NUMERICAL PROPERTIES OF THE LINEARLY CONSTRAINED QRD-RLS ADAPTIVE FILTER

J. Huo and Y. H. Leung

Australian Telecommunications Research Institute
Curtin University of Technology
Bentley, WA 6009, Australia

ABSTRACT

Shepherd and McWhirter proposed a QRD-RLS algorithm for adaptive filtering with linear constraints. In this paper, the numerical properties of this algorithm are considered. In particular, it is shown that the computed weight vector satisfies a set of constraints which are perturbed from the original ones, the amount of perturbation being dependent on the wordlength. The linearly constrained FLS algorithm of Resende *et al* is also studied. Simulation results show that this algorithm is numerically unstable, even in the absence of explosive divergence.

1. INTRODUCTION

Adaptive filters have found many applications in signal processing [4]. In applications such as adaptive beamforming, linear constraints are often imposed on the filter weights to attain a desired temporal and/or spatial response [3], [8]. When implemented digitally, it is well-known that adaptive filters can suffer from a number of numerical problems [4], caused by the accumulation of rounding errors which arises as a result of the inherently recursive nature of the adaptive algorithm and the necessarily finite precision of the digital implementation.

There exists a rich literature on the study of the numerical stability of the *unconstrained* RLS algorithms. Some of these algorithms are potentially unstable because of the implicit or explicit update of the inverse covariance matrix [1], [2], [9]. However, the QRD-RLS algorithm is stable [4], [5]. It is shown in [5] that if the filter input does not approach zero asymptotically, then the computed weights are bounded.

Although the numerical stability of the unconstrained algorithms are well understood, imposition of constraints may lead to extra numerical difficulties. An immediately obvious problem is that the deviation of the computed weights from the imposed constraints may grow unacceptably large [3]. We called this problem *constraint drift*.

In this paper, we analyze the constraint drift property of the linearly constrained QRD-RLS (LCQRD-RLS) algorithm proposed by Shepherd and McWhirter [7]. We begin with a brief description of the algorithm. After that, a geometrical interpretation is presented to show that the LCQRD-RLS algorithm is free from constraint drift. This is confirmed by simulation study. The simulation study also compares the numerical performances of the LCQRD-RLS algorithm against the linear constrained FLS (LCFLS) algorithm of [6]. It is shown that the LCFLS is numerically unstable.

2. QRD-RLS ALGORITHM FOR LINEARLY CONSTRAINED ADAPTIVE FILTERING

Consider the FIR filter characterized by its weights $\mathbf{w}(n)$ as shown in Figure 1. The linearly constrained RLS filter is defined by the following optimization problem

$$\min_{\mathbf{w}(n)}\left(\sum_{i=0}^{n}\beta^{n-i}\left|d(i)-\mathbf{w}^H(n)\mathbf{u}(i)\right|^2\right)$$

subject to $$\mathbf{C}^H\mathbf{w}(n) = \mathbf{m} \qquad \text{(P.1)}$$

where $d(n)$ is the desired signal; $\mathbf{w}(n)$ is the $N\times 1$ filter weight vector; $\mathbf{u}(n) = [u(n)\ \ldots\ u(n-N+1)]^T$ is the $N\times 1$ input data vector; $0\le\beta\le 1$ is the forgetting factor; $\mathbf{C}$ is the $N\times K$ constraint matrix (assumed full rank), and $\mathbf{m}$ is the $K\times 1$ constraint vector.

In [7], Shepherd and McWhirter showed that the above optimization problem can be solved by the systolic array structure depicted in Figure 2. The whole network consists two sections: a frozen network and a canonical systolic array. The frozen network restricts the computed weight vector onto the constraints while the canonical section carries out the least-squares minimization. The signals $\hat{\mathbf{u}}(n)$ and $\hat{y}$ are defined within the equation box in Figure 2. The quantities U, V and m' can be obtained by applying a unitary transformation (QR decomposition) to the augmented constraint matrix.

$$Q\left[C^H \ -\mathbf{m}\right] = \left[\mathbf{U}\ \mathbf{V}\ \mathbf{m}'\right] \qquad (1)$$

where the matrix U is required to be upper triangular. The matrix on the right side of (1) is stored in the first K rows of the systolic array. This forms the frozen network.

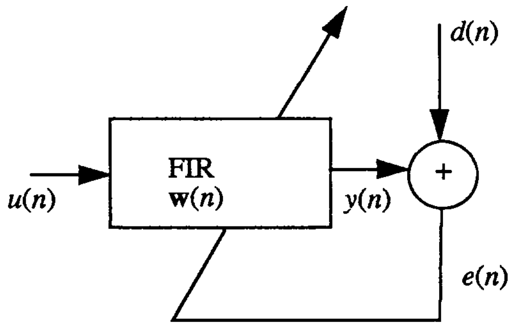

Figure 1. Adaptive FIR Filter

In operation, the input data are first processed by the frozen network to produce $\left[\hat{\mathbf{u}}(n)\ \hat{y}(n)\right]$. This is then fed to the canonical section. The a posteriori residual is produced at the output of the final cell. Note that for proper operation of the array, the input data must be delayed one snapshot per column. The weights can be obtained from the systolic array by backward substitution.

Detailed derivation of the algorithm and description of the function of the systolic array are given in [7].

3. CONSTRAINT DRIFT

As mentioned above, the accumulation of rounding errors may push the computed weights far off the constraint plane. In [3], Frost studied the constraint drift problem of the linearly constrained LMS algorithm by making a geometrical interpretation. Here, we follow a similar approach to analyze the LCQRD-RLS algorithm.

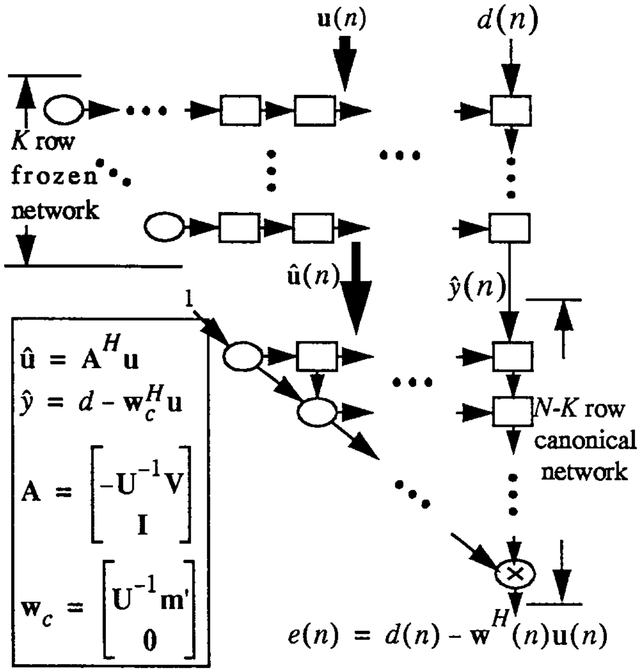

Figure 2. Systolic Array for Linearly Constrained Least-Squares Adaptive Filtering

Using the definitions in Figure 2, we can write the weight vector at iteration n as the sum of two orthogonal components as follows

$$\begin{aligned}
\mathbf{w}(n) &= \mathbf{A}\mathbf{w}_b(n) + \mathbf{w}_c \\
&= \begin{bmatrix} -\mathbf{U}^{-1}\mathbf{V} \\ \mathbf{I} \end{bmatrix} \mathbf{w}_b(n) + \begin{bmatrix} \mathbf{U}^{-1}\mathbf{m}' \\ 0 \end{bmatrix} \\
&= \left(\begin{bmatrix} -\mathbf{U}^{-1}\mathbf{V} \\ 0 \end{bmatrix} \mathbf{w}_b(n) + \begin{bmatrix} \mathbf{U}^{-1}\mathbf{m}' \\ 0 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ \mathbf{I} \end{bmatrix} \mathbf{w}_b(n) \\
&= (\mathbf{B}\mathbf{w}_b(n) + \mathbf{w}_c) + \mathbf{w}'_b(n)
\end{aligned} \qquad (2)$$

where $\mathbf{w}_b(n)$ is the unconstrained weight vector implied by the canonical section.

When finite precision arithmetic is used, rounding errors are inevitably introduced. Firstly, there are rounding errors in the stored values of the pre-computed frozen network. These introduce a fixed perturbation in the matrix A and the vector $\mathbf{w}_c$, and can be interpreted as a perturbation of the imposed constraints. Secondly, the preprocessing of the input data in the frozen network to compute $\hat{\mathbf{u}}(n)$ and $\hat{y}(n)$

will introduce additional rounding errors which can be viewed as the result of applying a perturbation on the input data. Thus the weight vector derived from the systolic array implementation of the LCQRD-RLS algorithm corresponds to the solution of problem (P.1) with perturbed constraint matrix, perturbed constraint vector and perturbed input data. The important point to note here is that the weight vector always satisfies the perturbed constraints exactly.

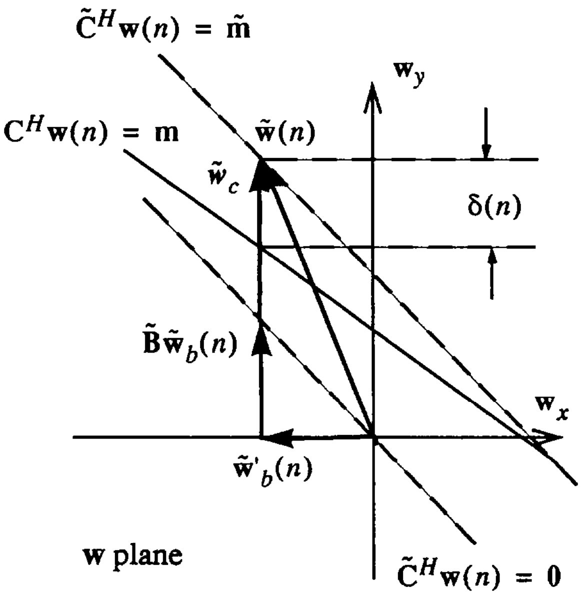

Figure 3. Geometrical Interpretation

Equation (2) and the above discussions can be visualized in Figure 3. $\tilde{x}$ denotes the computed quantity of x. The axes w_x represents the second term in equation (2) while w_y represents the first term. $C^H w(n) = m$ are the original constraints. However, due to rounding errors, the constraints that the computed weight vector actually satisfies are given by $\tilde{C}^H w(n) = \tilde{m}$. This results in the computed weight vector $\tilde{w}(n)$ deviating from the original constraints by $\delta(n)$.

From Figure 3 we see that the deviation of the computed weight vector from the imposed constraints depends on how close the perturbed constraints are to the original ones, and how large is the vector $\tilde{w}_b(n)$. To keep the perturbed constraints close to the imposed ones, we can simply use more bits in the frozen network. For $d(n)$ to be bounded, we require $\tilde{w}_b(n)$ to be bounded which is the case if the input data $u(n)$ do not approach zero asymptotically [5].

4. SIMULATION STUDY

In this section, we present the results of a simulation study which highlights the superior numerical performance of the LCQRD-RLS algorithm. In this study, the LCQRD-RLS and the LCFLS of [6] are both constrained to have unit response at frequencies 0.2π and 0.5π. The input signal consists of 3 sinusoids with unit amplitude at frequencies 0.2π, 0.5π and 0.325π, and a zero mean additive noise with variance 0.1. The desired signal $d(n)$ is set to zero for all n.

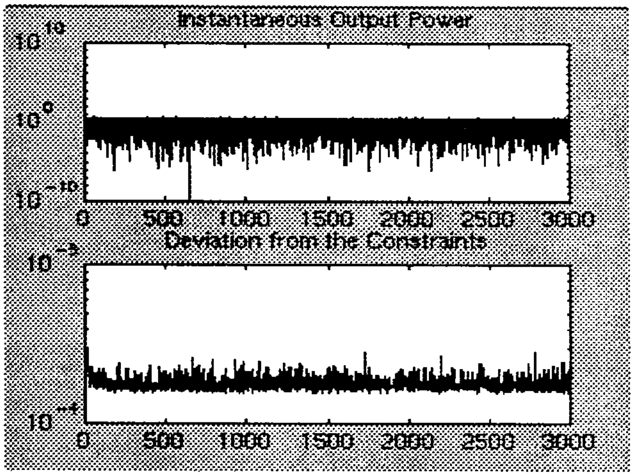

Figure 4. Constraint Drift -- LCQRD-RLS

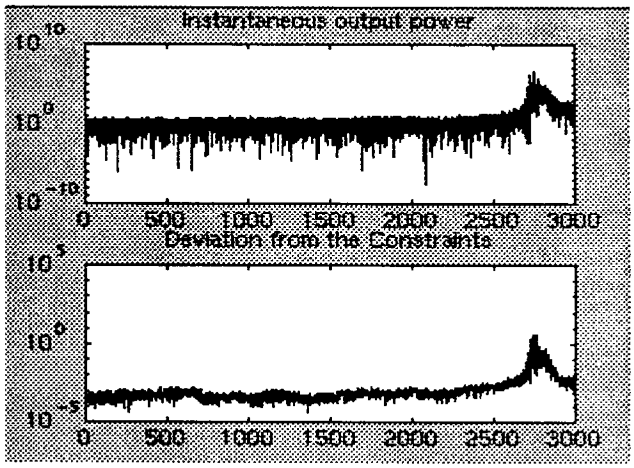

Figure 5. Constraint Drift -- LCFLS

For the systolic array, the precomputation of the frozen network and the backward substitution to extract the weight vector were carried out in full precision (1-bit sign, 11-bit exponent and 52-bit mantissa). All precomputed quantities in the LCFLS were also calculated with full precision. In contrast the computations in the recursive updates were carried out with only 10 bits in the mantissa.

Figure 4 presents the instantaneous output power of the systolic array and the deviation of the corresponding weight vector from the original constraints (defined as $\|m - C^T w\|$). No sign of divergence was observed in this figure.

For comparison, the same data were applied to the LCFLS and the results are displayed in Figure 5. As can be seen, the weight vector of the LCFLS suddenly deviates significantly from the constraints at about iteration 2700. This is accompanied by a sudden increase in output power. Detailed examination of the algorithm revealed that this divergent behavior is not caused by explosive divergence [4] but rather is due to the update of the Q matrix of this algorithm [6] becoming unstable numerically. This Q matrix is used to reinforce the constraints in the LCFLS algorithm.

5. CONCLUSIONS

In this paper, we analyzed the constraint drift property of the linearly constrained QRD-RLS algorithm. It is shown that since the computed weights given by this algorithm always satisfy a set of perturbed constraints exactly, so the LCQRD-RLS algorithm will not exhibit constraint drift. We also showed that the LCFLS algorithm is numerically unstable.

References

[1] G. E. Bottomley and S. T. Alexander, "A Theoretical Basis for The Divergence of Conventional Recursive Least Squares Filters", *Proc. ICASSP'89*, Glasgow, Scotland, pp. 908-911, 1989

[2] J. M. Cioffi, "Limited-Precision Effects in Adaptive Filtering", *IEEE Trans. Circuits and Sys.*, vol. CAS-34, no. 7, pp. 821-833, July 1987

[3] O. L. Frost, III, "An Algorithm for Linearly Constrained Adaptive Array Processing", *Proc. IEEE*, vol. 60, no. 8, pp. 926-935, Aug 1972

[4] S. Haykin, *Adaptive Filter Theory*, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 1996

[5] H. Leung and S. Haykin, "Stability of Recursive QRD-LS Algorithms Using Finite-Precision Systolic Array Implementation", *IEEE Trans. ASSP*, vol. 37, no. 5, pp. 760-763, May 1989

[6] L. S. Resende, J. M. T. Romano and M. G. Bellanger, "A Fast Least-Squares Algorithm for Linearly Constrained Adaptive Filtering", *IEEE Trans. Signal Processing*, vol. 44, no. 5, pp. 1168 - 1174, 1996

[7] T. J. Shepherd and J. G. McWhirter, "Adaptive Systolic Beamformer", in S. Haykin, J. Litva, and T. J. Shepherd (eds.), *Radar Array Processing*, pp. 153-247, Springer-Verlag Berlin, 1993

[8] B. D. Van Veen and K. M. Buckley, "Beamforming: A Versatile Approach to Spatial Filtering", *IEEE ASSP Mag.*, pp. 4-24, April 1988

[9] M. H. Verhaegen, "Round-off Error Propagation in Four Generally-applicable, Recursive, Least-squares Estimation Schemes", *Automatica*, vol. 25, no. 3, pp. 437-444, 1989