D3.4

A NOVEL STRUCTURE FOR ADAPTIVE LS FIR FILTERING BASED ON QR DECOMPOSITION

A.P.Varvitsiotis*, S. Theodoridis*, G. Moustakides**

* Dept. of Computer Eng. University of Patras, Patra, Rio, Greece.
** Computer Technology Institute, Patras.

ABSTRACT:

A very powerful technique for computing the LS estimates of an FIR filter's impulse response, is based on the QR factorization of the input data matrix. The method consists of two parts. First the input matrix is factorized into an orthogonal Q part and an upper triangular R part. The unknown coefficients are then obtained from a triangular linear system of equations. This paper presents a new algorithm for solving the above linear system, and it is appropriate for adaptive processing. This is achieved via a set of Givens rotations and a modified Faddeeva's scheme.

1. INTRODUCTION

Least squares FIR filtering is of major importance in many Signal Processing applications, such as Communications |1|, Spectral analysis |2|, Control and System idendification |3|. A major task in these problems is to compute the LS estimates of the unknown FIR system's impulse response, based on the minimization of the total squared error between the actual and a desired response signal, over a given time interval.

A very powerful technique for the efficient computation of the above estimates is the one exploiting the QR decomposition of the input data matrix |4|. According to this method QR decomposition is achieved via a sequence of Givens rotations (GR) and the unknown coefficients are computed from the resulting triangular matrix using back substitution (BS). It is well known that this method is well suited for implementation on a triangular systolic array structure,followed by a linear array to perform the back substitution. The disadvantages of such a scheme are a) a seperate linear array is required for BS, b) BS is not numerically robust and c) it is not appropriate for continuous adaptive operation, since new data cannot processed by the above array structure during BS.

Recently an alternative method for matrix operations has been suggested in |5|,called modified Fuddeeva algorithm, which alleviates the need for BS. The central idea of this method is first to triangularize the input data matrix via a series of GR and then use the obtained triangular matrix to compute the unknown coefficients,performing Gaussian elimination (GE) on a set of successive rows of the identity matrix. Although this scheme overcomes the need for backsubstitution still it is not appropriate for continuous adaptive operation and also the processing elements must be able to switch between rotational mode (GR) to multiply/add mode (GE).

In this paper, the algorithm given in |5| is first extended so that annullment of the identity matrix through the triangular matrix is achieved via GR. This results in scaling each of unknown parameters with a different scaling factor.However it is shown that each one of these scaling factors can be readily obtained in terms of GR parameters at practically no extra computational cost. Furthermore, if the array is to compute the unknown system's impulse response adaptively, on a sample by sample basis,an alternative scheme is adopted. While input data enter the top row processing elements (PE) of the triangular array, the corresponding Q-transformed desired response vectors enter the rightmost collumn boundary PEs, for each time instant. This is equivalent to the annullment of successive Q-transformed desired response vectors through the transpose of the triangularized input data matrix. Descaling and computation of the unknown coefficients are performed at the boundary PEs along the diagonal.

2. FULL GR MODIFIED FADDEEVA ALGORITHM

Let us consider a linear set of equations

$$A\underline{x} = \underline{b} \qquad 2.1$$

where $\underline{x}$ and $\underline{b}$ are collumn vectors and A is an mxm matrix. In Faddeva's algorithm matrix triangularization and computation of the quantity $\underline{c}^t A^{-1} \underline{b} + d$, with $\underline{c}$, d being a vector and a scalar respectively, are performed in a single step. This is achieved by triangularizing, via Gaussian elimination (GE), the augmented square matrix

$$\begin{bmatrix} A & \underline{b} \\ -\underline{c}^t & d \end{bmatrix} \qquad 2.2$$

In mathematical form Gaussian elimination may be expressed as

$$W \begin{bmatrix} A & \underline{b} \\ -\underline{c}^t & d \end{bmatrix} = \begin{bmatrix} R & \underline{b}' \\ 0 & d' \end{bmatrix} \qquad 2.3$$

with

$$W = \begin{bmatrix} L & 0 \\ \underline{w}^t & 1 \end{bmatrix} \qquad 2.4$$

where L is lower and R upper triangular matrices. From 2.3, 2.4 we get

$$d' = d + \underline{c}^t A^{-1} \underline{b} \qquad 2.5$$

CH2673-2/89/0000-0904 $1.00 © 1989 IEEE

Thus by setting d=0 and $\underline{c}$ equal to various unit vectors $\underline{i}_k=|0,\dots 1,\dots 0|^t$, one can find the elements of $A^{-1}\,\underline{b}$.

Let us now assume that matrix A has already been triangularized. Annulment of $\underline{c}^t$ through A will be performed via Givens rotations, instead of Gaussian elimination, due to the superior numerical performance of the former technique. The linear transformation imposed by a sequence of Givens rotations, which rotate $\underline{c}^t$ into a vector of zeros, has the form of an (m+1)x(m+1) matrix Q of the form |1|.

$$Q=\gamma\begin{bmatrix} Z & -Z\underline{\beta} \\ \underline{\beta}^t & 1 \end{bmatrix} \qquad 2.6$$

where γ is the product of the cosines of the M rotation angles, Z is an mxm lower triangular matrix and $\underline{\beta}$ is a collumn vector. From 2.6 and following similar arguments as above we obtain

$$Q\begin{bmatrix} A & \underline{b} \\ -\underline{c}^t & d \end{bmatrix} = \gamma\begin{bmatrix} Z(A+\underline{\beta}\,\underline{c}^t) & Z(\underline{b}-\underline{\beta}d) \\ \underline{\beta}^t A-\underline{c}^t & \underline{\beta}^t\,\underline{b}+d \end{bmatrix} = \begin{bmatrix} R' & \underline{b}' \\ \underline{o}^t & d' \end{bmatrix} \qquad 2.7$$

where

$$d' = (\underline{c}^t A^{-1}\,\underline{b} + d) \qquad 2.8$$

Eq. 2.8 is the same as 2.5 except the scaling factor γ. If instead of $\underline{c}^t$ a collection of vectors is used |5|, 2.8 is generalized to

$$\underline{d}' = \Gamma(CA^{-1}\,\underline{b} + \underline{d}) \qquad 2.9$$

where

$$\Gamma = \mathrm{diag}\,(\gamma_1,\gamma_2,\dots\gamma_m) \qquad 2.10$$

However the parameters γ_i are easily computed a cross the diagonal elements of the corresponding array structure, as it has been suggested in |6|. Thus for $\underline{d}=\underline{0}$ and C=I, 2.9 suggests that

$$\underline{x} \equiv A^{-1}\,\underline{b} = \Gamma^{-1}\,\underline{d}' \qquad 2.11$$

As it is suggested in |5| 2.9 can be used to perform a number of different matrix computations.

3. MODIFIED FADDEEVA ALGORITHM FOR ADAPTIVE OPERATION.

In Least Squares FIR filtering matrix A of the previous section is the upper triangular matrix of the QR factorization of the input data matrix, $\underline{b}$ is the Q transformed vactor of the desired response and $\underline{x}$ is the LS estimate of the impulse response |1|. In an adaptive mode of operation matrix A is continuously updated on a sample by sample basis. Thus in order to annul matrix I by rotating it through A, adaptation of A has temporarily to cease. This is obviously an undesirable feature. Here an alternative scheme is suggested that overcomes this problem. In order to compute $A^{-1}\,\underline{b}$, we choose to annul $-\underline{b}^t$ through A^t |7|. The (m+1)x (m+1) transformation matrix can be written as

$$Q = \begin{bmatrix} Q_{11} & \underline{q}_{12} \\ \underline{q}_{21}^t & q_{22} \end{bmatrix} \qquad 3.1$$

$$Q\begin{bmatrix} A^t \\ -\underline{b}^t \end{bmatrix} = \begin{bmatrix} Q_{11}\,A^t - \underline{q}_{12}\,\underline{b}^t \\ \underline{q}_{21}^t\,A^t - q_{22}\,\underline{b}^t \end{bmatrix} = \begin{bmatrix} Q_{11}\,A^t - \underline{q}_{12}\,\underline{b}^t \\ \underline{o}^t \end{bmatrix} \qquad 3.2$$

Hence the required solution is given by

$$\underline{x} = A^{-1}\,\underline{b} = \underline{q}_{21}/q_{22} \qquad 3.3$$

Next we shall show that $\underline{q}_{21}/q_{22}$ can be expressed directly in terms of the rotation angles, which result from the annulment of $-\underline{b}^t$ through A^t via Givens rotations.

Lemma.
Define

$$Q^{(1)} = \begin{bmatrix} c_1 & \underline{o}^t & s_1 \\ \underline{o} & I_{m-1} & \underline{o} \\ -s_1 & \underline{o}^t & c_1 \end{bmatrix} \qquad 3.4a$$

$$\gamma^{(1)} = c_1 \qquad 3.4b$$

and the recursive relations

$$Q^{(k+1)} = \begin{bmatrix} I_k & \underline{o} & 0 & \underline{o} \\ \underline{o}^t & c_{k+1} & \underline{o}^t & s_{k+1} \\ 0 & \underline{o} & I_{m-k-1} & \underline{o} \\ \underline{o}^t & -s_{k+1} & \underline{o}^t & c_{k+1} \end{bmatrix} Q^{(k)}$$

$$\gamma^{(k+1)} = c_{k+1}\,\gamma^{(k)} \qquad 3.5b$$

where I_k is the k-dimensional identity matrix and c_k, s_k are cosines and sines, respectively, of the Givens rotations. Show that

$$Q^{(k)} = \gamma^{(k)}\begin{bmatrix} Z^{(k)} & 0 & -Z^{(k)}\,\underline{\beta}^{(k)} \\ 0 & \gamma^{-(k)}\,I_{m-k} & \underline{o} \\ \underline{\beta}^{(k)t} & \underline{o}^t & 1 \end{bmatrix} \qquad 3.6a$$

where

$$\underline{\beta}^{(k)t} = \left[\frac{-s_1}{c_1}, \frac{-s_2}{c_1c_2}, \dots, \frac{-s_k}{c_1c_2\dots c_k}\right] \qquad 3.6b$$

Proof:

a) Obvious for k=1

b) Assume that for all k such that $1\le k\le n<m$, $Q^{(k)}$ is given by 3.6a. Rewrite for k=n 3.6a as

$$Q^{(n)}=\gamma^{(n)}\begin{bmatrix} Z^{(n)} & \underline{o} & 0 & -Z^{(n)}\underline{\beta}^{(n)} \\ \underline{o}^t & \gamma^{-(n)}\underline{o}^t & & o \\ 0 & \underline{o} & \gamma^{-(n)} I & o \\ \underline{\beta}^{(n)t} & o & \underline{o}^t & 1 \end{bmatrix} \quad 3.7$$

Then using 3.5a the following is obtained.

$$Q^{(n+1)} = c_{n+1}\gamma^{(n)} \left[D \vdots E \right] \quad 3.8a$$

with

$$D = \begin{bmatrix} c_{n+1}^{-1} Z^{(n)} & \underline{o} \\ s_{n+1} c_{n+1}^{-1} \underline{\beta}^{(n)t} & \gamma^{-(n)} \\ 0 & \underline{o} \\ \underline{\beta}^{(n)t} & -s_{n+1} c_{n+1}^{-1} \gamma^{-(n)} \end{bmatrix} \quad 3.8b$$

$$E=\begin{bmatrix} 0 & -c_{n+1}^{-1} Z^{(n)} \underline{\beta}^{(n)} \\ \underline{o}^t & s_{n+1} c_{n+1}^{-1} \\ c_{n+1}^{-1}\gamma^{-(n)} I & \underline{o} \\ \underline{o}^t & 1 \end{bmatrix} \quad 3.8c$$

Define

$$Z^{(n+1)} = \begin{bmatrix} c_{n+1}^{-1} Z^{(n)} & \underline{o} \\ Z_{n+1} c_{n+1}^{-1} \underline{\beta}^{(n)t} & \gamma^{-(n)} \end{bmatrix} \quad 3.9a$$

and

$$\underline{\beta}^{(n+1)t} = \left[\underline{\beta}^{(n)t} - s_{n+1} c_{n+1}^{-1} \gamma^{-(n)} \right] \quad 3.9b$$

Then if $Z^{(n)}$ is lower triangular, $Z^{(n+1)}$ is also lower triangular. Moreover taking into account that c_i and s_i are cosines and sines

$$-Z^{(n+1)} \underline{\beta}^{(n+1)} = \begin{bmatrix} -c_{n+1}^{-1} Z^{(n)} \underline{\beta}^{(n)} \\ s_{n+1}/c_{n+1} \end{bmatrix} \quad 3.10$$

Also

$$\underline{\beta}^{(k)t} = \left[\frac{-s_1}{c_1}, \frac{-s_2}{c_1 c_2}, \ldots, \frac{-s_k}{c_1 c_2 \cdots c_k} \right] \quad 3.11$$

From the above lemma and the fact that Q in 3.1 results from a series of rotations of the type described in the lemma, it is readily shown that 3.3 becomes

$$\underline{x} \equiv R^{-1} b = J \left[\frac{-s_1}{c_1}, \frac{-s_2}{c_1 c_2}, \ldots \frac{-s_m}{c_1 c_2 \cdots c_m} \right]^t \quad 3.12$$

where the exchange matrix J results from the fact that the above lemma refers to annulment through an upper triangular matrix, while A^t is a lower triangular matrix.

Figure 1 shows the triangular systolic structure, which realizes the above algorithm. There are two opposite data flows. Data enter the top row and are then reflected by the rightmost column cells back to angle-computing PEs. Moreover the computational hyperplanes for each function are parallel, but schedule vectors assume opposite directions. This means that the computations on a given input data row are propagated from top left to the bottom right element and vice versa, while the matrix elements involved in each computation should remain available in the respective elements. As a consequence, each (i,j) element must include a FIFO queue of size 2(2m-j-i)+1, where the triangular matrix element r_{ij} is saved until used in the second angle computation/rotation. Note that the coefficients are produced doubly skewed, due to the length of the data path between two successive boundary PEs. Each processing element must be able to perform two full angle computations (boundary elements) or rotations (intermidiate elements). In this way the updates of the triangular matrix and of the filters' coefficients take place concurrently.

Conclusion: A new scheme has been discussed for the adaptive computation of an FIR filter's impulse response. The algorithm can be implemented via a systolic triangular array without the need of a backsubstitution step.

References:

1 S. Haykin, "Adaptive filter theory", Prentice Hall, 1986.

2 J.G. Proakis, D. Manolakis, "Introduction to Digital Signal Processing", MacMillan, 1988.

3 G.C. Goodwin, K.S. Sin, "Adaptive filtering prediction and Control", Prentice Hall, 1984.

4 M.W. Gentleman, H.T. Kung, "Matrix triangularization by systolic arrays", Proc. SPIE, vol. 298, Real time processing IV, pp. 298-303.

5 J.G. Nash, S. Hanses, "Modified Faddeeva Algorithm for Concurrent Execution of linear algebraic operations", IEE Trans. on Computers, vol. 37, No. 2, pp. 129-136, Feb. 1988.

6 J.G. McWhirter, "Systolic array for recursive Least mean Squares minimization", Electronics Letters, vol. 19, No. 18, pp. 729-730, Sep. 1983.

7 P. Dewilde, J. Annevelick, "Hierarchical design of Processor Arrays applied to a new pipelined matrix solver", VLSI Signal Processing, S.Y. Kung, J.G. Nash (eds), pp. 106-116, IEEE Press, 1986.

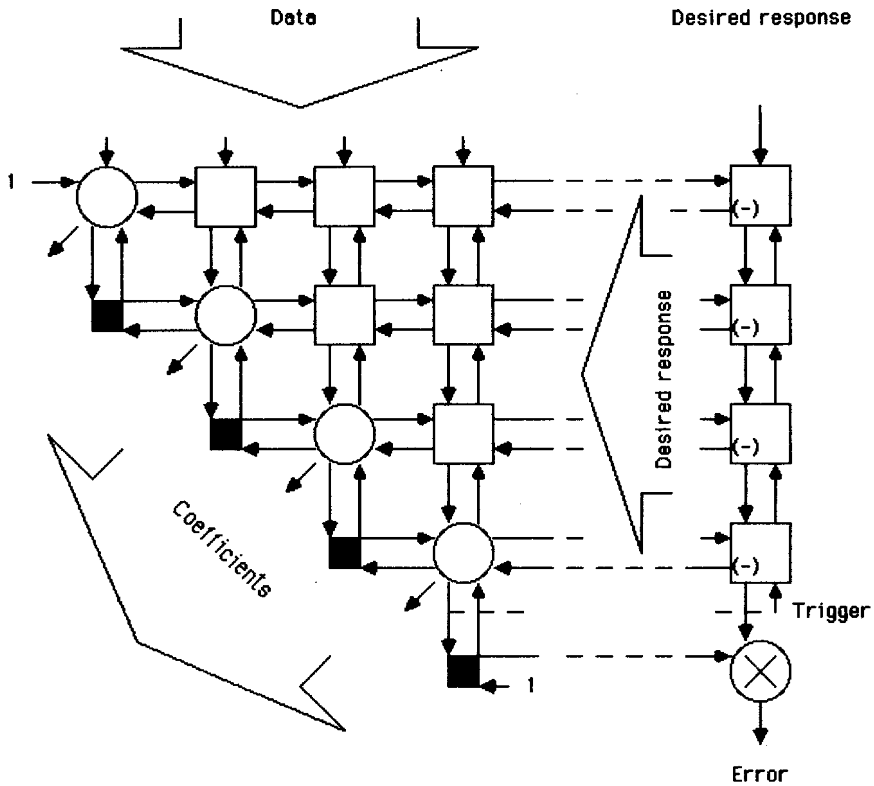

Figure 1. The proposed array for adaptive computation of the unknown coefficients on a sample by sample basis.