UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

MEMORY ENHANCED PSO-BASED OPTIMIZATION APPROACH FOR SMART ANTENNAS CONTROL IN COMPLEX INTERFERENCE SCENARIO

M. Benedetti, R. Azaro, and A. Massa

August 2007

Technical Report DIT-07-051

Memory Enhanced PSO-Based Optimization Approach for Smart Antennas Control in Complex Interference Scenarios

M. Benedetti, R. Azaro, *Member*, *IEEE*, and A. Massa, *Member*, *IEEE*

Department of Information and Communication Technologies

University of Trento, Via Sommarive 14, I-38050 Trento - Italy

Tel. +39 0461 882057, Fax +39 0461 882093

E-mail: *andrea.massa@ing.unitn.it,*

{*manuel.benedetti, renzo.azaro*}*@dit.unitn.it*

Web-page: *http://www.eledia.ing.unitn.it*

Memory Enhanced PSO-Based Optimization Approach for Smart Antennas Control in Complex Interference Scenarios

M. Benedetti, R. Azaro, and A. Massa

Abstract

In the framework of control methods for adaptive phased-arrays, this paper deals with complex communication scenarios by considering a memory-enhanced cooperative algorithm. Compared to existing approaches where far-field interferences are taken into account, the proposed analysis considers a more realistic situation where the jamming sources are located either in the near-field or in the far-field of the receiving antenna. In order to carefully address the arising challenges and to effectively deal with such complex environments, an optimization approach based on an enhanced *PSO*-based algorithm is used. The obtained results seem to confirm the effectiveness of the proposed technique in terms of both signal-to-noise ratio and computational costs and complexity.

Index Terms:

Smart Antennas, Adaptive Control, Optimization Techniques, Particle Swarm Optimizer, Phased Arrays.

1 Introduction

The continuous evolution of communication systems requires the development and customization of techniques based on the idea of diversity [1]. In the framework of antennas design, such a theory has been applied for developing smart systems able to improve the quality of the received signal and to suppress the effects of interfering sources. The concept of spatial diversity [2] has led to the coupling of array theory with adaptive control and therefore to the design of antenna architectures able to maximize the system performances (i.e., the signal-to-interference-plus-noise ratio) by tuning dynamically the weights of the array elements.

The mathematical theory of adaptive systems has been originally proposed by Applebaum in [3] dealing with linear arrays of isotropic sources in the presence of far-field (FF) narrow-band signals (i.e., a desired signal and a set of jammers). The array weights, adapted for placing nulls in the far-field pattern in the directions of interference, are obtained by multiplying the quiescent weights by the inverse of the sampled covariance matrix formed from the complex signals received at each element in the array.

Alternatively, the adaptive control has been also recast as an optimization problem by defining a suitable cost functional to be maximized. Originally, deterministic techniques based on gradient methods (e.g., the least mean square (LMS) algorithm by Widrow *et al.* [4][5]) have been proposed, but the resulting approaches were still characterized by several non-negligible drawbacks. Because of the need of estimating the covariance matrix of the desired signal from the measurements of the received signals at each element of the array, the array must have an expensive receiver or a correlator at each element. Unfortunately, most arrays (or the simplest/cheapest) have a single receiver at the output of the summer and the receivers (when available) would require sophisticated calibrations. On the other side, these methods consider variable analog amplitude and phase weights, but phased arrays usually have only digital beam steering phase shifters at the elements and the feed network (fixed) determines the amplitude values. Therefore, the continuous phase values calculated by the adaptive algorithms are only approximated and the quantization error limits the null placement.

In order to reduce the complexity and the costs of adaptive systems, the possibility of

implementing a phase-only control (i.e., adjusting the phase shifter setting) for reducing the total output power measured by the receiver at the output of the summer has been investigated [6]. A significant improvement on this technique has been proposed by *Haupt* [7] who used a Genetic Algorithm (GA) to adjust some of the least significant bits of the beam steering phase shifters for minimizing the total output power thus removing the interfering signals from the output of the array.

Notwithstanding the success and successive experimental implementation [8], such a GA-based approach did not take into account constantly changing conditions and the need of a readaptation to new environments once the population converged. Such a problem has been overcome in successive works by *Weile and Michielssen* [9] or *Donelli* et al. [10] by using diploidy and dominance or cooperative algorithms [i.e., the particle swarm optimizer (PSO)].

In such a framework, this paper is aimed at assessing the effectiveness and reliability of an enhanced PSO-based technique in the presence of more complex working conditions. In particular, the signals impinging on the array are characterized by randomly variable directions and generated by electromagnetic sources located at difference distances from the antenna system. More in detail, the source of the desired signal is assumed to be very far from the system, whereas the distance of the interfering sources from the array varies from the near to the far zone. Such a situation turns out to be quite realistic since it could model/describe an "*info-mobility*" scenario where a moving network node (e.g., a car or a pedestrian) communicates with a base station or another node of the mobile network. In this situation, a neighboring node (i.e., close to the receiving system) would be considered as a near-field jamming source.

In order to properly address such a topic, the PSO-based approach is added with enhanced learning capabilities. Similarly to [11], the enhanced strategy is characterized by the use of memory-based operators, which perform an exchange of information between the swarm and a set of reference solutions (defining the "*memory*" of the process) iteratively updated. Furthermore, the memory mechanism is further exploited (and customized to the cooperative optimizer at hand) by introducing a new term in the PSO velocity equation.

The paper is organized as follows. The mathematical formulation is presented in Sect. 2 where the adaptive antenna control is recast as the minimization of the total power of the array in terms of the quantized phase weights. The optimization procedure is detailed in Sect. 3 and the results of a numerical validation are shown in Sect. 4. Finally, some conclusions are drawn (Sect. 5).

2 Mathematical Formulation

Let us consider an array of N elements (Figure 1). The narrowband signal received by the n-th element of the array at the time-step[(1)] t_ℓ, $\ell = 1, ..., L$, can be expressed as follows

$$s_n^{(r)}(t_\ell) = a^{(r)}(t_\ell)\, e^{j\varphi_n^{(r)}} \qquad n = 1, ..., N;\ l = 1, ..., L \tag{1}$$

where $a^{(r)}(t_\ell) = h^{(r)}(t_\ell)\, e^{j2\pi f t_\ell}$, $h^{(r)}(t_\ell)$ and f being the slowly-varying envelope of the received signal and the carrier frequency, respectively. Moreover, $\varphi_n^{(r)}$ is the phase term of the received signal coming from the angular coordinates (θ_r, ϕ_r) that identify the direction-of-arrival (DoA) of the received signal. Under far-field conditions [12], the phase term of (1) turns out to be

$$\varphi_n^{(r)} = \frac{2\pi}{\lambda}\left(u_r x_n + v_r y_n + q_r z_n\right) \tag{2}$$

where $u_r = \sin\theta_r \cos\phi_r$, $v_r = \sin\theta_r \sin\phi_r$, and $q_r = \cos\theta_r$, and (x_n, y_n, z_n) are the Cartesian coordinates of the n-th element of the array.

By considering co-channel interferences, $s_n^{(r)}$ is the result of the summation of the desired signal $s_n^{(d)}$, a set of I jammers $\left\{s_{i,n}^{(g)};\ i = 1, ..., I\right\}$, and an uncorrelated background noise [or noise signal $s_n^{(o)}$] characterized by an average power equal to σ^2,

$$s_n^{(r)}(t_\ell) = s_n^{(d)}(t_\ell) + \sum_{i=1}^{I} s_{i,n}^{(g)}(t_\ell) + s_n^{(o)}(t_\ell) \tag{3}$$

(1) A time-step is a slot of time, between two consecutive snapshots ($\triangle t_{\ell+1}$ and $\triangle t_\ell$), characterized by the presence of a desired signal and a fixed number of interfering signals with invariant DoAs: $t_\ell \triangleq \triangle t_{\ell+1} - \triangle t_\ell$.

where $s_n^{(d)}(t_\ell) = a^{(d)}(t_\ell)\, e^{j\varphi_n^{(d)}}$ and $s_{i,n}^{(g)}(t_\ell) = a_{i,n}^{(g)}(t_\ell)\, e^{j\varphi_{i,n}^{(g)}}$. Analogously to (2), $\varphi_n^{(d)} = \frac{2\pi}{\lambda}(u_d x_n + v_d y_n + q_d z_n)$, while

$$\varphi_{i,n}^{(g)} = \frac{2\pi}{\lambda}\left[\rho_i - \sqrt{(\rho_i u_i - x_n)^2 + (\rho_i v_i - y_n)^2 + (\rho_i q_i - z_n)^2}\right] \quad n = 1, ..., N;\ i = 1, ..., I \tag{4}$$

to model [13] the phase term of the i-th interference source located at $(\rho_i,\, \theta_i\,, \phi_i)$ either in the far-field or in the near-field depending on the value of ρ_i (Fig. 2).

As far as the signal $s^{(e)}$ available at the the output of the summer is concerned, it appears that (see Fig. 1)

$$s^{(e)}(t_\ell) = \sum_{n=1}^{N} W_n s_n^{(r)}(t_\ell) \tag{5}$$

where $W_n = w_n e^{j\beta_n}$ is the n-th complex weight. Consequently, the total output power measured by the single receiver is equal to [3][14]

$$\mathcal{P}(t_\ell) = \mathcal{P}_\ell(\underline{W}) \triangleq \sum_{n=1}^{N} w_n e^{j\beta_n} \sum_{p=1}^{N} w_p e^{-j\beta_p} \Omega_{p,n}^r(t_\ell) \tag{6}$$

that is a function of $\underline{W} = \{W_n;\, n = 1, ..., N\}$, $\Omega_{p,n}^r(t_\ell)$ being the (p, n)-entry of the covariance matrix of the received signal.

In order to minimize the total output power thus removing the interfering signals from the output of the array, the array coefficients are iteratively updated for taking into account constantly changing (i.e., at each time-step) conditions and the need of a readaptation to new environments. Moreover, a time-varying phase-only control is implemented to reduce the complexity and the costs of the adaptive system. In particular, the following optimization problem

$$\underline{\beta}^{opt}(t_\ell) = arg\left\{min_{\underline{\beta}}\left[\mathcal{P}(t_\ell)\right]\right\} \tag{7}$$

is solved by means of the enhanced PSO-based strategy (Sect. 3) to determine the optimal setting of the phases, $\underline{\beta} = \{\beta_n;\, n = 1, ..., N\}$, since amplitude coefficients $\{w_n;\, n = 1, ..., N\}$ are fixed quantities (e.g., uniform amplitudes or distributed according to Dolph-Chebychev pattern).

3 Memory Enhanced PSO-based Optimization ($PSOM$)

3.1 Structure of the Binary PSO Optimization

The PSO [16][17] has been introduced by Eberhart and Kennedy in the last decade [18]. It is a multiple-agent optimization approach based on the imitation of the social behavior of groups of animals in search of food. A swarm of P particles, which models a set of P trial solutions, is defined and its evolution in the solution space is controlled by means of a set of updating equations that take into account and exploit the history of the swarm.

In this paper, following the implementation guidelines of the PSO-based strategy proposed in [10] and concerned with N-sized phased-arrays in the presence of simplified far-field interferences, the solution space is binarized for allowing the use of digital beam steering phase shifters. The trajectories of each particle in the binary space are determined by evaluating the changes in the probability that a coordinate will take on a zero or one value.

Because of the complexity of the scenario at hand, the learning capabilities of the approach have been enhanced by defining a memory mechanism as well as an innovative updating relationship aimed at exploiting the "*history*" of the optimization for speeding up the convergence to the optimal solution and the adaptability of the control to the time-varying conditions.

As far as the mapping between the problem at hand and the swarm structure is concerned, let us refer to a phased-array controlled by B-bits digital phase shifters. Therefore, the p-th trial solution turns out to be the sequence of the quantized phase values [10]

$$\underline{\mathcal{B}}_p = \{\beta_{b,p,n} \in \{0,1\}\,;\, n = 1,...,N;\, b = 1,...,B\}\,. \tag{8}$$

Concerning the particle description, $\underline{\mathcal{B}}_p$ defines the position of the p-th element of the swarm in the solution space and the velocity $\underline{\mathcal{V}}_p$

$$\underline{\mathcal{V}}_p = \{\mathcal{V}_{b,p,n};\, n = 1,...,N;\, b = 1,...,B\} \tag{9}$$

models the capacity of the particle to fly from a given position $\underline{\mathcal{B}}_p^{k_\ell}$ to another position $\underline{\mathcal{B}}_p^{k_\ell+1}$ of the solution space, k_ℓ being the iteration index at the ℓ-th time-step t_ℓ. Moreover, $\mathcal{V}_{b,p,n}$ is the probability that $\beta_{b,p,n}$ takes value 1.

The swarm samples the solution space by means of a binary PSO-based strategy. At each iteration k_ℓ ($k_\ell = 1, ..., K$) of every time-step t_ℓ, the P trial solutions are ranked according to their "fitness" to the environmental scenario by computing (6) in correspondence with $\underline{\mathcal{B}}_p^{k_\ell}$, $\mathcal{P}_p^{k_\ell} = \mathcal{P}\left(\underline{\mathcal{B}}_p^{k_\ell}\right)$. Such an operation leads to the definition of the personal best particle $\underline{\xi}_p^{\ell} = \arg\left\{\min_{h_\ell=1,...,k_\ell}\left[\mathcal{P}\left(\underline{\mathcal{B}}_p^{h_\ell}\right)\right]\right\}$ and of the global best particle $\underline{\zeta}^{\ell} = \arg\left\{\min_{p=1,...,P}\left[\mathcal{P}\left(\underline{\xi}_p^{\ell}\right)\right]\right\}$. Starting from the initial population randomly generated around the "desired signal" particle (i.e., $\underline{\mathcal{B}}_p^{k_\ell} = \left\{\beta_{b,p,n}^{k_\ell} \text{ such that } \beta_n = \varphi_n^{(d)};\, n = 1, ..., N\right\}$, $k_\ell = 1$ and $p = 1$), the set of solutions iteratively evolves by modifying the particles positions according to the binary-position updating equation [10]:

$$\beta_{b,p,n}^{k_\ell+1} = \begin{cases} 1 & if\ r_{b,p,n}^{k_\ell} < \Im\left(\mathcal{V}_{b,p,n}^{k_\ell}\right) \\ 0 & otherwise \end{cases} \tag{10}$$

where $\Im(\,.\,)$ is the sigmoid function

$$\Im\left(\mathcal{V}_{b,p,n}^{k_\ell}\right) = \frac{1}{1+\exp\left(-\beta_{b,p,n}^{k_\ell}\right)} \tag{11}$$

$r_{b,p,n}^{k_\ell}$ being a random number drawn from an uniform distribution between 0 and 1. As far as the velocity update is concerned, it is obtained by applying the *Thresholding Operator* $\Lambda(\cdot)$ to the result $\mathcal{X}_{b,p,n}^{k_\ell}$ of the *Memory-Based Velocity Operator* $\mathcal{U}\{\cdot\}$(13)

$$\mathcal{V}_{b,p,n}^{k_\ell} = \Lambda\left\{\mathcal{X}_{b,p,n}^{k_\ell}\right\} = \begin{cases} -\mathcal{V}_{max} & \mathcal{X}_{b,p,n}^{k_\ell} < \mathcal{V}_{max} \\ \mathcal{X}_{b,p,n}^{k_\ell} & -\mathcal{V}_{max} \le \mathcal{X}_{b,p,n}^{k_\ell} \le \mathcal{V}_{max} \\ \mathcal{V}_{max} & \mathcal{X}_{b,p,n}^{k_\ell} > \mathcal{V}_{max} \end{cases}. \tag{12}$$

During a time step t_ℓ, the iterative process stops when a maximum number of iterations K is reached, $k_\ell = K$, (i.e., when the maximum reaction time T_{resp} of the system is elapsed, $T_{resp} = K \times T_{k_\ell}$, T_{k_ℓ} being the iteration CPU-time) or if the *optimality criterion* of the

system performance is attained [i.e., $\mathcal{P}\left(\underline{\zeta}^{\ell}\right) \leq \gamma_{opt}$, γ_{opt} being a user-defined threshold]. Whatever the termination condition, $\underline{\zeta}^{\ell}$ is assumed as the problem solution concerned with the ℓ-th time-step, t_{ℓ}.

3.2 Memory-Based Learning and Updating Strategy

In order to define a fast "*reaction*" of the control to the environmental changes, a customized and integrated strategy based on a *memory mechanism* has been implemented through the definition of suitable operators acting during the iterative procedure ($k_{\ell} = 1, ..., K$) and in the whole time-varying process (t_{ℓ}; $\ell = 1, ..., L$).

The *memory mechanism* lies on the definition of a "system memory" composed by a finite-length buffer $\mathcal{M} = \{\underline{\varsigma}^{m}; m = 1, ..., M\}$ (M being the buffer length). At each time-step, the *Storage Operator* allows an exchange of information from the swarm to the memory of the system. In correspondence with a new time-step ($t_{\ell} \leftarrow t_{\ell+1}$), the solutions stored in $\mathcal{M}$ are ranked according to their fitness values such that $\mathcal{P}_{\ell+1}\left(\underline{\varsigma}_1\right) \geq ..... \geq \mathcal{P}_{\ell+1}\left(\underline{\varsigma}_M\right)$. Then, at the end of the time-step, the system memory is updated as follows: $\underline{\varsigma}_1 = \underline{\zeta}^{\ell+1}$ if $\mathcal{P}_{\ell+1}\left(\underline{\zeta}^{\ell+1}\right) < \mathcal{P}_{\ell+1}\left(\underline{\varsigma}_1\right)$. In a complementary fashion, the operator $\Pi\{\cdot\}$ controls the exploitation of the system memory to improve the swarm reaction to the changes of the interference scenario. Unlike [11], a simpler activation mechanism is implemented by defining a user-fixed lower bound for the system performances, γ_{wor}. More in detail, when $\mathcal{P}\left(\underline{\zeta}^{\ell}\right) > \gamma_{wor}$ then the worst particle is replaced by the best solution stored in $\mathcal{M}$ ($\underline{\gamma}^{\ell+1} \leftarrow \underline{\varsigma}_M$, being $\underline{\gamma}^{\ell+1} = arg\left\{max_{p=1,...,P}\left[\mathcal{P}\left(\underline{\mathcal{B}}_{p}^{k_{\ell+1}}\right)\right]\right\}$).

Although such a learning strategy effectively uses the available information on the system history, certainly the exploitation of the information contained in $\mathcal{M}$ at each iteration k_{ℓ} of the swarm evolution would allow a more punctual and immediate use of the acquired knowledge on the behavior of the environment. Towards this purpose, the *Memory-Based Velocity Operator* $\mathcal{U}\{\cdot\}$ is defined as the composition of four terms

$$\mathcal{X}_{b,p,n}^{k_{\ell}} = \mathcal{I}\left\{\mathcal{V}_{b,p,n}^{k_{\ell}-1}\right\} + \mathcal{S}\left\{\beta_{b,p,n}^{k_{\ell}}, \xi_{b,p,n}^{\ell}\right\} + \mathcal{G}\left\{\beta_{b,p,n}^{k_{\ell}}, \zeta_{b,n}^{\ell}\right\} + \mathcal{A}\left\{\varsigma_{b,n}^{m}; m = 1, ..., M\right\}. \quad (13)$$

The first velocity component, usually referred to as *inertia*, is given by

$$\mathcal{I}\left\{\mathcal{V}_{b,p,n}^{k_\ell-1}\right\} = \alpha \mathcal{V}_{b,p,n}^{k_\ell-1}. \tag{14}$$

It models the tendency of a particle to continue in the same direction it is traveling. In general, the inertial weight α takes a constant value [19] or it decreases during the iterative process to favor a local searching at the end of the optimization [20][21].

The second term is called *self-knowledge* and it causes the attraction of the particle towards the best position previously reached for an amount proportional to a fixed constant coefficient c_1 (*cognition coefficient*) and a random number r_1 from an uniform distribution between 0 and 1

$$\mathcal{S}\left\{\beta_{b,p,n}^{k_\ell},\, \xi_{b,p,n}^{\ell}\right\} = c_1 r_1 \left(\xi_{b,p,n}^{\ell} - \beta_{b,p,n}^{k_\ell}\right). \tag{15}$$

Complementary to the self-knowledge component, the *group-knowledge* term models a linear attraction towards the optimal position achieved so far

$$\mathcal{G}\left\{\beta_{b,p,n}^{k_\ell},\, \zeta_{b,n}^{\ell}\right\} = c_2 r_2 \left(\zeta_{b,n}^{\ell} - \beta_{b,p,n}^{k_\ell}\right) \tag{16}$$

c_2 being the *social coefficient* and $r_2 \in [0,\, 1]$.

Because of the time-varying scenario and the need to reduce the reaction time taking into account the similarities among the environmental conditions at different time-steps, the fourth velocity component (indicated as "ambient-knowledge") is accordingly defined as follows

$$\mathcal{A}\left\{\varsigma_{b,n}^{m};\, m = 1, ..., M\right\} = c_3 r_3 \frac{\sum_{m=1}^{M}\left[\varsigma_{b,n}^{m} e^{-H\frac{(m-1)}{M}}\right]}{M} \tag{17}$$

H, c_3 being two constant weighting parameters and r_3 is another random number. In such a manner, the particle velocity is influenced by a *historical term* related to the optimal solutions at different time-steps and in correspondence with various interference scenarios.

4 Numerical Validation

In this section, the results of several numerical tests are reported in order to assess the potentialities and current limitations of the proposed approach. The first subsection

deals with the *calibration* of the *PSO*-based procedure and it is aimed at defining the optimal configuration of the key parameters of the optimization algorithm. The latter is concerned with the description of the performances of the adaptive control in complex scenarios characterized by the presence of near-field interference sources, as well.

4.1 Calibration of the Optimization Algorithm

The key parameters of the optimization algorithm have been selected through numerical simulations. They have been fixed to those values that allow a favorable trade-off between the rate of convergence towards a suitable solution and the capability of usefully exploring the whole solution space. Moreover, due to the intrinsic statistical nature of the approach, each test case or experiment has been run several times to assess the quality of the solution as well as its statistical significance.

The reference geometry consisted of a linear array of $N = 20$ z-oriented and $\lambda/2$-spaced dipoles lying on the x-axis. The amplitudes of the array weights have been chosen according to the Dolph-Chebyshev distribution. In the following, such a geometry will be referred to as *linear array*.

The inertial weight α has been heuristically tuned by verifying the effectiveness of the adaptive control in correspondence with different rules of variation or setting. Towards this end, an interference scenario characterized by jamming signals with directions randomly distributed and arrival-times modeled by means of a Poisson's process [11] has been considered. With reference to Fig. 3, where a representative sample of a stochastic realization of the interference generation process is pictorially described, a random number I of jamming signals [Fig. 3(a)] with *DoA*s uniformly distributed in $\phi \in [0; 180]$ [Fig. 3(b)] has been considered (*Poisson's scenario*). The power of the jamming sources has been fixed to $30\,dB$ above the power of the desired signal (the power of background noise has been assumed equal to $\sigma^2 = -30\,dB$). Moreover, the positions of the jamming sources have been randomly chosen between $5\,\lambda$ and $100\,\lambda$. In such a noisy environment, the choice of a swarm of $P = 30$ particles is a good trade-off between convergence rate and quality of the adaptive control as confirmed by Fig. 4 where the plot of the average value of the signal-to-interference-plus-noise ratio ($SINR$) [9] versus P is reported to provide

a quality rating of the algorithm performance[2].

Different choices of α have been analyzed (Tab. I) taking into account the guidelines suggested in the reference literature. Firstly, a dynamic law has been used by decreasing the inertial weight from 0.9 up to 0.4 in the range of iterations ($k_\ell = 1, ..., K$, $K = 1000$) of a time-step t_ℓ. In general, such a choice allows a better balance between global and local exploration during the minimization encouraging the global and the local search at the start and at the end of the optimization, respectively. However, when solving (7) and as confirmed by the indexes in Tab. I and related to the $SINR$ averaged over L time-steps, better performances have been attained by choosing a small and constant value of the inertial weight ($\alpha = 0.1$). Such a choice usually favors the reaction and the adaptability of the control to the environmental changes thus improving the convergence rate of the algorithm. Consequently, the faster the control reaches a set of suitable weights the lower becomes the response time with a reduction of the amount of iterations needed for each time-step without penalizing the effectiveness of the optimization process. Therefore, starting from such an indication and after an exhaustive and statistically relevant set of numerical tests, K has been set to 20 iterations whatever the interference scenario.

As far as the tuning of the "self-knowledge", of the "group-knowledge" and of the "ambient-knowledge" terms is concerned, a large number of simulations has been performed by considering the guidelines recommended by the PSO literature [16][17] as references and by taking into account other experimentations in similar optimization frameworks [22][19][23]. The hyperspace of possible setups of the parameters c_1, c_2, c_3, and H has been sampled to find the most suitable setting to allow an efficient PSO-based optimization. As a representative example, let us refer to Fig. 5 where the plot of the averaged $SINR$ along a slice of the PSO parameters hyperspace ($H = 10$ and $c_1 = 2.0$) is shown. The maximum value of such a quality index is situated at $c_2 = 2c_3 = 2.0$ and such a parameters configuration has been assumed in the following analyses/experiments.

Finally, the control parameters of the "memory mechanism" have been tuned. Because of the novelty of the proposed implementation, no indications are available. Thus, three dif-

[2] Unfortunately, the $SINR$ cannot be used by the control algorithm to rank trial solutions, but only as a quality index. As a matter of fact, there is no way to calculate the signal-to-interference-plus-noise ratio for the system architecture assumed in this paper (Fig. 1). Therefore, the total output power measured by the receiver is used as the index of the "fitness to the environment" of each particle.

ferent scenarios have been considered. Besides the Poisson's environment, two synthetic and customized interference configurations have been generated to verify the effectiveness of the approach in fully exploiting similarities and occurrences of jamming signals. The former (*intermittent scenario*) coincides with that proposed by Weile *et al.* in [9]. The latter (*deterministic scenario*) considers a cluster of interferences whose *DoA*s are supposed to be invariant during a large number of iterations (Tab. II).

In order to evaluate the sensitivity of the system to the memory dimension (i.e., the buffer length M), let us analyze the behavior of the following index

$$\Delta = \frac{\langle SINR_{M=\mathcal{Y}} \rangle - \langle SINR_{M=0} \rangle}{\langle SINR_{M=0} \rangle} \times 100 \tag{18}$$

where $\mathcal{Y}$ is the current value of M and the operator $\langle \, . \, \rangle$ stands for the average value. Concerning the *deterministic scenario*, the obtained results are summarized in Tab. III. As it can be noticed, the efficiency of the control improves in correspondence with an increase of the dimension of the buffer, until a saturation verifies when $M \geq 20$ (i.e., $\frac{M}{P} = 0.67$). As a matter of fact, $M = 20$ seems to be the best choice since it allows a non-negligible improvement in the control capabilities ($\Delta = 39.7$) without significantly affecting the computational burden. To further confirm such a conclusion, the analysis has been extended to the whole set of scenarios. Figure 6 shows the plots of the $SINR$ with ($M = 20$) and without ($M = 0$) memory versus t_ℓ ($\ell = 1, ..., L$; $L = 900$). As expected, the most relevant enhancement holds for the deterministic configuration, even though the learning capabilities of the approach impacts in a non-negligible way in correspondence with the "intermittent" configuration and the Poisson's scenario, as well. Moreover, the obtained results confirm that the introduction of a memory buffer and of an enhanced strategy for the velocity updating turns out in a fully exploitation of the existing (when negligible or limited too) correlations among different time-steps.

4.2 Testing of the Optimization Algorithm

By assuming the optimal setting of the $PSOM$ parameters defined after the "calibration" phase, this sub-section presents the results of a study aimed at evaluating the performance

of the adaptive control in various situations and scenarios. Such a study considered a comparative assessment, as well. As a matter of fact, the enhanced PSO-based control has been compared with other state-of-the-art procedures in terms of both quality indexes and computational costs.

The first analysis is devoted at evaluating the dependence of the adaptive control on the locations of the interference sources and the receiving system architecture. As a result, it appears that the performances of the $PSOM$ are notably affected from the number of bits B of the digital phase shifters especially in correspondence with small values of the distance ρ_i. Such an event is pointed out in Fig. $7(a)$ where the behavior of Φ_{av} versus B for different values of ρ_i is summarized (Poisson's scenario). Φ_{av} is a quality index defined as

$$\Phi_{av} = \frac{\langle SINR_{Full} \rangle - \langle SINR_{FF} \rangle}{\langle SINR_{FF} \rangle} \times 100$$

where the subscripts $(Full)$ and (FF) indicate that the $SINR$ has been computed with the array weights determined by minimizing (6) and using (4) or (2) for modeling the jammers, respectively.

As expected and confirming the effectiveness of the "Full" formulation in dealing with near-field interferences, Φ_{av} increases when ρ_i becomes smaller and smaller. Moreover, the value of Φ_{av} grows as B increases up to $B = 8$. As a matter of fact, when $B \geq 10$ the binary-solution-space considerably enlarges and it appears to be too large for allowing fast convergence and reliable results.

For comparison purposes, Figure $7(b)$ shows the results obtained setting $B = 8$ with the $PSOM$ approach, the Applebaum-based ideal method [3], the Applebaum technique with discrete phases (DPA), the Least Mean Square algorithm (LMS), the LMS with discrete phases $(DPLMS)$, the PSO-based approach proposed in [10] (PSO), and the learned real-time GA [24] $(LRTGA)$. As it can be noticed, the proposed approach outperforms both $DPLMS$ and $LRTGA$, as well as the PSO. Moreover, its behavior turns out to be quite close to that of the DPA whatever the jammers locations, despite a lower architectural complexity. Furthermore, the $PSOM$ achieves better signal-to-noise ratios than LMS when $\frac{\rho_i}{\lambda} < 400$, while for farther interferences the LMS allows slightly better performances, but with multiple receivers one at each array element.

As a representative example, Figure 8 shows the behavior of the $SINR$ for a realization of the Poisson's scenario ($L = 900$) under the assumption that ρ_i is randomly distributed in the range [5λ, 100λ] and the interferences do not change in $K^{(PSOM)} = 20$ iterations. Moreover, the control methods have been arrested after the same T_{resp}. Consequently, K has been fixed to 3000 when using the LMS algorithm since the number of complex floating point operations per iteration is $\mathcal{O}(N)$, while the floating point operations needed by $PSOM/PSO/LRTGA$ are of the order of $\mathcal{O}(P^2 \times B \times N)$. As far as the $LRTGA$ is concerned, it is of about 4 times computationally heavier than the PSO-based methods (Tab. IV). Therefore, each GA-based optimization loop has been terminated at $K^{(LRTGA)} = \frac{K^{(PSO)}}{4}$.

In Figure 8(a), the results obtained with the FF formulation are given in terms of the signal-to-interference-plus-noise ratio ($SINR_{FF}$), whereas Fig. 8(b) shows the $SINR$ behavior when using the complete formulation ($SINR_{Full}$). Except for the ideal approach and whatever the control technique, the system performance improves by resorting to the *Full formulation* as outlined by the plot of the index Φ [Fig. 9(c)] given by

$$\Phi = \frac{SINR_{Full} - SINR_{FF}}{SINR_{FF}} \times 100.$$

On the other hand, the PSO-based approaches generally outperform other optimization methods as well as the LMS-based techniques. Furthermore, they turn out to be very close or better than the DPA approach [Fig. 8(b)]. As a matter of fact, $\langle SINR_{Full}^{PSOM} \rangle = 29.90$ and $\langle SINR_{Full}^{PSO} \rangle = 28.82$ versus $\langle SINR_{Full}^{DPA} \rangle = 27.05$ (Tab. V).

The second test case deals with the same scenario of Fig. 8, but with a lower response time T_{resp}. As a matter of fact, the optimization loops have been terminated at $K^{(PSOM)} = K^{(PSO)} = 5$, $K^{(LRTGA)} = 2$, and $K^{(LMS)} = 750$, respectively. Unlike both Applebaum and LMS-based approaches, the results from stochastic strategies significantly change. Whatever the formulation, the average values of $SINR$ reduce of about $3 \div 6\,dB$ as indicated in Tab. V and Tab. VI. However, the $PSOM$ still favorably compares with the other digital optimization methods (i.e., $DPLMS$, PSO, and $LRTGA$) [Figs. 9(a)-(c) and Tab. VI].

The last experiment is concerned with a more complex situation. Let us consider a planar

array of $N = 61$ z-oriented and $\lambda/2$-spaced dipoles [11] with uniform amplitudes. At each time-step t_ℓ, a random number of I jamming signals with Poisson-modeled arrival-times and DoAs uniformly distributed in $\theta \in [0; 180]$ and $\phi \in [0; 180]$ ($3D$-*Poisson scenario*) impinges on the array. Likewise the Poisson's scenario, each jammer is characterized by a power of $30\,dB$ above the desired signal power and the locations of the interference sources are random variables uniformly distributed between $5\,\lambda$ and $100\,\lambda$.

Figure $10(a)$ shows the plot of the $SINR$ value in a window of $L = 100$ time-steps. As expected, the complete formulation allows a more effective adaptive control ($\left\langle SINR_{Full}^{PSOM} \right\rangle = 38.02$ vs. $\left\langle SINR_{FF}^{PSOM} \right\rangle = 26.84$). As far as the comparative assessment is concerned, Figure $10(b)$ points out that on average the efficiency of the $PSOM$ tends to that of the DPA ($\left\langle SINR_{Full}^{PSOM} \right\rangle = 38.02$ vs. $\left\langle SINR_{Full}^{DPA} \right\rangle = 38.31$) and it overcomes the LMS-based strategies, the PSO as well as the $LRTGA$ ($\left\langle SINR_{Full}^{LMS} \right\rangle = 31.31$, $\left\langle SINR_{Full}^{DPLMS} \right\rangle = 28.62$, $\left\langle SINR_{Full}^{PSO} \right\rangle = 31.63$, and $\left\langle SINR_{Full}^{LRTGA} \right\rangle = 29.43$).

For completeness, Figure 11 shows the color level representations of the quiescent beam pattern [Fig. $11(a)$] and both near-field [13] [Fig. $11(b)$ - $\rho_{oss} = 25\,\lambda$, Fig. $11(c)$ - $\rho_{oss} = 59\,\lambda$] and far-field [Fig. $11(d)$] distributions generated by the adaptive planar array at the $\ell = 28$-th snapshot when two interference sources located at ($\theta_1 = 62^o$, $\phi_1 = 89^o$, $\rho_1 = 25\,\lambda$) and ($\theta_2 = 42^o$, $\phi_2 = 39^o$, $\rho_2 = 59\,\lambda$) radiates.

5 Conclusions

This paper has investigated both the theoretical and numerical aspects of the use of digital phase-shifters only weighting for adaptive null steering in complex interference scenarios. It has demonstrated the application of a PSO-based control equipped with enhanced memory features for the adaptation of the antenna array to minimize the total output power at the receiver. The mathematical formulation of the approach and the algorithmic sequence of the enhanced adaptive control have been carefully described. The numerical validation has been carried out by considering different array geometries and various interference configurations.

The $PSOM$-based approach demonstrated:

- an enhanced efficiency of the adaptive control (*Full* vs. *FF* formulation);
- a favorable trade-off among architectural complexity of the receiver, computational load, and fast readaptation to changing environmental conditions;
- a robustness to both near-field and far-field interferences.

As far as the main novelties of this paper are concerned, they can be summarized as follows:

- the mathematical formulation of the smart control able to model time-varying scenarios characterized by randomly located jamming sources;
- the enhanced *PSO*-based approach, which has been suitably designed to profitably exploit the memory mechanism.

Future developments and research activities will be aimed at improving the model of the interference scenario. For example, by considering the presence of scatterers in the closeness of the antenna or different statistical descriptions. Moreover, it would be interesting to study the performance of the adaptive control under conditions when array elements are expected to fail [25]. In principle, no changes to the proposed algorithm would be required and certainly, the memory mechanism could aid under such conditions, as well.

References

[1] C. B. Dietrich, W. L. Stutzman, K. Byung-Ki, and K. Dietze, “Smart antennas in wireless communications: base-station diversity and handset beamforming,” *IEEE Antennas Propagat. Mag.*, vol. 52, pp. 142-151, Oct. 2000.

[2] M. Chryssomallis, "Smart antennas," *IEEE Antennas Propagat. Mag.*, vol. 42, pp. 129-136, Jun. 2000.

[3] S. P. Applebaum, "Adaptive arrays," *IEEE Trans. Antennas Propagat.*, vol. 24, pp. 585-598, Sep. 1976.

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

[5] L. C. Godara, “Improved LMS Algorithm for Adaptive Beamforming,” *IEEE Trans. Antennas Propagat.*, vol. 38, pp. 1631-1635, Oct. 1990.

[6] C.A. Baird and G. G. Rassweiler, “Adaptive sidelobe nulling using digitally controlled phase-shifters,” *IEEE Trans. Antennas Propagat.*, vol. 24, pp. 638-649, Sep. 1976.

[7] R. L. Haupt, "Phase-only adaptive nulling with a genetic algorithm," *IEEE Trans. Antennas Propagat.*, vol. 45, pp. 1009-1015, Jun. 1997.

[8] R. L. Haupt and H. Southall, "Experimental adaptive nulling with a genetic algorithm," *Microwave J.*, vol. 42, pp. 78-89, 1999.

[9] 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, pp. 1424-1433, Oct. 2001.

[10] M. Donelli, R. Azaro, F. De Natale, and A. Massa, “An innovative computational approach based on a particle swarm strategy for adaptive phased-arrays control,” *IEEE Trans. Antennas Propagat.*, vol. 54, pp. 888-898, Mar. 2006.

[11] 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, pp. 2919-2924, Nov. 2004.

[12] C.A. Balanis, *Antenna Theory*. New York: Wiley, 1996.

[13] A. J. Fenn, "Evaluation of adaptive phased array antenna far-field nulling performance in the near-field region," *IEEE Trans. Antennas Propagat.*, vol. 38, pp. 173-185, Feb. 1990.

[14] L. C. Godara, *Smart Antennas*. Boca Raton: CRC Press, 2004.

[15] R. T. Compton Jr., *Adaptive Antennas*. Englewood Cliffs, New Jersey, Prentice Hall, 1988.

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

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

[18] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," *Proc. Sixth Int. Symp. Micro-Machine and Human Science* (Nagoya, Japan), pp. 39-43, 1995.

[19] M. Donelli and A. Massa, "Computational approach based on a particle swarm optimizer for microwave imaging of two-dimensional dielectric scatterers," *IEEE Trans. Microwave Theory Techn.,* vol. 53, pp. 1761-1776, May 2005.

[20] 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.

[21] 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.

[22] D. W. Boeringer and D. H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," *IEEE Trans. Antennas and Propagat.*, vol. 52, pp. 771-779, Mar. 2004.

[23] M. Donelli, G. Franceschini, A. Martini, and A. Massa, "An integrated multiscaling strategy based on a particle swarm algorithm for inverse scattering problems," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pp. 298-312, Feb. 2006.

[24] S. Caorsi, M. Donelli, A. Lommi, and A. Massa "A real-time approach to array control based on a learned genetic algorithm," *Microwaves Opt. Technol. Lett.,* vol. 36, pp. 235-238, Feb. 2003.

[25] B.-K. Yeo and Y. Lu, "Array failure correction with a genetic algorithm," *IEEE Trans. Antennas and Propagat.*, vol. 47, pp. 823-828, May 1999.

Figure Captions

- **Figure 1.** Architecture of the adaptive array with a single receiver at the output of the summer.

- **Figure 2.** Geometry of the scenario under test.

- **Figure 3.** Poisson's interference scenario. (a) Number of interference signals I and (b) distribution of the angles of arrival of the jammers versus the time-step index.

- **Figure 4.** *Calibration Phase*. Averaged $SINR$ for different sizes of the swarm, P.

- **Figure 5.** *Calibration Phase.* Behavior of the averaged $SINR$ versus c_2 and c_3 ($c_1 = 2$, $H = 10$).

- **Figure 6.** *Calibration Phase.* Behavior of the $SINR$ versus the time-step index for different interference scenarios with ($M = 20$) and without memory mechanism ($M = 0$).

- **Figure 7.** *Testing Phase* (Poisson's interference scenario). Behavior of Φ_{av} versus ρ_i for (a) different values of B ($PSOM$) and in correspondence with (b) different control techniques ($B = 8$).

- **Figure 8.** *Testing Phase* (Poisson's interference scenario, $\rho_i \in [5\lambda, 100\lambda]$ - *Linear Array*). Plots of (a) $SINR_{Full}$, (b) $SINR_{FF}$, and (c) Φ versus the time-step index for different adaptive control methods [$K^{PSOM} = 20$].

- **Figure 9.** *Testing Phase* (Poisson's interference scenario, $\rho_i \in [5\lambda, 100\lambda]$ - *Linear Array*). Plots of (a) $SINR_{Full}$, (b) $SINR_{FF}$, and (c) Φ versus the time-step index for different adaptive control methods [$K^{PSOM} = 5$].

- **Figure 10.** *Testing Phase* (Poisson's interference scenario, $\rho_i \in [5\lambda, 100\lambda]$ - *Planar Array*). (a) Plots of $SINR_{Full}$ and $SINR_{FF}$ versus the time-step index when using $PSOM$. (b) Comparison between different control methods.

- **Figure 11.** *Testing Phase* (3D Poisson's interference scenario, $\rho_i \in [5\lambda, 100\lambda]$ - *Planar Array*). (a) Quiescent beam pattern. Beam patterns generated at the $\ell = 28$-th snapshot when (b) $\rho_{obs} = 25\lambda$, (c) $d_{obs} = 59\lambda$ and (d) in the FF region.

Table Captions

- **Table I.** *Calibration Phase.* Impact of the inertial weight setting α on the system performance (Δ).
- **Table II.** Descriptive parameters of the *Deterministic Scenario*.
- **Table III.** *Calibration Phase.* Impact of the dimension M of the memory buffer on the system performance (Δ).
- **Table IV.** Computational costs of the digital optimization approaches (CPU *Intel* $P4$, $2.8\,GHz$, $512\,MB$ RAM). $\mathcal{T} \triangleq \frac{T_{k_\ell}}{min\left(T_{k_\ell}\right)}$.
- **Table V.** *Testing Phase* (Poisson's interference scenario, $\rho_i \in [5\lambda,\, 100\lambda]$ - *Linear Array*). Average values of $SINR_{Full}$ and of $SINR_{FF}$ for different adaptive control methods $[K^{(PSOM)} = 20]$.
- **Table VI.** *Testing Phase* (Poisson's interference scenario, $\rho_i \in [5\lambda,\, 100\lambda]$ - *Linear Array*). Average values of $SINR_{Full}$ and of $SINR_{FF}$ for different adaptive control methods $[K^{(PSOM)} = 5]$.

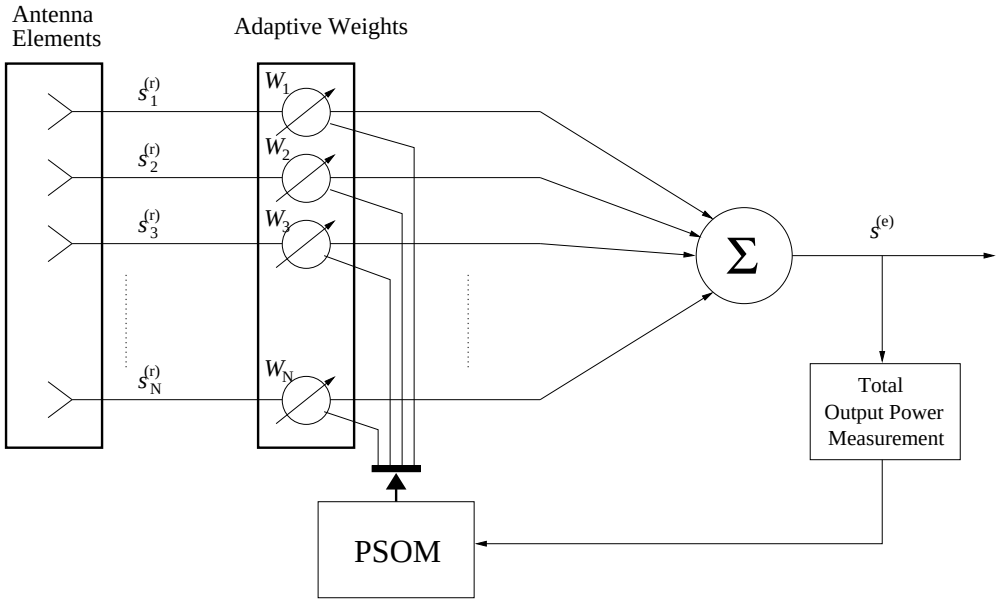


Fig. 1 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

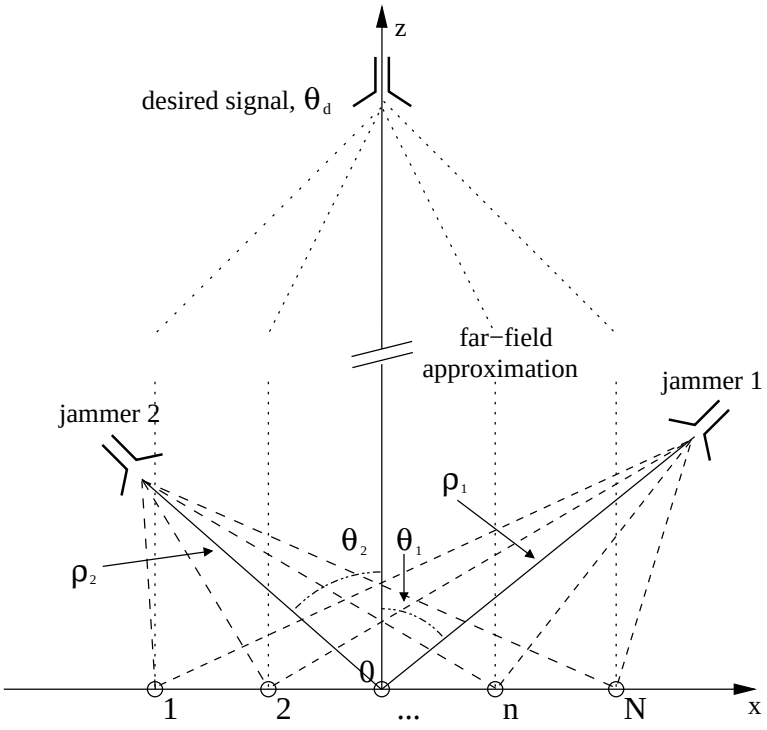
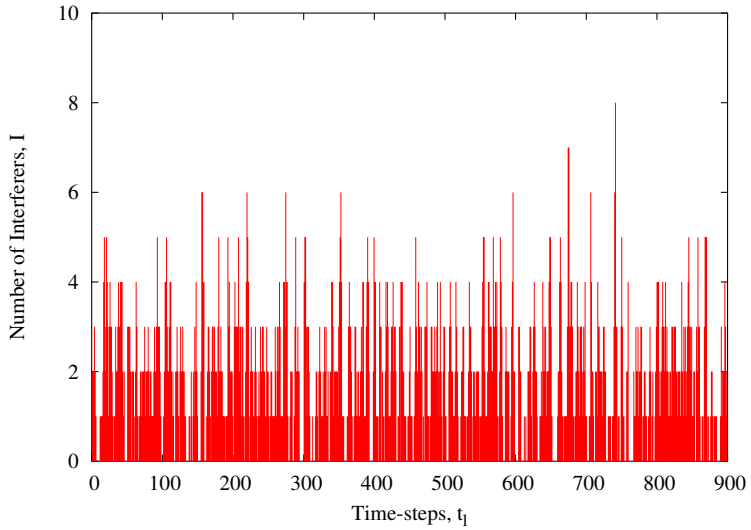
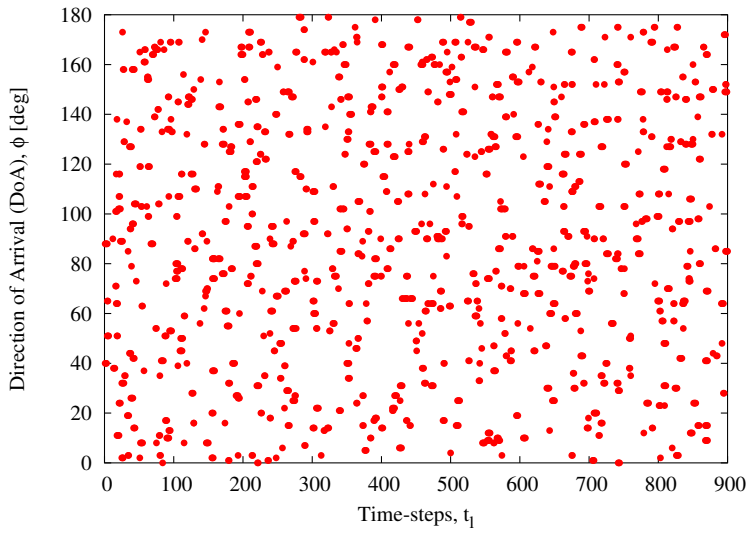


Fig. 2 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."



(a)



(b)

Fig. 3 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

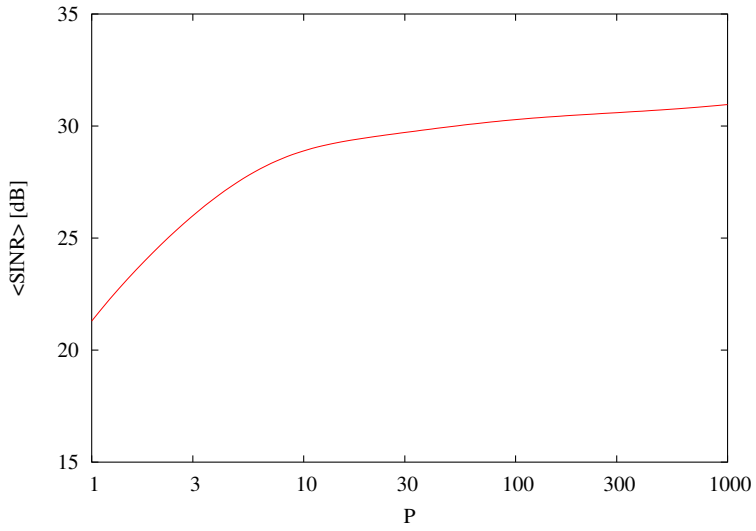


Fig. 4 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

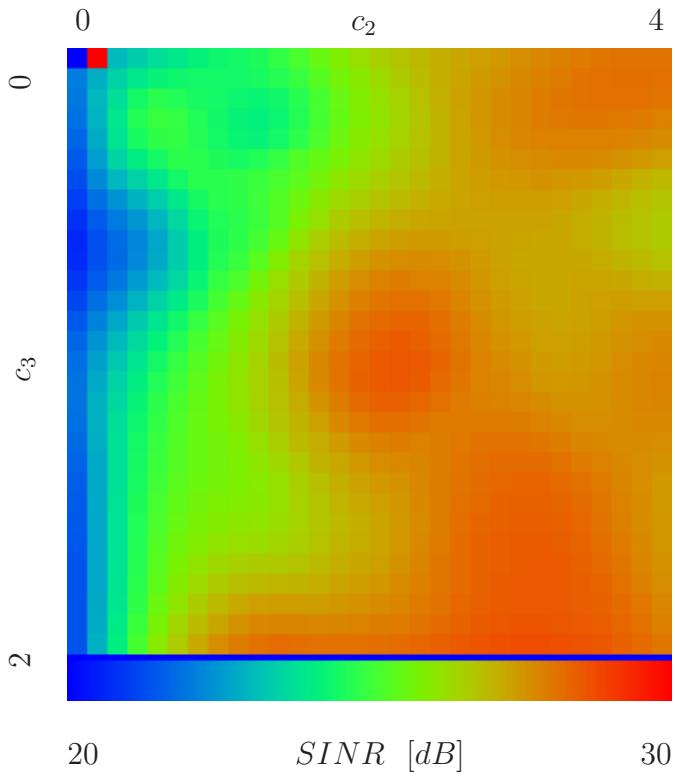


Fig. 5 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

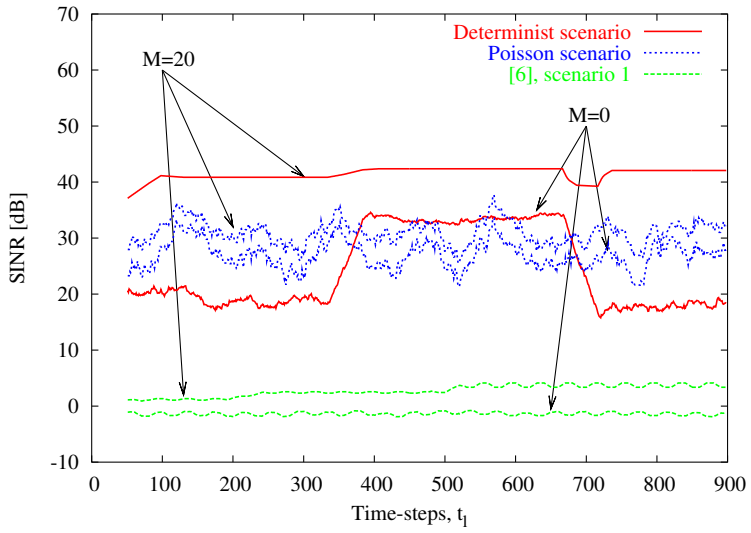
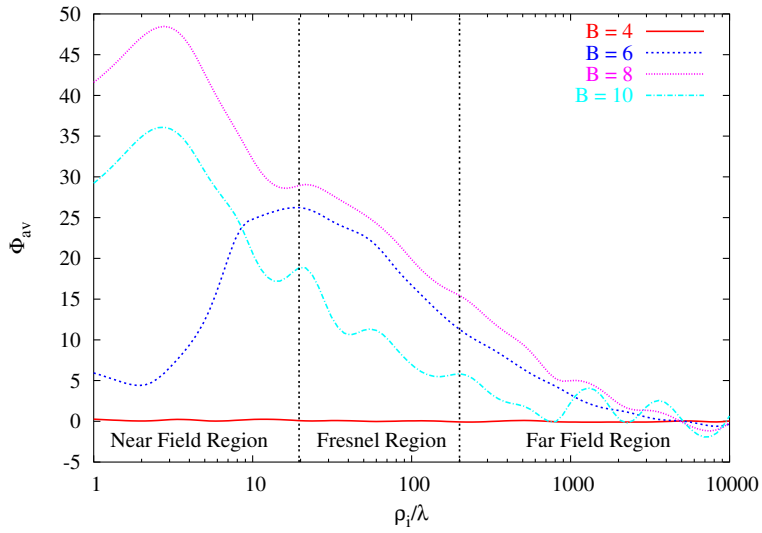
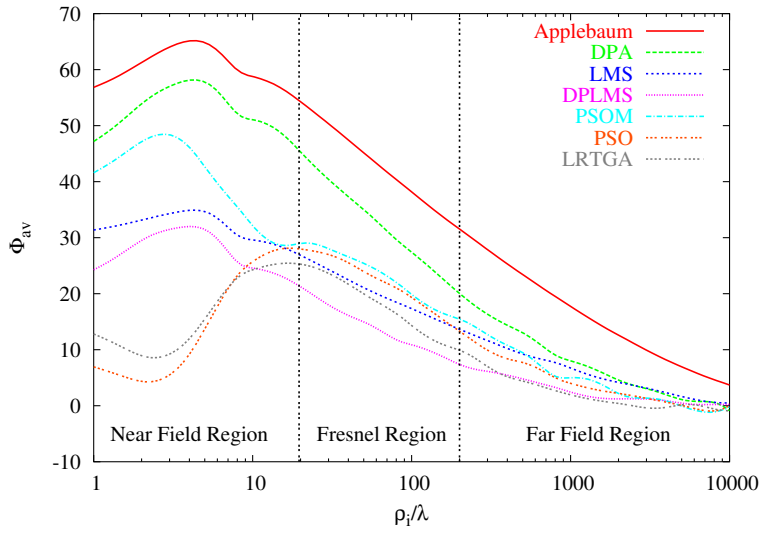


Fig. 6 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

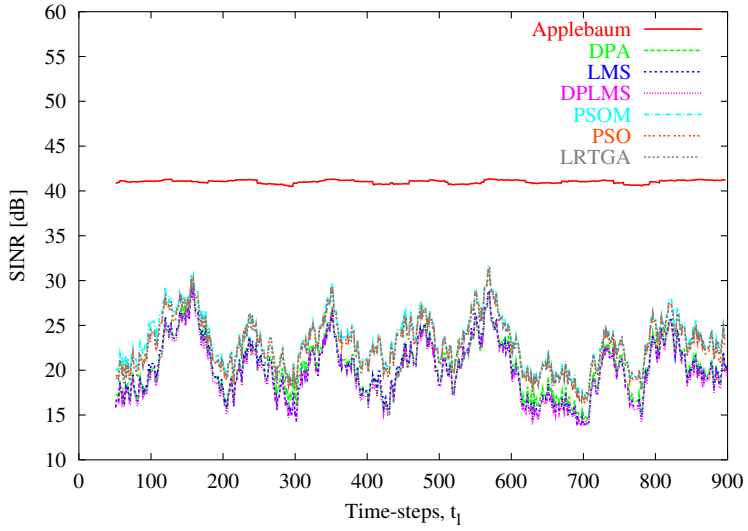


(a)

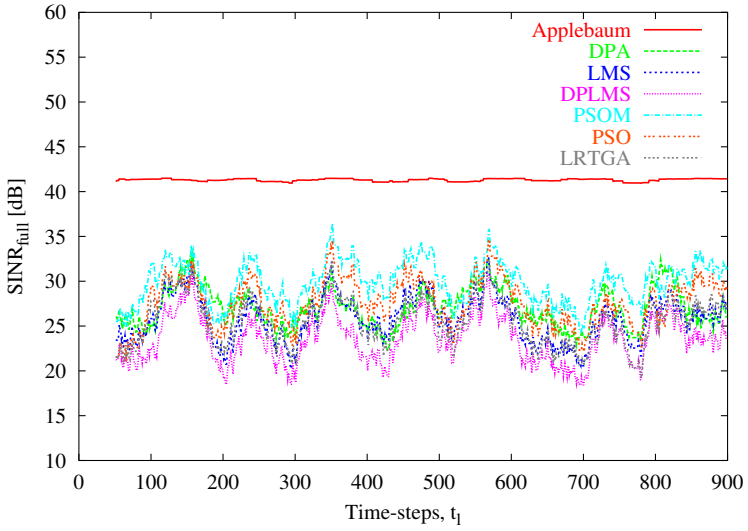


(b)

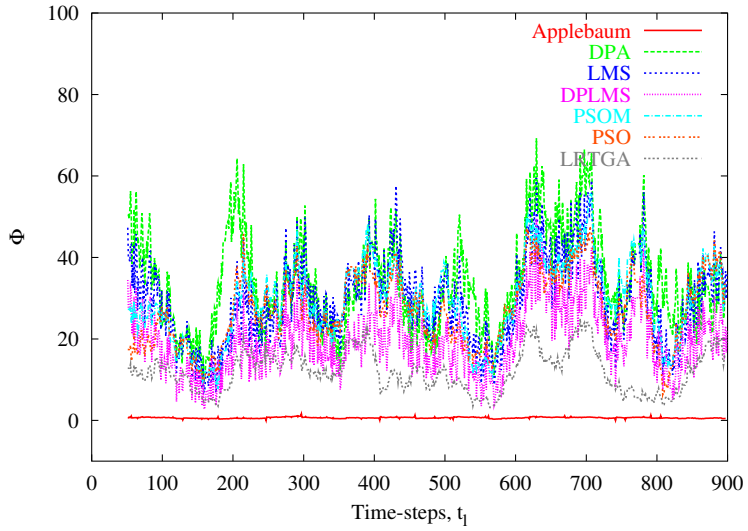
Fig. 7 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."



(a)

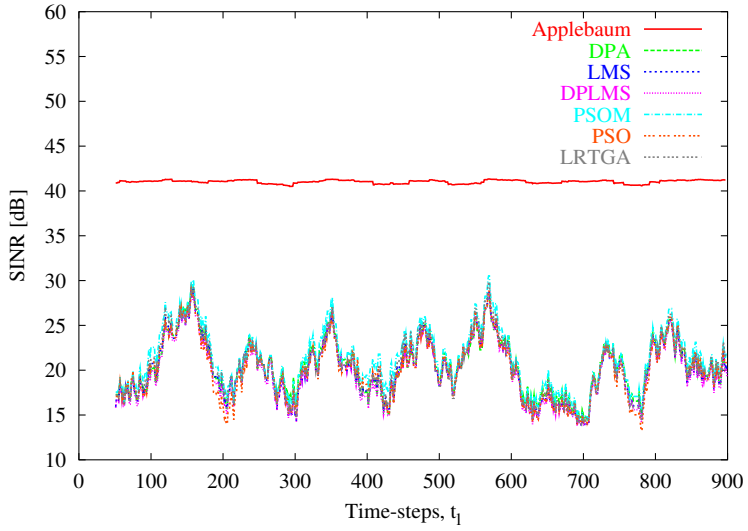


(b)

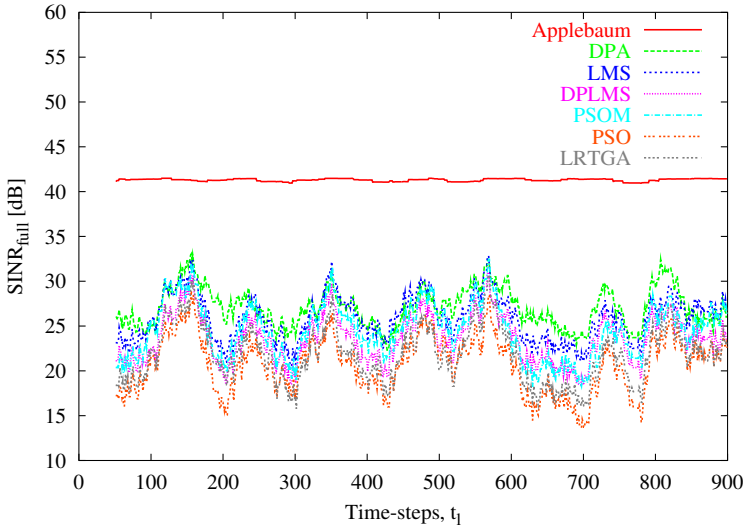


(c)

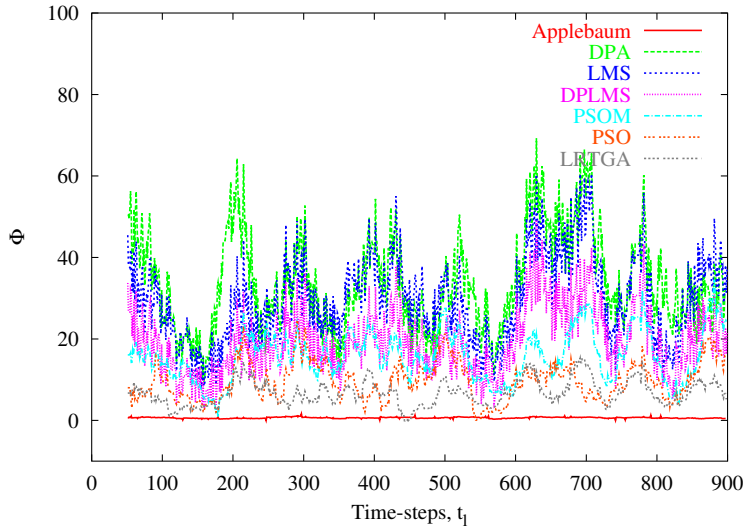
Fig. 8 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."



(*a*)

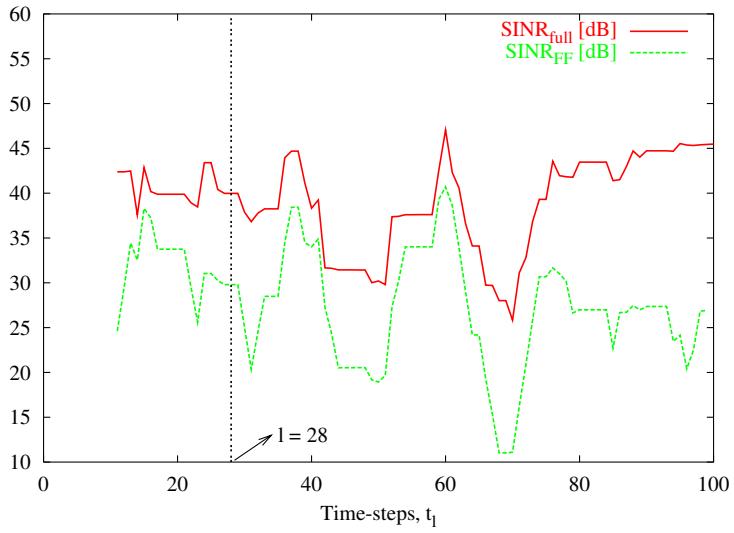


(*b*)

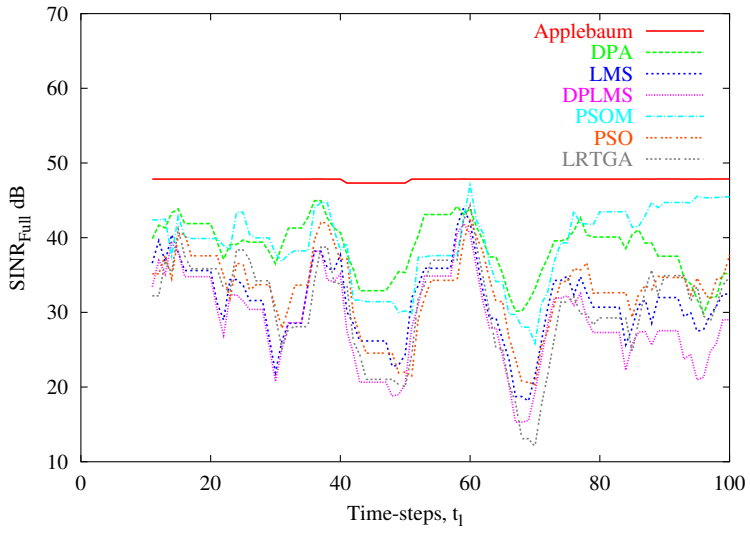


(*c*)

Fig. 9 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."



(a)



(b)

Fig. 10 - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

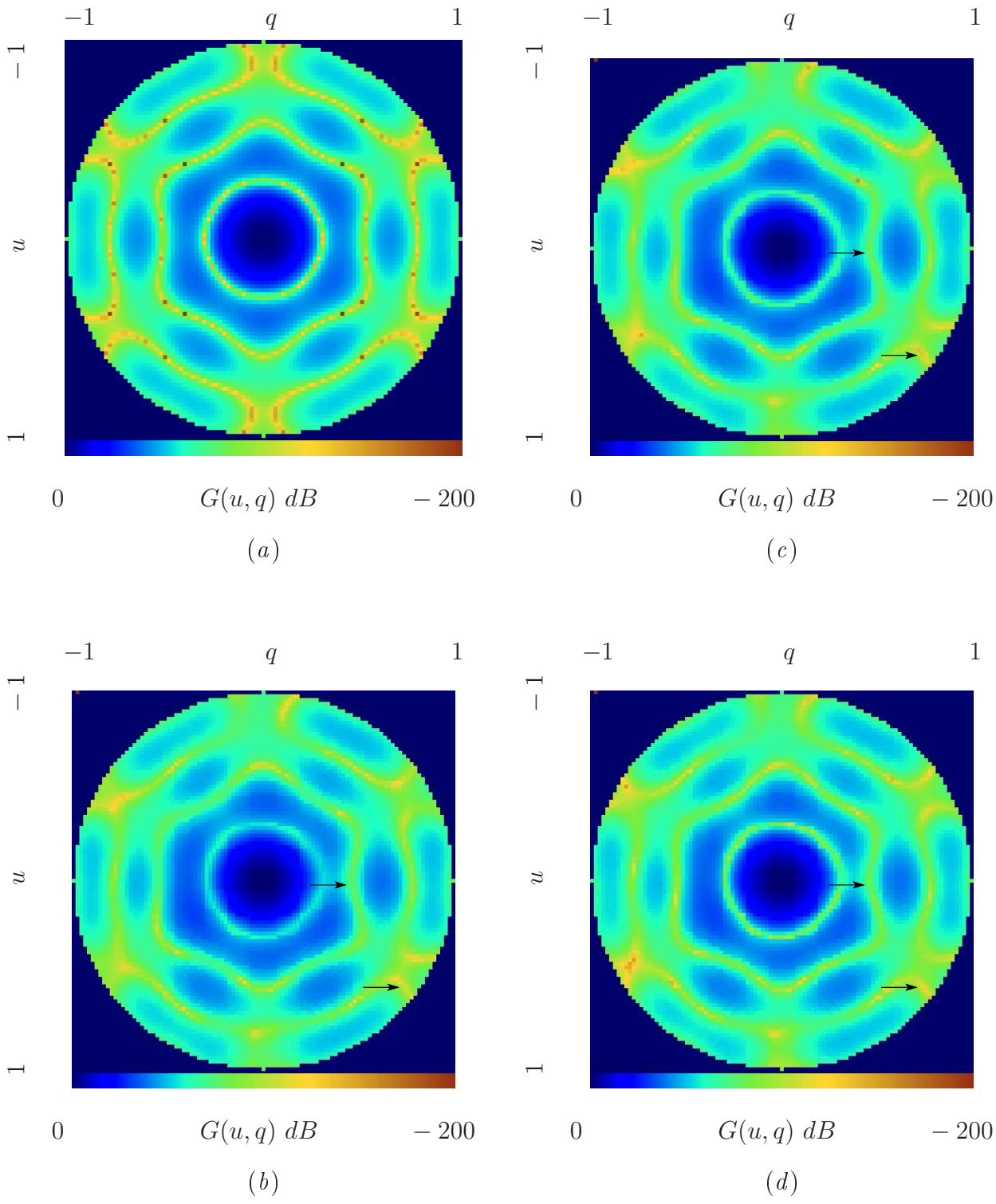


Fig. 11 (*I*) - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

α	$\langle SINR \rangle \ \ [dB]$
$0.4 \rightarrow 0.9$	13.81
0.9	13.73
0.4	13.86
0.1	13.93
0.01	13.84

Table I - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

ℓ	θ_1	ϕ_1	$\rho_1\,[\lambda]$
$0 \longrightarrow 330$	90	165	5
$330 \longrightarrow 660$	90	120	10
$660 \longrightarrow 990$	90	42	7

Table II - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

M	Δ
5	10.9
10	27.3
20	39.7
40	41.1

Table III - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

Control Algorithm	T_{k_ℓ} $[ms]$	$\mathcal{T}$
PSOM	1.62	1.02
PSO	1.59	1.0
LRTGA	6.48	4.07

Table IV - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

	$\langle SINR_{Full} \rangle \quad [dB]$	$\langle SINR_{FF} \rangle \quad [dB]$
Applebaum	42.80	42.52
DPA	27.05	20.44
LMS	25.82	20.09
$DPLMS$	23.54	19.82
$PSOM$	29.90	23.25
PSO	28.82	22.81
$LRTGA$	25.56	22.87

Table V - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."

	$\langle SINR_{Full} \rangle$ $[dB]$	$\langle SINR_{FF} \rangle$ $[dB]$
Applebaum	42.80	42.52
DPA	27.05	20.44
LMS	25.82	20.09
$DPLMS$	23.54	19.82
$PSOM$	24.48	21.24
PSO	21.98	20.07
$LRTGA$	21.84	20.55

Table VI - M. Benedetti *et al.*, "Memory Enhanced PSO-based ..."