CONSTRAINED QUASI-NEWTON ALGORITHM FOR CDMA MOBILE COMMUNICATIONS

M. L. R. de Campos,[1] *S. Werner,*[2] *J. A. Apolinário Jr.,*[1] *and T. I. Laakso*[2]

[1]Instituto Militar de Engenharia
Departamento de Engenharia Elétrica
Praça Gal. Tibúrcio, 80
22.290-270, Rio de Janeiro, BRAZIL
campos@aquarius.ime.eb.br
apolin@aquarius.ime.eb.br

[2]Helsinki University of Technology
Laboratory of Telecommunications Technology
Espoo, FINLAND
Stefan.Werner@hut.fi
Timo.Laakso@hut.fi

Abstract—*This work presents the derivation of the constrained version of a Quasi-Newton (QN) algorithm. Application of such algorithm to the problem of direct-sequence code-division multiple access (DS-CDMA) is investigated through simulations. The algorithm is compared to other constrained adaptation algorithms known in the literature in terms of convergence rate for a similar misadjustment level. Two scenarios describing synchronous and asynchronous transmission are used in simulations of interference suppression in DS-CDMA communication, showing the great capabilities of the algorithm in such applications.*

I. INTRODUCTION

Multiuser interference is the main problem in high-capacity CDMA systems. Efficient algorithms are needed especially for CDMA mobile reception where complexity issue is of importance. Multiuser interference cancellation algorithms [1] are in general too complex, and furthermore, codes related to undesired users may not be known. A promising technique is adaptive multiuser detection, where constrained adaptive filters can be used for interference cancellation [2]–[6].

Two traditional methods for implementing constrained adaptive filters are the approach proposed by Frost [7] and the generalized side-lobe canceller (GSC) proposed by Griffiths and Jim [8]. In the GSC the constrained optimization problem is solved by a structure which utilizes an unconstrained adaptation algorithm. Although both techniques are widely used due to the simplicity of the least mean squares (LMS) algorithm, performance is strongly dependent on the eigenvalue spread of the input-signal autocorrelation matrix or, in other words, performance degrades if the input signal is highly correlated. An alternative approach may be to use the least-squares technique, as proposed in [9]. The drawback, in this case, arises when stability of the least-squares algorithm may not be guaranteed in finite-precision arithmetic implementation [10].

An algorithm which presents convergence speed comparable to that of the recursive least squares (RLS) algorithm, but is guaranteed to be stable even under high input-signal correlation and fixed-point short-wordlength arithmetic is the Quasi-Newton (QN) algorithm [10]. In this paper, we show the extension of the QN algorithm to the constrained optimization problem.

This paper is organized as follows: Section II presents a description of the constrained adaptation algorithm proposed by Frost [7] and the extension to its normalized version with variable step-size [3]. In Section III the constrained QN algorithm is derived and Section IV shows simulation results in typical DS-CDMA applications. Conclusions are summarized in Section V.

II. CONSTRAINED ADAPTIVE FILTERING

In this section we briefly describe two constrained algorithms, the classical Frost algorithm [7] and the newly developed constrained version of the NLMS [3]. Both algorithm solves the optimization problem $\hat{\mathbf{w}} = \arg \min_{\mathbf{w}}(\mathbf{u}^T\mathbf{w})$ subjected to the constraint equation $\mathbf{C}^T\mathbf{w} = \mathbf{F}$.

This work is part of a research project of the Institute of Radio Communication (IRC) funded by Technology Development Center (TEKES), NOKIA Research Center, Telecom Finland, The Helsinki Telephone Company, Finland, CAPES, CNPq, and FAPERJ, Brazil.

0-7803-5030-8/98/$10.00 ©1998 IEEE

A. Frost algorithm

The Frost algorithm [7] was originally derived for adaptive array processing. It is an LMS-type algorithm and the idea is to ensure the constraints at each iteration. Instead of using the mean-squared error, Frost used the unconstrained Langrangian as the cost function for deriving the LMS algorithm. The resulting algorithm has been widely used in for example array processing.

The constrained algorithm can be written as

$$\mathbf{w}(n) = \mathbf{P}\left[\mathbf{w}(n-1) - \mu e(n)\mathbf{u}(n)\right] + \mathbf{g}$$

$$e(n) = \mathbf{u}^T\mathbf{w}(n-1)$$

$$\mathbf{P} = \mathbf{I} - \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T$$

$$\mathbf{g} = \mathbf{C}(\mathbf{C}^T\mathbf{C})^{-1}\mathbf{F}$$

$$\mathbf{w}(0) = \mathbf{F}$$

where μ is the step size parameter, $\mathbf{C}$ is the constraint matrix, $\mathbf{F}$ is vector of constraint values, $\mathbf{P}$ is a projection matrix and $\mathbf{g}$ is fixed vector satisfying the constraint.

The expression inside the brackets of vector-update is the unconstrained LMS update of the weight vector. In general, this update does not lie in the constraint plane. In order to move the unconstrained update back onto the constraint plane, it is first projected onto the constraint subspace by the matrix $\mathbf{P}$, i.e., all components perpendicular to the plane $\mathbf{C}^T\mathbf{w} = 0$ are removed. Finally, the vector is moved back to the constraint plane by adding the fixed vector $\mathbf{g}$. The updated weight vector satisfies the constraints within the numerical precision used in the implementation.

B. Constrained NLMS

The constrained NLMS was derived in [7] the algorithm is given by

$$\mathbf{w}(n) = \mathbf{P}\left[\mathbf{w}(n-1) - \mu\frac{e(n)}{\mathbf{u}(n)\mathbf{P}\mathbf{u}(n)}\mathbf{u}(n)\right] + \mathbf{g}$$

where $\mathbf{P}$ and $\mathbf{g}$ are given by the same equations as in the Frost algorithm. The expression inside the brackets is just the unconstrained NLMS update for the rotated vector $\bar{\mathbf{u}} = \mathbf{P}\mathbf{u}$. This follows directly from the property of the projection matrix $\mathbf{P}^T\mathbf{P} = \mathbf{P}$. The constrained version of the NLMS offers faster convergence rate compared to the Frost algorithm. However, as with the unconstrained NLMS it suffers from a high misadjustment. In order to get a tradeoff between fast convergence and low misadjustment one may use a time-varying step-size. For example, the optimal step-sequence derived in [3] can be used. The step-sequence is given by (see [3] for details)

$$\mu(n) = \mu(n-1)\frac{1 - \frac{\mu(n-1)}{K}}{1 - \frac{\mu^2(n-1)}{K}}$$

where K is the number of coefficients in the adaptive filter.

III. QN ALGORITHM

Given an input-signal vector $\mathbf{u}(n)$ and a desired signal $d(n)$, the conventional QN algorithm [10] may be implemented as shown in Table I. The initial value $\mathbf{R}^{-1}(0)$ can be any positive definite matrix, usually chosen as $\mathbf{R}^{-1}(0) = \gamma \cdot \mathbf{I}$ with $\gamma > 0$, and α is a positive constant used to control speed of convergence and misadjustment.

TABLE I
THE UNCONSTRAINED QN ALGORITHM

$$e(n) = d(n) - \mathbf{w}^T(n-1)\mathbf{u}(n)$$

$$\mathbf{t}(n) = \mathbf{R}^{-1}(n-1)\mathbf{u}(n)$$

$$\tau(n) = \mathbf{u}^T(n)\mathbf{t}(n)$$

$$\mu(n) = \frac{1}{2\tau(n)}$$

$$\mathbf{R}^{-1}(n) = \mathbf{R}^{-1}(n-1) + \frac{[\mu(n) - 1]}{\tau(n)}\mathbf{t}(n)\mathbf{t}^T(n)$$

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \alpha\frac{e(n)}{\tau(n)}\mathbf{t}(n)$$

This algorithm is a normalized Newton-type algorithm which employs an estimate of the inverse of the input-signal autocorrelation matrix that is robust and remains positive definite (PD) even for highly correlated input signals and short wordlength arithmetic. Convergence in the mean and in the mean-squared of the coefficient vector for normalized Newton-type algorithms is guaranteed for $0 < \alpha < 2$ provided that $\mathbf{R}^{-1}(n)$ is PD [11].

Derivation of the constrained version of the QN algorithm may be carried out in the same way as the constrained RLS algorithm proposed by Resende et al. [9]. A deterministic objective function minimized at each iteration by the QN algorithm is constructed and modified to incorporate the constraint via the method of Lagrange multipliers [10]:

$$\xi(n) = \sum_{i=1}^{n} [1 - \mu(i)] \times \left[\frac{d(i) - \mu(i)\mathbf{u}^T(i)\mathbf{w}(i-1)}{1 - \mu(i)} - \mathbf{u}^T(i)\mathbf{w}(n)\right]^2 + \frac{1}{2}\mathbf{w}^T(n)\mathbf{R}(0)\mathbf{w}(n) + \mathcal{L}^T\left[\mathbf{C}^T\mathbf{w}(n) - \mathbf{F}\right] \quad (1)$$

where $\mathcal{L}$ is the Lagrange multiplier, and $\mathbf{C}^T\mathbf{w}(n) = \mathbf{F}$ is the constraint.

Minimization of (1) is achieved for

$$\mathbf{w}(n) = \mathbf{w}(n-1) + 2e(n)\mathbf{T}(n)\mathbf{R}^{-1}(n)\mathbf{u}(n) + \mathbf{m}(n)\left[\mathbf{F} - \mathbf{C}^T\mathbf{w}(n-1)\right] \quad (2)$$

where

$$\mathbf{T}(n) = \mathbf{I} - \mathbf{R}^{-1}(n)\mathbf{C}\left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1}\mathbf{C}^T$$

and

$$\mathbf{m}(n) = \mathbf{R}^{-1}(n)\mathbf{C}\left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1}$$

The last term on the right-hand side of (2) must be maintained in finite-precision-arithmetic implementation in order to prevent accumulation of quantization errors [7]. Details of the derivation are provided in Appendix A.

Computational complexity of the algorithm is comparable to that of the constrained RLS algorithm derived in [9] and the only significant difference between the two algorithms is in the estimate of the input-signal autocorrelation matrix.

IV. SIMULATION RESULTS

Example A: Linearly-Constrained Mini-mum-Variance Filtering of Sinusoids

In this example we consider the received signal consisting of three sinusoids in white noise. The QN, NLMS, RLS, and Frost's algorithms were compared. The example was taken from [9] and the equations are reproduced here for clarity.

The received signal is

$$u(n) = \sin(0.3n\pi) + \sin(0.325n\pi) + \sin(0.7n\pi) + r(n)$$

where $r(n)$ is white noise with power chosen such that the SNR is 40 dB.

The filter is constrained to pass components at frequencies 0.1 rad/s and 0.25 rad/s undistorted. This results in four constraints such that

$$\mathbf{C}^T = \begin{bmatrix} 1 & \cos(0.2\pi) & \ldots & \cos[(N-1)0.2\pi] \\ 1 & \cos(0.5\pi) & \ldots & \cos[(N-1)0.5\pi] \\ 1 & \sin(0.2\pi) & \ldots & \sin[(N-1)0.2\pi] \\ 1 & \sin(0.5\pi) & \ldots & \sin[(N-1)0.5\pi] \end{bmatrix}$$

and

$$\mathbf{F}^T = [1 \quad 1 \quad 0 \quad 0]$$

The optimum filter coefficient vector is given by

$$\mathbf{w}_{\text{opt}} = \begin{bmatrix} -0.4132 \\ 0.2964 \\ -1.0324 \\ -0.2535 \\ -0.5921 \\ -0.7046 \\ -0.6467 \\ -0.8854 \\ 0.2681 \\ -0.7307 \\ -0.0580 \end{bmatrix}$$

All algorithms tested in this example were tuned for best convergence rate and a common misadjustment level after convergence. For the NLMS and Frost's algorithm $\mu = 0.1$, for the QN algorithm $\alpha = 1$, and for the RLS algorithm $\lambda = 0.99$.

The Euclidean norm of the coefficient-error vectors of all algorithms are depicted in Figure 1 as a result of an ensemble average of 200 simulations.

We can clearly verify the superior performance of the QN algorithm as compared to the gradient-based algorithms, achieving a convergence rate closer to that of the RLS algorithm. Therefore, its usefulness is clear in applications where the RLS algorithm presents divergence and gradient-based algorithms do not converge as fast as required.

Example B: Interference Suppression in a CDMA Communication System

In this example we apply the constrained QN algorithm to the case of single-user detection in a DS-CDMA mobile communication system. The case of synchronous transmission will be treated in Section B.1, and asynchronous transmission in Section B.2

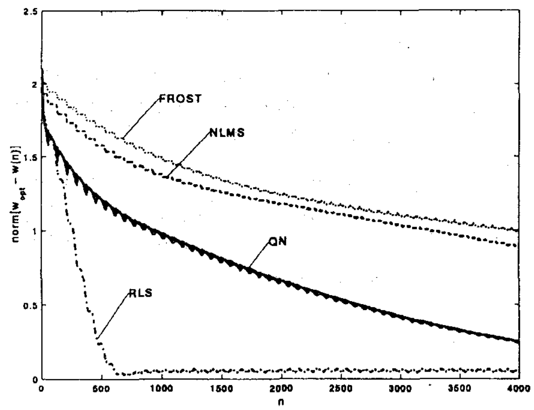

Figure 1: Coefficient-vector deviation for Example A (sum of sinusoids).

A. Synchronous Transmission

We assume synchronous transmission of K users. The received continuous-time signal is given by:

$$u(t) = \sum_{n=-\infty}^{\infty} \sum_{i=1}^{K} A_i b_i(n) s_i(t - nT_b) + r(t)$$

where for the ith user, A_i is the amplitude, $b_i(n) \in \{-1, 1\}$ is the *nth* information bit with duration T_b, $r(t)$ is additive white Gaussian noise, and $s_i(t)$ is a spreading sequence of the form

$$s_i(t) = \sum_{j=1}^{G} s_i^j p(t - jnT_c)$$

where $G = T_b/T_c$ is the number of chips per bit (spreading gain), $s_i^j \in \{-1, 1\}$, and $p(t)$ is the chip waveform, assumed rectangular with unit energy and duration T_b, i.e., $s_i(t) = 0$ for $t \notin [0, T_b]$.

The received signal is passed through a chip-matched filter and is sampled at chip rate. If the samples during the *nth* bit interval are collected into vectors

$$\mathbf{u}^T(n) = [u(nG+1) \quad \dots \quad u(nG+G)]$$

and

$$\mathbf{r}^T(n) = [r(nG+1) \quad \dots \quad r(nG+G)]$$

we can write the received discrete-time signal as

$$\mathbf{u}(n) = \mathbf{SAb}(n) + \mathbf{r}(n)$$

where $\mathbf{S} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_K]$ is the spreading matrix containing the sampled spreading sequences of the users, $\mathbf{b}(n) = [b_1(n) \quad b_2(n) \quad \dots \quad b_K(n)]$, $\mathbf{A} = \text{diag}[A_1 \quad A_2 \quad \dots \quad A_K]$, and $\mathbf{r}(n)$ is the sampled noise sequence.

The received signal vector is then passed through the receiver filter $\mathbf{w}(n-1)$ producing the decision variable $z(n) = \mathbf{u}^T(n)\mathbf{w}(n-1)$. The bit estimate corresponding to the user of interest is formed by taking the sign of $z(n)$, i.e., $\hat{b}_1(n) = \text{sign}[z(n)]$.

The filter coefficients were obtained as described in the previous section under the constraint that the desired user code is not distorted throughout the process. For this example, $\mathbf{C} = \mathbf{s}_1$, $\mathbf{F} = 1$, and $d(n) = 0$.

In order to test the algorithm we assumed a system of $K = 10$ users with spreading sequences taken as Gold codes of length 15 [6]. Signal-to-noise ratio (SNR) for user number one was set to 15 dB and relative power of interfering users are set to 15 dB, i.e., $10\log(P_i/P_1) = 15$.

Figure 2 shows the learning curve for the QN algorithm with $\alpha = 0.1$ compared to those of the normalized LMS (NLMS) algorithm with the time-varying step-size described in [3] and the algorithm originally proposed by Frost in [7] with step-size $\mu = 3 \cdot 10^{-4}$.

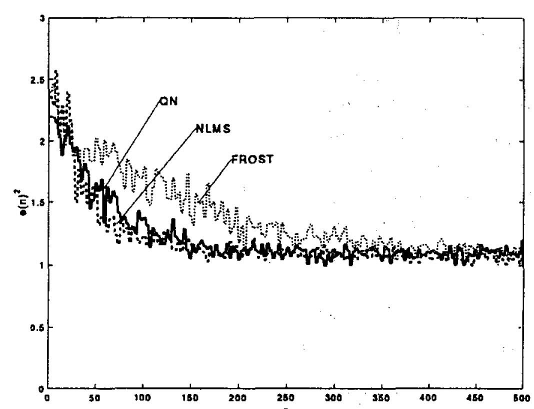

Figure 2: Mean-squared error for Example B.1 (synchronous CDMA).

From the figure we can see that QN and NLMS algorithms present similar convergence properties in this example. They converge in the mean to their steady state in approximately 160 iterations, whereas the Frost algorithm takes over 450 iterations.

B. Asynchronous Transmission

The received signal for asynchronous transmission is given by

$$u(t) = \sum_{n=-\infty}^{\infty} \sum_{i=1}^{K} A_i b_i(n) s_i(t - \tau_i - nT_b) + r(t)$$

where τ_i is the propagation delay. We also assumed synchronization with the user of interest, i.e., $\tau_1 = 0$, and interference comes from two consecutive symbols from the other users [2]. The signal is passed through a chip-matched filter and is sampled at chip rate.

The system used in the simulations was the same as that in $B.1$, i.e., there were 10 users using Gold codes of length 15. SNR for user number one was 15 dB and the relative power of the interfering users were 15 dB.

The step-size used for the QN algorithm was $\alpha = 0.4$ and for Frost's algorithm $\mu = 10^{-3}$. The NLMS algorithm used the variable step-size.

As can be seen from Figure 3, the QN algorithm outperformed both NLMS and Frost's algorithms as far as convergence speed was concerned.

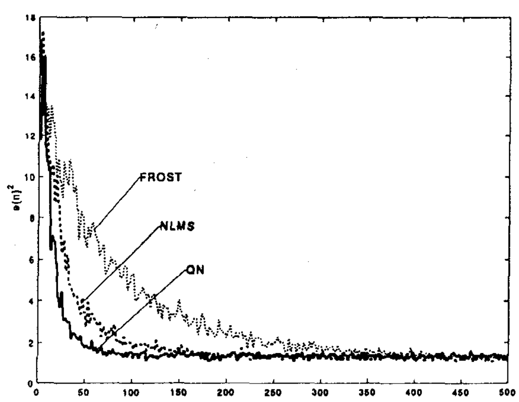

Figure 3: Mean-squared error for Example B.2 (asynchronous CDMA).

V. CONCLUSIONS

This paper introduced the constrained Quasi-Newton (QN) algorithm based on the so-called Frost structure. The conventional QN algorithm which can be used in the GSC structure was briefly presented and the derivation of the constrained version was derived.

The algorithms were applied to CDMA mobile reception and the simulations showed faster convergence rate when compared to the LMS Frost structure with similar misadjustment. It was shown that for a high input-signal correlation, the QN algorithm outperforms the LMS and NLMS algorithms.

APPENDIX A

In this section the derivation of the constrained version of the QN algorithm is presented. By differentiating $\xi(n)$ in (1) with respect to each element of $\mathbf{w}(n)$ and setting the result equal to zero, we have

$$\mathbf{R}(n)\mathbf{w}(n) = \bar{\mathbf{p}}(n)$$

where

$$\begin{aligned}\mathbf{R}(n) &= \sum_{i=1}^{n} 2\left[1 - \mu(i)\right] \mathbf{u}(i)\mathbf{u}^T(i) + \mathbf{R}(0) \\ &= \mathbf{R}(n-1) + 2\left[1 - \mu(n)\right] \mathbf{u}(n)\mathbf{u}^T(n)\end{aligned} \quad \text{(A.1)}$$

and

$$\begin{aligned}\bar{\mathbf{p}}(n) &= \mathbf{p}(n) + \mathbf{C}\mathcal{L} \\ &= \sum_{i=1}^{n} 2\left[d(i) - \mu(i)\mathbf{u}^T(i)\mathbf{w}(i-1)\right]\mathbf{u}(i) \\ &\quad + \mathbf{C}\mathcal{L} \\ &= \mathbf{p}(n-1) + \mathbf{C}\mathcal{L} \\ &\quad + 2\left[d(n) - \mu(n)\mathbf{u}^T(n)\mathbf{w}(n-1)\right]\mathbf{u}(n)\end{aligned} \quad \text{(A.2)}$$

with

$$\mathcal{L} = \left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1}\left[\mathbf{F} - \mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{p}(n)\right]$$

Therefore,

$$\begin{aligned}\mathbf{w}(n) &= \mathbf{R}^{-1}(n)\Big\{\mathbf{p}(n) + \mathbf{C}\left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1} \\ &\quad \times \left[\mathbf{F} - \mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{p}(n)\right]\Big\} \\ &= \mathbf{T}(n)\mathbf{R}^{-1}(n)\mathbf{p}(n) + \mathbf{m}(n)\mathbf{F}\end{aligned} \quad \text{(A.3)}$$

where

$$\mathbf{T}(n) = \mathbf{I} - \mathbf{R}^{-1}(n)\mathbf{C}\left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1}\mathbf{C}^T$$

and

$$\mathbf{m}(n) = \mathbf{R}^{-1}(n)\mathbf{C}\left[\mathbf{C}^T\mathbf{R}^{-1}(n)\mathbf{C}\right]^{-1}$$

The inverse of the estimate of the input-signal autocorrelation matrix can be obtained by applying the matrix inversion lemma [12] to (A.1):

$$\mathbf{R}^{-1}(n) = \mathbf{R}^{-1}(n-1) + \frac{[\mu(n)-1]}{\tau(n)}\mathbf{t}(n)\mathbf{t}^T(n) \tag{A.4}$$

In order to obtain a recursive solution, we may substitute (A.2) and (A.4) in (A.3) to find, after some algebraic manipulation,

$$\begin{aligned}\mathbf{w}(n) &= \mathbf{w}(n-1) + 2e(n)\mathbf{T}(n)\mathbf{R}^{-1}(n)\mathbf{u}(n) \\ &\quad + \mathbf{m}(n)\left[\mathbf{F} - \mathbf{C}^T\mathbf{w}(n-1)\right]\end{aligned}$$

which is the updating equation for the constrained QN algorithm. This formula is similar to that obtained by Resende et al. in [9] with the main difference being the estimate of the input-signal autocorrelation matrix.

REFERENCES

[1] M. K. Varansi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Trans. Communications*, vol. 38, pp. 509–519, Apr. 1990.

[2] M. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Trans. Information Theory*, vol. 41, pp. 944–960, July 1995.

[3] J. A. Apolinário Jr., S. Werner, T. Laakso, and P. S. R. Diniz, "Constrained normalized adaptive filtering for CDMA mobile communications," *accepted by European Signal Processing Conference*, Rhodes, Greece, Sept. 1998.

[4] S. Werner, T. Laakso, and J. Lilleberg, "Multiple-antenna receiver for CDMA mobile reception," *accepted by International Conference on Communications*, Atlanta, USA, June 1998.

[5] J. B. Schodorf and D. B. Williams, "A blind adaptive interference cancellation scheme for CDMA systems," in *Proc. Twenty-Ninth Asilomar Conference on Signals, Systems, & Computers*, Pacific Grove, USA, 1995.

[6] S. C. Park and J. F. Doherty, "Generalized projection algorithm for blind interference suppression in DS/CDMA communications," *IEEE Trans. Circuits and Systems — Part II*, vol. 44, pp. 453–460, June 1997.

[7] O. L. Frost III, "An algorithm for linearly constrained adaptive array processing," *Proceedings of the IEEE*, vol. 60, pp. 926–935, Aug. 1972.

[8] L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas and Propagation*, vol. AP-30, pp. 27–34, Jan. 1982.

[9] 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, pp. 1168–1174, May 1996.

[10] M. L. R. de Campos and A. Antoniou, "A new quasi-Newton adaptive filtering algorithm," *IEEE Trans. Circuits and Systems — Part II*, vol. 44, pp. 924–934, Nov. 1997.

[11] P. S. R. Diniz, M. L. R. de Campos, and A. Antoniou, "Analysis of LMS-Newton adaptive filtering algorithms with variable convergence factor," *IEEE Trans. Signal Processing*, vol. 43, pp. 617–627, Mar. 1995.

[12] S. Haykin, *Adaptive Filter Theory*. New Jersey: Prentice-Hall, Englewood-Cliffs, 2nd ed., 1991.