AN INNOVATIVE COMPUTATIONAL APPROACH BASED ON A PARTICLE SWARM STRATEGY FOR ADAPTIVE PHASED-ARRAYS CONTROL

M. Donelli, R. Azaro, F. De Natale, and A. Massa

March 2006

Technical Report # DISI-11-008

An Innovative Computational Approach based on a Particle Swarm Strategy for Adaptive Phased-Arrays Control

Massimo Donelli, Renzo Azaro, Francesco De Natale, and Andrea Massa

Department of Information and Communication Technology
University of Trento, Via Sommarive 14, 38050 Trento - Italy
Tel. +39 0461 882057, Fax +39 0461 882093
E-mail: {*andrea.massa*, *francesco.denatale*}*@ing.unitn.it,*
{*massimo.donelli*, *renzo.azaro*}*@dit.unitn.it*
Web-site: *http://www.eledia.ing.unitn.it*

An Innovative Computational Approach based on a Particle Swarm Strategy for Adaptive Phased-Arrays Control

Massimo Donelli, Renzo Azaro, Francesco De Natale, and Andrea Massa

Abstract

In this paper a new approach to the control of phased arrays is presented and assessed. Starting from the adaptive array theory, a particle swarm strategy is used to tune the phase coefficients of the array in order to adaptively minimize/avoid the effects of interfering signals at the receiver. To show the effectiveness of the proposed approach, a selected set of numerical examples, concerned with linear as well as planar arrays, is presented. Furthermore, to evaluate the advantages of the PSO-based strategy over state-of-art methods, a comparative study is carried out by analyzing the performance of the method in terms of both the signal-to-interference-plus-noise-ratio and resulting beam pattern. The achieved results, even though preliminary, seem to confirm that the PSO-based approach satisfactorily works and it generally outperforms previously proposed/state-of-art phase-only adaptive control strategies.

Key words:

Adaptive Control, Phased Arrays, Particle Swarm Optimizer.

1 Introduction

In the last years there has been a growing interest in the design and application of phased-arrays for remote sensing radar systems [1], military and commercial communication equipments [2]. Reconfigurable phased-array antennas, which are able to receive in a specific and tunable direction by excluding other undesired signals coming from different incidence angles, are commonly used in several fields such as airport surveillance, missile detection and tracking [3]. Adaptive phased-arrays separate the desired signal from interfering ones by continuously tuning the weights of the array in order to place nulls in the beam pattern and to avoid the effects of interferences. In the scientific literature (see [4][5] and the references cited therein), several methodologies for the adaptive phase-array control have been proposed aimed at defining suitable strategies for the optimal synthesis of array weights in terms of amplitude as well as phase. In most of them, the values of the complex weights for placing suitable (i.e., the directions of the interferences) nulls in the far field pattern of the antenna are determined by multiplying the quiescent weights by the inverse of the covariance matrix obtained from the signals received at each element of the array. Although mathematically elegant and fast, these strategies turn out to be impractical or very difficult to be implemented because of the costs of the hardware requirements. As a matter of fact, these solutions require an expensive receiver or correlator at each element of the array. Unfortunately, several implementations have a single receiver at the output of the summer. Thus, there is the need of an antenna customized for the algorithm in hand. Moreover, the receivers require a sophisticated calibration. As far as the use of variable analog amplitude and phase weights at each element of the array is concerned, a phased array has usually only digital beam steering phase shifter at each element and the amplitude coefficients are determined by a fixed feed network.

Consequently, in order to limit the cost of the hardware equipment by using commercially available components, there is the need of operating in the digital domain by only acting on the phase terms of the array elements. Such a choice turns out to be a cheap solution for a real implementation because it resorts to existing (i.e., non-customized) and standard array architectures and components (digital phase shifters) without expensive additions as adjustable amplitude weights or correlators. Within such a framework, effective approaches have been proposed where the control problem has been reformulated as an optimization one and successively solved through an evolutionary strategy based on a genetic algorithm (GA). In [6], the author describes an innovative method for the adaptive phase-nulling of linear arrays where a GA adjusts some of the least significant bits of the beam steering phase shifters to minimize the total output power. To face the problem of the readaption of GA to new environments, once its population has converged, *Weile et al.* [7] proposed an algorithm, which uses diploid individuals and dominance relation to exploit learning and memory capabilities. As far as the exploitation of learning and memory capabilities are concerned, *Massa et al.* developed an enhanced evolutionary method (indicated in the following as CGA) for linear [8] as well as planar geometries [9] where customized operators and strategies have been implemented to allow fast convergence and improved adaptation capabilities.

Recently, a new stochastic algorithm called particle swarm optimizer (PSO) [10] has been shown to be a valuable addition to the electromagnetic design engineer's toolbox [11][12][13][14]. The conceptual bases of the GA and PSO rest upon two different paradigms. Unlike GAs, the PSO is based upon the cooperation among the agents/trial-solutions and not on their competition. In general, one advantage of the PSO over the GA is its more "*easy*" implementation. As a matter of fact, a GA requires the definition of a suitable strategy for the application of the genetic operators (e.g., the choice of the crossover and mutation probabilities) as well as the choice of the best implementation of the operator for the application in hand (e.g., tournament selection or proportional selection, single-point crossover or multi-point crossover). On the contrary, the updating equations for the PSO are sequentially applied at each individual and the process for selecting the best operator is eliminated.

Moreover, other differences occur for the calibration of the control parameters. For GA, the most important are the population size, the crossover probability, and the gene-mutation and chromosome-mutation rates. For PSO, the swarm size, the inertial weight and the acceleration terms. In general, manipulating these latter is easier than changing various operators and their occurrence. Furthermore, there exist many comprehensive studies on the effects of PSO parameters that makes their selection even easier (see [13][15] and the references therein).

Another key-issue for the application of a PSO-based procedure to the phased-array real-time tuning lies in its ability to control the convergence of the optimization as well as its stagnation. Stagnation occurs in the GA when eventually all the agents possess the same genetic code. In that case, the gene pool is so homogeneous that, as pointed out in [16] and detailed in [13], although crossover and mutation rate can affect the convergence of the optimization there is little or no possibility to explore other regions of the solution space and only a lucky mutation can generate a different optimal individual thus avoiding stagnation. Unlike GAs, the PSO allows a more significant level of control by decreasing the *inertial weight* (χ) during the optimization process. As a matter of fact, higher values χ produce relatively straight particle trajectories resulting in a good global search characteristic. On the other hand, small values for χ encourage a local searching and the homogeneity of the swarm. Consequently, it is generally useful decreasing the inertial coefficients during the minimization process to avoid a premature stagnation and to allow a refined local search only at the end of the optimization [17][18].

Therefore, it might be profitable to evaluate the effectiveness of the PSO in dealing with real-time control where GAs have found great success and widespread implementation. For such a purpose, this paper proposes a computational approach based on a customized PSO for the phased-array dynamic control. In general, standard versions of the particle swarm deal with continuous real values where trajectories in the solution space are defined as changes in positions on some number of dimensions. For this application, a binary version of the optimizer is designed. In such a case, the particle position coordinates encode the vertexes of a multi-dimensional hypercube and the trajectories are changes in the probability that a coordinate will take on zero or one value.

The paper is structured as follows. In Section 2, a suitable cost function is defined according to the Applebaum's theory to recast the phased-array control as an optimization problem. Then, a detailed description of the PSO-based computational procedure is reported (Sect. 3). As far as the assessment is concerned, the results of a selected set of numerical experiments, concerned with linear as well as planar structures, are shown in Sect. 4. Final comments and conclusions are drawn in Sect. 5.

2 Mathematical Formulation

Let us consider an array of M elements equally-spaced on a planar lattice, as shown in Figure 1. The desired received signal at the m-th element can be expressed as:

$$S_m^d(t) = p_d(t)\, e^{j\beta_m^d} \quad m = 1, \dots, M \tag{1}$$

where $\beta_m^d = \frac{2\pi}{\lambda}(u_d x_m + v_d y_m + z_d z_m)$, $u_d = sin\theta_d\, cos\phi_d$, $v_d = sin\theta_d\, sin\phi_d$, $z_d = cos\theta_d$, and x_m, y_m, z_m are the Cartesian coordinates of the m-th element of the array. Moreover, λ is the free-space wavelength and θ_d and ϕ_d are the polar coordinates that define the direction of arrival (DOA) of the desired signal characterized by an envelope $p_d(t)$.

By considering a real environment, a variable number of interfering signals $S_i(t)$, $i = 1, ..., I$ also impinges on the array and they contribute at the m-th sensor with an additive term

$$S_m^i(t) = p_i(t)\, e^{j\beta_m^i} \quad \begin{array}{c} m = 1, \dots, M \\ i = 1, \dots, I \end{array} \tag{2}$$

where $\beta_m^i = \frac{2\pi}{\lambda}(u_i x_m + v_i y_m + z_i z_m)$ and $p_i(t)$ is the envelope of the j-th interference centered at the same angular frequency ω_d of the desired signal. Moreover, desired as well as interference signals are assumed to be narrow-band ones.

The description of the scenario under analysis is completed by considering a noise contribution modeled with an additive gaussian process (AWGN) characterized by a power $\wp_n$.

Under these assumptions, the covariance matrices of dimension $M \times M$ related to the desired ($\Im \Leftarrow d$) or to the i-th interfering signal ($\Im \Leftarrow i$), can be expressed as follows

$$\Phi_{\Im} = E\left\{ \sum_{m=1}^{M} \sum_{n=1}^{M} S_m^{\Im *}(t) S_n^{\Im}(t) \right\} \tag{3}$$

where $E\{.\}$ indicates the expectation operator, and the asterisk $*$ denotes the complex conjugate. As far as the covariance matrix of the noise is concerned, it turns out to be

$$\Phi_n = p_n 1^M \tag{4}$$

1^M being the M-dimensional identity matrix.

Moreover, the covariance matrix of the undesired signal at the receiver is equal to

$$\Phi_u = \sum_{i=1}^{I} \Phi_i + \Phi_n \tag{5}$$

and its power contribution results [4]

$$\wp_u = \frac{1}{2} \underline{W}^{T*} \Phi_u \underline{W} \tag{6}$$

where T means transpose and $\underline{W}$ is given by

$$\underline{W} = \left\{ w_m e^{j\phi_m};\; m = 1, ..., M \right\} \tag{7}$$

w_m being the m-th amplitude coefficient, while ϕ_m indicates the phase shift of the m-th array element.

Furthermore, the power contribution of the desired signal at the receiver is given by

$$\wp_d = \frac{1}{2} p_d^2(t) \left| \underline{W}^T \underline{U}(\theta_d, \phi_d) \right|^2 \tag{8}$$

where $\underline{U}(\theta_d, \phi_d)$ is a vector whose m-th element is equal to $U_m(\theta_d, \phi_d) = e^{j\beta_m^d}$. Thus, according to (6) and (8), the signal-to-interference-plus-noise-ratio (SINR) turns out to be

$$\Psi(\underline{W}) \triangleq \frac{\wp_d}{\wp_u} = \frac{p_d^2(t) \left| \underline{W}^T \underline{U}(\theta_d, \phi_d) \right|^2}{\underline{W}^{T*} \Phi_u \underline{W}} \tag{9}$$

and its maximization, with respect to $\underline{W}$, represents the goal of any adaptive control procedure. Unfortunately, since Φ_u and $p_d(t)$ are not known and they cannot be directly measured, (9) is not available. However, as shown in [7], the problem can be recast as the maximization of a computable cost function $f(\underline{W})$

$$f(\underline{W}) = \frac{\left| \underline{W}^T \underline{U}(\theta_d, \phi_d) \right|^2}{\underline{W}^{T*} \Phi_t \underline{W}} \tag{10}$$

where $\Phi_t = \Phi_d + \sum_{i=1}^{I} \Phi_i + \Phi_n$ is a quantity measurable at the receiver. Because of the time-varying nature and complexity of (10), it is convenient to address the problem by means of a suitable global optimization procedure. Towards this end, an innovative strategy based on a new paradigm in the electromagnetic community will be presented in the following section.

3 The optimization approach

The PSO is a multiple-agents optimization algorithm developed by Kennedy and Eberhart [19][10] in 1995 that imitates the social behavior of groups of insects and animals such as swarms of bees, flocks of birds, and shoals of fish. The standard PSO implementation considers a swarm of P trial solutions (called *particles*). Each particle flies in the solution space by improving its position according to suitable updating equations. on the basis of information on each particle's previous best performance and the best previous performance of its neighbors. For real-number spaces, the trajectories of the particles are defined as changes in the positions on some number of dimensions. In the binary version, trajectories are changes in the probability that a coordinate will take on a zero or one value [10][20].

The following steps summarize the application of the binary PSO strategy by focusing on its customization to the phase-only on-line adaptive array control. Let us consider a generic timestep (t) corresponding to a fixed interference scenario.

- **Step 0 - Coding**. Since phased-arrays are taken into account and the m-th element of the array is controlled through a L-bit digital phase shifter, the p-th binary trial solution

$$\underline{\zeta}^p = \left\{\varphi_m^{p,l};\ l = 1, \dots, L;\ m = 1, \dots, M\right\} \quad p = 1, ..., P \tag{11}$$

codes a sequence of quantized phase values ϕ_m^p, $m = 1, ..., M$

$$\phi_m^p = \left(\frac{\phi_{max} - \phi_{min}}{2^L - 1}\right) \sum_{l=1}^{L} 2^{l-1} \varphi_m^{p,l} + \phi_{min} \tag{12}$$

where ϕ_{max} and ϕ_{min} are the maximum and minimum range bounds corresponding to the parameter ϕ_m, and $\varphi_m^{p,l}$ is the binary bit in the l-th place along the encoded representation of the ϕ_m^p parameter.

With each "*position*" vector $\underline{\zeta}^p$ is associated a velocity vector $\underline{v}^p = \left\{v_m^{p,l};\ l = 1, \dots, L;\ m = 1, \dots, M\right\}$, which models the capacity of the particle to fly from a given position $\underline{\zeta}_k^p$ to another position $\underline{\zeta}_{k+1}^p$ in a successive iteration of the space-solution sampling process. Moreover, each $v_m^{p,l}$ represents the probability of $\varphi_m^{p,l}$ taking value 1. If $v_m^{p,l}$ is higher, $\varphi_m^{p,l}$ is more likely to assume the value 1, and lower values favor the 0 choice.

- **Step 1 - Initialization.** At the beginning of the maximization process ($k = 0$) and according to the PSO working strategy, the positions of the P particles of the swarm $\Gamma_0 = \left\{\underline{\zeta}_0^p;\, p = 1, ..., P\right\}$ as well as their velocities $V_0 = \{\underline{v}_0^p;\, p = 1, ..., P\}$ are randomly generated according to the following rules

$$\varphi_{m,0}^{p,l} = \begin{cases} 1 & if\ \rho_{m,0}^{p,l} \geq 0.5 \\ 0 & otherwise \end{cases}, \quad v_{m,0}^{p,l} = \begin{cases} 1 & if\ \sigma_{m,0}^{p,l} \geq 0.5 \\ 0 & otherwise \end{cases} \tag{13}$$

  where $\rho_{m,k}^{p,l}$ and $\sigma_{m,k}^{p,l}$ are random numbers drawn from a uniform distribution between 0 and 1.

- **Step 2 - Fitness evaluation.** The degree of "optimality" of each particle is evaluated at the k-th iteration by computing its cost function value $\underline{f}_k^p = f\left\{\underline{W}\left(\underline{\zeta}_k^p\right)\right\}$. Moreover, the best position reached up to now by the p-th particle as well as the optimal position in the overall swarm are stored in the "*previous best*" particle $\underline{\xi}_k^p = arg\left(max_{h=1,...,k}\left[f\left\{\underline{W}\left(\underline{\zeta}_h^p\right)\right\}\right]\right)$ and in the "*global best*" particle $\underline{\varsigma}_k = arg\left(max_{p=1,...,P}\left[f\left\{\underline{W}\left(\underline{\xi}_k^p\right)\right\}\right]\right)$, respectively.

- **Step 3 - Iteration updating.** The iteration index is updated, $k = k + 1$.

- **Step 4 - Termination criterion.** If the maximum number of iterations K (admissible in a timestep t) is reached ($k = K$) or if the optimal fitness is under a given threshold η ($f\left\{\underline{W}\left(\underline{\varsigma}_k\right)\right\} \leq \eta$), then the optimization process is stopped and $\underline{\varsigma}_k$ is assumed as the problem solution. Moreover, in order to improve the "*reaction*" of the algorithm to environmental changes occurring between consecutive timesteps, fully exploiting the similarities among the scenario conditions at different timesteps, the optimal particle is stored in a finite-length buffer (B being the buffer length) whose elements are updated at each timestep t as follows: $\left.\underline{\gamma}_B\right\rfloor_t = \underline{\varsigma}_k$ and $\left.\underline{\gamma}_b\right\rfloor_t = \left.\underline{\gamma}_{b+1}\right\rfloor_{t-1}$,

$b = 1, ..., B$. These B trial solutions are evaluated during the optimization when the reliability of the system (in terms of $SINR$) degrades and according to the *learning strategy* described in [9].

Otherwise, the Step 5 is done.

- **Step 5 - Velocity updating.** The velocity of each particle is updated according to the following relation:

$$v_{m,k}^{p,l} = \begin{cases} v_{max} & if\ v_{m,k}^{p,l} > v_{max} \\ -v_{max} & if\ v_{m,k}^{p,l} < -v_{max} \\ v_{m,k}^{p,l} & otherwise \end{cases} \tag{14}$$

where v_{max} is a constant clamping value [10], which is similar to the mutation rate in GAs. Unlike continuous-valued PSO, where increasing the clamping value enlarges the region of the solution space explored by a particle, in the binary version, a smaller v_{max} allows a higher mutation rate. In general, v_{max} is set at 4.0 to ensure that there is always some chance that $\varphi_{m}^{p,l}$ will change state. Moreover,

$$v_{m,k}^{p,l} = \chi v_{m,k-1}^{p,l} + a_1 \rho_1 \left\{ \xi_{m,k}^{p,l} - \varphi_{m,k}^{p,l} \right\} + a_2 \rho_2 \left\{ \delta_{m}^{l} - \varphi_{m,k}^{p,l} \right\} \tag{15}$$

where $\underline{\delta} = arg\left(max_{b=1,...,B} \left[f \left\{ \underline{W}\left(\underline{\gamma}_b \right) \right\} \right] \right)$; ρ_1 and ρ_1 are two positive random numbers drawn from a uniform distribution with a predefined upper limit often set so that $\rho_1 + \rho_2 = 4.0$; a_1 and a_2 are constants called *cognition* and *social* acceleration [15], respectively.

- **Step 6 - Position updating.** The particle position is then updated as follows

$$\varphi_{m,k}^{p,l} = \begin{cases} 1 & if\ \rho_{m,k}^{p,l} < S\left(v_{m,k}^{p,l} \right) \\ 0 & otherwise \end{cases} \tag{16}$$

where $S\left(.\right)$ indicates the sigmoid function

$$S\left(v_{m,k}^{p,l}\right)=\frac{1}{1+exp\left(-v_{m,k}^{p,l}\right)} \tag{17}$$

Therefore, the probability that $\varphi_{m,k}^{p,l}=1$ is equal to $S\left(v_{m,k}^{p,l}\right)$ and that $\varphi_{m,k}^{p,l}=0$ is equal to $\left[1-S\left(v_{m,k}^{p,l}\right)\right]$. Moreover, the probability that $\varphi_{m,k}^{p,l}$ changes its value ($0\rightarrow 1$ or $1\rightarrow 0$) turns out to be equal to $\left\{S\left(v_{m,k}^{p,l}\right)\left[1-S\left(v_{m,k}^{p,l}\right)\right]\right\}$.

Then go to Step 2.

4 Numerical Results

In this section, the capabilities of the proposed PSO-based array control procedure will be assessed by presenting the results of a selected set of numerical experiments.

In the numerical validation, the proposed approach has been evaluated by considering linear (Section 4.1) and planar (Sections 4.2-4.3) arrays working in a realistic scenario characterized by a background Gaussian noise of $30\,dB$ below the desired signal (coming from $\theta_d=0^o$) and interferences of amplitude $30\,dB$ above the desired signal uncorrelated with each other, the desired signal, and the noise.

Moreover, the obtained results have been compared with those obtained with other state-of-the-art procedures: (a) the optimal Applebaum approach [4] (where array amplitudes and phases are simultaneously tuned), (b) a modified version of the Applebaum approach where the Applebaum's phases are quantized (DPA), (c) the customized version of a GAs-based control strategy proposed in [21] (CGA), and (d) the Least-Mean-Square algorithm [5] (LMS).

Whatever the phased-array control method, the array weights have been set to $w_m=1$, $m=1,...,M$, while the phase coefficients have been iteratively tuned in the range $\phi_{min}=0^o$ and $\phi_{max}=360^o$ making use of digital phase shifters characterized by $N_{bit}=6$ as in [7].

As far as the multiple-agents control strategies are concerned (GAs and PSO), a population of $P = M$ trial solutions has been considered and $K = 20$. Moreover, the following configuration has been heuristically (and according to the guidelines suggested in the related literature [10][13]) assumed for the PSO parameters: $a_1 = a_2 = 2.0$, $B = \frac{P}{10}$, and χ linearly varying from 0.9 to 0.4 over the course of the optimization run. As an example, Fig. 2 shows the results (in terms of averaged $SINR$ over $T = 100$ timesteps) of the calibration process of χ when dealing with the geometry and scenario described at Sect. 4.1.

On the other hand, the same parametric configuration chosen in [9] has been adopted for the GA-based procedure.

4.1 Linear Array

The first test case deals with the run-time optimization of a $M = 20$ elements half-wavelength-spaced linear array. During the repeated executions (since the results expressed in terms of $SINR$ are based on the execution of the algorithm for 50 independent realizations of the noise and interference stochastic processes), the directions θ_i of jamming signals have been supposed to be random variables uniformly distributed with an arrival-time modeled through a Poisson random model [22] characterized by $\lambda = 1$ and with a life-time of two time-steps [8] (see Fig. 3 where a representative example of a stochastic realization of the interference arrival process is given). In particular, Fig. 3(a) - red crosses - shows a sample of arrival angles of interfering signals at each timestep. For completeness, the number of interfering signals received at each time-step is depicted in Fig. 3(b).

The behavior of the $SINR$ (computed by calculating a running average over 50 past timesteps) as a function of the timestep index is shown in Figure 4(a) where the results obtained with the PSO-based procedure are compared with those of others state-of-the-art methods. As can be observed, the plots point out the effectiveness of the PSO-based procedure. The average $SINR$ over all generations and simulations is equal to

$av\,\{SINR\}|_{PSO} = 19.61\,dB$ for PSO and $av\,\{SINR\}|_{CGA} = 14.93\,dB$ for CGA-method (Tab. I). This difference of about $4\,dB$ is quite significant, especially given that most points of the two graphs in Fig. $4(a)$ fall in the range $[7\,dB,\ 22\,dB]$ (a $15\,dB$ dynamic range).

For sake of thoroughness, Figs. $4(c)$-(e) give some indications on the synthesized far-field patterns starting from the quiescent pattern displayed in Fig. $4(b)$. These results refer to different timesteps where representative jamming configurations occur. More in detail, such figures are concerned with the following scenarios: two jamming signals at $\theta_1 = -28^o$ and $\theta_2 = 30^o$ [$t = 396$ - Fig. $4(c)$], a single interference at $\theta_1 = 68^o$ [$t = 604$ - Fig. $4(d)$], and three jammers at $\theta_1 = 40^o$, $\theta_2 = -86^o$, and $\theta_3 = 83^o$ [$t = 802$ - Fig. $4(c)$]. In general, multiple-agents approaches (CGA and PSO) quickly place nulls in the far-field pattern (near or at the same angular position where the interferences impinge on the array) only slightly perturbing the main lobe thus outperforming other deterministic or sub-optimal techniques. Moreover, the improvement in the $SINR$ results from the deeper nulls which are placed by the PSO-based approach.

4.2 Planar Arrays

The second experiment is concerned with a $M = 10 \times 10$ half-wavelength-spaced planar array whose geometry is the same as that analyzed in [9]. Concerning the angular coordinates of the jammers, they have been modeled by considering the same stochastic model of the previous example and Fig. $3(a)$ shows an example of their distribution versus the timestep index t (ϕ - green crosses, θ - red crosses).

In order to give some information on the results of the comparative study, Figure 5 shows the behavior of the running average of the $SINR$ during $T = 250$ timesteps. As shown in Table II, the PSO-based approach generally outperforms other techniques. More in detail, the achieved enhancement can be quantified on average in $5\,dB$ over the CGA-based procedure and of about $10\,dB$ in comparison with the DPA sub-optimal approach. Such an event can be better appreciated by observing the resulting far-field patterns.

As representative examples, Fig. 6 shows some samples of the elevation beam patterns at various timesteps: (a) $t = 55$ [the jammer impinges on the array from the direction $(\theta_1 = -59^o,\ \phi_1 = 173^o)$], (b) $t = 105$ [the jammer impinges on the array from the direction $(\theta_1 = 43^o,\ \phi_1 = 132^o)$], and $t = 205$ (c)-(d) [two jammers impinge on the array from the directions $(\theta_1 = -39^o,\ \phi_1 = 8^o)$ and $(\theta_1 = 46^o,\ \phi_1 = 60^o)$]. By comparing the plots relating to different methodologies and as expected from the indications drawn from Fig. 5, it turns out that the PSO-based procedure is able to place nulls exactly in correspondence with interfering signals (as pictorially shown in Fig. 6 where a color-level representation of the synthesized beam patterns is displayed), while other methodologies (without considering the optimal synthesis) generally make an error of some degrees with respect to the correct position [e.g., $\Delta\theta|_{CGA} \simeq 5^o$ and $\Delta\theta|_{LMS} \simeq 9^o$ - Fig. 6(a)] or reduce the null depth [see Fig. 6(b)].

Moreover, by comparing the quiescent beam pattern [Fig. 7(d)] and those synthesized with the PSO at various timesteps [Figs. 7(a)-7(c)], it is interesting to note that even though the interferences impinge on a high side-lobe of the original pattern the remaining part of this is preserved after the array-control synthesis.

To further assess the effectiveness of the proposed approach in dealing with planar configurations, the last experiment deals with the elliptical geometry of $M = 160$ elements proposed in [23] and shown in Figure 8.

Once again, the feasibility and capabilities of the PSO-based approach are confirmed as pointed out by the plots of the running average $SINR$s shown in Fig. 9 and quantified on average in Tab. III.

For completeness, Fig. 10 shows the beam patterns synthesized by using the PSO strategy as well as the quiescent one [Fig. 10(d)].

5 Conclusions

In this paper, the feasibility and effectiveness of the application of an innovative multiple-agents technique to the adaptive and continuous optimization of antenna arrays controlled with only digital phase shifters have been investigated. Starting from an assigned quiescent pattern and by assuming some *a-priori* knowledge on the desired signal (i.e., the direction of arrival), the proposed approach demonstrated a non-negligible improvement over state-of-the-art techniques.

The approach is based on the Applebaum's theory and it recasts the control-problem as an optimization one to be iteratively solved through the maximization of a suitable cost function. Towards this end, a new evolutionary paradigm has been adopted and customized (i.e., the binary particle swarm optimizer) since the particle swarm optimizer was generally used in continuous spaces and, at best of our knowledge, never in the binary version in presence of a varying scenario. Thus, a suitable strategy has been implemented taking into account the need of an enhancement in the "memory" features of the original algorithm for dealing with a time-varying and complex scenario.

Future work will be aimed at further assessing the capabilities of PSO-based procedure in dealing with different scenarios, such as multi-path interference, and with different background noise models for enabling a timely and effective development of smart-sensors and smart-environments in the framework of applied sciences.

Acknowledgments

A. Massa wishes to thank E. Vico for useful discussions and suggestions. Moreover, the authors are grateful to Ing. S. Piffer and Ing. E. Zeni for kindly providing some numerical results of computer simulations.

This work has been partially supported in Italy by the *Center of REsearch And Telecommunication Experimentations for NETworked communities* (CREATE-NET) and by the "*WILMA - Wireless Internet and Location Management Architecture*" - Fondo Progetti

References

[1] G. Gaille, E. Vourch, M. J. Martin, J. R. Mosig, and M. Polegre, "Conformal array antenna for observation platforms in low earth orbit," *IEEE Antennas Propagat. Mag.*, vol. 44, pp. 103-104, Jun. 2003.

[2] W. Gregorwich, "Conformal airborne arrays," *Aerospace Conference 1997*, vol. 4, pp. 463-470, Feb. 1-8, 1997.

[3] R. C. Hansen, *Phased array antennas*, New York: John Wiley & sons Publisher, 1998.

[4] S. B. Applebaum, "Adaptive arrays," *IEEE Trans. Antennas Propagat.*, vol. 24, no. 5, pp. 585-598, Sept. 1976.

[5] B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems," *IEEE Proc.,* vol. 55, pp. 2143-2159, 1967.

[6] R. L. Haupt, "Phase-Only adaptive nulling with a genetic algorithm," *IEEE Trans. Antennas Propagat.*, vol. 45, no. 6, pp. 1009-1015, Jun. 1997.

[7] D. S. Weile and E. Michielssen, "The control of adaptive antenna arrays with genetic algorithms using dominance and diploidy," *IEEE Trans. Antennas Propagat.*, vol. 49, no. 10, pp. 1424-1433, Oct. 2001.

[8] C. Sacchi, F. De Natale, M. Donelli, A. Lommi, and A. Massa, "Adaptive antenna array control in the presence of interfering signals with stochastic arrivals: assessment of a GA-based procedure," *IEEE Trans. Wireless Communicat.*, vol. 3, no. 4, pp. 1031-1036, Jul. 2004.

[9] A. Massa, M. Donelli, F. De Natale, S. Caorsi, and A. Lommi, "Planar antenna array control with genetic algorithms and adaptive array theory," *IEEE Trans. Antennas Propagat.*, vol. 52, no. 11, pp. 2919-2924, Nov. 2004.

[10] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*. San Francisco: Morgan Kaufmann Publishiers, 2001.

[11] D. Gies and Y. Rahmat-Samii, "Particle swarm optimization for reconfigurable phase-differentiated array design," *Microwave Opt. Technology Lett.*, vol. 38, pp. 168-175, Aug. 2003.

[12] D. W. Boeringer and D. H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," *IEEE Trans. Antennas Propagat.*, vol. 52, no. 3, pp. 771-779, Mar. 2004.

[13] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Trans. Antennas Propagat.*, vol. 52, pp. 397-407, Feb. 2004.

[14] J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids: Optimization of a profiled corrugated horn antenna," in *IEEE Antennas Propagat. Soc. Int. Symp. Dig.*, vol. 1, pp. 314-317, 2002.

[15] R. C. Eberhart and Y. Shi, "Particle Swarm Optimization: Developments, Applications and Resources," *Proc. Congress on Evolutionary Computation 2001,* Seul, Corea, pp. 81-86, 2001.

[16] Y. Rahmat-Samii, "Frontiers in evolutionary optimization techniques applied to antenna designs: genetic algorithm (GA) and particle swarm optimization (PSO)," in *Proc. 13ème Journées Internationales de Nice sur les Antennes* (*JINA 2004*), France, Nice, pp. 29-33, 8-10 Nov. 2004.

[17] R. C. Eberhart and Y. Shi, “Comparing inertia weights and constriction factors in particle swarm optimization,” in *Proc. Congress on Evolutionary Computation 2000,* Piscataway, USA, pp. 84-88, 2000.

[18] Y. Shi and R. C. Eberhart, “Empirical study of particle swarm optimization,” in *Proc. Congress on Evolutionary Computation 1999,* Washington, USA, pp. 1945-1950, 1999.

[19] J. Kennedy and R. C. Eberhart, “Particle swarm optimization,” in *Proc. IEEE Int. Conference Neural Networks*, vol. IV, Perth, Australia, Nov./Dec. 1995, pp. 1942-1948.

[20] J. Kennedy and R. C. Eberhart, “A discrete binary version of the particle swarm algorithm,” in *Proc. IEEE Int. Conference Systems, Man, and Cybernetics*, vol. 5, Oct. 1997, pp. 4104-4108.

[21] S. Caorsi, M. Donelli, A. Lommi, and A. Massa, “A real-time approach to array control based on a learned genetic algorithm,” *Microwave Opt. Technol. Lett.*, vol. 36, no. 4, Feb. 2003.

[22] A. Carlson, *Communication Systems*. New York: McGraw-Hill, 1987.

[23] J. Rodriguez, R. Munoz, H. Estevez, F. Ares, and E. Moreno, “Synthesis of planar arrays with arbitrary geometry generating arbitrary footprint patterns,” *IEEE Trans. Antennas Propagat.*, vol. 52, no. 9, pp. 2484 -2488, Sept. 2004.

Figure Captions

- **Figure 1.** Problem geometry.

- **Figure 2.** Calibration Process (χ) - Linear Array. Averaged $SINR$ over $T = 100$ timesteps for different values of χ_{max} ($\chi_{min} = 0.4$) and χ_{min} ($\chi_{max} = 0.9$).

- **Figure 3.** Interference Scenario. (a) Distribution of angles of arrival of interfering signals arrival versus timestep number and (b) plot of the number of interference signals per time-step life-time.

- **Figure 4.** Adaptive control of linear arrays ($M = 20$). (a) Running average of the SINR versus k obtained by means of the Applebaum's method (solid line), the DPA method (dashed line), the LMS approach (point-dashed line), the CGA-based technique (dotted line), and the PSO strategy (small dashed line). Quiescent elevation beam pattern (b). Samples of synthesized elevation beam patterns at (c) $t = 396$, (d) $t = 604$, and (e) $t = 802$.

- **Figure 5.** Adaptive control of planar arrays (Square array - $M = 100$). Running average of the SINR versus t obtained by means of the Applebaum's method (solid line), the Applebaum's method with discrete phase-coefficients (DPA) (dashed line), the LMS approach (point-dashed line), the CGA-based technique (dotted line), and the PSO strategy (small dashed line).

- **Figure 6.** Adaptive control of planar arrays (Square array - $M = 100$). Samples of synthesized elevation beam patterns at (a) $t = 55$ ($\phi = 173^o$), (b) $t = 105$ ($\phi = 132^o$), and $t = 205$ (c) $\phi = 8^o$ and (d) $\phi = 60^o$.

- **Figure 7.** Adaptive control of planar arrays (Square array - $M = 100$). Samples of synthesized beam patterns at (a) $t = 55$, (b) $t = 105$, and (c) $t = 205$. Quiescent beam pattern (d).

- **Figure 8.** Elliptical array layout ($M = 160$).

- **Figure 9.** Running average of the SINR versus t obtained by means of the Applebaum's method (solid line), the DPA method (dashed line), the LMS approach (point-dashed line), the CGA-based technique (dotted line), and the PSO strategy (small dashed line).

- **Figure 10.** Adaptive control of planar arrays (Elliptical array - $M = 160$). Samples of synthesized beam patterns at (a) $t = 5$, (b) $t = 155$, and (c) $t = 205$. Quiescent beam pattern (d).

Table Captions

- **Table I.** Adaptive control of linear arrays ($M = 20$). SINR statistics.

- **Table II.** Adaptive control of planar arrays (Square array - $M = 100$). SINR statistics.

- **Table III.** Adaptive control of planar arrays (Elliptical array - $M = 160$). SINR statistics.

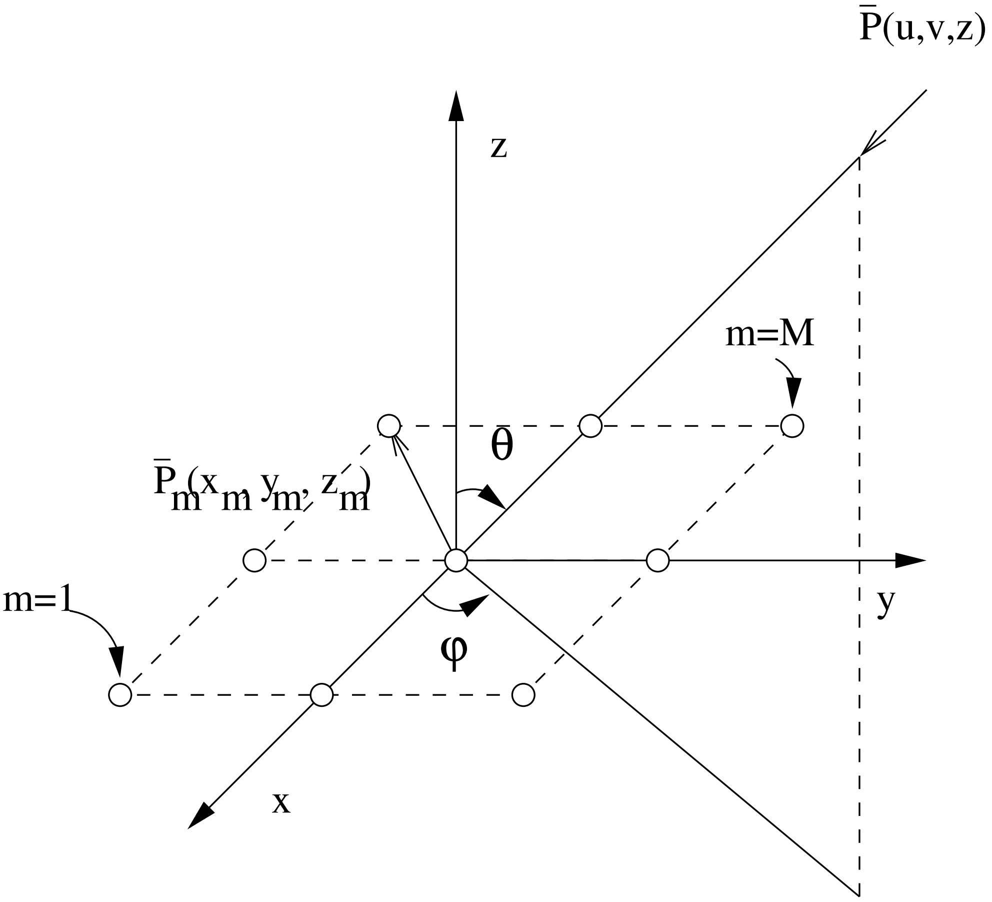


Fig. 1 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

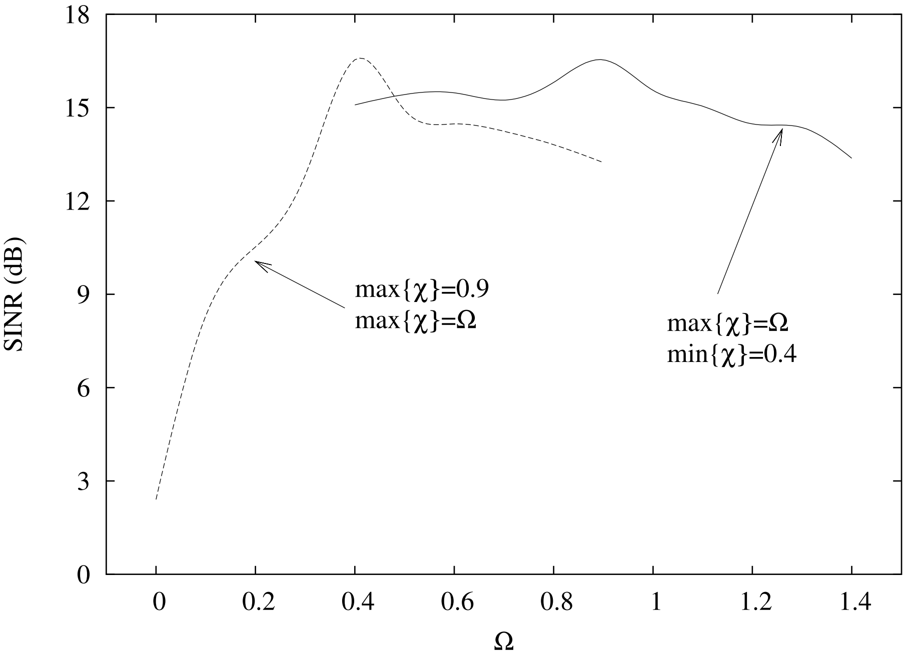
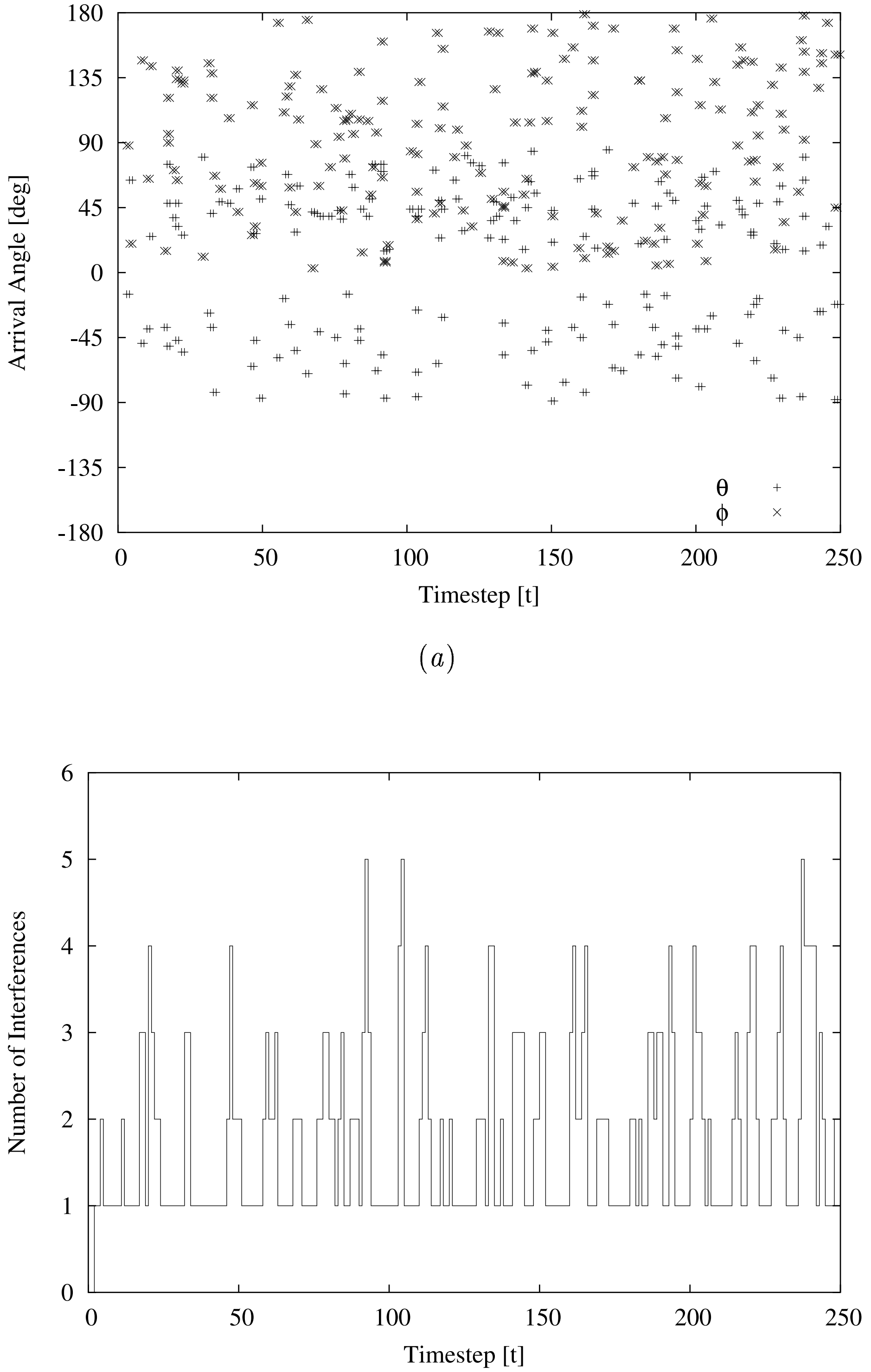


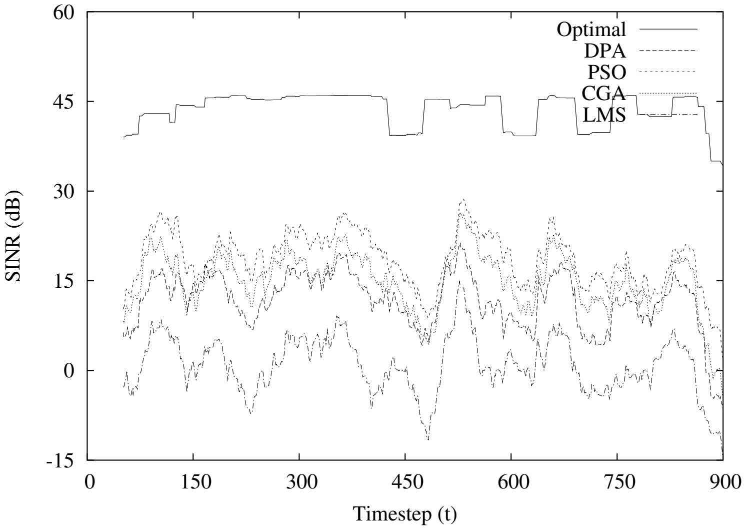
Fig. 2 - M. Donelli *et al.*, “An Innovative Computational Approach based on ...”



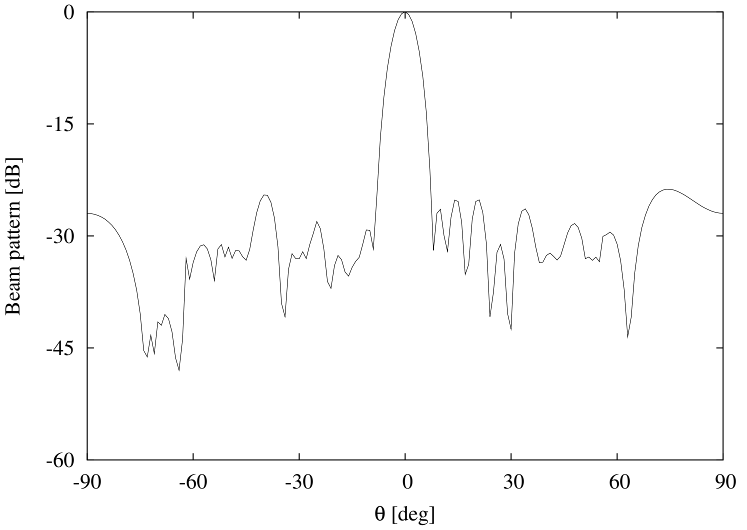
(a)

(b)

Fig. 3 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."



(a)



(b)

Fig. 4(*I*) - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

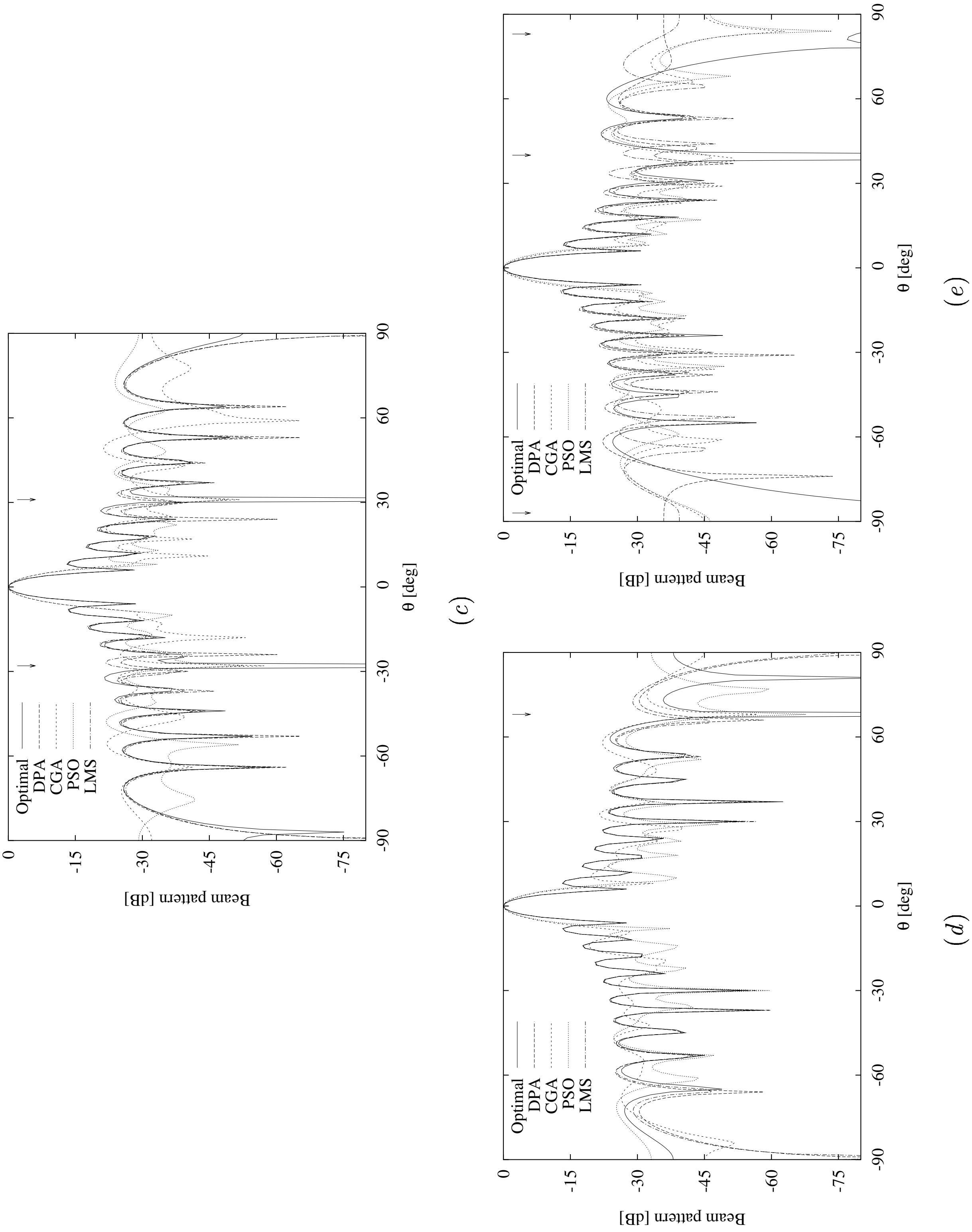


Fig. 4(*II*) - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

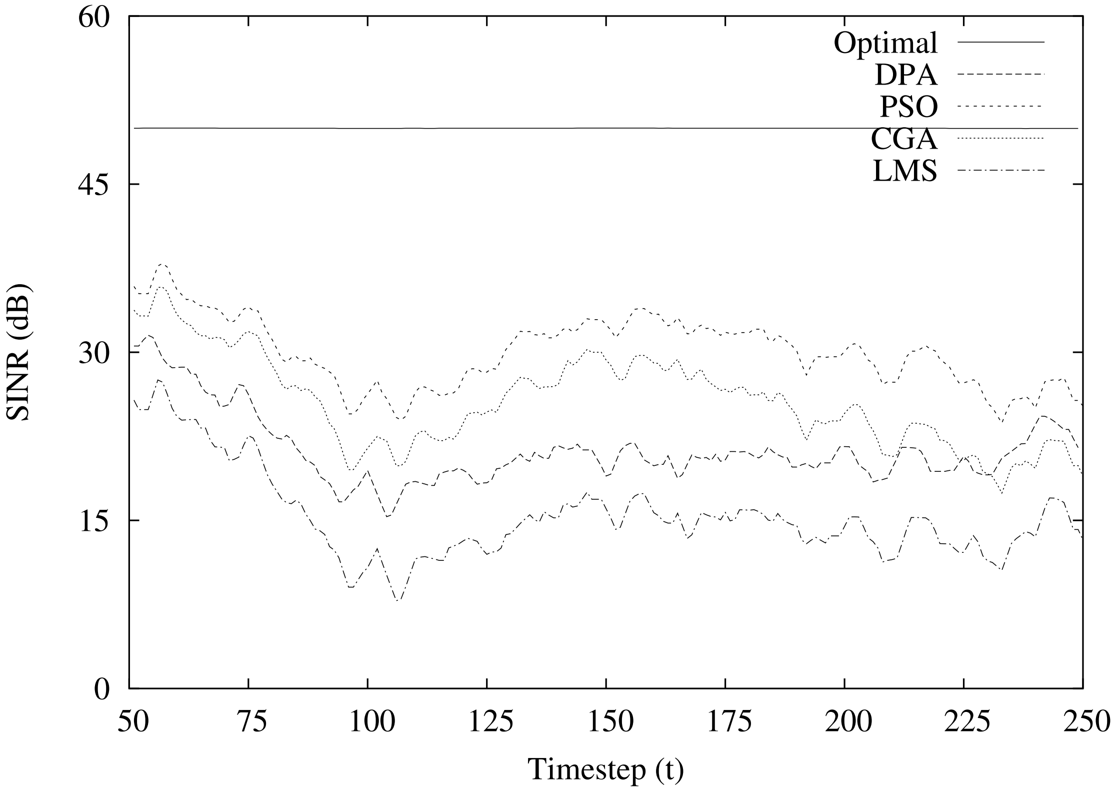
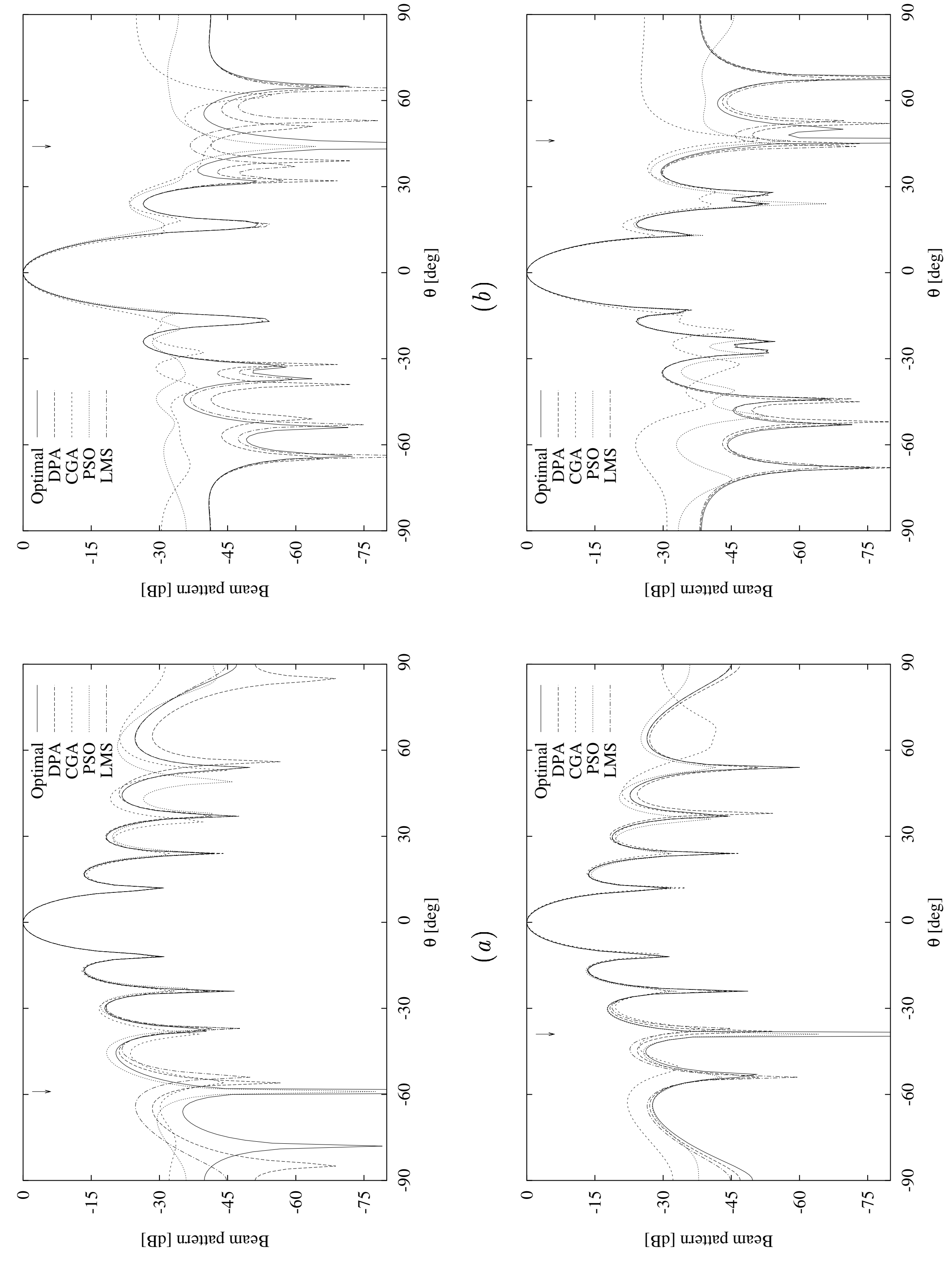


Fig. 5 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."



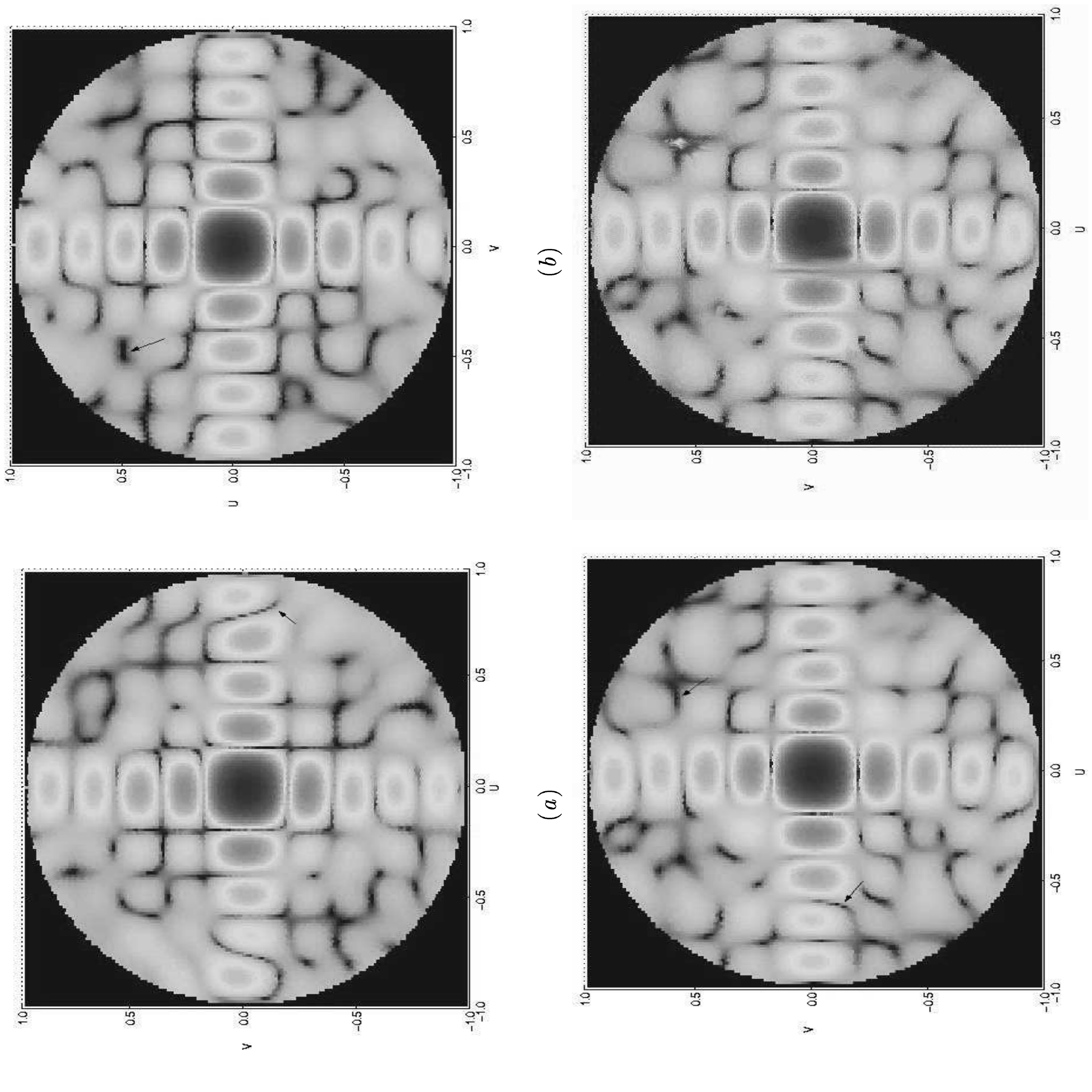
(*a*)

(*b*)

(*c*)

(*d*)

Fig. 6 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."



(*a*)

(*b*)

(*c*)

(*d*)

Fig. 7 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

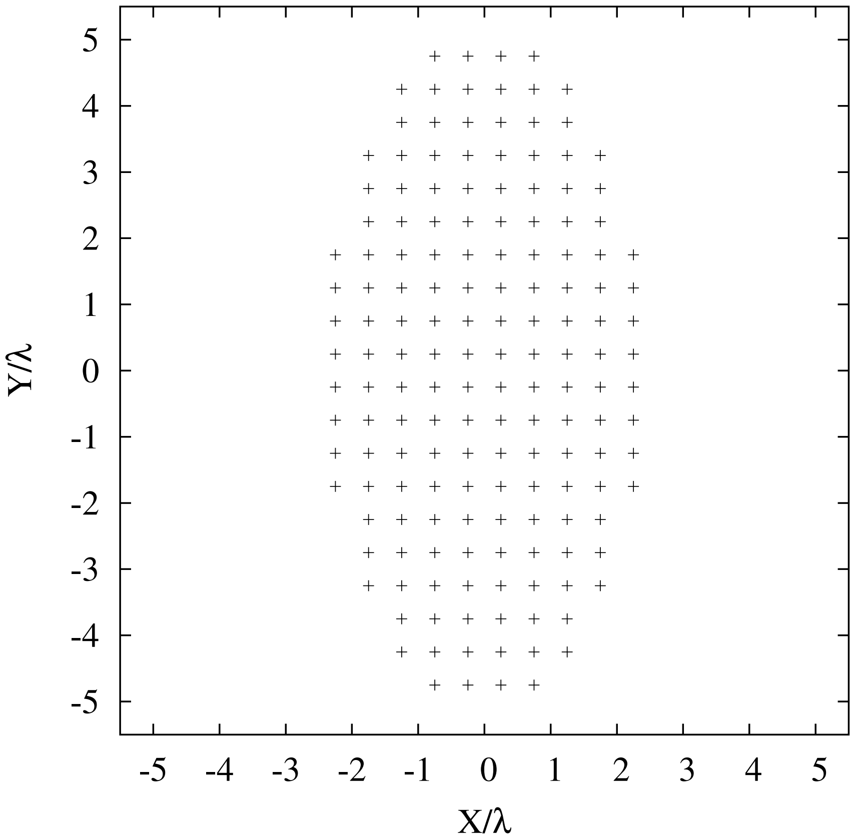


Fig. 8 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

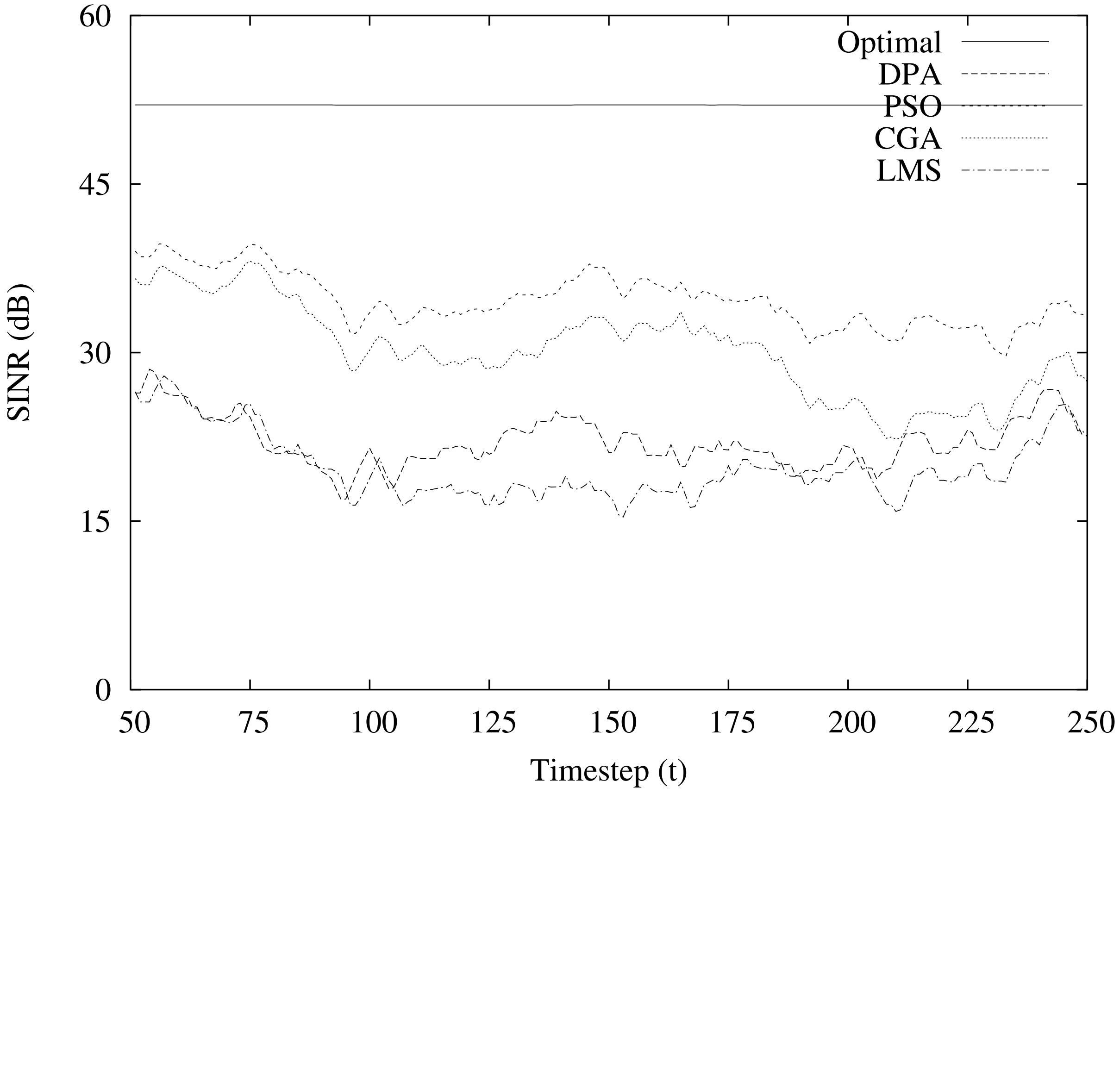
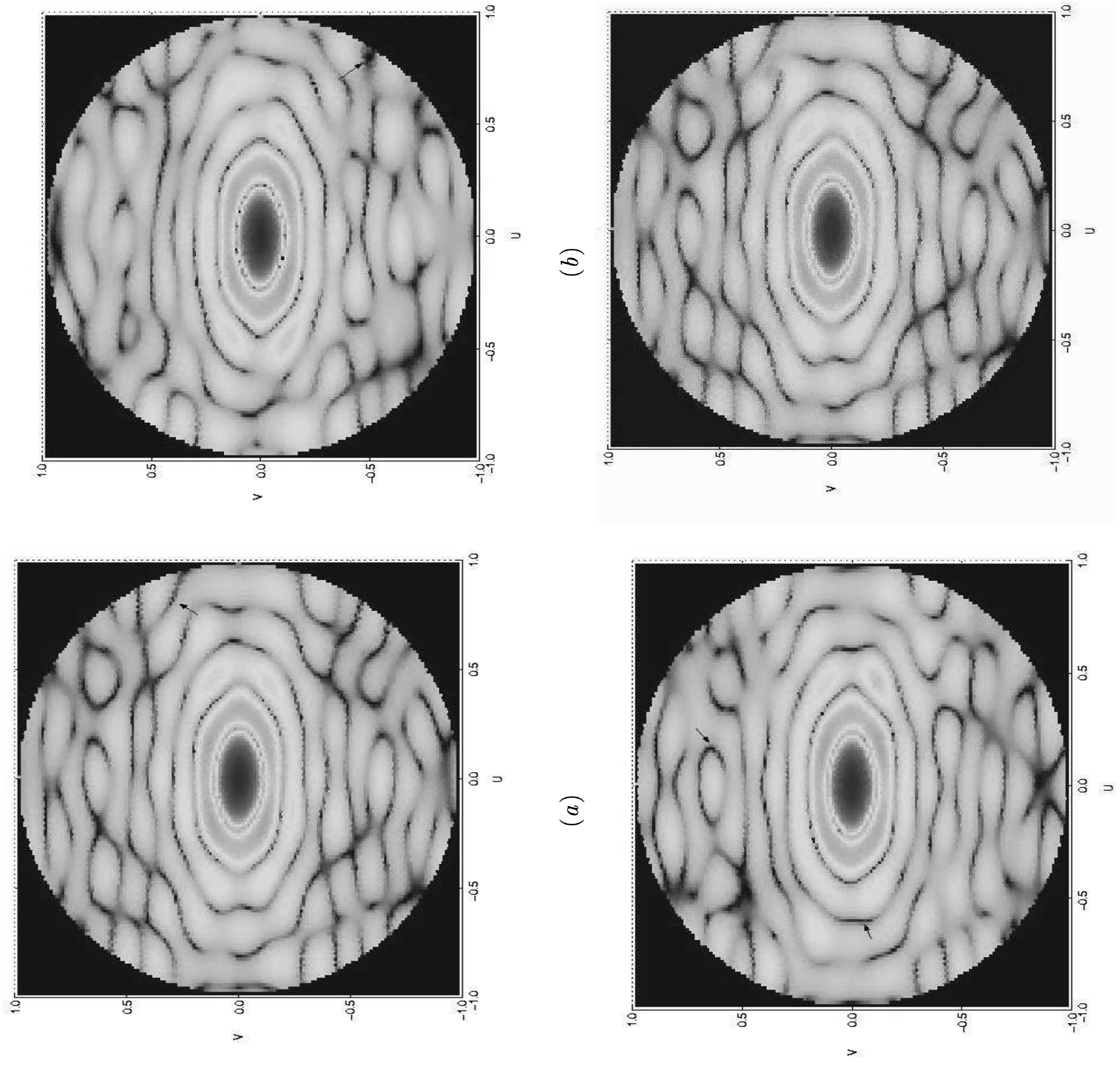


Fig. 9 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."



(*a*) (*b*) (*c*) (*d*)

Fig. 10 - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

	LMS	DPA	CGA	PSO	$Optimal$
$av\{SINR\}$	0.87	11.64	14.93	19.61	34.18
σ_{SINR} $\left[\times 10^{-1}\right]$	22.81	22.25	24.92	21.53	51.08

Tab. I - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

	LMS	DPA	CGA	PSO	$Optimal$
$av\{SINR\}$	15.24	21.24	25.59	30.03	49.98
σ_{SINR} $[\times 10^{-1}]$	16.32	9.72	16.87	9.74	0.006

Tab. II - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."

	LMS	*DPA*	*CGA*	*PSO*	*Optimal*
$av\{SINR\}$	19.78	20.07	30.17	35.69	52.02
σ_{SINR} $[\times 10^{-1}]$	8.30	5.02	16.43	5.59	0.005

Tab. III - M. Donelli *et al.*, "An Innovative Computational Approach based on ..."