This work is focused in the study of the steady state mean

[1]Note that only those based on Backward prediction errors (right sided column of Table 1) will be addressed hereafter.

J. A. Apolinário Jr. and C. A. Medina S. are with the Departamento de Engenharia Elétrica, Instituto Militar de Engenharia, Rio de Janeiro, Brazil, (e-mails:apolin@ieee.org and csmedina@epq.ime.eb.br)

P. S. R. Diniz is with the Programa de Engenharia Elétrica, COPPE/Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil, (e-mail: diniz@lps.ufrj.br)

* The author thanks FAPERJ for partial support of this work under grant number E-26/171.307/2001

** This work was partially supported by CAPES.

squared value of the internal variables of this class of Fast QRD-based algorithms which are well known for their good numerical behavior and low computational complexity. Since these algorithms present similar performances in finite precision, specially when using a reasonably large number of bits of wordlength, they are all currently subject of research. Particularly in the case of fixed-point arithmetic implementations, information about the range of their internal variables–such as those offered by an infinite precision analysis–is very interesting for a practical implementation.

It is also worth mentioning that finite-precision analysis requires the estimates of the mean square values found in this work, some of them obtained here and others collected from the technical literature. The relevance of the infinite precision analysis can be clearly observed in [9], where the section "*Quantization Error and Stability Analysis,*" addressing the finite precision analysis of the conventional QRD–RLS algorithm, was only possible with the results of the infinite precision analysis carried out in the previous section.

Since in an infinite precision environment many variables are identical for all Fast QR algorithms based on Backward Prediction Errors mentioned in Table 1, the use of results from other works was possible. We have used theoretical expressions for the mean square values of different variables from the analysis of the conventional QR–RLS algorithm performed by Diniz and Siqueira in 1995 [9]. We have also used results for variables of the *a Posteriori* Fast QR algorithm based on Backward Prediction Errors [2] in paper by Siqueira, Diniz, and Alwan [10] published in 1994. Finally, we have used some expressions derived in a work carried by Miranda, Aguayo, and Gerken in 1997 [11] concerning the variables of the Fast QR algorithm based on *a Priori* Backward Prediction Errors [3].

The main contributions of this work, besides the new theoretical expressions developed, are concerned to the unified framework in which all FQR algorithms based on Backward Prediction Errors were addressed and all their infinite precision analysis were presented using the same notation.

This paper is organized as follows. In Section 2, we present an overview of the Fast QR algorithms based on backward prediction errors. Then, in Sections 3 and 4, the infinite precision analysis concerning the steady state mean square values of each internal variable is presented. In Section 5, the validation of the analytical results obtained is carried out through computer simulations. Finally, some conclusions are summarized and the detailed algorithmic implementations are presented in the appendixes.

2. THE FQR ALGORITHMS BASED ON BACKWARD PREDICTION ERRORS

The RLS algorithms minimize the following cost function

$$\xi(k) = \sum_{i=0}^{k} \lambda^{k-i} e^2(i) = \boldsymbol{e}^T(k)\boldsymbol{e}(k) = \| \boldsymbol{e}(k) \|^2 \qquad (1)$$

where each component of the error vector[2] $\boldsymbol{e}(k)$ is the *a posteriori* error at instant i weighted by $\lambda^{(k-i)/2}$ (λ is the forgetting factor). The vector $\boldsymbol{e}(k)$ is given by

$$\boldsymbol{e}(k) = \boldsymbol{d}(k) - \boldsymbol{X}(k)\boldsymbol{w}(k) \qquad (2)$$

In the above equation, the weighted desired or reference signal vector $\boldsymbol{d}(k)$, the coefficient vector $\boldsymbol{w}(k)$, and the input data matrix $\boldsymbol{X}(k)$ are defined as follows.

$$\boldsymbol{d}(k) = \begin{bmatrix} d(k) \\ \lambda^{1/2} d(k-1) \\ \vdots \\ \lambda^{k/2} d(0) \end{bmatrix} \qquad (3)$$

$$\boldsymbol{w}(k) = \begin{bmatrix} w_0(k) \\ w_1(k) \\ \vdots \\ w_N(k) \end{bmatrix} \qquad (4)$$

$$\boldsymbol{X}(k) = \begin{bmatrix} \boldsymbol{x}^T(k) \\ \lambda^{1/2} \boldsymbol{x}^T(k-1) \\ \vdots \\ \lambda^{k/2} \boldsymbol{x}^T(0) \end{bmatrix} \qquad (5)$$

where N is the filter order (number of coefficients minus one), $\boldsymbol{x}(k)$ is the input signal vector $[x(k)\, x(k-1) \cdots x(k-N)]^T$ —samples before instant 0 are considered equal to zero—, and $\boldsymbol{w}(k)$ is the coefficient vector. The premultiplication of the above equation by the orthonormal matrix $\boldsymbol{Q}(k)$ triangularizes $\boldsymbol{X}(k)$ without affecting the cost function.

$$\begin{aligned} \boldsymbol{e}_q(k) = \boldsymbol{Q}(k)\boldsymbol{e}(k) &= \begin{bmatrix} \boldsymbol{e}_{q_1}(k) \\ \boldsymbol{e}_{q_2}(k) \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{d}_{q_1}(k) \\ \boldsymbol{d}_{q_2}(k) \end{bmatrix} - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{U}(k) \end{bmatrix} \boldsymbol{w}(k) \end{aligned} \qquad (6)$$

The weighted-square error in (1) is minimized by choosing $\boldsymbol{w}(k)$ such that the term $\boldsymbol{d}_{q2}(k) - \boldsymbol{U}(k)\boldsymbol{w}(k)$ is zero. Equation (6) can be written in a recursive form while avoiding the ever increasing order for the vectors and matrices involved [8]:

$$\begin{bmatrix} e_{q_1}(k) \\ \boldsymbol{d}_{q_2}(k) \end{bmatrix} = \boldsymbol{Q}_\theta(k) \begin{bmatrix} d(k) \\ \lambda^{1/2} \boldsymbol{d}_{q_2}(k-1) \end{bmatrix} \qquad (7)$$

where e_{q_1} is the first element of $\boldsymbol{e}_{q_1}$ and $\boldsymbol{Q}_\theta(k) = \prod_{i=N}^{0} \boldsymbol{Q}_{\theta_i}(k)$ is a sequence of Givens rotations that annihilates the elements of the input vector $\boldsymbol{x}(k)$ in the equation

$$\begin{bmatrix} \boldsymbol{0}^T \\ \boldsymbol{U}(k) \end{bmatrix} = \boldsymbol{Q}_\theta(k) \begin{bmatrix} \boldsymbol{x}^T(k) \\ \lambda^{1/2} \boldsymbol{U}(k-1) \end{bmatrix} \qquad (8)$$

and,

$$\boldsymbol{Q}_{\theta_i}(k) = \begin{bmatrix} cos\theta_i(k) & \boldsymbol{0}^T & -sin\theta_i(k) & \boldsymbol{0}^T \\ \boldsymbol{0} & \boldsymbol{I}_{N-i} & \boldsymbol{0} & \boldsymbol{0} \cdots \boldsymbol{0} \\ sin\theta_i(k) & \boldsymbol{0}^T & cos\theta_i(k) & \boldsymbol{0}^T \\ \boldsymbol{0} & \boldsymbol{0} \cdots \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I}_i \end{bmatrix} \qquad (9)$$

[2] Note that scalars are represented by italic letters while vectors are written in bold face italic.

The following relation which is also used in the conventional QR algorithm is obtained by postmultiplying $\boldsymbol{e}_q^T(k)\boldsymbol{Q}(k)$ by the pinning vector $[1\ 0\ \cdots\ 0]^T$.

$$e(k) = e_{q_1}(k)\prod_{i=0}^{N}\cos\theta_i(k) = e_{q_1}(k)\gamma(k) \quad (10)$$

where $\gamma(k)$ is the first element of the first row of $\boldsymbol{Q}_\theta(k)$.

Matrix $\boldsymbol{Q}_\theta(k)$ in (7) can be partitioned as

$$\boldsymbol{Q}_\theta(k) = \begin{bmatrix} \gamma(k) & -\gamma(k)\boldsymbol{a}^T(k) \\ \boldsymbol{f}(k) & \boldsymbol{E}(k) \end{bmatrix} \quad (11)$$

where, using (11) in (8) and recalling that $\boldsymbol{Q}_\theta(k)$ is orthonormal, it is possible to prove that, for the case of lower triangularization of $\mathbf{U}(k)$ (backward prediction errors update), $\boldsymbol{f}(k) = [\boldsymbol{U}(k)]^{-T}\boldsymbol{x}(k)$ is the normalized *a posteriori* backward prediction error vector [3], $\boldsymbol{a}(k) = \boldsymbol{U}^{-T}(k-1)\boldsymbol{X}(k)/\sqrt{\lambda}$ is the normalized *a priori* backward prediction error vector [3], and $\boldsymbol{E}(k) = \lambda^{1/2}[\boldsymbol{U}(k)]^{-T}[\boldsymbol{U}(k-1)]^T$.

The update of the *a posteriori* and the *a priori* backward prediction error vectors, $\mathbf{f}(k)$ and $\mathbf{a}(k)$ respectively, leads to two different algorithms, the so-called FQR_POS_B and FQR_PRI_B algorithms. The update equations of these vectors are given by

$$\begin{bmatrix} \frac{e_b(k+1)}{\|\boldsymbol{e}_b(k+1)\|} \\ \boldsymbol{f}(k+1) \end{bmatrix} = \boldsymbol{Q}'_{\theta_f}(k+1)\begin{bmatrix} \boldsymbol{f}(k) \\ \frac{e_f(k+1)}{\|\boldsymbol{e}_f(k+1)\|} \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} \frac{e'_b(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_b(k)\|} \\ \boldsymbol{a}(k+1) \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k)\begin{bmatrix} \boldsymbol{a}(k) \\ \frac{e'_f(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_f(k)\|} \end{bmatrix} \quad (13)$$

where $\boldsymbol{Q}'_{\theta f}(k)$ is a sequence of Givens rotations that generates $\|\ \boldsymbol{e}_f(k)\ \|$ and can be obtained through the following equation.

$$\begin{bmatrix} \mathbf{0} \\ \|\ \boldsymbol{e}_f^{(0)}(k+1)\ \| \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k+1)\begin{bmatrix} \boldsymbol{d}_{fq_2}(k+1) \\ \|\ \boldsymbol{e}_f(k+1)\ \| \end{bmatrix} \quad (14)$$

3. MEAN SQUARE VALUES OF COMMON VARIABLES (FQR_POS_B AND FQR_PRI_B ALGORITHMS)

The matrix equations of the two implementations of Fast QR algorithms mentioned before are listed in Tables 4 and 5. As can be seen from these tables, several equations are exactly the same. In this section, we summarize the mean square values of all variables found in both algorithms.

Mean Square Values of $cos\theta_i(k)$ and $sin\theta_i(k)$

The following results can be found in [9].

$$E[cos^2\theta_i(k)] \approx \lambda \quad (15)$$

$$E[sin^2\theta_i(k)] \approx 1-\lambda \quad (16)$$

Mean Square Value of $e_{fq_1}^{(i)}(k)$

The following result was first derived in [10].

$$E\left\{[e_{fq_1}^{(i)}(k)]^2\right\} \approx \sigma_x^2\left(\frac{2\lambda}{1+\lambda}\right)^i \quad (17)$$

Mean Square Value of $d_{fq2_i}(k)$

The following result was obtained from [10].

$$E\left\{[d_{fq2_i}(k)]^2\right\} \approx \frac{\sigma_x^2}{1+\lambda}\left(\frac{2\lambda}{1+\lambda}\right)^{N+1-i} \quad (18)$$

Mean Square Value of $\|\ \boldsymbol{e}_f^{(i)}(k)\ \|$

The following result can be found in [10].

$$E[\|\ \boldsymbol{e}_f^{(i)}(k)\ \|^2] \approx \frac{\sigma_x^2}{1-\lambda}\left(\frac{2\lambda}{1+\lambda}\right)^i \quad (19)$$

Mean Square Values of $cos\theta'_{f_i}(k)$ and $sin\theta'_{f_i}(k)$

The following results were also derived in [10].

$$E[cos^2\theta'_{f_i}(k)] \approx \frac{2\lambda}{1+\lambda} \quad (20)$$

$$E[sin^2\theta'_{f_i}(k)] \approx \frac{1-\lambda}{1+\lambda} \quad (21)$$

Mean Square Value of $\gamma^{(i)}(k)$

It is known from the technical literature that $\gamma(k) = \prod_{i=0}^{N} cos\theta_i(k)$. If we use (15) and (16), and assume independence between $cos\theta_i(k)$ and $cos\theta_j(k)$, $i \neq j$, it is easy to find the following expression.

$$E\left\{[\gamma^{(i)}(k)]^2\right\} \approx \lambda^i \quad (22)$$

The same expression was also obtained in [11] using a different approach.

Mean Square Value of $d_{q2_i}(k)$

The following result was first introduced in [9].

$$E[d^2_{q2_{N+1-i}}(k)] \approx \left[\frac{2\lambda}{1+\lambda}\right]^i \left[\frac{\sigma_x^2}{1-\lambda}w_{0,i}^2 + \frac{\sigma_x^2}{1+\lambda}\sum_{j=i+1}^{N} w_{0,j}^2\right] \quad (23)$$

where $w_{0,i}^2 = E[w_i^2(k)]$. Observe that although $w_{0,i}$ is not available, a rough estimate of $\sigma_x^2 w_{0,i}^2$ can be obtained based on the power of the reference signal [9].

Mean Square Value of $e_{q_1}^{(i)}(k)$

From the joint process estimation part of the FQR_POS_B algorithm, we take the expressions of $e_{q_1}^{(i)}(k+1)$ and $d_{q2_{N+2-i}}(k+1)$, and use them to derive the expected value of $[e_{q_1}^{(i)}(k+1)]^2 + [d_{q2_{N+2-i}}(k+1)]^2$. By assuming stationarity, we find the following relation.

$$\boxed{\begin{aligned} E\left\{[e_{q_1}^{(i)}(k)]^2\right\} = \ & E\left\{[e_{q_1}^{(i-1)}(k)]^2\right\} \\ & -(1-\lambda)E[d^2_{q2_{N+2-i}}(k)] \end{aligned}} \quad (24)$$

where $E\left\{[e_{q_1}^{(0)}(k)]^2\right\} = \sigma_d^2 = \sigma_x^2\sum_{i=0}^{N} w_{o,i}^2 + \sigma_n^2$ is the variance of the reference signal and σ_n^2 is the variance of the measurement noise (it is assumed here that the algorithm is applied in a sufficient-order identification problem, i.e., the unknown FIR system has the same order as the adaptive filter).

Finally, from the last equation of the algorithm and assuming that $e_{q_1}(k)$ and $\gamma(k)$ are uncorrelated, we have $E[e^2(k)] \approx \lambda^{N+1} E[e^2_{q_1}(k)]$. Since from (24) and (23), assuming $\lambda \approx 1$, we have that $E[e^2_{q_1}(k)] \approx \sigma_n^2$, the following expression results.

$$\boxed{E[e^2(k)] \approx \lambda^{N+1}\sigma_n^2} \tag{25}$$

4. MEAN SQUARE VALUES OF SPECIFIC VARIABLES OF EACH ALGORITHM

4.1 MEAN SQUARE VALUES OF INTERNAL VARIABLES OF THE FQR_POS_B ALGORITHM

For this algorithm, from the derivation of (12), it can be observed that the last element of $\boldsymbol{f}(k+1)$, given by $\frac{x(k+1)}{\|\boldsymbol{e}_f^{(0)}(k+1)\|}$, was precalculated in a previous step. This fact leads to two slightly different versions of the same algorithm. The first one is based on this prior knowledge of the last element of $\boldsymbol{f}(k+1)$ while the second is based on the straightforward computation of $\boldsymbol{f}(k+1)$ and requires the calculation of $\frac{e_f(k+1)}{\|\boldsymbol{e}_f(k+1)\|}$.

The first version of this algorithm was introduced in [6] and its detailed description is presented in Appendix A. The second version of this algorithm was introduced in [2] and its detailed description is given is Appendix B.

For the infinite precision results of the FQR_POS_B Algorithm, all variables have the same notation used in its detailed description.

Mean Square Value of $f_i(k)$

From the implementation of the step "Obtaining $\boldsymbol{Q}_\theta(k+1)$" (see Table 4 and Appendix A or B) we obtain the following expression.

$$f_{N+2-i}(k+1) = \gamma^{(i-1)}(k+1) sin\theta_{i-1}(k+1) \tag{26}$$

By taking the expected value of (26) squared and using the approximations (16) and (22), we obtain the following expression.

$$\boxed{E[f_i^2(k)] \approx \lambda^{N+1-i}(1-\lambda)} \tag{27}$$

Mean Square Value of aux_i

The implementation of the step "Obtaining $f(k+1)$" can be carried out in two different ways, as mentioned in the beginning of this section. These implementations can be found in Appendixes A and B, respectively.

In the **first version** of this algorithm, since we take the expressions for $f_{N+2-i}(k+1)$ and aux_i and use them to calculate $E[f^2_{N+2-i}(k)] + E[aux_i^2]$, it is straightforward to realize that $E[aux_i^2] + E[f^2_{N+2-i}(k+1)] = E[aux^2_{i-1}] + E[f^2_{N+i-1}(k)]$. Since $f_{N+1}(k+1) = aux_0$, it is easy to figure out that $E[aux_i^2] = E[f^2_{N+1-i}(k)]$; then, the following expression results.

$$\boxed{E[aux_i^2] \approx \lambda^i(1-\lambda)} \tag{28}$$

For the **second version** of this algorithm, since we use the expressions for $f_{i-1}(k+1)$ and aux_i to calculate $E[f^2_{i-1}(k+1)] + E[aux_i^2]$, it is straightforward to show that $E[aux^2_{i-1}] - E[f^2_{i-1}(k+1)] = E[aux_i^2] - E[f_i^2(k)]$. Since $f_{N+1}(k+1) = aux_{N+1}$, it follows that $E[aux_i^2] = E[f_i^2(k)]$; therefore:

$$\boxed{E[aux_i^2] \approx \lambda^{N+1-i}(1-\lambda)}. \tag{29}$$

4.2 MEAN SQUARE VALUES OF INTERNAL VARIABLES OF THE FQR_PRI_B ALGORITHM

For the FQR_PRI_B algorithm, it is observed from the derivation of (13) that the last element of $\boldsymbol{a}(k+1)$ had been previously calculated. This observation led to two slightly different versions of the same algorithm. The first version of this algorithm is based on the prior knowledge of the last element of $\boldsymbol{a}(k+1)$ (or $a_{N+1}(k+1) = \frac{x(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_f^{(0)}(k)\|}$) and was first presented in [4]. The second version of the FQR_PRI_B algorithm is based on the straightforward computation of $\boldsymbol{a}(k+1)$ according to (13) and requires the calculation of $\frac{e'_f(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_f(k)\|}$.

The first version of the FQR_PRI_B algorithm was introduced in [4] and its detailed description is presented in Appendix C. The second version of this algorithm was introduced in [3] and its detailed description is given is Appendix D.

For the infinite precision results of the FQR_PRI_B Algorithm, all variables have the same notation used in its detailed description.

Mean Square Value of $a_i(k)$

From the implementation of the step "Obtaining $\boldsymbol{Q}_\theta(k+1)$" (see Table 5 and appendix C or D), we obtain the following expression.

$$a^2_{N+2-i}(k+1) = [\gamma^{(i)}(k+1)]^{-2} - [\gamma^{(i-1)}(k+1)]^{-2} \tag{30}$$

By taking the expected value of (30), using the approximation of (22), and employing the averaging principle [12, 13], it is possible to obtain

$$E[a_i^2(k)] \approx \lambda^{-(N+2-i)}(1-\lambda). \tag{31}$$

This expression is also available in [11].

Mean Square Value of aux_i

The implementation of the step "Obtaining $a(k+1)$" can be carried out in two distinct ways, as also discussed in the beginning of this section. These implementations are detailed in Appendixes C and D, respectively.

In the **first version** of this algorithm we use the expressions for $a_{N+2-i}(k+1)$ and aux_i to calculate $E[a^2_{N+2-i}(k+1)] + E[aux_i^2]$. It is then straightforward to infer that $E[aux_i^2]+E[a^2_{N+2-i}(k+1)] = E[aux^2_{i-1}]+E[a^2_{N+1-i}(k)]$. Since $a_{N+1}(k+1) = aux_0$, it is possible to conclude that $E[aux_i^2] = E[a^2_{N+1-i}(k)]$ leading—see (31)—to the following result.

$$\boxed{E[aux_i^2] \approx \lambda^{-(i+1)}(1-\lambda)} \qquad (32)$$

For the **second version** of this algorithm we use the expressions for $a_{i-1}(k+1)$ and aux_i to calculate $E[a^2_{i-1}(k+1)] + E[aux_i^2]$. This leads to $E[aux^2_{i-1}] + E[a_i^2(k)] = E[aux_i^2] + E[a^2_{i-1}(k+1)]$. Since $a_{N+1}(k+1) = aux_{N+1}$, it is easy to conclude that $E[aux_i^2] = E[a_i^2(k)]$; as a consequence, from (31), the following expression results.

$$\boxed{E[aux_i^2] \approx \lambda^{-(N+2-i)}(1-\lambda)} \qquad (33)$$

5. SIMULATION RESULTS

In this section we consider a system identification example where the input signal is a zero-mean random Gaussian process with variance $\sigma_x^2 = 10^{-3}$, the measurement noise is Gaussian with variance $\sigma_n^2 = 10^{-7}$, the desired signal is obtained through a fourth-order filter. In an ensemble of 1000 runs, each with 5000 samples, only the 4000 last output samples were used to calculate the mean square value. The chosen λ was 0.95.

The four algorithms were used in the simulation in order to compare the simulated with the theoretical results. From these results, Table 2 shows the total errors between the theoretical and simulated values for the non-common variables. This error was computed, for each algorithm, as the sum of the absolute value of the difference between the simulated values (in dB) and the theoretical values (in dB). As can be seen from this table, the lowest error corresponds to the FQR_POS_B Version 1 algorithm. This only means that we can predict the mean squared values slightly better for this algorithm than for the others. All detailed results are shown in Table 3. As can be observed from this table, the predicted mean square values for all internal variables are very close to their measured values.

Table 2. Total error for non-common variables.

Algorithm	Error
FQR_PRI_B Version 1	2.3415281914
FQR_PRI_B Version 2	2.5444828614
FQR_POS_B Version 1	1.5464361322
FQR_POS_B Version 2	1.6079328534

Table 3. Mean Square Values of Internal Variables.
(* represents common variables)

	Simulation	Theoretical
	$e^{(i)}_{fq_1}(k)$	$^*\ e^{(i)}_{fq_1}(k)$
i=0	$0.1001463\ 10^{-2}$	$0.1000000\ 10^{-2}$
i=1	$0.0977555\ 10^{-2}$	$0.0974359\ 10^{-2}$
i=2	$0.0954073\ 10^{-2}$	$0.0949375\ 10^{-2}$
i=3	$0.0931882\ 10^{-2}$	$0.0925032\ 10^{-2}$
i=4	$0.0909946\ 10^{-2}$	$0.0901314\ 10^{-2}$
i=5	$0.0888874\ 10^{-2}$	$0.0878203\ 10^{-2}$
	$d_{fq2_i}(k)$	$^*\ d_{fq2_i}(k)$
i=1	$0.421465504\ 10^{-3}$	$0.462212139\ 10^{-3}$
i=2	$0.438696191\ 10^{-3}$	$0.474375616\ 10^{-3}$
i=3	$0.443879956\ 10^{-3}$	$0.486859185\ 10^{-3}$
i=4	$0.469505010\ 10^{-3}$	$0.499671269\ 10^{-3}$
i=5	$0.477679154\ 10^{-3}$	$0.512820513\ 10^{-3}$
	$\Vert \boldsymbol{e}^{(i)}_f(k) \Vert$	$^*\ \Vert \boldsymbol{e}^{(i)}_f(k) \Vert$
i=0	0.020026731	0.02
i=1	0.019549052	0.019487179
i=2	0.019079547	0.018987508
i=3	0.018635667	0.018500649
i=4	0.018196971	0.018026273
i=5	0.017775505	0.017564061
	$\gamma^{(i)}(k)$	$^*\ \gamma^{(i)}(k)$
i=0	1.001001001	1
i=1	0.953151588	0.95
i=2	0.907438523	0.9025
i=3	0.863791227	0.857375
i=4	0.822138408	0.81450625
i=5	0.782402799	0.773780938
	$aux_i PRI\_BV.1$	$aux_i PRI\_BV.1$ **eq.(32)**
i=0	0.055449104	0.052631579
i=1	0.058259792	0.055401662
i=2	0.061240975	0.058317539
i=3	0.064403535	0.061386883
i=4	0.067740128	0.064617772

Continues on pp. 8

6. CONCLUSIONS

In this paper, four versions of Fast QR Decomposition algorithms based on backward prediction errors have been analyzed in infinite precision environment. These algorithms are generally good choices among the Fast RLS algorithms due to their low computational complexity and proved stability when implemented with finite precision arithmetic.

Closed-form formulae for the estimation of the mean square values of the internal variables were obtained and theoretical results were compared with computer simulations, confirming the accuracy of the analysis.

These expressions are keys for a proper implementation of these algorithms using fixed-point arithmetic processors, since the number of bits for each internal variable could be determined by its estimated mean squared value. In addition, they are required in the finite-precision analysis of the FQR_PRI_B and FQR_POS_B algorithms which, so far, is not available in the literature.

REFERENCES

[1] J. M. Cioffi, "The Fast Adaptive ROTOR's RLS Algorithm," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. ASSP-38, pp. 631–653, Apr. 1990.

[2] P. A. Regalia and M. G. Bellanger, "On the Duality between Fast QR Methods and Lattice Methods in Least Squares Adaptive Filtering," *IEEE Transactions on Signal Processing*, vol. SP-39, pp. 879–891, Apr. 1991.

[3] M. D. Miranda and M. Gerken, "A Hybrid QR-Lattice

Least Squares Algorithm Using *a Priori* Errors," in *Proceedings of the 38 Midwest Symposium on Circuits and Systems*, pp. 983–986, Rio de Janeiro, RJ, Brazil, Aug. 1995.

[4] A. A. Rontogiannis and S. Theodoridis, "New Fast Inverse QR Least Squares Adaptive Algorithms," *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing*, pp. 1412–1415, Detroit, MI, USA, 1995.

[5] J. A. Apolinário Jr. and P. S. R. Diniz, "A New Fast QR Algorithm Based on *a Priori* Errors," *IEEE Signal Processing Letters*, vol. 4, pp. 307–309, Nov. 1997.

[6] J. A. Apolinário Jr., M. G. Siqueira, and P. S. R. Diniz, "On Fast QR Algorithms Based on Backward Prediction errors: New Results and Comparisons," *Proceedings of the First Balkan Conference on Signal Processing, Communications, Circuits, and Systems*, Istanbul, Turkey, June 2000.

[7] M. D. Miranda and M. Gerken, "A Hybrid Least Squares QR-Lattice Algorithm Using *a Priori* Errors," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2900–2911, Dec. 1997.

[8] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Norwell, MA, USA, Kluwer Academic Press, 1997.

[9] P. S. R. Diniz and M. G. Siqueira, "Fixed-point error analysis of the QR-recursive least square algorithm," *IEEE Transactions on Circuits and Systems — Part II*, v. 42, pp. 334–348, May 1995.

[10] M. G. Siqueira, P. S. R. Diniz, and A. Alwan, "Infinite Precision Analysis of the Fast QR Decomposition RLS Algorithm," *Proceedings of the IEEE International Symposium on Circuits and Systems*, London, UK, 1994.

[11] M. D. Miranda, L. Aguayo, and M. Gerken, "Performance of the *a Priori* and *a Posteriori* QR-LSL Algorithms in a Limited Precision Environment," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Munich, Germany, April 1997.

[12] C. Samsom and V. Reddy, "Fixed Point Error Analysis of the Normalized Ladder Algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 31, pp. 177–1191, October 1983.

[13] N. E. Hubing and S. T. Alexander, "Statistical Analysis of Initialization Methods for RLS Adaptive Filters," *IEEE Transactions on Signal Processing*, v. 39, pp. 1793–1804, August 1991.

José Antonio Apolinário Jr. was born in Taubaté, Brazil, in January 1960. He graduated from the Military Academy of Agulhas Negras (AMAN), Resende, Brazil, in 1981, and received the B. Sc. degree from the Military Institute of Engineering (IME), Rio de Janeiro, Brazil, in 1988, and the M. Sc. degree from the University of Brasília (UnB), Brasília, Brazil, in 1993, and the D. Sc. degree from the Federal University of Rio de Janeiro (COPPE/UFRJ) in 1998, all in electrical engineering.

He is currently with the IME, heading the Department of Electrical Engineering. He was a Visiting Professor at the Escuela Politécnica del Ejército, Quito, Ecuador, in 1999 and 2000, and a Visiting Researcher at the Helsinki University of Technology, Espoo, Finland, in 1997. His research interests comprise many aspects of the digital signal processing incluiding adaptive systems and speech and array processing.

Dr. Apolinário has organized the Rio de Janeiro Chapter of the IEEE Communications Society, where he has been the Chapter Chair since 2001.

César Augusto Medina S. was born in Cuenca, Ecuador in December 1976. He received the Electronics Eng. degree from the Army Polytechnical School (ESPE), Quito, Ecuador, in 2000. He is currently with the Military Institute of Engineering (IME), Rio de Janeiro, Brazil as a M. Sc. student in eletrical engineering. His interests comprise aspects of the digital signal processing, adaptive filters, and speech processing.

Paulo S. R. Diniz was born in Niterói, Brazil. He received the Electronics Eng. degree (Cum Laude) from the Federal University of Rio de Janeiro (UFRJ) in 1978, the M.Sc. degree from COPPE/UFRJ in 1981, and the Ph.D. from Concordia University, Montreal, P.Q., Canada, in 1984, all in electrical engineering.

Since 1979 he has been with the Department of Electronic Engineering (the undergraduate dept.) UFRJ. He has also been with the Program of Electrical Engineering (the graduate studies dept.), COPPE/UFRJ, since 1984, where he is presently a Professor. He served as Undergraduate Course Coordinator and as Chairman of the Graduate Department. He is one of the three senior researchers and coordinators of the National Excellence Center in Signal Processing. He has also received the Rio de Janeiro State Scientist award, from the Governor of Rio de Janeiro state.

From January 1991 to July 1992, he was a visiting Research Associate in the Department of Electrical and Computer Engineering of University of Victoria, Victoria, B.C., Canada. He also holds a visiting Professor position at Helsinki University of Technology. From January 2002 to June 2002, he was a Melchor Chair Professor in the Department of Electrical Engineering of University of Notre Dame, Notre Dame, IN, USA. His teaching and research interests are in analog and digital signal processing, adaptive signal processing, digital communications, wireless communications, multirate systems, stochastic processes, and electronic circuits. He has published several refereed papers in some of these areas and wrote the books *ADAPTIVE FILTERING: Algorithms and Practical Implementation*, Kluwer Academic Publishers, Second Edition 2002, and *DIGITAL SIGNAL PROCESSING: System Analysis and Design*," Cambridge University Press, Cambridge, UK, 2002 (with E. A. B. da Silva and S. L. Netto).

He was the Technical Program Chair of the 1995 MWSCAS held in Rio de Janeiro, Brazil. He has been on the technical committee of several international conferences including ISCAS, ICECS, EUSIPCO and MWSCAS. He has served Vice President for region 9 of the IEEE Circuits and Systems Society and as Chairman of the DSP technical committee of the same Society. He is also a Fellow of IEEE (*for fundamental contributions to the design and implementation of fixed and adaptive filters and Electrical Engineering Education*). He has served as associate editor for the following Journals: IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing from 1996 to 1999, IEEE Transactions on Signal Processing from 1999 to 2002, and the Circuits, Systems and Signal Processing Journal from 1998 to 2002. He was a distinguished lecturer of the IEEE Circuits and Systems Society for the year 2000 to 2001.

Table 3. Mean Square Values of Internal Variables (Cont.).

Simulation	Theoretical
$aux_i PRI\_BV.2$	$aux_i PRI\_BV.2$ **eq.(33)**
i=0 0.071272803	0.068018707
i=1 0.067740128	0.064617772
i=2 0.064403535	0.061386883
i=3 0.061240975	0.058317539
i=4 0.058259792	0.055401662
i=5 0.055449104	0.052631579
$aux_i POS\_BV.2$	$aux_i POS\_BV.2$ **eq.(29)**
i=0 0.037887263	0.038689047
i=1 0.039734548	0.040725313
i=2 0.041652272	0.04286875
i=3 0.043648002	0.045125
i=4 0.045713712	0.0475
i=5 0.047849413	0.05
$d_{q2_i}(k)$	$^* d_{q2_i}(k)$
i=0 0.000182101	0.000180263
i=1 0.000749849	0.000744770
i=2 0.000213413	0.000214218
i=3 0.015857762	0.015814596
i=4 0.000616829	0.000646154
$e_{q1}^{(i)}(k+1)$	$^* e_{q1}^{(i)}(k+1)$
i=0 $0.881162016\ 10^{-3}$	$0.88\ 10^{-3}$
i=1 $0.850300906\ 10^{-3}$	$0.847692308\ 10^{-3}$
i=2 $0.057352799\ 10^{-3}$	$0.056962525\ 10^{-3}$
i=3 $0.046681956\ 10^{-3}$	$0.046251623\ 10^{-3}$
i=4 $0.009183574\ 10^{-3}$	$0.009013137\ 10^{-3}$
i=5 0.000077200	0
$cos\theta'_{f_i}(k)$	$^* cos\theta'_{f_i}(k)$
i=0 0.977811931	0.974358974
i=1 0.977604023	0.974358974
i=2 0.978317517	0.974358974
i=3 0.978072023	0.974358974
i=4 0.978399190	0.974358974
$sin\theta'_i(k)$	$^* sin\theta'_i(k)$
i=0 0.023189070	0.025641026
i=1 0.023396978	0.025641026
i=2 0.022683484	0.025641026
i=3 0.022928978	0.025641026
i=4 0.022601811	0.025641026
$a_i(k)PRI\_B$	$a_i(k)PRI\_B$ **eq.(31)**
i=0 0.071273163	0.068018707
i=1 0.067741817	0.064617772
i=2 0.064404320	0.061386883
i=3 0.061240011	0.058317539
i=4 0.058259075	0.055401662
i=5 0.055449104	0.052631579
$f_i(k)POS\_B$	$f_i(k)POS\_B$ **eq.(27)**
i=0 0.037887517	0.038689047
i=1 0.039735609	0.040725313
i=2 0.041652819	0.04286875
i=3 0.043647296	0.045125
i=4 0.045713065	0.0475
i=5 0.047849413	0.05
$aux_i POS\_BV.1$	$aux_i POS\_BV.1$ **eq.(28)**
i=0 0.047849413	0.0475
i=1 0.045713712	0.045125
i=2 0.043648002	0.04286875
i=3 0.041652272	0.040725313
i=4 0.039734548	0.038689047
$cos\theta_i(k)$	$^* cos\theta_i(k)$
0.953151588	0.95
$sin\theta_i(k)$	$^* sin\theta_i(k)$
0.047849413	0.05
$e(k)$	$^* e(k)$
$6.09\ 10^{-8}$	$7.74\ 10^{-8}$

Table 4. The FQR_POS_B equations.

FQR_POS_B

for each k

{ 1. Obtaining $\boldsymbol{d}_{fq_2}(k+1)$:

$$\begin{bmatrix} e_{fq_1}(k+1) \\ \boldsymbol{d}_{fq_2}(k+1) \end{bmatrix} = \boldsymbol{Q}_\theta(k) \begin{bmatrix} x(k+1) \\ \lambda^{1/2}\boldsymbol{d}_{fq_2}(k) \end{bmatrix}$$

2. Obtaining $\| \boldsymbol{e}_f(k+1) \|$:

$$\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2}$$

3. Obtaining $\boldsymbol{Q}'_{\theta f}(k+1)$:

$$\begin{bmatrix} \boldsymbol{0} \\ \| \boldsymbol{e}_f^{(0)}(k+1) \| \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k+1) \begin{bmatrix} \boldsymbol{d}_{fq_2}(k+1) \\ \| \boldsymbol{e}_f(k+1) \| \end{bmatrix}$$

4. Obtaining $\boldsymbol{f}(k+1)$:

$$\begin{bmatrix} \frac{e_b(k+1)}{\|\boldsymbol{e}_b(k+1)\|} \\ \boldsymbol{f}(k+1) \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k+1) \begin{bmatrix} \boldsymbol{f}(k) \\ \frac{e_f(k+1)}{\|\boldsymbol{e}_f(k+1)\|} \end{bmatrix}$$

5. Obtaining $\boldsymbol{Q}_\theta(k+1)$:

$$\begin{bmatrix} 1 \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{Q}_\theta^T(k+1) \begin{bmatrix} \gamma(k+1) \\ \boldsymbol{f}(k+1) \end{bmatrix}$$

6. Joint Process Estimation:

$$\begin{bmatrix} e_{q_1}(k+1) \\ \boldsymbol{d}_{q_2}(k+1) \end{bmatrix} = \boldsymbol{Q}_\theta(k+1) \begin{bmatrix} d(k+1) \\ \lambda^{1/2}\boldsymbol{d}_{q_2}(k) \end{bmatrix}$$

7. Updating the output error:

$$e(k+1) = e_{q_1}(k+1)\gamma(k+1)$$

}

Table 5. The FQR_PRI_B equations.

FQR_PRI_B

for each k

{ 1. Obtaining $\boldsymbol{d}_{fq_2}(k+1)$:

$$\begin{bmatrix} e_{fq_1}(k+1) \\ \boldsymbol{d}_{fq_2}(k+1) \end{bmatrix} = \boldsymbol{Q}_\theta(k) \begin{bmatrix} x(k+1) \\ \lambda^{1/2}\boldsymbol{d}_{fq_2}(k) \end{bmatrix}$$

2. Obtaining $\boldsymbol{a}(k+1)$:

$$\begin{bmatrix} \frac{e'_b(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_b(k)\|} \\ \boldsymbol{a}(k+1) \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k) \begin{bmatrix} \boldsymbol{a}(k) \\ \frac{e'_f(k+1)}{\sqrt{\lambda}\|\boldsymbol{e}_f(k)\|} \end{bmatrix}$$

3. Obtaining $\| \boldsymbol{e}_f(k+1) \|$:

$$\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2}$$

4. Obtaining $\boldsymbol{Q}'_{\theta f}(k+1)$:

$$\begin{bmatrix} \boldsymbol{0} \\ \| \boldsymbol{e}_f^{(0)}(k+1) \| \end{bmatrix} = \boldsymbol{Q}'_{\theta f}(k+1) \begin{bmatrix} \boldsymbol{d}_{fq_2}(k+1) \\ \| \boldsymbol{e}_f(k+1) \| \end{bmatrix}$$

5. Obtaining $\boldsymbol{Q}_\theta(k+1)$:

$$\begin{bmatrix} 1/\gamma(k+1) \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{Q}_\theta(k+1) \begin{bmatrix} 1 \\ -\boldsymbol{a}(k+1) \end{bmatrix}$$

6. Joint Process Estimation:

$$\begin{bmatrix} e_{q_1}(k+1) \\ \boldsymbol{d}_{q_2}(k+1) \end{bmatrix} = \boldsymbol{Q}_\theta(k+1) \begin{bmatrix} d(k+1) \\ \lambda^{1/2}\boldsymbol{d}_{q_2}(k) \end{bmatrix}$$

7. Updating the output error:

$$e(k+1) = e_{q_1}(k+1)\gamma(k+1)$$

}

Appendix A: The Detailed FQR_POS_B Version 1 Algorithm

FQR_POS_B – Version 1 [6]

Initialization:
ϵ = small positive value;
$\| \boldsymbol{e}_f(k) \| = \epsilon$;
$\boldsymbol{d}_{fq2}(k) =$ zeros$(N+1, 1)$;
$\boldsymbol{d}_{q2}(k) =$ zeros$(N+1, 1)$;
$cos\boldsymbol{\theta}(k) =$ ones$(N+1, 1)$;
$sin\boldsymbol{\theta}(k) =$ zeros$(N+1, 1)$;
$\boldsymbol{f}(k) =$ zeros$(N+1, 1)$;
for $k = 1, 2, \ldots$
$\{ \;\; e_{fq_1}^{(0)}(k+1) = x(k+1);$
for $i = 1 : N+1$
$\{ \;\; e_{fq_1}^{(i)}(k+1) = cos\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k)\lambda^{1/2} d_{fq2_{N+2-i}}(k);$
$d_{fq2_{N+2-i}}(k+1) = sin\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k)\lambda^{1/2} d_{fq2_{N+2-i}}(k);$
$\}$
$e_{fq_1}(k+1) = e_{fq_1}^{(N+1)}(k+1);$
$\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2};$
$\| \boldsymbol{e}_f^{(N+1)}(k+1) \| = \| \boldsymbol{e}_f(k+1) \|;$
for $i = 1 : N+1$
$\{ \;\; \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \| = \sqrt{\| \boldsymbol{e}_f^{(N+2-i)}(k+1) \|^2 + d_{fq2_i}^2(k+1)};$
$cos\theta'_{f_{N+1-i}}(k+1) = \| \boldsymbol{e}_f^{(N+2-i)}(k+1) \| \, / \, \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|;$
$sin\theta'_{f_{N+1-i}}(k+1) = d_{fq2_i}(k+1) / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|;$
$\}$
$aux_0 = x(k+1) / \| \boldsymbol{e}_f^{(0)}(k+1) \|;$
$f_{N+1}(k+1) = aux_0;$
for $i = 1 : N$
$\{ \;\; f_{N+1-i}(k+1) = \frac{f_{N+2-i}(k) - sin\theta'_{f_{i-1}}(k+1) aux_{i-1}}{cos\theta'_{f_{i-1}}(k+1)};$
$aux_i = -sin\theta'_{f_{i-1}}(k+1) f_{N+1-i}(k+1) + cos\theta'_{f_{i-1}}(k+1) aux_{i-1};$
$\}$
$\gamma^{(0)}(k+1) = 1;$
for $i = 1 : N+1$
$\{ \;\; sin\theta_{i-1}(k+1) = f_{N+2-i}(k+1) / \gamma^{(i-1)}(k+1);$
$cos\theta_{i-1}(k+1) = \sqrt{1 - sin^2\theta_{i-1}(k+1)};$
$\gamma^{(i)}(k+1) = cos\theta_{i-1}(k+1) \gamma^{(i-1)}(k+1);$
$\}$
$\gamma(k+1) = \gamma^{(N+1)}(k+1);$
$e_{q_1}^{(0)}(k+1) = d(k+1);$
for $i = 1 : N+1$
$\{ \;\; e_{q_1}^{(i)}(k+1) = cos\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k+1)\lambda^{1/2} d_{q2_{N+2-i}}(k);$
$d_{q2_{N+2-i}}(k+1) = sin\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k+1)\lambda^{1/2} d_{q2_{N+2-i}}(k);$
$\}$
$e_{q_1}(k+1) = e_{q_1}^{(N+1)}(k+1);$
$e(k+1) = e_{q_1}(k+1)\gamma(k+1);$
$\}$

Appendix B: The Detailed FQR_POS_B Version 2 Algorithm

FQR_POS_B – Version 2 [2]

Initialization:
ϵ = small positive value;
$\| \boldsymbol{e}_f(k) \| = \epsilon$;
$\boldsymbol{d}_{fq2}(k) =$ zeros$(N+1,1)$;
$\boldsymbol{d}_{q2}(k) =$ zeros$(N+1,1)$;
$cos\boldsymbol{\theta}(k) =$ ones$(N+1,1)$;
$sin\boldsymbol{\theta}(k) =$ zeros$(N+1,1)$;
$\boldsymbol{f}(k) =$ zeros$(N+1,1)$;
for $k = 1, 2, \ldots$
{ $e_{fq_1}^{(0)}(k+1) = x(k+1)$;
  for $i = 1 : N+1$
  { $e_{fq_1}^{(i)}(k+1) = cos\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k)\lambda^{1/2} d_{fq2_{N+2-i}}(k)$;
    $d_{fq2_{N+2-i}}(k+1) = sin\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k)\lambda^{1/2} d_{fq2_{N+2-i}}(k)$;
  }
  $e_{fq_1}(k+1) = e_{fq_1}^{(N+1)}(k+1)$;
  $\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2}$;
  $\| \boldsymbol{e}_f^{(N+1)}(k+1) \| = \| \boldsymbol{e}_f(k+1) \|$;
  for $i = 1 : N+1$
  { $\| \boldsymbol{e}_f^{(N+1-i)}(k+1) \| = \sqrt{\| \boldsymbol{e}_f^{(N+2-i)}(k+1) \|^2 + d_{fq2_i}^2(k+1)}$;
    $cos\theta'_{f_{N+1-i}}(k+1) = \| \boldsymbol{e}_f^{(N+2-i)}(k+1) \| / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|$;
    $sin\theta'_{f_{N+1-i}}(k+1) = d_{fq2_i}(k+1) / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|$;
  }
  $aux_0 = \frac{\gamma(k) e_{fq_1}(k+1)}{\| \boldsymbol{e}_f(k+1) \|}$;
  for $i = 1 : N+1$
  { $f_{i-1}(k+1) = cos\theta'_{f_{N+1-i}}(k+1) f_i(k) - sin\theta'_{f_{N+1-i}}(k+1) aux_{i-1}$;
    $aux_i = sin\theta'_{f_{N+1-i}}(k+1) f_i(k) + cos\theta'_{f_{N+1-i}}(k+1) aux_{i-1}$;
  }
  $\frac{e_b(k+1)}{\| \boldsymbol{e}_b(k+1) \|} = f_0(k+1)$;
  $f_{N+1}(k+1) = aux_{N+1}$;
  $\gamma^{(0)}(k+1) = 1$;
  for $i = 1 : N+1$
  { $sin\theta_{i-1}(k+1) = f_{N+2-i}(k+1)/\gamma^{(i-1)}(k+1)$;
    $cos\theta_{i-1}(k+1) = \sqrt{1 - sin^2\theta_{i-1}(k+1)}$;
    $\gamma^{(i)}(k+1) = cos\theta_{i-1}(k+1)\gamma^{(i-1)}(k+1)$;
  }
  $\gamma(k+1) = \gamma^{(N+1)}(k+1)$;
  $e_{q_1}^{(0)}(k+1) = d(k+1)$;
  for $i = 1 : N+1$
  { $e_{q_1}^{(i)}(k+1) = cos\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k+1)\lambda^{1/2} d_{q2_{N+2-i}}(k)$;
    $d_{q2_{N+2-i}}(k+1) = sin\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k+1)\lambda^{1/2} d_{q2_{N+2-i}}(k)$;
  }
  $e_{q_1}(k+1) = e_{q_1}^{(N+1)}(k+1)$;
  $e(k+1) = e_{q_1}(k+1)\gamma(k+1)$;
}

Appendix C: The Detailed FQR_PRI_B Version 1 Algorithm

FQR_PRI_B – Version 1 [4]

Initialization:
ϵ = small positive value;
$\| \boldsymbol{e}_f^{(0)}(k) \| = \epsilon$;
$\| \boldsymbol{e}_f(k) \| = \epsilon$;
$\boldsymbol{d}_{fq2}(k) =$ zeros$(N+1,1)$;
$\boldsymbol{d}_{q2}(k) =$ zeros$(N+1,1)$;
$cos\boldsymbol{\theta}(k) =$ ones$(N+1,1)$;
$cos\boldsymbol{\theta}'_f(k) =$ ones$(N+1,1)$;
$sin\boldsymbol{\theta}(k) =$ zeros$(N+1,1)$;
$sin\boldsymbol{\theta}'_f(k) =$ zeros$(N+1,1)$;
$\boldsymbol{a}(k) =$ zeros$(N+1,1)$;
for $k = 1, 2, \ldots$
$\{$ $e_{fq_1}^{(0)}(k+1) = x(k+1)$;
for $i = 1 : N+1$
$\{$ $e_{fq_1}^{(i)}(k+1) = cos\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k) \lambda^{1/2} d_{fq2_{N+2-i}}(k)$;
$d_{fq2_{N+2-i}}(k+1) = sin\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k) \lambda^{1/2} d_{fq2_{N+2-i}}(k)$;
$\}$
$e_{fq_1}(k+1) = e_{fq_1}^{(N+1)}(k+1)$;
$aux_0 = \frac{x(k+1)}{\lambda^{1/2} \| \boldsymbol{e}_f^{(0)}(k) \|}$;
$a_{N+1}(k+1) = aux_0$;
for $i = 1 : N$
$\{$ $a_{N+1-i}(k+1) = \frac{a_{N+2-i}(k) - sin\theta'_{f_{i-1}}(k) aux_{i-1}}{cos\theta'_{f_{i-1}}(k)}$;
$aux_i = -sin\theta'_{f_{i-1}}(k) a_{N+1-i}(k+1) + cos\theta'_{f_{i-1}}(k) aux_{i-1}$;
$\}$
$\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2}$;
$\| \boldsymbol{e}_f^{(N+1)}(k+1) \| = \| \boldsymbol{e}_f(k+1) \|$;
for $i = 1 : N+1$
$\{$ $\| \boldsymbol{e}_f^{(N+1-i)}(k+1) \| = \sqrt{\| \boldsymbol{e}_f^{(N+2-i)}(k+1) \|^2 + d_{fq2_i}^2}(k+1)$;
$cos\theta'_{f_{N+1-i}}(k+1) = \| \boldsymbol{e}_f^{(N+2-i)}(k+1) \| / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|$;
$sin\theta'_{f_{N+1-i}}(k+1) = d_{fq2_i}(k+1) / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|$;
$\}$
$1/\gamma^{(0)}(k+1) = 1$;
for $i = 1 : N+1$
$\{$ $1/\gamma^{(i)}(k+1) = \sqrt{[1/\gamma^{(i-1)}(k+1)]^2 + a_{N+2-i}^2(k+1)}$;
$cos\theta_{i-1}(k+1) = \frac{1/\gamma^{(i-1)}(k+1)}{1/\gamma^{(i)}(k+1)}$;
$sin\theta_{i-1}(k+1) = \frac{a_{N+2-i}(k+1)}{1/\gamma^{(i)}(k+1)}$;
$\}$
$\gamma(k+1) = 1/[1/\gamma^{(N+1)}(k+1)]$;
$e_{q_1}^{(0)}(k+1) = d(k+1)$;
for $i = 1 : N+1$
$\{$ $e_{q_1}^{(i)}(k+1) = cos\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k+1) \lambda^{1/2} d_{q2_{N+2-i}}(k)$;
$d_{q2_{N+2-i}}(k+1) = sin\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k+1) \lambda^{1/2} d_{q2_{N+2-i}}(k)$;
$\}$
$e_{q_1}(k+1) = e_{q_1}^{(N+1)}(k+1)$;
$e(k+1) = e_{q_1}(k+1) \gamma(k+1)$;
$\}$

Appendix D: The Detailed FQR_PRI_B Version 2 Algorithm

FQR_PRI_B – Version 2 [3]

Initialization:

ϵ = small positive value;

$\| \boldsymbol{e}_f^{(0)}(k) \| = \epsilon;$

$\| \boldsymbol{e}_f(k) \| = \epsilon;$

$\boldsymbol{d}_{fq2}(k) = \text{zeros}(N+1,1);$

$\boldsymbol{d}_{q2}(k) = \text{zeros}(N+1,1);$

$cos\boldsymbol{\theta}(k) = \text{ones}(N+1,1);$

$cos{\boldsymbol{\theta}'}_f(k) = \text{ones}(N+1,1);$

$sin\boldsymbol{\theta}(k) = \text{zeros}(N+1,1);$

$sin{\boldsymbol{\theta}'}_f(k) = \text{zeros}(N+1,1);$

$\boldsymbol{a}(k) = \text{zeros}(N+1,1);$

for $k = 1, 2, \ldots$

$\{ \quad e_{fq_1}^{(0)}(k+1) = x(k+1);$

for $i = 1 : N+1$

$\{ \quad e_{fq_1}^{(i)}(k+1) = cos\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k) \lambda^{1/2} d_{fq2_{N+2-i}}(k);$

$d_{fq2_{N+2-i}}(k+1) = sin\theta_{i-1}(k) e_{fq_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k) \lambda^{1/2} d_{fq2_{N+2-i}}(k);$

$\}$

$e_{fq_1}(k+1) = e_{fq_1}^{(N+1)}(k+1);$

$aux_0 = \frac{e_{fq_1}(k+1)}{\gamma(k)\lambda^{1/2} \| \boldsymbol{e}_f(k) \|};$

for $i = 1 : N+1$

$\{ \quad a_{i-1}(k+1) = cos{\theta'}_{f_{N+1-i}}(k) a_i(k) - sin{\theta'}_{f_{N+1-i}}(k) aux_{i-1};$

$aux_i = sin{\theta'}_{f_{N+1-i}}(k) a_i(k) + cos{\theta'}_{f_{N+1-i}}(k) aux_{i-1};$

$\}$

$\frac{e'_b(k+1)}{\lambda^{1/2} \| \boldsymbol{e}_b(k) \|} = a_0(k+1);$

$a_{N+1}(k+1) = aux_{N+1};$

$\| \boldsymbol{e}_f(k+1) \| = \sqrt{e_{fq_1}^2(k+1) + \lambda \| \boldsymbol{e}_f(k) \|^2};$

$\| \boldsymbol{e}_f^{(N+1)}(k+1) \| = \| \boldsymbol{e}_f(k+1) \|;$

for $i = 1 : N+1$

$\{ \quad \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \| = \sqrt{\| \boldsymbol{e}_f^{(N+2-i)}(k+1) \|^2 + d_{fq2_i}^2(k+1)};$

$cos{\theta'}_{f_{N+1-i}}(k+1) = \| \boldsymbol{e}_f^{(N+2-i)}(k+1) \| / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|;$

$sin{\theta'}_{f_{N+1-i}}(k+1) = d_{fq2_i}(k+1) / \| \boldsymbol{e}_f^{(N+1-i)}(k+1) \|;$

$\}$

$1/\gamma^{(0)}(k+1) = 1;$

for $i = 1 : N+1$

$\{ \quad 1/\gamma^{(i)}(k+1) = \sqrt{[1/\gamma^{(i-1)}(k+1)]^2 + a_{N+2-i}^2(k+1)};$

$cos\theta_{i-1}(k+1) = \frac{1/\gamma^{(i-1)}(k+1)}{1/\gamma^{(i)}(k+1)};$

$sin\theta_{i-1}(k+1) = \frac{a_{N+2-i}(k+1)}{1/\gamma^{(i)}(k+1)};$

$\}$

$\gamma(k+1) = 1/[1/\gamma^{(N+1)}(k+1)];$

$e_{q_1}^{(0)}(k+1) = d(k+1);$

for $i = 1 : N+1$

$\{ \quad e_{q_1}^{(i)}(k+1) = cos\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) - sin\theta_{i-1}(k+1) \lambda^{1/2} d_{q2_{N+2-i}}(k);$

$d_{q2_{N+2-i}}(k+1) = sin\theta_{i-1}(k+1) e_{q_1}^{(i-1)}(k+1) + cos\theta_{i-1}(k+1) \lambda^{1/2} d_{q2_{N+2-i}}(k);$

$\}$

$e_{q_1}(k+1) = e_{q_1}^{(N+1)}(k+1);$

$e(k+1) = e_{q_1}(k+1)\gamma(k+1);$

$\}$