Hybrid Algorithms for Adaptive Array Systems

D. PAPADIMITRIOU and I.O. VARDIAMBASIS
Microwave Communications & Electromagnetic Applications Laboratory,
Telecommunications Division, Department of Electronics,
Technological Educational Institute (T.E.I.) of Crete - Chania Branch,
Romanou 3, Chalepa, 73133 Chania, Crete.
GREECE

Abstract: - In recent years, mobile communications have caused an explosive growth to the number of wireless users. This growth has triggered an enormous demand not only for capacity but also for better coverage and quality of services with priority on interference cancellation. This need lead scientists in using adaptive array antennas. The aim of this paper is to study the adaptive beamforming approaches as part of smart antenna technology. This work's focus is on the investigation of various adaptive algorithms and their ability to automatically respond to an unknown interference environment by steering nulls and reducing side lobe levels in the direction of interference, while keeping desired signal beam characteristics. The investigation covers three well known algorithms [Sample Matrix Inversion (SMI), Least Mean Squares (LMS), Normalized Least Mean Squares (N-LMS)], along with the combination of SMI and LMS and the combination of SMI and N-LMS, which are proposed here.

Keywords: - Smart antennas, Beamforming, Adaptive arrays, Sample Matrix Inversion algorithm, Least Mean Squares algorithm, Mobile communications.

1 Introduction

Rapid growth of the capacity needed by mobile communication systems, leads to increased density of radio sources. Consequently, undesired noise appears in the signal environment, which tends to reduce communication systems' performance. In order to overcome this problem, adaptive arrays are used, because they can discriminate the interference sources in the wireless channel suppressing them automatically. Thus, system's performance is improved, without any knowledge about the location of interference-sources.

Fig. 1a shows an adaptive array system in the form of an adaptive linear combiner, which is the basic building block of almost all adaptive filters, thus having a wide variety of practical applications [1]-[5]. The necessary learning system is generally characterized as signal operator with adjustable parameters by an adaptive algorithm, which adjusts automatically all system's parameters in order to optimise its performance. The adjustable parameters of Fig. 1 are weights (indicated by arrowed circles), while the input signals are stochastic. The adaptive (adjusting weights) algorithm is using information obtained from inputs. Its efficiency depends on minimizing the input information usage and maximizing solution's quality (achieving parameter adjustments close to optimum). However, input information minimization corresponds to slow adaptive convergence, fact which is the learning systems' trade off.

Adaptive array systems adjust their directional parameters, so as to maximize the signal to noise ratio (SNR), because the desired signal is received by the array along with many interference signals. Then using an adaptive algorithm, the variable system's weights automatically adjust, and the system manages to have a main lobe in the direction of the desired signal and at the same time to reject any interference incident from other directions.

In the following paragraphs, three known adaptive algorithms (SMI, LMS, N-LMS) and two new hybrid algorithms are used in an adaptive array system in order to demonstrate their convergence speed and stability, with respect to noise cancellation. In Section 2 the configuration of an adaptive array system is given and the used adaptive algorithms are introduced. Finally in Section 3, the simulation parameters and results are described along with relative conclusions.

2 Adaptive Beamforming

2.1 Beamforming principles

Consider the adaptive beamforming configuration shown in Fig. 1a, which is a uniform linear array (ULA) with M isotropic elements, placed along x-axis of Fig. 1b. The output $y(t)$ of this array system is the weighted sum of the received signals $s_\ell(t)$ ($\ell = 1,2,...,q$) and the noise $n(t)$. The weights w_m ($m = 1,2,...,M$) are computed iteratively based on the array output $y(t)$ and the reference signal $d(t)$, which is approximately the desired transmitted signal. The reference signal is known to the receiver by using training block data, while its format varies and depends upon the implemented system. Also, the correlation between the reference and the desired signals influences the accuracy and the convergence of the selected algorithm.

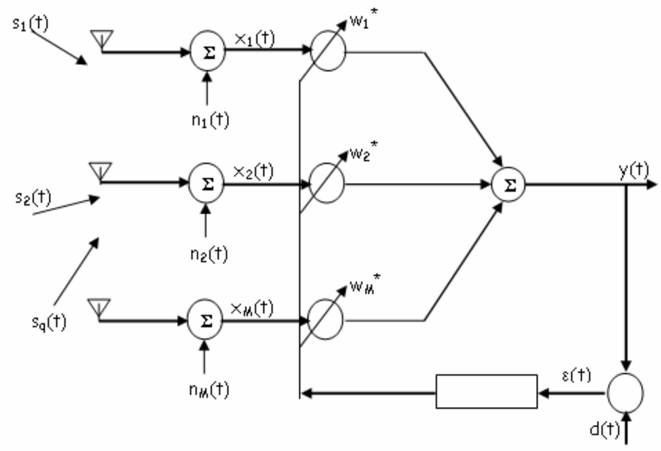

Figure 1 (a): An adaptive array system.

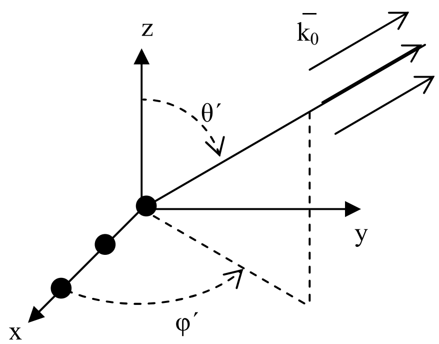

Figure 1 (b): The wave incident upon the structure.

Assuming that $x_M(t)$ is the input data to the M-th antenna element at time t, and w_M is the weight of the M-th element, the array's input data and weight factors can be written in vector form:

$$\overline{x}(t) = [x_1(t)\ x_2(t) .... x_M(t)]^T, \qquad (1a)$$

$$\overline{w} = [w_1\ w_2 .... w_M]^T, \qquad (1b)$$

where superscript T denotes the transpose operation. Using this notation, the array output $y(t)$ is:

$$y(t) = \overline{w}^H \cdot \overline{x}(t), \qquad (1c)$$

where superscript H denotes the Hermittian (complex conjugate) transpose. So, in order to compute the optimum weight vector $\overline{w}_{opt}$, the input data vector $\overline{x}(t)$ has to be determined.

Considering as $s(t)$ either the desired or the interfering input signal, which is obliquely incident upon the antenna structure with elevation angle $\pi/2-\theta'$ and azimuth angle φ' (as in Fig. 1(b)), the input data vector $\overline{x}(t)$ can be expressed as:

$$\overline{x}(t) = \begin{bmatrix} e^{-j2\pi f_c \cdot \tau_1(\theta',\varphi')} \\ e^{-j2\pi f_c \cdot \tau_2(\theta',\varphi')} \\ \cdots \\ e^{-j2\pi f_c \cdot \tau_M(\theta',\varphi')} \end{bmatrix} \cdot s(t) = \overline{a}(\theta',\varphi') \cdot s(t), \qquad (2)$$

where $\overline{a}(\theta',\varphi')$ is termed the array propagation vector or the steering vector for the particular values of (θ',φ'), f_c represents the carrier frequency, $\tau_m(\theta',\varphi') = d \cdot \sin(\theta') \cdot \cos(\varphi')/c$ stands for the time delay due to the distance d between array elements, and c is the velocity of light in free space.

Since there are q narrowband signals $s_\ell(t)$ impinging upon the array with different arrival directions $(\theta_\ell,\varphi_\ell)$, the input data vector may be expressed as:

$$\overline{x}(t) = \sum_{\ell=1}^{q} \overline{\alpha}_\ell(\theta_\ell,\varphi_\ell) \cdot s_\ell(t) + \overline{n}(t), \qquad (3)$$

where $\overline{n}(t)$ denotes the $M \times 1$ vector of noise at the M array elements. Using matrix notation:

$$\overline{x}(t) = \overline{A}(\theta',\varphi') \cdot \overline{S}(t) + \overline{n}(t), \qquad (4)$$

where $\overline{A}(\theta',\varphi') = \left[\overline{a}_1(\theta_1,\varphi_1)\ \overline{a}_2(\theta_2,\varphi_2) .... \overline{a}_q(\theta_q,\varphi_q)\right]$ is the $M \times q$ matrix of steering vectors and $\overline{S}(t) = [s_1(t)\ s_2(t) ...\ s_q(t)]^T$ is the $q \times 1$ vector of impinging signals.

If $s_d(t)$ denotes the desired signal arriving from the direction (θ_d,φ_d) upon the array and $s_j(t)$ ($j = 1,...,N_u$) denotes anyone of the $N_u (= q-1)$ undesired interfering signals arriving from (θ_j,φ_j), (3) may be rewritten as follows:

$$\overline{x}(t) = \overline{a}(\theta_d,\varphi_d) \cdot s_d(t) + \sum_{j=1}^{N_u} \overline{a}(\theta_j,\phi_j) \cdot s_j(t) + \overline{n}(t) \qquad (5)$$

where it should be noted that the directions of arrival are known a priori using a direction of arrival (DOA) algorithm.

Therefore, (5) requires an adaptive algorithm in order to estimate $s_d(t)$ from $\overline{x}(t)$ while minimizing the error between the array output $y(t)$ and the reference signal $d^*(t)$ approximating the desired signal $s_d(t)$. The signal $d^*(t)$ may be constructed at the receiver during the training period (the complex conjugate expression of the reference signal stands only for mathematical convenience).

If $y(k)$ and $d^*(k)$ denote the sample signals of $y(t)$ and $d^*(t)$ at time instant t_k, the array system's error is given by $\varepsilon(k)=d^*(k)-y(k)$ and the mean square error (MSE) $E[\varepsilon^2(k)]$ is given by:

$$E\left[\varepsilon^2(k)\right]=E\left[\left(d^*(k)-y(k)\right)^2\right]=E\left[\left(d^*(k)-\overline{w}^H\cdot\overline{x}(k)\right)^2\right]\Rightarrow$$

$$\Rightarrow E\left[\varepsilon^2(k)\right]=E\left[d^2(k)\right]-2\cdot\overline{w}^H\cdot\overline{r}+\overline{w}^H\cdot\overline{R}\cdot\overline{w} \quad (6)$$

where $\overline{r}=E[d^*(k)\cdot\overline{x}(k)]$ is the $M\times 1$ cross-correlation matrix between the desired signal and the input data vector and $\overline{R}=E[\overline{x}(k)\cdot\overline{x}^H(k)]$ is the $M\times M$ covariance matrix of the input data vector. The minimum mean square error can be obtained by setting the gradient vector of (6), with respect to w, equal to zero, i.e.

$$\nabla_w\left(E\left[\varepsilon^2(k)\right]\right)=0\Rightarrow -2\cdot\overline{r}+2\cdot\overline{R}\cdot\overline{w}=0. \quad (7)$$

Therefore the optimum solution for the weights is given by:

$$\overline{w}_{opt}=\overline{R}^{-1}\cdot\overline{r}, \quad (8)$$

which is referred to as the optimum Weiner solution.

2.2 Adaptive algorithms

2.2.1 Least Mean Squares (LMS) algorithm

A common adaptive algorithm that is well known [4] is the least mean squares (LMS) algorithm. It is based on the steepest-descent method, which is an optimization method [6] working recursively to compute and update the weights vector. The optimum value of weights vector means that the system's success to minimize the MSE. The successive corrections of the weights vector are in the direction of the negative of the gradient vector. Using steepest-descent method, the weight vector equation, at time instant t_k, is given [2] by:

$$\overline{w}(k+1)=\overline{w}(k)+\frac{1}{2}\mu\cdot\left[-\nabla\left(E\{\varepsilon^2(k)\}\right)\right], \quad (9)$$

where gain constant μ (step size) controls the convergence characteristics of the algorithm. Using (7), follows that:

$$\overline{w}(k+1)=\overline{w}(k)+\mu\cdot\left[\overline{r}-\overline{R}\cdot\overline{w}(k)\right]. \quad (10)$$

Using the instantaneous estimates (at t_k) of covariance matrices $\overline{R}$ and $\overline{r}$, instead of their actual values, the weights can then be updated as:

$$\overline{w}(k+1)=\overline{w}(k)+\mu\cdot\overline{x}(k)\cdot\left[d^*(k)-\overline{x}^H(k)\cdot\overline{w}(k)\right]\Rightarrow$$

$$\Rightarrow\overline{w}(k+1)=\overline{w}(k)+\mu\cdot\overline{x}(k)\cdot\varepsilon^*(t). \quad (11)$$

Note that this is a continuously adaptive approach, where the weights are updated as the data are sampled, so that the resulting weight vector sequence converges to the optimum solution. The LMS algorithm is initiated with the arbitrary value $\overline{w}(0)=0$. In order the weight vector to have stable convergence, μ must take values in the interval:

$$0<\mu<\lambda_{max}^{-1}, \quad (12)$$

where λ_{max} is the largest eigenvalue of the covariance matrix $\overline{R}$. The convergence of the algorithm is inversely proportional to the eigenvalue spread of the correlation matrix. If μ is chosen to be large, then the algorithm's convergence is faster, but may be less stable around the minimum value. On the contrary a small value of μ leads to slow convergence.

2.2.2 Sample Matrix Inversion (SMI) algorithm

One way to succeed a faster convergence rate is to use the sample matrix inversion algorithm, since it employs direct inversion of the covariance matrix $\overline{R}$ [7]. If a priori the desired and the interfering signal are known, then the optimum weights can be calculated directly using SMI. Supposing that the signals are known and the signal environment undergoes frequent changes, the adaptive system must update the weights vector continually, in order to meet the new requirements. Here it is considered that these changes do not happen during the processing of the block data, although the covariance matrices $\overline{R}$ and $\overline{r}$ arise in a finite observation interval:

$$\hat{\overline{R}}=\sum_{i=N_1}^{N_2}\overline{x}(i)\cdot\overline{x}^H(i), \quad (13a)$$

$$\hat{\overline{r}}=\sum_{i=N_1}^{N_2}d^*(i)\cdot\overline{x}(i), \quad (13b)$$

where N_1 and N_2 are the lower and upper limits of the observation interval, respectively. The weights vector can now be estimated by the equation:

$$\hat{\overline{w}}=\hat{\overline{R}}^{-1}\cdot\hat{\overline{r}}. \quad (14)$$

There is always a residual error in the SMI algorithm (since it is based on estimation):

$$\overline{\varepsilon}=\hat{\overline{R}}\cdot\overline{w}_{opt}-\hat{\overline{r}}. \quad (15)$$

This error is usually greater when compared to LMS error. One of disadvantages of this algorithm is that its stability depends on the ability to invert the large covariance matrix. In order to avoid a covariance matrix's singularity, the zero mean noise is added to the array response vector. Another disadvantage that has to be encountered is that even though the algorithm has faster convergence, huge matrix inversions lead to computational complexities that cannot be easily overcame.

2.2.3 Normalized Least Mean Squares (N-LMS) algorithm

Though other algorithms, such as the RLS one, may be more complex, less stable and more difficult to implement, they often converge much faster than the LMS. Improving the convergence speed, but keeping the simplicity of the LMS, has been the major consideration of much algorithmic research, resulting in the normalized least mean squares (N-LMS) algorithm. This algorithm, which has been studied widely and has various applications to mobile communications [8], is a variation of the constant-step-size LMS algorithm, using data-dependent step size at each iteration. For example, at the n-th iteration the step size is given by:

$$\mu(t) = \frac{\mu_0}{\overline{x}^H(t)\cdot\overline{x}(t) + a}, \qquad (16)$$

where μ_0 is a constant (usually ½) and a is a very small number introduced to prevent division by zero, if the product $\overline{x}^H(t)\cdot\overline{x}(t)$ is very small. This algorithm does not need an estimation of the correlation matrix's eigenvalues or trace, in order to select the maximum permissible step size. The N-LMS algorithm normally has better convergence performance and less signal sensitivity compared to the normal LMS algorithm.

2.2.4 Hybrid algorithm

In this work two hybrid algorithms combining the LMS or the N-LMS algorithm with the SMI one are proposed. Because both algorithms are characterized by similar concepts, only the hybrid LMS is introduced here. As mentioned before, LMS is a simple algorithm with possible slow convergence. LMS is a continuous algorithm, so it is well suited for continuous transmission systems. On the other hand, SMI has a very faster convergence speed, as it uses inversion of the correlation matrix, but is requiring a signal environment changing slightly during the process of a data block. Therefore, there is need for an algorithm that: a) is simple to implement, b) has fast convergence and c) is not computational intensive.

In the LMS algorithm, as discussed previously, weights are initialized arbitrarily with $\overline{w}(0) = 0$ and then are updated using (11). Due to arbitrary weights initialization, the LMS takes longer to converge. In order to speed up convergence, an initial weights vector, that has been come through the SMI algorithm, is used. So, using only a small block of incoming data derives the weights vector. Therefore the initial weights vector is:

$$\overline{w}_{in} = \hat{\overline{R}}_p \cdot \hat{\overline{r}}_p, \qquad (17)$$

where the covariance and correlation matrices are:

$$\hat{\overline{R}}_p = \sum_{i=1}^{p} \overline{x}(i)\cdot\overline{x}^H(i), \qquad (18a)$$

$$\hat{\overline{r}}_p = \sum_{i=1}^{p} d^*(i)\cdot\overline{x}(i), \qquad (18b)$$

where p is the block size, taken to be small in order to encounter the probability of changes in the signal environment. Besides, large block will need large matrix's inversion, resulting in a computationally complex method. After the initial weights vector derivation, the LMS algorithm is implemented. At time instant t_k, the weights vector update is:

$$\overline{w}(k+1) = \overline{w}(k) + \mu\,\overline{x}(k)\left[d^*(k) - \overline{x}^H(k)\cdot\overline{w}(k)\right] \qquad (19)$$

with initial weights vector $\overline{w}(0) = \overline{w}_{in}$.

When the LMS algorithm begin adaptation, the antenna beam has already steered close to the approximate direction of the desired signal. Therefore LMS algorithm takes less time to converge. After that, even if the signal environment changes, the hybrid algorithm is able to encounter these changes.

3 Simulation and Results

In this section, the performance of the algorithms described in the previous section, is shown and compared. For simulation purposes, a 9-element linear array antenna is used, with its individual elements placed along the x-axis with a half-wave length distance between them. The full wavelength distance $\lambda = c/f_c$ is computed with $f_c = 900\,\text{MHz}$ as the working frequency of the communication system. Fig. 2 shows that increasing the elements of the antenna array, the system manages to steer its main lobe in the direction of the interesting signal and to reject the incident interferences from other propagation directions. So, for the sake of simplicity and cost we use a 9-element linear array antenna.

The desired signal is a simple sinusoidal signal with frequency f_1. All interfering signals are also sinusoidal signals with frequency f_2 close to f_1.

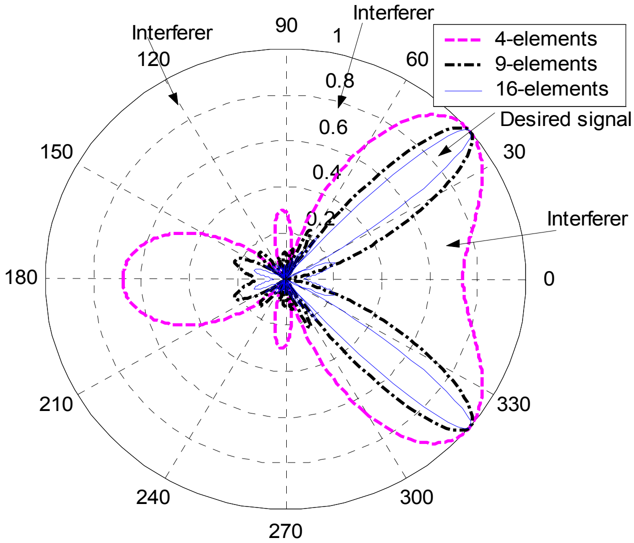

Figure 2. Beam patterns for various antenna elements.

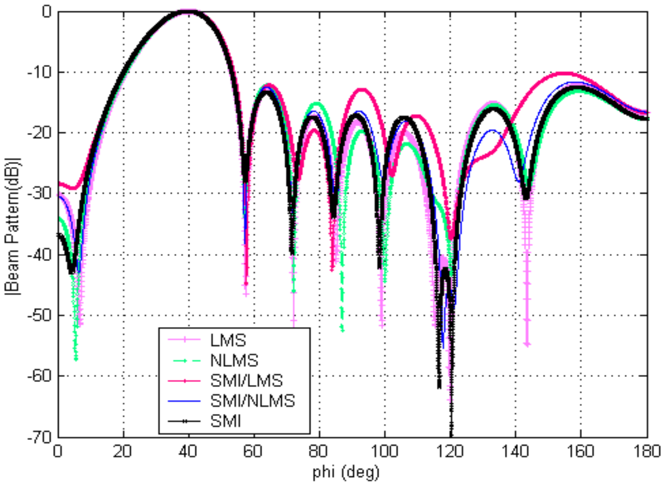

Figure 3. Beam patterns vs. azimuth angle φ for the presented algorithms, when $\theta = \pi/2$.

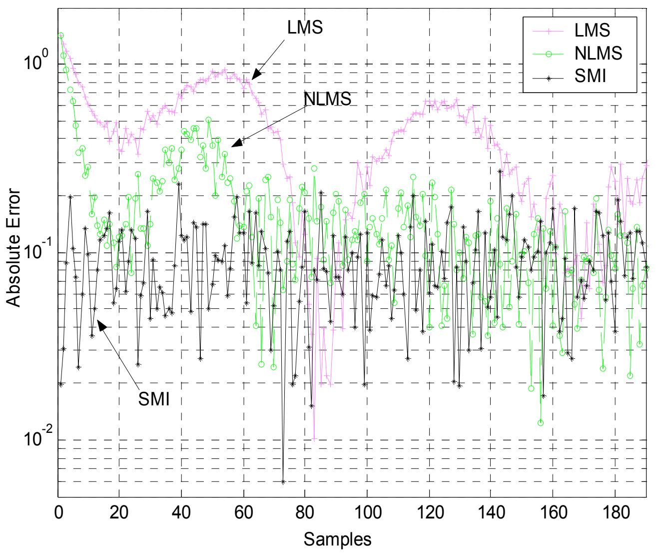

Figure 4. Absolute error of SMI, LMS and N-LMS algorithms vs. the number of samples used.

The SNR is taken to be 10 dB, while the signal to interference ratio (SIR) is taken to be –5 dB for all interferes. The signal of interest is considered to be impinging to the antenna array from the direction (θ_0, φ_0), while interference signals are impinging from different directions (θ_i, φ_i), where elevation (or azimuth) incidence angles θ (or φ) take values in the interval $\pi/2<\theta<\pi$ (or $0<\varphi<2\pi$). In the present simulations we consider 3 interferers and white gaussian noise with zero mean and variance $\sigma^2 = 0.1$. Finally for testing we use a block of 3,000 samples.

Fig. 3 shows the φ-plane of beam patterns in case of the presented algorithms. It is assumed that signal of interest impinges to the antenna array from direction (100°, 40°), while the three interference signals from coming from directions (97°, 10°), (92°, 80°), and (97°, 120°) (all used in radians). As can be seen, all algorithms manage to steer the main lobe in the direction of interest and force nulls to the directions of all interferers.

Fig. 4, which presents the absolute error of the adaptive algorithms, shows that the SMI algorithm converges immediately, even though it remains unstable. The rest four algorithms are more stable than the SMI, while the faster convergent algorithm appears to be the normalized LMS (N-LMS).

The convergence speed is shown in Fig. 5, where the mean square error is plotted. The normalized-LMS convergences faster than LMS. At the same time, the hybrid NLMS excels over the hybrid LMS. At the end the mean square error of NLMS and hybrid NLMS value less than that of LMS and hybrid LMS, respectively.

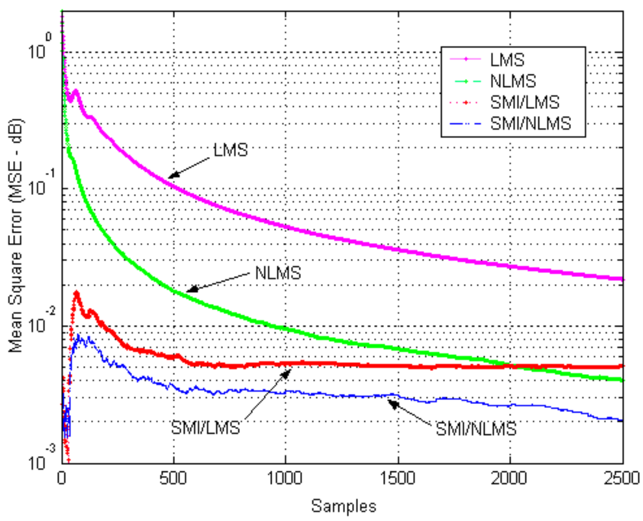

Figure 5. MSE vs. the number of samples, for the proposed algorithms (1st scenario).

In Fig. 6, the mean square error is plotted for the case that the white gaussian noise is added to the signal, before this impinges to the antenna array (second scenario). The NLMS has faster convergence than LMS, and the same also happens with hybrid NLMS and hybrid LMS. But the mean square error results in higher values than the values of the previous scenario. This, however, was expected because the system cannot significantly determine the transmitted signal, and this is why communication systems use training symbols. As a consequence in Fig. 7 the weight estimation is characterized by instability, especially for the NLMS algorithms, since the weights updating are more dependent on the incoming signal than for the rest algorithms.

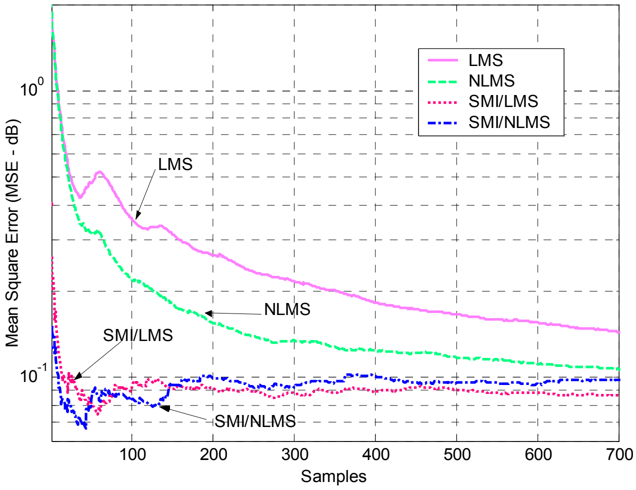

Figure 6: MSE vs. the number of samples, for the proposed algorithms (2nd scenario).

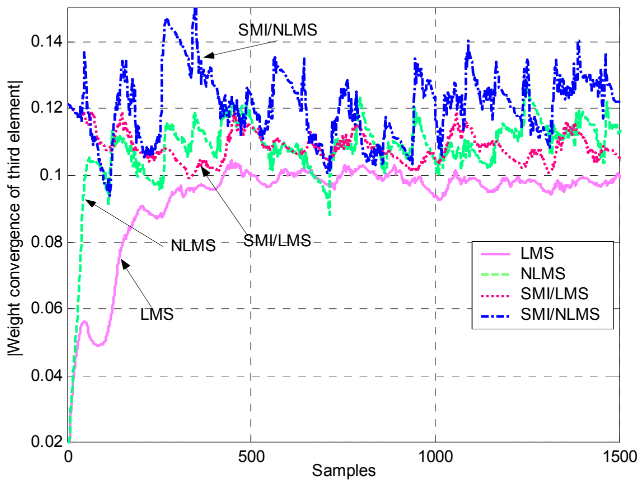

Figure 7. A weight convergence vs. the number of samples, for the proposed algorithms (2nd scenario).

4 Conclusions

In this work, the SMI, LMS, N-LMS, hybrid LMS and hybrid N-LMS adaptive algorithms were investigated. Their ability to respond automatically to an unknown interference environment by steering nulls in the direction of interferences and main lobe in the direction of interest, was presented. Simulation results were also provided in order to understand various aspects, such as convergence and stability, of these algorithms.

Two scenarios were developed. In the first, where noise was added after the signal impingement to the antenna, SMI was found to have the faster convergence with higher complexity. Between the N-LMS and LMS, it was found that the former has faster convergence, but lower MSE value. Based on SMI, N-LMS and LMS, two hybrid algorithms with the simplicity of LMS-family and the convergence speed of SMI were proposed. In the second scenario, where the white gaussian noise was added on the signal before impingement to the antenna array, even if all algorithms manage to converge, their mean square errors result in higher values than the values of the previous scenario.

Acknowledgments

This work was supported by the Greek Ministry of National Education and Religious Affairs and the European Union under the ΕΠΕΑΕΚ II projects: "Archimedes – Support of Research Groups in TEI of Crete – Smart antenna study & design using techniques of computational electromagnetics and pilot development & operation of a digital audio broadcasting station at Chania (SMART-DAB)" and "Reformation of the Electronics Dept's syllabus".

References

[1] B. Widrow and S.D. Stearns, *Adaptive Signal Processing,* Prentice Hall, 1985.
[2] S. Haykin, *Introduction to Adaptive Filters,* Macmillan Publishing Company, 1985.
[3] R.T Compton, *Adaptive Antennas: Concepts and Applications,* Prentice Hall, 1988.
[4] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 1991.
[5] B. Widrow and E. Walach, *Adaptive Inverse Control*, Prentice Hall, 1996.
[6] W. Murray, *Numerical Methods for Unconstrained Optimization*, Academic Press, 1972.
[7] I.S. Reed, J.D. Mallett and L.E. Brennen, "Rapid convergence rate in adaptive arrays", *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-10, pp.853-863, Nov. 1974.
[8] D.T.M. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms", *IEEE Trans. Signal Process.,* vol. 41, pp. 2811-2825, 1993.