Joint Use of Constant Modulus and Least Squares Criteria in Linearly-Constrained Communication Arrays

Viktor I. DJIGAN

Advanced Research Laboratory, ELVEES R&D Center of Microelectronics, POB 19, Centralny Prospect, Zelenograd, Moscow, Russia 124460

djigan@elvees.ru

Abstract. *This paper considers the application of the linear constraints and RLS inverse QR decomposition in adaptive arrays based on constant modulus criterion. The computational procedures of adaptive algorithms are presented. Linearly constrained least squares adaptive arrays, constant modulus adaptive arrays and linearly constrained constant modulus adaptive arrays are compared via simulation. It is demonstrated, that a constant phase shift in the array output signal, caused by desired signal orientation and array weights, is compensated in a simple way in linearly constrained constant modulus adaptive arrays.*

Keywords

Adaptive array, constant modulus criterion, RLS, inverse QR decomposition, linear constraints.

1. Introduction

Today antenna arrays [1–5] are widely used as receiving and transmitting antennas in wireless communication systems [6], because they have a number of useful properties such as non-mechanical scanning, the increasing of the desired signal to thermal noise ratio (SNR) and the ability to increase the desired signal to interference ratio (SIR). Arrays with the latter property are called Adaptive Arrays (AA). AA automatically create dips in the directional pattern (DP) in the directions of interference sources and keep the required gain in the direction of a desired signal source, if the directions do not coincide. It means that AA are a space filter and their DP is an amplitude-angle response, which is changed in accordance with the angle distribution of the signal and interference sources.

In wireless communication systems, arrays with small numbers of antennas N are usually used. In this case, not only simple adaptive algorithms with linear computational complexity $O(N)$, but the computationally complex algorithms with quadratic complexity $O(N^2)$, can be readily implemented in modern Digital Signal Processors (DSP) chips [7], [8], and can be used in AA.

Most adaptive filters usually require a reference signal. If no reference is provided, the linearly constrained adaptive algorithms can be used [9]. These algorithms, however, are sensitive to interferences coherent with the desired signal [10]. The interference example is a multipath propagation of the desired signal.

Other adaptive filtering algorithms without a reference signal are Constant Modulus (CM) criterion ones [11]. These algorithms are widely used in blind channel equalisers and less in AA [12].

Should there occur a few CM signals, due to multipath, for example, the adaptive filter may capture interference and suppress the desired signal [13]. The steering of the main lobe of DP of the AA in the direction of the desired signal source by means of initial values of weights may be inefficient because the array weights are changed during adaptation.

It was demonstrated in [14] that if the direction of the desired signal source is known, the application of linear constraints to CM adaptive algorithm allows AA to operate efficiently, if coherent interferences are received. In this case, the constraints keep the main lobe of DP in the direction of the desired signal source during adaptation.

The algorithm [14] has the complexity $O(N)$ and is based on gradient descent search strategy. It is known that such algorithms have a slower convergence and larger residual errors compared to Recursive Least Squares (RLS) algorithms with $O(N^2)$ complexity [15].

Besides, the cost function in the CM criterion is not a quadratic one. As a result, the gradient algorithms often follow local solutions of the adaptive filtering problem.

2. Problem Formulation and Solution

This paper considers the application of linearly-constrained RLS algorithms based on inverse QR decomposition (IQRD) in CM criterion AA.

CM criterion adaptive algorithms are used for the processing of signals with constant modulus envelope. An example is Quadrature Phase Shift Keying (QPSK) modulated signals, which are used in digital data transmitting. Any of the QPSK modulated data symbols a_i have the property $|a_i| = \sqrt{a_i^* a_i} = s = const$, where the superscript $*$ means complex conjugation. The value of s is known in the receiver. CM criterion is formulated as

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

and the adaptive CM algorithms are denoted as CM(p,q). Here $E[\ ]$ is averaging operation, $y(k)=\mathbf{h}^H_N(k\text{-}1)\mathbf{x}_N(k)$ is the array output signal, Fig. 1, $\mathbf{h}_N(k)=[h_1(k),\ldots,h_n(k),\ldots, h_N(k)]^T$ is the weights vector, $\mathbf{x}_N(k)=[x_1(k),\ldots, x_n(k),\ldots, x_N(k)]^T$ is a vector of space-time sampled signals, k is the discrete time (sample number), the superscripts H and T denote Hermitian transpose and transposition of a vector or a matrix. Vectors and matrices are denoted by bold lowercase and uppercase characters, respectively; the character N in subscripts indicates vector (N) or square matrix ($N\times N$) dimensions.

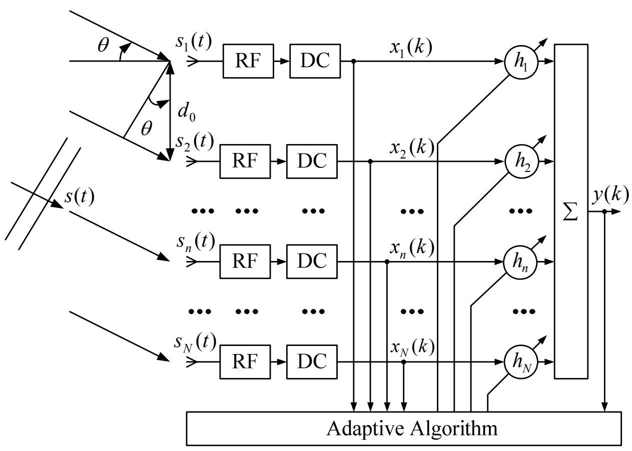

Fig. 1. Adaptive array: RF is radiofrequency amplifier, DC is down converter.

CM criterion (1) is nonlinear. Based on [16], it is shown in [17, 18], that for q=2, equation (1) can be converted to

$$J'(k) = \sum_{i=1}^{k} \lambda^{k-i} \left|s^p - \mathbf{h}_N^H(k)\mathbf{z}_N(i)\right|^2 \tag{2}$$

where $(1\text{-}0.4/N)\leq\lambda<1$ is the forgetting factor and $\mathbf{z}_N(k) = \mathbf{x}_N(k)\, \mathbf{x}^H_N(k)\, \mathbf{h}_N(k\text{-}1)\, |\mathbf{x}^H_N(k)\, \mathbf{h}_N(k\text{-}1)|^{p-2}$. That is, cost function (2) is quadratic in the space of weights $\mathbf{h}_N(k)$.

If p=2, then $\mathbf{z}_N(k) = \mathbf{x}_N(k)\, \mathbf{x}^H_N(k)\, \mathbf{h}_N(k\text{-}1)$. In this case $\mathbf{z}_N(k)$ calculation requires minimum arithmetic operations ($2N$ multiplications and N additions). Here, the transient response of the adaptive algorithm is also minimal, because $\mathbf{z}_N(k)$ has the smaller memory.

The above assumption allows the use of any of the RLS algorithms with quadratic complexity $O(N^2)$ as a CM(2,2) algorithm to minimize cost function (2). In [18] the RLS based on Matrix Inversion Lemma (MIL) was used. In [19] it is shown that such a RLS algorithm in CM AA can be unstable. The multichannel RLS algorithms [15] on the IQRD base are used in [19] as alternative solutions.

3. Linearly-Constrained CM(2,2) RLS

In this section the joint use of a multichannel linearly-constrained RLS algorithm and IQRD algorithms in CM(2,2) is considered.

The objective of the linearly-constrained least square filtering is to minimize the energy of the errors $s^2 - \mathbf{h}^H_N(k\text{-}1)\, \mathbf{z}_N(k)$, i.e. to minimize the cost function (2) under the constraint:

$$\mathbf{C}_{NJ}^H \mathbf{h}_N(k) = \mathbf{f}_J\,. \tag{3}$$

Here, $\mathbf{C}_{NJ}$ and $\mathbf{f}_J$ are matrix and vector of J linear constraints. The solution of the problem is the vector of adaptive filter weights [15]:

$$\begin{aligned}\mathbf{h}_N(k) &= \mathbf{R}_N^{-1}(k)\mathbf{r}_N(k) + \mathbf{R}_N^{-1}(k)\mathbf{C}_{NJ} \times \\ &\times \left[\mathbf{C}_{NJ}^H \mathbf{R}_N^{-1}(k)\mathbf{C}_{NJ}\right]^{-1}\left[\mathbf{f}_J - \mathbf{C}_{NJ}^H \mathbf{R}_N^{-1}(k)\mathbf{r}_N(k)\right],\end{aligned} \tag{4}$$

where $\mathbf{R}_N(k) = \sum_{i=1}^{k} \lambda^{k-i}\mathbf{z}_N(i)\mathbf{z}_N^H(i)$ is the correlation matrix, $\mathbf{r}_N(k) = \sum_{i=1}^{k} \lambda^{k-i}\mathbf{z}_N(i)s^2(i)$ is the cross correlation vector of $\mathbf{z}_N(k)$ and $s^2(k)$.

Matrix $\mathbf{C}_{NJ}$ is created as shown below. The flat wave $s(t)$ is received by AA from the direction θ, see Fig. 1. In linear AA the distance between antennas is d_0. It is usually chosen as a half of the wave length λ_0 of the carrier frequency f_0, i.e. $d_0=0.5\ \lambda_0=0.5v/f_0$, where v the velocity of electromagnetic waves in free space, which equals to the velocity of light.

The delay of the signal in the n-th antenna relative to the number 1 reference antenna is defined as $\tau_n=d_0(n\text{-}1)\sin(\theta)/v$, and the phase is defined as

$$\begin{aligned}\omega_0\tau_n &= 2\pi d_0 f_0(n-1)\sin(\theta)/v = \\ &= 2\pi d_0(n-1)\sin(\theta)/\lambda = \psi_n\,.\end{aligned}$$

Based on these relationships, the columns of the matrix $\mathbf{C}_{NJ}$ are defined as

$$\begin{aligned}\mathbf{c}_N^{(j)}(\theta) &= \left[c_1(\theta_j),\cdots,c_n(\theta_j),\cdots,c_N(\theta_j)\right]^T = \\ &= \left[e^{i\psi_1^j},\cdots,e^{i\psi_n^j},\cdots,e^{i\psi_N^j}\right]^T.\end{aligned}$$

Equation (3) means that $F(\theta_j)=\mathbf{c}^H_N(\theta_j)\mathbf{h}_N(k)=f_j=|f_j|e^{i\varphi(j)}$, i.e. DP $F(\theta_j)$ of AA in the direction θ_j equals to the j-th element of the constraint vector $\mathbf{f}_j$. Since we are interested in receiving only one desired signal from the direction θ, the constraint matrix becomes a vector $\mathbf{c}_N(\theta)$ and the vector of constraints becomes a scalar f. In this case the linearly-constrained RLS algorithm [15] for AA, see Fig. 1, is simplified as below.

$$\textbf{Initialisation: } \mathbf{x}_N(0)=\mathbf{0}_N, \mathbf{R}_N^{-1}(0)=\delta^{-2}\mathbf{I}_N,$$

0) $\boldsymbol{\gamma}_N(0)=\mathbf{R}_N^{-1}(0)\mathbf{c}_N, \mathbf{q}_N(0)=\boldsymbol{\gamma}_N\left[\mathbf{c}_N^H\boldsymbol{\gamma}_N\right]^{-1},$
$\mathbf{h}_N(0)=\mathbf{q}_N(0)f$

For $k=1,2,\ldots,K$

1) $\mathbf{z}_N(k)=\mathbf{x}_N(k)\mathbf{x}_N^H(k)\mathbf{h}_N(k-1)$

2) Computation of $\mathbf{g}_N(k)$

3) $\nu(k)=\mathbf{c}_N^H\mathbf{g}_N(k)$

4) $\upsilon^*(k)=\mathbf{z}_N^H(k)\mathbf{q}_N(k-1)$

5) $\mathbf{q}'_N(k)=\left[\mathbf{q}_N(k-1)-\mathbf{g}_N(k)\upsilon^*(k)\right]\times \times\left[1+\dfrac{\nu(k)\upsilon^*(k)}{1-\upsilon^*(k)\nu(k)}\right]$

6) $\mathbf{q}_N(k)=\mathbf{q}'_N(k)+\mathbf{c}_N\left(\mathbf{c}_N^H\mathbf{c}_N\right)^{-1}\left[1-\mathbf{c}_N^H\mathbf{q}'_N(k)\right]$

7) $\alpha_N(k)=s^2-\mathbf{h}_N^H(k-1)\mathbf{z}_N(k)$

8) $\mathbf{h}'_N(k)=\mathbf{h}_N(k-1)+\mathbf{g}_N(k)\alpha_N^*(k)$

9) $\mathbf{h}_N(k)=\mathbf{h}'_N(k)+\mathbf{q}_N(k)\left[f-\mathbf{c}_N^H\mathbf{h}'_N(k)\right]$

End for k

To compute Kalman gain $\mathbf{g}_N(k)$, a stable procedure based on IQRD of CM(2,2) RLS [15], [19] can be used. The computations with square root Givens rotations are presented below.

0) $\textbf{Initialisation: } \widetilde{\mathbf{R}}_N^{-H}(0)=\sqrt{\delta^{-2}\mathbf{I}_N}$

1) $\mathbf{u}_N^{(0)H}(k)=\mathbf{0}_N^T, \quad b_N^{(0)}(k)=1$

For $i=1,2,\ldots,N$

2) $a_{N,i}(k)=\lambda^{-0.5}\widetilde{\mathbf{R}}_N^{-H}(k-1)\Big|_{i,1:i}\mathbf{z}_N(k)\Big|_{1:i}$

3) $b_N^{(i)}(k)=\sqrt{\left[b_N^{(i-1)}(k)\right]^2+a_{N,i}^*(k)a_{N,i}(k)}$

4) $s_{N,i}(k)=a_{N,i}^*(k)/b_N^{(i)}(k)$

5) $c_{N,i}(k)=b_N^{(i-1)}(k)/b_N^{(i)}(k)$

For $j=1,2,\ldots,i$

6) $\widetilde{R}_{N,i,j}^{-H}(k)=\lambda^{-0.5}c_{N,i}(k)\widetilde{R}_{N,i,j}^{-H}(k-1)-s_{N,i}^*(k)u_{N,j}^{(i-1)*}(k)$

7) $u_{N,j}^{(i)*}(k)=\lambda^{-0.5}s_{N,i}(k)\widetilde{R}_{N,i,j}^{-H}(k-1)+c_{N,i}(k)u_{N,j}^{(i-1)*}(k)$

End for j

End for i

8) $\mathbf{g}_N(k)=\mathbf{u}_N^{(N)}(k)/b_N^{(N)}(k)$

Another procedure of IQRD Kalman gain computation by means of square root free Givens rotations is presented as well.

0) $\textbf{Initialisation: } \overline{\mathbf{R}}_N^{-H}(0)=\sqrt{\delta^{-2}\mathbf{I}_N}, \mathbf{K}_N^R(0)=\mathbf{I}_N$

1) $\overline{\mathbf{u}}_N^{(0)H}(k)=\mathbf{0}_N^T, \quad K_N^{B(0)}(k)=1$

For $i=1,2,\ldots,N$

2) $\overline{a}_{N,i}(k)=\overline{\mathbf{R}}_N^{-H}(k-1)\Big|_{i,1:i}\mathbf{z}_N(k)\Big|_{1:i}$

3) $K_N^{B(i)}(k)=K_N^{B(i-1)}(k)+\lambda^{-1}K_{N,i}^R(k-1)\overline{a}_{N,i}^*(k)\overline{a}_{N,i}(k)$

4) $\overline{s}_{N,i}(k)=\lambda^{-1}K_{N,i}^R(k-1)\overline{a}_{N,i}^*(k)/K_N^{B(i)}(k)$

5) $\overline{c}_{N,i}(k)=K_N^{B(i-1)}(k)/K_N^{B(i)}(k)$

For $j=1,2,\ldots,i$

6) $\overline{R}_{N,i,j}^{-H}(k)=\overline{R}_{N,i,j}^{-H}(k-1)-\overline{a}_{N,i}(k)\overline{u}_{N,j}^{(i-1)*}(k)$

7) $\overline{u}_{N,j}^{(i)*}(k)=\overline{s}_{N,i}(k)\overline{R}_{N,i,j}^{-H}(k-1)+\overline{c}_{N,i}(k)\overline{u}_{N,j}^{(i-1)*}(k)$

End for j

8) $K_{N,i}^R(k)=\lambda^{-1}K_{N,i}^R(k-1)\overline{c}_{N,i}(k)$

End for i

9) $\mathbf{g}_N(k)=\overline{\mathbf{u}}_N^{(N)}(k)$

4. Algorithm Complexity

Linearly-constrained algorithms with different procedures of the computation of vector $\mathbf{g}_N(k)$ are mathematically identical to each other and the only difference is the computational complexity. If floating point arithmetic is used, and the AA have the same parameters and process the same signals, then the algorithms compute the same output signal and the same weights of AA. These output parameters differ by the computational errors only because each algorithm uses its own sequence of computations with its own number of arithmetic operations.

The estimation of the computational complexity of the above linearly-constrained algorithm (excluding Kalman gain calculation) is $12N+1$ multiplications, $10N+5$ additions and 1 division. The computation of the vector $\mathbf{c}_N(\mathbf{c}^H_N\mathbf{c}_N)^{-1}$ is omitted in the estimation. The computation does not depend on the iteration number k and similarly to initialisation, it can be done in advance.

The complexity of Kalman gain computation by means of square root IQRD includes N square roots due to the computation of the variable $b_N(k)$ during N steps $i=1,\ldots,N$ at each k-th iteration of the adaptive filtering algorithm. The computations also include $3N^2+7N$ multiplications, $1.5N^2+2.5N$ additions and $N+1$ divisions. The complexity of Kalman gain computation by means of square root-free IQRD equals $2.5N^2+7N$ multiplications, $1.5N^2+2.5N$ additions and N divisions.

As a comparison, the complexity of the Kalman gain computations by means of RLS based on MIL [15], see below, equals $4N^2+2N$ multiplications, $3N^2+N+1$ additions and 1 division.

0) $\textbf{Initialisation:}\ \mathbf{R}_N^{-1}(0) = \delta^{-2}\mathbf{I}_N$

1) $\mathbf{g}_N(k) = \dfrac{\mathbf{R}_N^{-1}(k-1)\mathbf{z}_N(k)}{\lambda + \mathbf{z}_N^H(k)\mathbf{R}_N^{-1}(k-1)\mathbf{z}_N(k)}$

2) $\mathbf{R}_N^{-1}(k) = \lambda^{-1}\left[\mathbf{R}_N^{-1}(k-1) - \mathbf{g}_N(k)\mathbf{z}_N^H(k)\mathbf{R}_N^{-1}(k-1)\right]$

The complexity of Kalman gain computation by means of IQRD based on Householder transform [15], see below, equals $4N^2+3N+3$ multiplications, $3N^2+N+2$ additions, 1 square root and 2 divisions.

0) $\textbf{Initialisation:}\ \mathbf{A}_N^{-H}(0) = \sqrt{\delta^{-2}\mathbf{I}_N}$

1) $\widetilde{\mathbf{a}}_N(k) = \lambda^{-0.5}\mathbf{A}_N^{-H}(k-1)\mathbf{z}_N(k)$

2) $b_N(k) = \sqrt{1+\widetilde{\mathbf{a}}_N^H(k)\widetilde{\mathbf{a}}_N(k)}$

3) $\gamma_N(k) = \left[b_N(k) + b_N^2(k)\right]^{-1}$

4) $\widetilde{\mathbf{u}}_N(k) = \mathbf{A}_N^{-1}(k-1)\widetilde{\mathbf{a}}_N$

5) $\hat{\mathbf{u}}_N(k) = \lambda^{-0.5}\gamma_N(k)\widetilde{\mathbf{u}}_N(k)$

6) $\mathbf{A}_N^{-H}(k) = \lambda^{-0.5}\mathbf{A}_N^{-H}(k-1) - \widetilde{\mathbf{a}}_N(k)\hat{\mathbf{u}}_N^H(k)$

7) $\mathbf{g}_N(k) = \lambda^{-0.5}\widetilde{\mathbf{u}}_N(k)/b_N^2(k)$

Initial values of the matrix $\mathbf{R}_N^{-1}(0)$ and the matrices $\widetilde{\mathbf{R}}_N^{-1}(0)$, $\overline{\mathbf{R}}_N^{-1}(0)$ are related as: $\mathbf{R}_N^{-1}(0) = \widetilde{\mathbf{R}}_N^{-1}(0)\widetilde{\mathbf{R}}_N^{-H}(0)$, $\mathbf{R}_N^{-1}(0) = \overline{\mathbf{R}}_N^{-1}(0)\overline{\mathbf{R}}_N^{-H}(0)$ and $\mathbf{R}_N^{-1}(0) = \mathbf{A}_N^{-1}(0)\mathbf{A}_N^{-H}(0)$. The parameter $\delta^2 \geq 0.01\sigma_z^2$ is used for initial regularization of $\mathbf{R}_N(k)$ [20]. Here σ_z^2 is a variance of $z(k)$ in vector $\mathbf{z}_N(k)$.

Thus, the total complexity of the considered linearly-constrained CM(2,2) RLS algorithm depends on a procedure of vector $\mathbf{g}_N(k)$ computation. The use of the procedure depends on the computing recourses and the stability of the procedure in given operational conditions.

In order to compare the algorithm's complexity, we have to present square roots and divisions in terms of multiplications and additions. In most of DSP, the operations are not supported by hardware and they are implemented in software by means of table and analytical methods.

For example, in applied libraries of "Multicore" DSP [8] the initial values of square roots and divisions are accumulated in a table and improved iteratively by means of Newton-Raphson method [21]. Square root requires 13 multiplications and 3 additions. Division requires 7 multiplications and 3 additions.

The full computational complexity of the considered algorithms in terms of multiplications (MUL) and additions (ADD) is presented in Fig. 2 and Fig. 3 for N=2,…,32. Here, the above complexities of square roots and divisions are used.

Relative increase of the complexity to the same algorithms without constraints [19] is presented in Fig. 4 and Fig. 5. As we can see, the contribution of constraints in the total complexity decreases when N is increased.

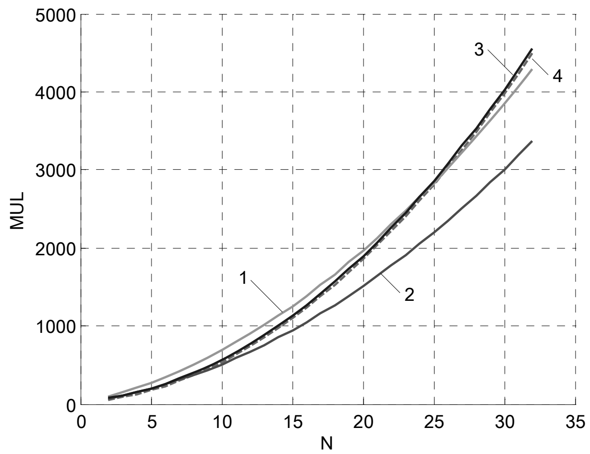

Fig. 2. Linearly-constrained adaptive array complexity: multiplications.

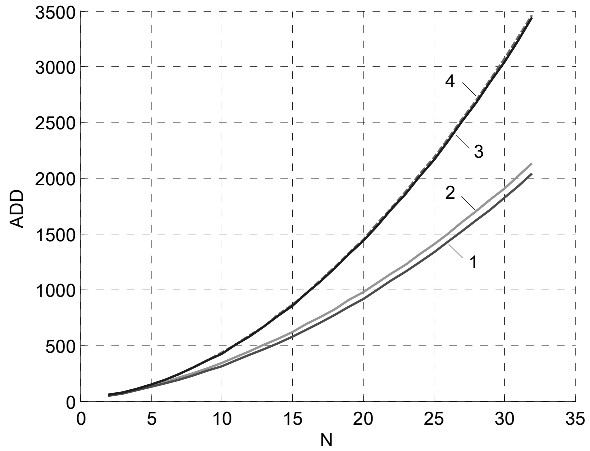

Fig. 3. Linearly-constrained adaptive array complexity: additions.

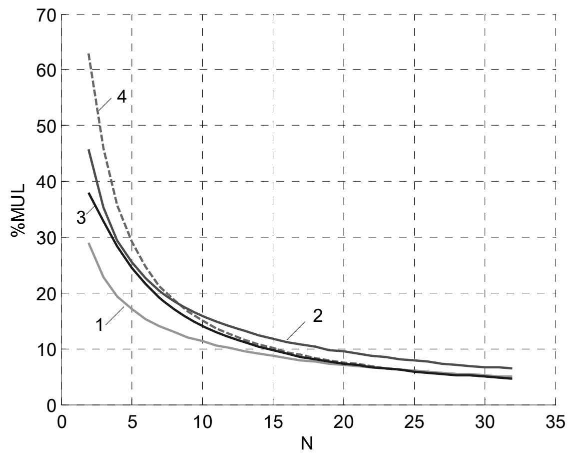

Fig. 4. Relative complexity of linearly-constrained adaptive array: multiplications.

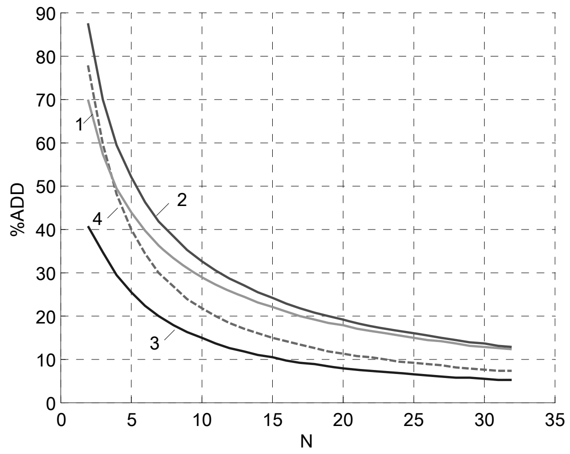

Fig. 5. Relative complexity of linearly-constrained adaptive array: additions.

The curves in Fig. 2 – Fig.5 are denoted as follows: curve 1 belongs to linearly-constrained array with square root Givens IQRD Kalman gain calculation, curve 2 belongs to linearly-constrained array with square root free Givens IQRD Kalman gain calculation, curve 3 belongs to linearly-constrained array with MIL Kalman gain calculation and curve 4 belongs to linearly-constrained array with Householder transform Kalman gain calculation.

If N=8, the linearly-constrained algorithm with square root free IQRD Kalman gain calculation requires about K_T=2680 instructions of DSP 1892BM3T (“Multicore-12”, MC-12) whose clock frequency is 100 MHz. The DSP has Single Instruction Single Data architecture. It executes about $100\cdot 10^6/K_T \approx 37.5\cdot 10^3$ algorithm iterations per second. If the iterations occur at double data rate, the AA can be used for QPSK-4 modulation in a communication system with a data rate of about $2\cdot 37.5\cdot 10^3/2$=37.5 kbit/s. If DSP 1892BM2T (“Multicore-24”, MC-24) with Single Instruction Multiple Data architecture is used the data rate is approximately doubled. A new DSP MCF-0428 (“Multi-Force”) of the family, whose computational power is 33 times higher that of DSP MC-12, allows to build AA for data rate up to $33\cdot 37.5\cdot 10^3$=1.24 Mbit/s. If the linear constraints are not used then the AA supports QPSK-4 data receiving at 45 kbit/s (MC-12), at 90 kbit/s (MC-24) and 1.5 Mbit/s (MCF-0428).

5. Simulation

The efficiency of the considered algorithms is demonstrated via simulation, which was conducted in base-band in accordance with Fig. 1. The desired signal was QPSK-4 with $|a_i|$=1. The desired signal direction was θ_s=0°, and two interferences were placed at $\theta_{J_1} = 21^\circ$ and $\theta_{J_2} = -38^\circ$ directions. Correlated interference J_1 was a copy of the desired signal delayed by half of an information symbol. The interference level was 3 dB lower than the desired signal. Interference J_2 was simulated by white noise, the level of which was 20 dB higher than the desired signal. SNR in channel was -30 dB. Algorithm parameters were selected as N=8, δ^2=0.01 and λ=0.9999. All computations were conducted in floating point arithmetic. To provide the mentioned delay of correlated interference, the information symbol was sampled twice per duration. The samples corresponded to algorithm iterations k. The number of iterations was $K=15\cdot 10^4$. In the simulation the Kalman gains were calculated by means of square root-free Givens rotation IQRD. Three algorithms were considered, a linearly constrained RLS algorithm, a CM(2,2) RLS algorithm, and a linearly constrained CM(2,2) RLS algorithm.

A linearly-constrained RLS algorithm differs from a linearly-constrained CM(2,2) RLS algorithm in the following way: vector $\mathbf{z}_N$ is substituted by $\mathbf{x}_N$ and $\alpha_N(k) = -\mathbf{h}^H_N(k\text{-}1)\,\mathbf{x}_N(k)$. CM(2,2) RLS [19] was initialized in two ways as $\mathbf{h}_N(0) = [N^{-1}, \mathbf{0}^T_{N-1}]^T$ and $\mathbf{h}_N(0) = N^{-1}\mathbf{i}_N$, where $\mathbf{i}_N$ is the unity vector. As the DP of antennas was assumed to be omnidirectional, all signals were received by antenna 1 at the beginning of the adaptation in the first case. In the second case the direction of the main beam of DP coincided with the desired signal direction θ_s=0° and all signals were received by all antennas. In the linearly-constrained CM(2,2) RLS algorithm the weight vector $\mathbf{h}_N(0)$ was $\mathbf{q}_N(0) = N^{-1}\mathbf{i}_N$, because $f = 1$.

The simulation results, Fig. 6 – Fig. 9, demonstrate the property of a linearly-constrained CM(2,2) RLS algorithm comparing to linearly-constrained RLS and CM(2,2) RLS algorithms. In the figures the DP of an array with main beam direction θ_s=0° is shown in grey and AA DP in steady-state is shown in black.

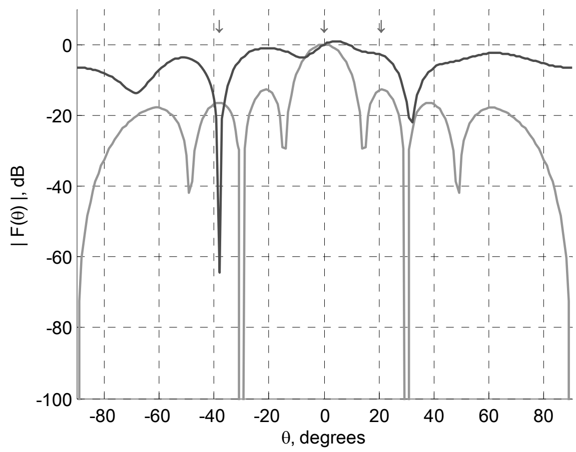

Fig. 6. Simulations results: linearly-constrained RLS AA.

Fig. 6 demonstrates that a linearly-constrained RLS algorithm provides the constraint $f = 1$ in the direction θ_s=0° and suppresses uncorrelated interference in the direction $\theta_{J_2} = -38^\circ$. The suppressing of correlated interference in the direction $\theta_{J_1} = 21^\circ$ is insufficient (about 3 dB). The final DP is destroyed and signal constellation at AA output is destroyed as well, see Fig. 10.

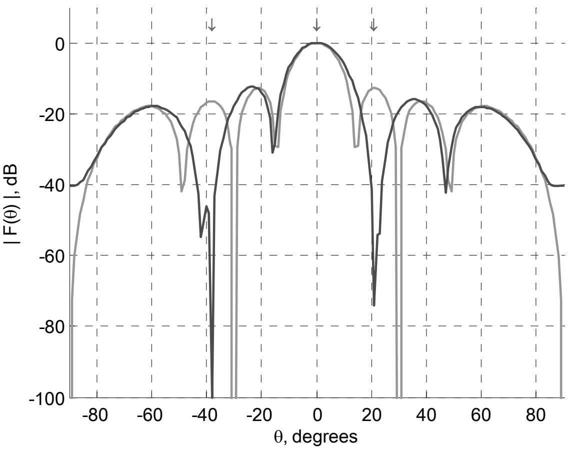

Fig. 7. Simulations results: CM(2,2) RLS AA initialized as $\mathbf{h}_N(0) = \left[N^{-1}, \mathbf{0}_{N-1}^T\right]^T$.

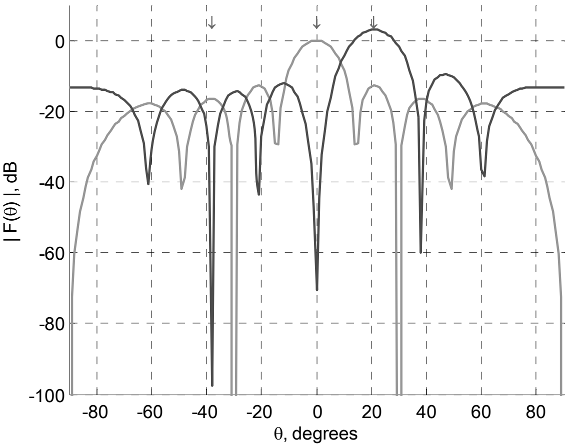

Fig. 8. Simulations results: CM(2,2) RLS AA initialized as $\mathbf{h}_N(0) = N^{-1}\mathbf{i}_N$.

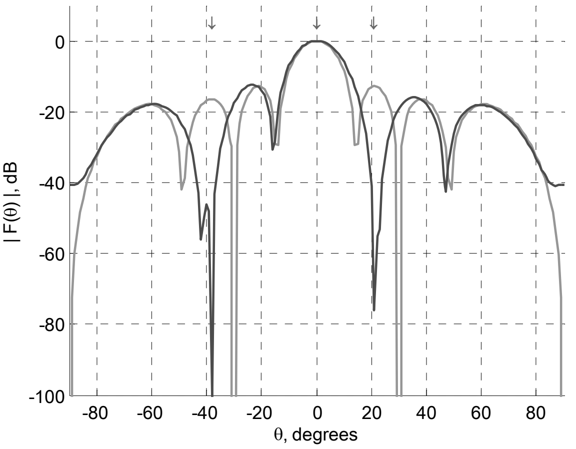

Fig. 9. Simulations results: linearly-constrained CM(2,2) RLS AA.

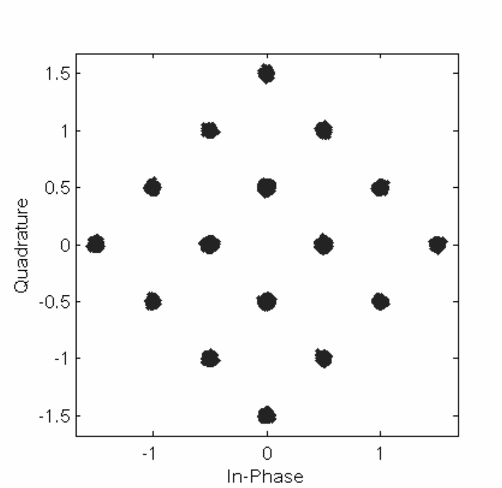

Fig. 10. Constellation: linearly-constrained RLS AA.

CM(2,2) RLS algorithm, Fig. 7, suppresses both interferences. During adaptation it acquires the main lobe of DP in direction θ_s=0°, i.e. the direction of the CM signal with maximal power. However the constellation at AA output has a fix phase shift (Fig. 11) $\varphi_s = arctg \frac{\mathrm{Im}[F(\theta_s)]}{\mathrm{Re}[F(\theta_s)]}$.

The phase shift leads to wrong detection of information symbols and has to be removed. The removing is usually done by an additional phase-locked loop [11].

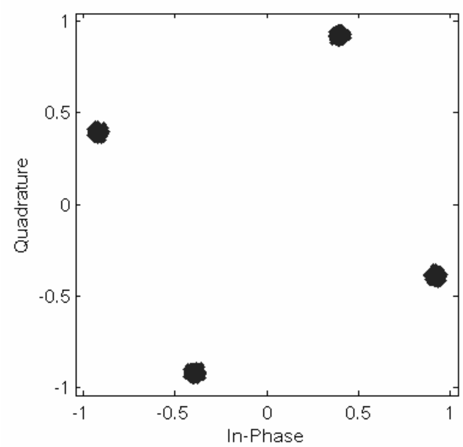

Fig. 11. Constellation: CM(2,2) RLS AA initialized as $\mathbf{h}_N(0) = \left[N^{-1}, \mathbf{0}_{N-1}^T\right]^T$.

The CM(2,2) RLS algorithm, Fig. 8, due to a different initialization has DP with the main lobe in the direction θ_s=0° in the beginning of adaptation. However, during adaptation the algorithm captures correlated interference. As a result, the main lobe of DP is redirected towards the interference. To provide the condition $|a_i|$=1 at AA output the gain of DP is increased by 3 dB in the direction of coherent interference. Constellation at AA output is also rotated as $\varphi_{J_1} = arctg \dfrac{\mathrm{Im}[F(\theta_{J_1})]}{\mathrm{Re}[F(\theta_{J_1})]}$, Fig. 12.

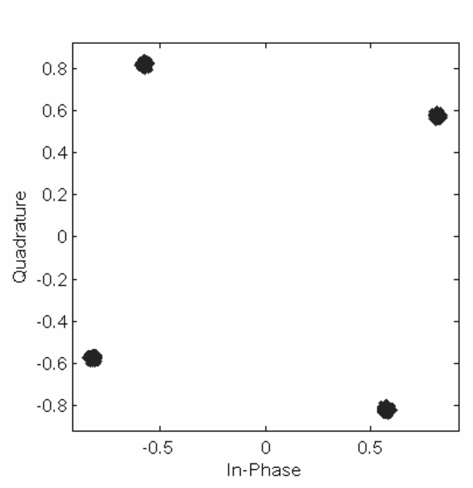

Fig. 12. Constellation: CM(2,2) RLS AA initialized as $\mathbf{h}_N(0) = N^{-1}\mathbf{i}_N$.

If one was to use linear constraints in CM(2,2) RLS algorithm, it is possible to compensate the above mentioned phase shift in a simple way. It can be done if the constraint f is real-valued, i.e. $F(\theta_s) = \mathbf{h}^H_N \mathbf{c}_N(\theta_s) = |f_s|$. Simulation demonstrates that if the desired signal direction θ_s is known, the linear constraint $f = |f_s|$ ensures the required orientation of the DP main lobe and the correct orientation of the AA output constellation which coincides with that of the transmitted data alphabet, Fig. 13, at each iteration.

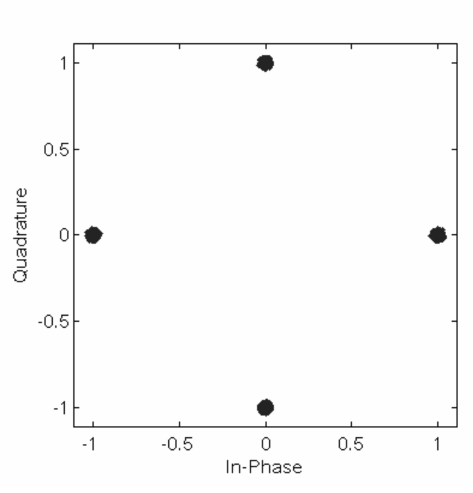

Fig. 13. Constellation: linearly-constrained CM(2,2) RLS AA.

DP at Fig. 7 and Fig. 9 are about the same because the norms of a weights vector $\rho_N(K) = \|\mathbf{h}_N(K)\|_2$ are also about the same. Weight vector for Fig. 7 is $\mathbf{h}_N(K) = [0.0461 + i\,0.0858, 0.0532 + i\,0.0878, 0.1129 + i\,0.1042, 0.1103 + i\,0.1030, 0.0614 + i\,0.1437, 0.0577 + i\,0.1423, 0.0686 + i\,0.0798, 0.0639 + i\,0.0717]^T$ and $\rho_N(K) = 0.3670$. Weight vector for Fig. 9 is $\mathbf{h}_N(K) = [0.0966 + i\,0.0115, 0.1025 + i\,0.0069, 0.1503 - i\,0.0327, 0.1477 - i\,0.0311, 0.1529 + i\,0.0323, 0.1498 + i\,0.0346, 0.1048 - i\,0.0102, 0.0955 - i\,0.0114]^T$ and $\rho_N(K) = 0.3672$.

This similar simulation was conducted for the different interference combinations. A linearly constrained algorithm provides a DP dip in the direction of coherent interference about -3 dB, and about -64 dB in the direction of white noise interference.

A CM(2,2) RLS algorithm initialized as $\mathbf{h}_N(0) = [N^{-1}, \mathbf{0}^T_{N-1}]^T$ and a linearly-constrained CM(2,2) RLS algorithm provide about -68…-76 dB of DP dip in coherent interference direction and -100…-105 dB of dip in white noise interference direction.

A CM(2,2) RLS algorithm, initialized as $\mathbf{h}_N(0) = N^{-1}\mathbf{i}_N$, provides about the same interference suppression. In cases where the algorithm captures coherent interference, the main lobe of DP is reoriented in the direction of the interference, the gain of the lobe is increased and the desired signal is suppressed. A DP dip in the desired signal direction is about -70 dB.

6. Conclusion

Thus, the paper considers the joint application of constant modulus and least squares criteria in linearly constrained AA. The simulation demonstrates the efficiency of the technology in AA receiving a QPSK-4 modulated signals. If the direction of the desired signal source is known, the application of a simple linear constraint to CM(2,2) RLS [19] allows to efficiently suppress coherent and white noise interferences, prevents the capture of coherent interferences and compensates constant phase shift in AA output signals. The achieved properties come at the price of some additional complexity in the algorithm, which drops as the number of AA antennas increases. Further development of the approach is presently conducted in the direction of multimodulus signal processing.

References

[1] BENENSON, L. S., ZHOURAVLEV, V. A., POPOV, S. V., POSTNOV G. A. *Antenna Arrays: Computation and Design Methods*. Moscow: Soviet Radio, 1966. (in Russian).

[2] BRATCHIKOV, A. N., VASIN, V. I., VASILENKO, O. O. etc. *Active Phased Arrays*. Moscow: Radioengineering, 2004. (in Russian).

[3] HUDSON, J. E. *Adaptive Array Principles*. Loughborough: Peter Peregrinus Ltd., 1981.

[4] HAYKIN, S. *AdaptiveFfilter Theory*. 4th edition. Prentice Hall, 2001.

[5] SAYED, A. H. *Fundamentals of Adaptive Filtering*. Hoboken, NJ: John Wiley and Sons, Inc, 2003.

[6] GODARA, L. C. Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations. *Proc. of the IEEE*, 1997, vol. 85, no.8, p. 1195–1245.

[7] DJIGAN, V. I. Applied library of adaptive algorithms. *Electronics: Science, Technology, Business,* 2006, no. 1, p. 60–65. (in Russian).

[8] SOLOKHINA, T., ALEXANDROV, Y., PETRICHKOVICH, J. Signal controllers of "ELVEES" company: first line of Russian DSP. *Electronics: Science, Technology, Business,* 2005, no. 7, p. 70–77. (in Russian).

[9] FROST, O. L. An algorithm for linearly constrained adaptive array processing. *Proc. of the IEEE*, 1972, vol. 60, no. 8, p. 926–935.

[10] SHAN, T.-J., KAILATH, T. Adaptive beamforming for coherent signals and interference. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 1985, vol. 33, no. 3, p. 527–536.

[11] TREICHLER, J., LARIMORE, M. New processing techniques based on the constant modulus adaptive algorithm. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 1985, vol. 33, no. 2, p. 420–431.

[12] GOOCH, R., LUNDELL, J. The CM array: An adaptive beamformer for constant modulus signals. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1986, vol. 11, p. 2523–2526.

[13] TREICHLER, J., LARIMORE, M. The tone capture properties of CMA-based interference suppressors. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 1985, vol. 33, no. 4, p. 946–958.

[14] RUDE, M.J. GRIFFITHS, L. J. Incorporation of linear constraints into the constant modulus algorithm. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1989, vol. 2, p. 968–971.

[15] DJIGAN, V. I. Multichannel RLS and fast RLS adaptive filtering algorithms. *Successes of Modern Radioelectronics*, 2004, no. 11, p. 48–77. (in Russian).

[16] YANG, B. Projection approximation subspace tracking. *IEEE Trans. Signal Processing*, 1995, vol. 43, no. 1, p. 95–107.

[17] CHEN ,Y. X., HE, Z. Y., NG, T. S., KWOK, P. C. K. RLS adaptive blind beamforming algorithm for cyclostationary signals. *Electronic*s, 1999, vol. 35, no.14, p. 1136–1138.

[18] CHEN, Y., LE-NGOC, T., CHAMPAGNE, B., XU, C. Recursive least squares constant modulus algorithm for blind adaptive array. *IEEE Trans. Signal Processing*, 2004, vol. 52, no. 5, p. 1452–1456.

[19] DJIGAN, V. I., PLETNEVA, I. D. Adaptive filtering algorithms based QR decomposition in digital arrays for communication. *Digital Signal Processing*, 2007, no. 4, p. 2–8. (in Russian).

[20] DJIGAN, V. I. On conditions of equivalence for different recursive least squares adaptive filtering algorithms. *Telecommunications*, 2006, no. 6, p. 6–11. (in Russian).

[21] WANG, L.-K., SCHULTE, M. J. Decimal floating-point division using Newton-Raphson iteration. In *Proc. of the 15-th IEEE International Conference on Application-Specific Systems, Architectures and Processors*, 2004, p. 84–95.

About Authors...

Victor I. DJIGAN was born in 1958 in the village of Sudilkov, Shepetovka District, Khmelnitsky Province, Ukraine. He received his B.Sc. degree with honours from the Vinnitsa Technical College of Electronic Devices in Ukraine in 1978; M.Sc. degree with honours from the Moscow Institute of Electronic Engineering (MIEE), Technical University in Russia in 1984; Ph.D. from MIEE in 1990, and D.Sc. degree from the Satellite Communication R&D Center in 2006, all in radio engineering. From 1984 onwards he has been holding different R&D positions in the MIEE, as well as in a number of Russian and international companies. These include companies such as Samsung Electronics, LG and PMC-Sierra outside Russia, where he worked in the fields of digital signal processing, communication, adaptive antenna arrays, acoustic echo and noise cancellation, transmitting line analysis and reflectometry. At present, he is a Principal Researcher in ELVEES R&D Center in Moscow, Russia, and a Scientific Secretary of the company. His scientific interests include adaptive signal processing, speech processing and digital communication. Dr. Djigan is the author of about 150 papers in the above-mentioned fields. He is a member of the Russian A.S. Popov Society for Radioengineering, Electronics & Communications and a Senior IEEE member (member in 96, Senior member in 2004). He has been also a member of the Editorial Board of the journal "Radioengineering: Proceedings of Czech and Slovak Technical Universities and URSI Committers" since 2005. Dr. Djigan is a Section Chair and a member of the Programme Committee of a number of Russian and International Conferences since 2004.