Correspondence

Recursive Least Squares Constant Modulus Algorithm for Blind Adaptive Array

Yuxin Chen, Tho Le-Ngoc, Benoit Champagne, and Changjiang Xu

Abstract—**We consider the problem of blind adaptive signal separation with an antenna array, based on the constant modulus (CM) criterion. An approximation to the CM cost function is proposed, which allows the use of the recursive least squares (RLS) optimization technique. A novel RLS constant modulus algorithm (RLS-CMA) is derived, where the modulus power of the array output can take on arbitrary positive real values (i.e., fractional values allowed). Simulations are performed to compare the performance of the proposed RLS-CMA to other well-known algorithms for blind adaptive beamforming. Results indicate that the RLS-CMA has a significantly faster convergence rate and better tracking ability.**

Index Terms—**Blind adaptive beamforming, blind adaptive signal separation, constant-modulus algorithm (CMA), recursive least-squares (RLS), wireless communications.**

I. Introduction

The problem of detecting and extracting communications signals from dense interference environments is particularly important in the design of modern wireless communications systems. Adaptive beamforming techniques, used in connection with antenna arrays, provide a potential solution to this problem by forming high-gain beams in the directions of arrival (DOA) of the signals of interest and, ideally, directing nulls in the DOAs of the interferences. Adaptive beamforming techniques often make use of a known training sequence. However, this consumes a large amount of the available spectrum, especially when the users move rapidly or the channel variation is severe.

To overcome this limitation, blind beamforming techniques based on the so-called constant modulus (CM) approach [1]–[7] have been widely used. Constant modulus algorithms (CMAs) for beamforming exploit the low modulus fluctuation exhibited by most communications signals to extract them from the array output. They are typically based on minimizing the mean qth power error between the array output, after a modulus nonlinearity, and a fixed real number. For our purpose, we find it convenient to present the CM cost function in the form

$$J(p,q) = E\left[(|y(n)|^p - 1)^q\right] \tag{1}$$

where $y(n)$ denotes the array output, $|\cdot|$ denotes the modulus function, $E[\cdot]$ denotes statistical expectation, and p and q are non-negative parameters. In this correspondence, we will focus on the special case $q = 2$.

Manuscript received February 10, 2003; revised July 1, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zhi Ding.

Y. Chen was with the Department of Electrical and Computer Engineering, McGill University, Montréal, QC, H3A 2A7 Canada. He is now with the Department of Electrical and Computer Engineering, Indiana University Purdue University, Indianapolis, IN 46202 USA (e-mail: yxchen@iupui.edu).

T. Le-Ngoc, B. Champagne, and C. Xu are with the Department of Electrical and Computer Engineering, McGill University, Montréal, QC, H3A 2A7 Canada (e-mail: tho@ece.mcgill.ca; champagne@ece.mcgill.ca; cjxu@tsp.ece.mcgill.ca).

Digital Object Identifier 10.1109/TSP.2004.826167

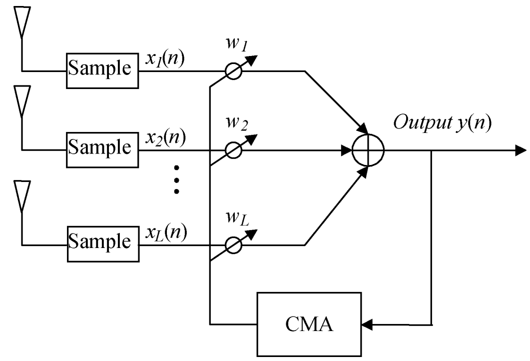

Fig. 1. Adaptive beamforming structure.

A CMA-based beamformer requires no special array geometry, knowledge of the array manifold, or the noise covariance matrix to adapt the array weights. In general, the CM cost function (1) is nonquadratic in the unknown array weights and is optimized iteratively over time via a stochastic gradient descent (SGD) approach. It is well known that SGD methods are quite sensitive to the selected step size and have a slow convergence rate. To overcome this limitation, Agee [3] proposed a least squares constant modulus algorithm (LS-CMA) for the $J(1,2)$ case, which is implemented by using a block-update iterative algorithm. Biedka *et al.* [4] analyzed the convergence behavior of the LS-CMA in interference cancellation applications and pointed out that the block size should be chosen carefully. Another important CM adaptive algorithm is the so-called orthogonalized CMA (O-CMA) [1], which is based on a Gauss–Newton iterative approach.

In this correspondence, we introduce an approximation into the CM cost function $J(p,2)$ that enables the use of the rapidly converging recursive least squares (RLS) algorithm for the array weight adaptation for any non-negative value of p. Simulations are performed to compare the performance of the RLS-CMA, SGD-CMA, LS-CMA, and O-CMA in the application of blind adaptive beamforming of communications signals. Results indicate that the RLS-CMA has the fastest convergence rate. It is of interest to note that the Furukawa *et al.* method [5] corresponds to a special case of our proposed algorithm (i.e. with $p = 1$), although they did not provide any specific interpretation. Here, we establish through simulations that among the above algorithms, the proposed RLS-CMA with $p = 2$ offers the best convergence property.

II. Data Model and CMA Array

A CMA array, with adjustable element weights, is shown in Fig. 1. Consider d independent sources, transmitting complex-valued signals $s_i(t), 1 \le i \le d$ with constant modulus waveforms $(|s_i(t)| = 1)$ in a wireless scenario. The signals are received by an array of L antennas, demodulated to baseband and sampled, resulting in the discrete-time signals $x_l(n), 1 \le l \le L$. The received discrete-time signals from each antenna elements are scaled by a complex weight $w_l(n), 1 \le l \le L$, and summed to form the array output $y(n)$. Referring to Fig. 1, an expression for the array output is given by

$$y(n) = \sum_{l=1}^{L} w_l^* x_l(n) \tag{2}$$

1053-587X/04$20.00 © 2004 IEEE

Authorized licensed use limited to: McGill University. Downloaded on June 2, 2009 at 16:49 from IEEE Xplore. Restrictions apply.

where superscript * denotes complex conjugate. Using vector format to denote the beamformer weights and the signals induced on the antenna elements, i.e.,

$$\mathbf{w} = [w_1, w_2, \cdots, w_L]^T \tag{3}$$

$$\mathbf{x}(n) = [x_1(n), x_2(n), \cdots, x_L(n)]^T \tag{4}$$

where superscript T stands for transpose, the output of the beamformer becomes

$$y(n) = \mathbf{w}^H \mathbf{x}(n) \tag{5}$$

where H denotes the complex conjugate transpose.

The purpose of the adaptive array is to extract the desired signal, say $s_1(t)$, by finding a suitable weight vector. In practical radio environments, multipath fading and interference can cause amplitude fluctuations in the received signals. The objective of the CMA is therefore to restore the array output to a constant envelope signal on average. This is accomplished by adjusting the weight vector $\mathbf{w}$ to minimize the cost function $J(p,q)$, as defined by (1).

For the sake of simplification, a stochastic gradient descent (SGD) algorithm is generally employed to minimize the cost function $J(p,q)$. Using complex matrix calculus and replacing the statistical expectation in (1) with an instantaneous estimation, the recursive update equation of the SGD-CMA for the special case when p is an even integer and $q = 2$ is obtained as follows:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu\left(\frac{p}{2}\right)\left[|y(n)|^p - 1\right]|y(n)|^{p-2}\, y^*(n)\mathbf{x}(n) \tag{6}$$

where μ is a positive step size parameter. The SGD-CMA (6) is used as a benchmark in the simulations reported in Section IV.

III. RLS-CMA Algorithm

In the above SGD-CMA (6), the step size μ should be carefully selected. A small step size will lead to slow convergence rate, whereas a large step size will result in the adapted weight oscillation. It is well known that adaptation algorithms based on the RLS technique generally have faster convergence rate than SGD algorithms. However, the cost function $J(p,q)$ (1) is nonquadratic in the array weights and cannot be solved by the standard RLS algorithm. Below, we introduce an approximation to $J(p,2)$, leading to a modified CM cost function, which in turns enables the use of the RLS algorithm.

Replacing the statistical expectation operator in (1) with an exponentially weighted time average sum and setting $q = 2$ yields

$$J = \sum_{k=1}^{n} \lambda^{n-k}\left(\left|\mathbf{w}^H(n)\mathbf{x}(k)\right|^p - 1\right)^2 \tag{7}$$

where λ is the forgetting factor, and $0 < \lambda \le 1$. Due to the presence of the modulus nonlinearity in (7), we immediately note that the cost function of interest here is nonquadratic in the array weight vector $\mathbf{w}(n)$, which prevents the application of the well-known recursive least-square optimization techniques. To overcome this limitation, we first rewrite (7) as

$$J = \sum_{k=1}^{n} \lambda^{n-k}\left(\mathbf{w}^H(n)\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{w}(n)\left|\mathbf{x}^H(k)\mathbf{w}(n)\right|^{p-2} - 1\right)^2. \tag{8}$$

Next, we note that for stationary or slowly varying signal environments, the difference between $\mathbf{x}^H(k)\mathbf{w}(n)$ and $\mathbf{x}^H(k)\mathbf{w}(k-1)$ is usually small when k is close to n, whereas larger differences for more distant values of n and k will be attenuated by the memory factor λ^{n-k}. Accordingly, it appears justified to consider the following approximation scheme, that is, replacing $\mathbf{x}^H(k)\mathbf{w}(n)$ by $\mathbf{x}^H(k)\mathbf{w}(k-1)$ in (8). Doing so, we obtain the modified cost function:

$$J' = \sum_{k=1}^{n} \lambda^{n-k}\left|\mathbf{w}^H(n)\left(\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{w}(k-1)\right) \times \left|\mathbf{x}^H(k)\mathbf{w}(k-1)\right|^{p-2} - 1\right|^2. \tag{9}$$

The main advantage of (9) over (8) is that at a given time n, the former is now *quadratic* in the unknown array weights $\mathbf{w}(n)$; the other weight vectors $\mathbf{w}(k-1)$ $(k = 1, \ldots, n)$ entering (9) are indeed available from previous iterations as they can be computed for $1 \le k \le n$ at the time instant n. When optimizing J' in (9) with respect to unknown weight vector $\mathbf{w}(n)$, these other weight vectors are treated as known, constant terms; it is therefore legitimate to view (9) as a quadratic function of $\mathbf{w}(n)$ during the optimization process at a given time n. We have been able to confirm the validity of the approximation $\mathbf{x}^H(k)\mathbf{w}(n) \approx \mathbf{x}^H(k)\mathbf{w}(k-1)$ via computer experiments (see also Section IV). The above approximation is conceptually similar to the ones used by Yang in the derivation of the projection approximation subspace tracking (PAST) algorithm [8] and by Chen *et al.* in the blind beamforming algorithm for cyclostationary signals [9], although the underlying applications in these works bear no direct connection with the present CM approach to signal separation.

Defining $\mathbf{z}(k) = (\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{w}(k-1))|\mathbf{x}^H(k)\mathbf{w}(k-1)|^{p-2}$, (9) can be rewritten more compactly as

$$J' = \sum_{k=1}^{n} \lambda^{n-k}\left|\mathbf{w}^H(n)\mathbf{z}(k) - 1\right|^2. \tag{10}$$

The above modified cost function J', which is an approximation of the original cost function J in (7), can now be solved iteratively for the optimum weight vector $\mathbf{w}(n)$ by using one of the many RLS algorithms available for such purpose in the literature (e.g., [10]).

Here, we solve (10) by using the standard RLS algorithm and call the resulting algorithm RLS-CMA. The complete set of equations necessary for the algorithm is described in Table I. Because the cost function J' (10) is quadratic in the unknown array weights and has the standard least-squares form, other RLS algorithms could be specialized just as well to the present application. For example, the QR-decomposition-based recursive algorithm, which is known to be numerically stable and robust, could be applied here.

TABLE I
RLS-CMA Algorithm

Initialization	$\mathbf{w}(0)=[1,\mathbf{0}_{1\times(L-1)}]^T$, $\mathbf{C}(0)=\delta^{-1}\mathbf{I}_{L\times L}$, δ= small positive constant
Approximation and RLS update (For each iteration *n=1,2,...*)	$\mathbf{z}(n) = \mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}(n-1)\left\|\mathbf{x}^H(n)\mathbf{w}(n-1)\right\|^{p-2}$ $\mathbf{h}(n) = \mathbf{z}^H(n)\mathbf{C}(n-1)$ $\mathbf{g}(n) = \mathbf{C}(n-1)\mathbf{z}(n)/(\lambda + \mathbf{h}(n)\mathbf{z}(n))$ $\mathbf{C}(n) = (\mathbf{C}(n-1) - \mathbf{g}(n)\mathbf{h}(n))/\lambda$ $e(n) = \mathbf{w}^H(n-1)\mathbf{z}(n) - 1$ $\mathbf{w}(n) = \mathbf{w}(n-1) + \mathbf{g}(n)e^*(n)$

The initial weight vector $\mathbf{w}(0)$ is set to $[1, \mathbf{0}_{1\times(L-1)}]^T$, i.e., the radiation pattern is omni-directional. The choice of δ is guided by similar considerations, as in the application of the standard RLS algorithm. Typically, δ is a small positive constant whose specific value

Authorized licensed use limited to: McGill University. Downloaded on June 2, 2009 at 16:49 from IEEE Xplore. Restrictions apply.

is adjusted empirically. The computational complexity of RLS-CMA is $(3L^2 + 6L + P - 2)$ complex multiplies per iteration.

It may be argued that the term $|\mathbf{x}^H(k)\mathbf{w}(k-1)|^{p-2}$ in (9), which results from the quadratic approximation of J', induces additional memory into the time evolution of the RLS-CMA when the modulus power $p \neq 2$. Hence, the convergence rate of RLS-CMA in the case $p = 2$ should be faster than that of the case $p \neq 2$. The simulation results in Section IV support that observation.

In [11], a robust CM array based on fractional lower order statistics (FLOS) was proposed to mitigate impulsive noise at the receiver and, at the same time, to restore the constant modulus character of the transmitted communication signals. The main advantage of this method is its robustness in various noise environments. The parameter p in the FLOS criterion is possibly fractional, and it is therefore easy to implement the latter recursively by using our proposed RLS-CMA technique.

Using Ljung's ordinary differential equation (ODE) framework [12], we have been able to show that the proposed RLS-CMA converges with probability 1 to the same local minimal point as the original CM cost function for the case $p = 2$. Due to lack of space, the results will be presented separately.

IV. Illustrative Simulation Results

A ten-element uniform linear array is employed. The interelement distance is set to half of the carrier wavelength of the desired signals. The performance of the adaptive beamformer is measured by the output signal-to-interference-plus-noise ratio (SINR), which is defined as

$$\text{SINR}_i = \frac{\mathbf{w}^H \mathbf{a}_i r_{s_i s_i} \mathbf{a}_i^H \mathbf{w}}{\mathbf{w}^H \mathbf{R}_I \mathbf{w}} \quad (11)$$

where $r_{s_i s_i}$ is the true power of the ith source, $\mathbf{a}_i$ is the associated transmission vector, and $\mathbf{R}_I$ is the true autocorrelation matrix of the interference (noise and other signals) in the environment.

Example 1: In this example, we compare the initial convergence rate and tracking ability of the proposed RLS-CMA and the SGD-CMA in the case $p = 2$, $q = 2$. To study tracking, we abruptly change the number of sources following the initial convergence period. In all simulations, we assume that all the sources have unit power, and the noise power is $\sigma_n^2 = 0.1$. The phase $\psi_i(n)$, $i \leq d$ of each source $s_i(n) = e^{j\psi_i(n)}$ is independently and uniformly distributed over $[-\pi, \pi]$, where d is the number of sources. In the first 5000 iterations, there are two sources with DOAs $\theta_1 = 10°$ and $\theta_2 = -30°$; in the subsequent 5000 iterations, there are two additional sources with DOAs $\theta_3 = -45°$ and $\theta_4 = 25°$.

In Fig. 2, the forgetting factor of the RLS-CMA is chosen as $\lambda = 0.99$, and the step size of the SGD-CMA is set to $\mu = 0.005$ and $\mu = 0.009$. In the initial convergence phase, the RLS-CMA has a much faster convergence rate than the SGD-CMA. The latter exhibits a two-stage convergence behavior: During the first stage, it quickly approaches a certain error level; however, in the second stage, it takes a much longer time to converge to the minimum MSE. By increasing the step size of the SGD-CMA, the first stage gets shorter, but the misdjustment becomes larger. After a sudden change (i.e., two sources added at iteration 5000), both the RLS-CMA and SGD-CMA can quickly track the change. However, the SGD-CMA has a lower steady-state SINR than the proposed RLS-CMA. This lower SINR becomes more pronounced as the SGD step size increases due to the corresponding increase in misadjustment. Fig. (3a) and (3b) shows the beam patterns of the proposed RLS-CMA and SGD-CMA ($\mu = 5 \times 10^{-3}$) at iterations 5000 and 10 000, respectively. The results indicate that the proposed RLS-CMA provides significantly deeper nulls in the directions of the other interfering sources.

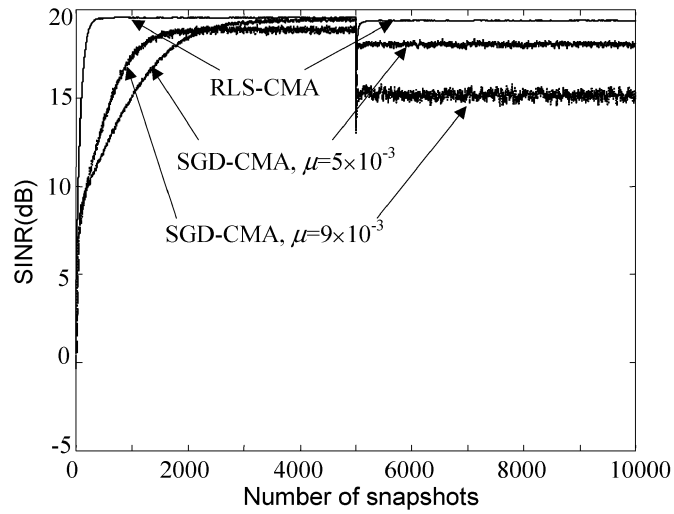

Fig. 2. SINR versus number of iterations for RLS-CMA and SGD-CMA in Example 1.

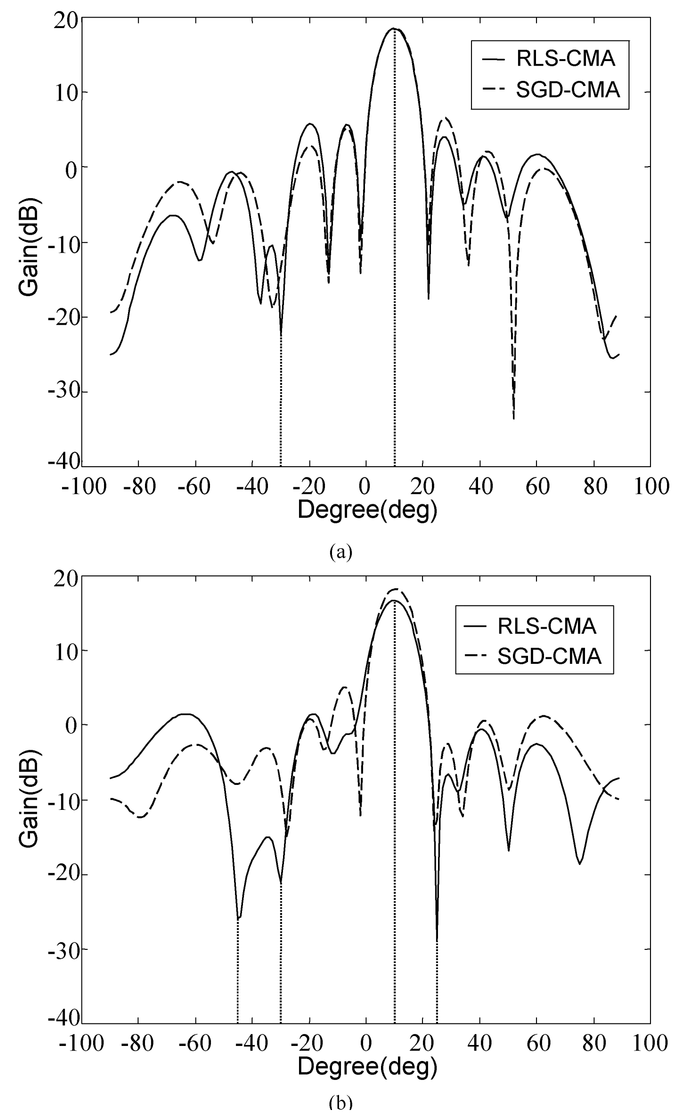

Fig. 3. Beam patterns obtained with the RLS-CMA and SGD-CMA in Example 1. (a) Beam pattern at iteration 5000. (b) Beam pattern at iteration 10 000.

Authorized licensed use limited to: McGill University. Downloaded on June 2, 2009 at 16:49 from IEEE Xplore. Restrictions apply.

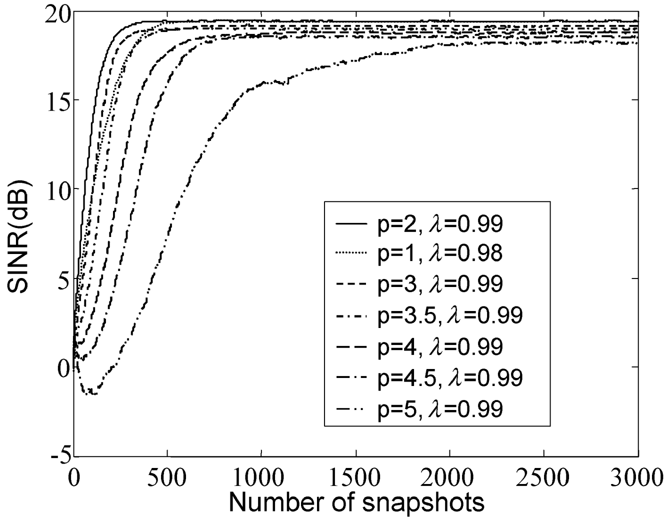

Fig. 4. SINR versus number of iterations for RLS-CMA in Example 2.

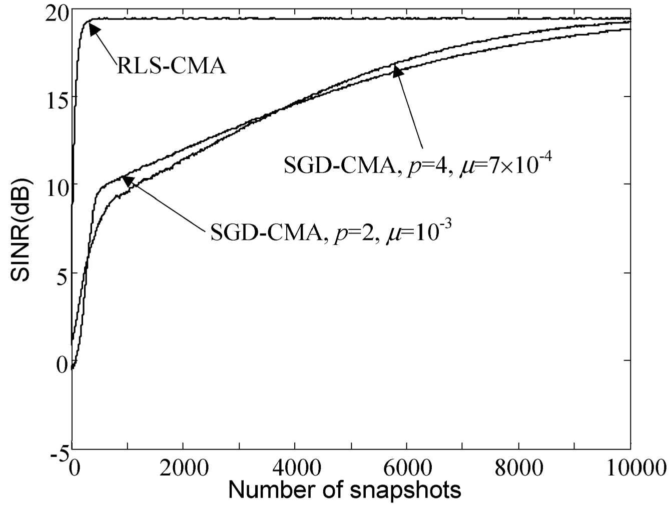

Fig. 5. SINR versus number of iterations for SGD-CMA in Example 2.

Example 2: In this example, we illustrate the effect of using different values of the modulus power p in the RLS-CMA and SGD-CMA. There are two sources with DOAs $\theta_1 = -10^\circ$ and $\theta_2 = 5^\circ$; the other simulation parameters are identical to those in Example 1. Results in this example are averaged over 100 independent runs. In Fig. 4, we compare the learning curves of the RLS-CMA with different p. The forgetting factor λ is set to 0.99, except when $p = 1$, where $\lambda = 0.98$. This ensures that the steady-state SINR for the cases $p = 1$ and $p = 2$ are identical so that a meaningful comparison of convergence rate can be made. As we discussed in Section III, the approximation made to the CM cost function in the case $p \neq 2$ adds additional memory to our RLS-CMA, as compared with the case $p = 2$. Hence, as shown in Fig. 4, the convergence rate of our algorithm degrades when $p \neq 2$.

For the SGD-CMA, the convergence rate is determined by the step size. Under the constraint of a fixed step size, increasing the modulus power p will lead to faster convergence, but the fluctuation in SINR will be larger when the algorithm reaches the steady state. For comparison purposes, different values of the step size were used in our experiments for the different choices of p so that similar convergence rates were observed. Fig. 5 shows the result so obtained. Again, the SGD-CMA is characterized by a two-stage convergence behavior. We also find that there is no significant difference between the learning curves of the SGD-CMA with different p. Once again, we note that the proposed RLS-CMA has a much faster convergence rate than the SGD-CMA.

Example 3: In this example, we make further comparisons of the RLS-CMA with the dynamic LS-CMA and the O-CMA, as shown in Fig. 6. The simulation conditions are the same as in Example 2. The parameter p is set to 2. In the dynamic LS-CMA, two different choices of block size are used, i.e., 30 and 60. In the O-CMA, two values of the step size, namely 3.5×10^{-3} and 4.5×10^{-3}, are simulated. SGD-CMA with step size of 10^{-3} is also included for the purpose of comparison. We note that the dynamic LS-CMA with small value of the block size (i.e., 30) has a relatively fast convergence rate. However, the fluctuations in SINR after convergence are significantly larger than with RLS-CMA. With the O-CMA, step-size values larger than 4.5×10^{-3} lead to divergence of the algorithm. The proposed RLS-CMA with parameter p set to 2 has the best convergence performance among all the algorithms tested.

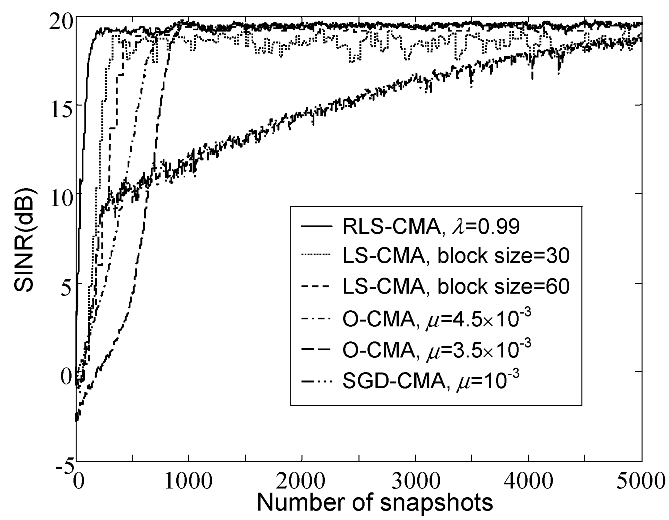

Fig. 6. Comparison of SINR versus number of iterations for RLS-CMA, dynamic LS-CMA, O-CMA and SGD-CMA in Example 3.

V. Conclusions

An approximation to the CM cost function was proposed that enabled the derivation of a novel RLS-CMA for blind adaptive beamforming. Comparative simulation experiments were conducted to investigate the convergence behavior and tracking ability of the new RLS-CMA. The performance of RLS-CMA was shown to be superior to that of SGD-CMA, both in terms of initial convergence rate and tracking ability under sudden change in the signal environment. The RLS-CMA also achieved the fastest convergence rate when compared with the well-known dynamic LS-CMA and O-CMA. Because the approximated cost function introduced in this work is quadratic in the unknown array weights, other types of RLS algorithms can also be utilized to improve the robustness of the method or to reduce the computational load. Finally, because of the simplified nature of the CM cost function in (1), we note that existing algorithms such as the SGD-CMA, LS-CMA, and O-CMA do not have the ability to lock on a particular source when multiple constant modulus signals are present. Clearly, this observation extends to the proposed RLS-CMA. Generally, this problem can be solved by using multistage methods [7].

References

[1] R. P. Gooch and J. D. Lundell, "The CM array: an adaptive beamformer for constant modulus signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1986, pp. 2523–2526.

[2] J. J. Shynk and R. P. Gooch, "The constant modulus array for cochannel signal copy and direction finding," *IEEE Trans. Signal Processing*, vol. 44, pp. 652–660, Mar. 1996.

Authorized licensed use limited to: McGill University. Downloaded on June 2, 2009 at 16:49 from IEEE Xplore. Restrictions apply.

[3] B. G. Agee, "The least squares CMA: a new technique for rapid correction of constant modulus signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1986, pp. 953–956.
[4] T. E. Biedka, W. H. Tranter, and J. H. Reed, "Convergence analysis of the least squares constant modulus algorithm in interference cancellation applications," *IEEE Trans. Signal Processing*, vol. 48, pp. 491–501, Mar. 2000.
[5] H. Furukawa, Y. Kamio, and H. Sasaoka, "Cochannel interference reduction and path-diversity reception technique using CMA adaptive array antenna in digital land mobile communications," *IEEE Trans. Veh. Technol.*, vol. 50, pp. 605–616, Mar. 2001.
[6] A. Mathur, A. V. Keerthi, and J. J. Shynk, "A variable step-size CM array algorithm for fast fading channels," *IEEE Trans. Signal Processing*, vol. 45, pp. 1083–1087, Apr. 1997.
[7] J. J. Shynk, A. V. Keethi, and A. Mathur, "Steady-state analysis of the multistage constant modulus array," *IEEE Trans. Signal Processing*, vol. 44, pp. 948–962, Apr. 1996.
[8] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Signal Processing*, vol. 43, pp. 95–107, Jan. 1995.
[9] Y. X. Chen, Z. Y. He, T. S. Ng, and P. C. K. Kwok, "RLS adaptive blind beamforming algorithm for cyclostationary signals," *Electron. Lett.*, vol. 35, no. 14, pp. 1136–1138, July 1999.
[10] S. Haykin, *Adaptive Filter Theory*, 2nd ed. NJ: Printice-Hall, 1991.
[11] M. Rupi, P. Tsakalides, C. L. Nikias, and E. Del Re, "Robust constant modulus arrays based on fractional lower-order statistics," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1999, pp. 2945–2948.
[12] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification*. Cambridge, MA: MIT Press, 1983.

Comments on "Numerical Evaluation of the Lambert W Function and Application to Generation of Generalized Gaussian Noise with Exponent 1/2"

D. A. Barry, L. Li, and D.-S. Jeng

***Abstract*—The Lambert W function appears in a wide variety of circumstances, including the recent application to signal processing referred to in the paper under discussion. Besides applications, a sizable body of mathematical analysis has been reported. The original paper presented a numerical algorithm for computation of W_{-1}. An existing, similar algorithm is presented. Iterative improvement of the W_{-1} estimates is also discussed, and issues concerning computational efficiency and possible sources of rounding error in fixed precision computational environments are identified. Existing, public-domain software takes into account all the identified numerical issues and produces estimates of W to near the precision available on the host machine.**

***Index Terms*—Algorithms, approximation methods, error estimation, finite word length effects, iterative methods, round-off errors.**

I. INTRODUCTION

The Lambert W function arises in a wide variety of mathematical, physical, chemical, and engineering contexts [1]. A recent summary of these is available [2]. In [3], a new use of the lower real branch of the Lambert W function W_{-1} is described; it arises in the inverse distribution function of generalized Gaussian noise with power 1/2. The main contributions of [3] were, first, to identify the relationship between W_{-1} and a particular case of the generalized Gaussian noise distribution and, second, to provide a numerical algorithm for computation of W_{-1}. Here, our focus is on the second contribution.

Manuscript received October 1, 2002; revised June 18, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Fredrik Gustafsson.

D. A. Barry is with the University of Edinburgh, Edinburgh EH9 3JL, U.K. (e-mail: d.a.barry@ed.ac.uk).

L. Li is with the School of Engineering, University of Queensland, St. Lucia, 4072, Australia (e-mail: l.li@uq.edu.au).

D.-S. Jeng is with the Department of Civil Engineering, the University of Sydney, Sydney, NSW 2006, Australia (e-mail: d.jeng@civil.usyd.edu.au).

Digital Object Identifier 10.1109/TSP.2004.826154

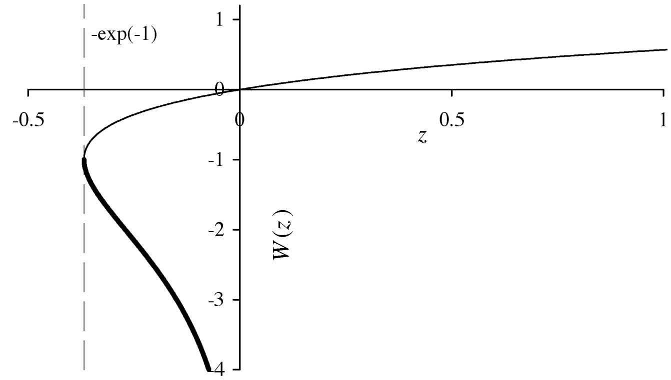

Fig. 1. Branches of the Lambert W function. The lower branch W_{-1} is given by the thick line; the upper branch W_0 is given by the thin line.

Given the widespread applications of the Lambert W function referred to above, further utilization of it in signal processing is a reasonable prospect. It is thus timely to mention the significant body of research pertaining to the W function in the mathematical literature, including numerical algorithms for its calculation. Specifically, this comment is intended first to extend and clarify some material presented in [3] regarding the W function and its approximations and, second, to alert readers that well-tested, arbitrary-precision numerical software is available for computation of *both* real branches of the Lambert W function, i.e., $W_{-1}(z), -\exp(-1) \le z < 0$ with $W_{-1}(z) \le -1$, and the principal branch $W_0(z), -\exp(-1) \le z$ with $W_0(z) \ge -1$ (see Fig. 1 for a plot of these branches). The available software has been thoroughly tested and, as such, is not prone to numerical problems such as rounding error that could occasionally result from the naïve application of the algorithm presented in [3].

II. DETAILED REMARKS

A. Available Software

We wish to make three remarks regarding existing software and algorithms used therein.

1) Software to compute real values of the Lambert W function is available in the public domain mathematical library *Netlib* as Algorithm 743 of the TOMS database (www.netlib.org/toms/743). The software, which is written in *FORTRAN* (both single and double precision versions), computes both real branches of the Lambert W function to whatever precision is available on the computing platform used. This algorithm and details of its application were described in [4] and [5]. A *FORTRAN*-to-*C* converter such as *f2c* (www.netlib.org/f2c) could be employed to help derive *C* or *C++* versions of the TOMS software.
2) The approximation in [4], which is simpler than that presented in [3], produces "one-shot" estimates of W_{-1} with

1053-587X/04$20.00 © 2004 IEEE

Authorized licensed use limited to: McGill University. Downloaded on June 2, 2009 at 16:49 from IEEE Xplore. Restrictions apply.