UNIVERSITY
OF TRENTO

DIPARTIMENTO DI INGEGNERIA E SCIENZA DELL'INFORMAZIONE

38123 Povo – Trento (Italy), Via Sommarive 14
http://www.disi.unitn.it

AN IMPROVED EXCITATION MATCHING METHOD BASED ON AN ANT COLONY OPTIMIZATION FOR SUBOPTIMAL-FREE CLUSTERING IN SUM-DIFFERENCE COMPROMISE SYNTHESIS

P. Rocca, L. Manica, and A. Massa

January 2011

Technical Report # DISI-11-026

An Improved Excitation Matching Method based on an Ant Colony Optimization for Suboptimal-Free Clustering in Sum-Difference Compromise Synthesis

P. Rocca, L. Manica, and A. Massa

Department of Information Engineering and Computer Science,

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

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

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

{*paolo.rocca*, *luca.manica*}*@dit.unitn.it*

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

An Improved Excitation Matching Method based on an Ant Colony Optimization for Suboptimal-Free Clustering in Sum-Difference Compromise Synthesis

P. Rocca, L. Manica, and A. Massa

Abstract

Dealing with an excitation matching method, this paper presents a global optimization strategy for the optimal clustering in sum-difference compromise linear arrays. Starting from a combinatorial formulation of the problem at hand, the proposed technique is aimed at determining the sub-array configuration expressed as the optimal path inside a directed acyclic graph structure modelling the solution space. Towards this end, an ant colony metaheuristic is used to benefit of its hill-climbing properties in dealing with the non-convexity of the sub-arraying as well as in managing graph searches. A selected set of numerical experiments are reported to assess the efficiency and current limitations of the ant-based strategy also in comparison with previous local combinatorial search methods.

Key words: Sum and Difference Patterns Synthesis, Monopulse Antennas, Linear Arrays, Excitation Matching, Directed Acyclic Graph, Ant Colony Optimization.

1 Introduction

In order to properly solve the "optimal compromise" problem in monopulse radar tracking array antennas, several techniques based on sub-arraying have been proposed to reduce the complexity of the feed network and to realize cheap and compact devices still maintaining a high angular resolution in tracking moving objects. First proposed in [1], the sub-arraying consists in optimizing, for a fixed and ideal sum mode, pre-specified sub-array layouts to synthesize a difference pattern having the slope around the central null as larger as possible (i.e., the highest accuracy of the radar localization) and well-shaped sidelobes for clutter and interferences rejection. In his work [1], *McNamara* mathematically determined the compromise solution through the solution of an over-determined system of linear equations coming from the excitation matching of the independently optimal sum [2][3] and difference [4][5] sets, while the sub-arraying was *a-priori* set. Unfortunately, such an approach usually requires many sub-arrays to get satisfactory results. Therefore, the synthesis of large arrays with a non-negligible number of elements turns out to be practically infeasible.

Alternative approaches based on evolutionary algorithms have been developed [6][7][8][9]. These techniques do not suffer from the ill-conditioning as the formulation in [1] and have shown to work effectively dealing with complex functionals. Moreover, they allow a simple and efficient inclusion of *a-priori* information and only require a suitable definition of the objective function to be optimized (e.g., sidelobe level [6][7][8][9] or directivity [10]). On the other hand, it cannot be neglected that the computational load and the memory requirements rise very rapidly when the number of array elements increases even if enhanced versions (e.g., [11]) are used.

Recently, a new approach still based on the optimal excitations matching has been proposed in [12]. Besides the methodological and algorithmic novelties introduced, the main result yielded is the proof that the compromise synthesis problem can be formulated as a combinatorial one where the dimension of the solution space grows as a binomial function of the number of array elements (and not exponentially as in classical optimization formulations). Moreover, only the sub-array aggregations are looked for, while the sub-array weights are obtained as a "free by-product". In order to solve the problem at hand, the solution space has been represented

through a tree structure where the best compromise solution corresponds to the minimum cost path. Moreover, an *ad-hoc* local search strategy (called ***BEM*)** has been implemented to effectively sample the solution space. In spite of the good results obtained in pattern matching [12][13], or boresight slope optimization [14], and *SLL* control [15] also further improved by means of a hybrid approach [16] aimed at exploiting the problem convexity [9][17] once the aggregation has been set, the whole procedure could suffer from a misleading clustering of the array elements that would deeply influence the second step (i.e., the weight computation) since the functional to be optimized is non-convex with respect to the sub-array memberships of the array elements. To avoid this drawback, global optimization is required for solving the clustering step since local searches could get stuck into local minima. However, "standard" evolutionary techniques or general purpose optimizers cannot be adopted because of their computational costs especially when dealing with high-dimension problems and *ad-hoc* algorithms must be used. Accordingly, this paper describes and analyzes the performance of a suitable state-of-the-art evolutionary strategy, namely the Ant Colony Optimizer (ACO) [18], whose intrinsic structure seems to be very appropriate to fully exploit a suitable defined graph-like model of the solution space. The preliminary assessment carried out in [19] and concerned with a tree-based representation of the solution space provided some indications on the effectiveness of the ACO in dealing with compromise problems. As a matter of fact, such an approach should in principle avoid the local minima of the cost function because of its *hill climbing* behavior as a global optimizer. On the other hand, it should perform better than other 'physically inspired' optimization algorithms because its intrinsic combinatorial nature able to fully adapt to the description of the solutions as an ensemble of contiguous partitions [12].

The outline of the paper is as follows. In Section 2, the synthesis problem of the optimal compromise among sum and difference patterns is formulated in terms of combinatorial optimization by also representing the solution space through an effective graph-like structure for dealing with high-dimensionality. Then, after a short review of the BEM (Sect. 2.1), the ACO for graph-searching is carefully described (Sect. 2.2). In Section 3, the results of a selected set of numerical experiments are reported in order to firstly describe the ACO behavior and then to point out its advantages and best features compared to the BEM. Finally, some conclusions

are drawn (Sect. 4).

2 Mathematical Formulation

Let us consider an isophoric linear array of N equally-spaced elements. The array excitations are supposed to be real and symmetric with respect to the antenna center, such that only $M = \frac{N}{2}$ elements are considered in the calculations. The problem of the compromise between sum and difference patterns can be mathematically formulated as the constrained minimization of a function $\Psi\left(\underline{A}, \underline{W}\right)$ [17] whose arguments are the sub-array vector $\underline{A} = \{a_m \in [1, Q]\,;\, m = 1, ..., M\}$, where $a_m = q$ if the m-th radiating element belongs to the q-th sub-array, and the real values of the Q sub-array weights $\underline{W} = \{w_{a_m};\, a_m = 1, ..., Q\}$. For a given and optimal sum mode set $\underline{\Sigma} = \{s_m = s_{-m};\, m = 1, ..., M\}$ [2][3], the problem solution is the definition of the two unknown vectors $\underline{A}$ and $\underline{W}$, such that the compromise difference excitations $\underline{B} = \{b_m = -b_{-m};\, m = 1, ..., M\}$ given by

$$b_m = s_m w_{a_m}\,;\, m = 1, ..., M \tag{1}$$

afford a difference pattern that satisfies an user-defined requirement (e.g., matching a reference difference pattern or difference pattern-features optimization as good trade-off among deep slope at boresight, low sidelobes, and narrow beamwidth). The coefficient w_{a_m}, $a_m = 1, ..., Q$, in (1) is the weight of the sub-array to which the n-th element belongs to. Figure 1 shows a sketch of the antenna feed network where only half array structure is shown. Dealing with an excitation matching problem [1], once the independently optimal sum $\underline{\Sigma}$ [2][3] and difference $\underline{\Delta} = \{d_m = -d_{-m};\, m = 1, ..., M\}$ [4][5] excitations are given, the compromise synthesis recasts as the minimization of the square distance between the optimal/target $\underline{\Delta}$ and actual/compromise $\underline{B}$ difference coefficients

$$\Psi\left(\underline{A}, \underline{W}\right) = \frac{1}{M} \left\| \underline{\Delta} - \underline{B} \right\|^2 . \tag{2}$$

In [12], it has been shown that after simple algebra Eq. (2) turns out to be a weighted summation of square distances

$$\Psi\left(\underline{A},\, \underline{W}\right) \;=\; \frac{1}{M}\sum_{m=1}^{M}\left[s_m\left(g_m - w_{a_m}\right)\right]^2 \tag{3}$$

where each term of the summation is weighted by the corresponding s_m^2 value, $\frac{1}{M}$ being a normalization coefficient. Moreover, $g_m = \frac{d_m}{s_m}$, $m = 1, \ldots, M$, are the so-called *optimal weights* [12]. Equation (3) defines a least square problem, where each term of the summation $(g_m - w_{a_m})^2$, $m = 1, ..., M$, is the square distance between the m-th *optimal weight* g_m (known) and the corresponding sub-array weight w_{a_m} (unknown).

By virtue of the fact that all terms in (3) are real-valued (i.e., $s_m,\, g_m,\, w_{a_m} \in \mathbb{R}$, $m = 1, ..., M$; $a_m = 1, ..., Q$) and the problem solution is a least square solution, the conclusions drawn in [21] also apply in such a case and can be profitably exploited to reach a suitable compromise solution. More specifically, (*a*) a least square partition that minimizes the cost function in (3) is a *contiguous partition* [1]; (*b*) the number of *essential* contiguous partition is finite and equal to $T^{(ess)} = \begin{pmatrix} M-1 \\ Q-1 \end{pmatrix}$; (*c*) the values of the sub-array weights w_{a_m}, $a_m = 1, \ldots, Q$, are equal to

$$w_{a_m}\left(\underline{A}\right) = \frac{\sum_{r=1}^{M} s_r^2 \delta_{a_r a_m} g_r}{\sum_{r=1}^{M} s_r^2 \delta_{a_r a_m}} \tag{4}$$

since, for a given contiguous partition $\underline{A}$, the point minimizing the sum of the square distances in each contiguous subset (i.e., a convex set containing the elements assigned to the same sub-array) is the weighted arithmetic mean of the corresponding g_m values. In (4), $\delta_{a_m a_r}$ is the Kronecker delta ($\delta_{a_m a_r} = 1$ if $a_r = a_m$ and $\delta_{a_m a_r} = 0$ otherwise). According to these guidelines, the problem solution only requires the synthesis of the optimal clustering $\underline{A}^{opt}$ since the sub-array weights $\underline{W}^{opt}$ are computed as a by-product through Eq. (4) [i.e., $w_{a_m}^{opt} = w_{a_m}\left(\underline{A}^{opt}\right)$, $a_m = 1, ..., Q$].

[1] A partition is called *contiguous* when given any three real elements (g_i, g_j, and g_k), where $g_i < g_j < g_k$, if g_i and g_k belongs to a subset, then also g_j has to belong to the same subset.

2.1 BEM for Graph-Searching

In [12], it has been shown how the solution space of the *contiguous partitions* can be represented in an effective fashion through a non-complete binary tree of depth $M-1$, wherein each level of the tree from the root to the leaves defines the sub-array membership for an element of the array. A more compact and non-redundant structure able to give a complete representation of the whole set of admissible sub-array configurations is based on a *Directed Acyclic Graph* (DAG) [20]. As a matter of fact, the non-complete binary tree of [12] can be reduced to an equivalent DAG by simply noticing that some parts of the tree recursively repeat themselves. Generally speaking, the DAG is a graph $G = (\underline{V}, \underline{E})$ composed by a set of $\underline{V}$ vertexes and $\underline{E}$ edges indicated in Fig. 2 (for the case when $M = 10$ and $Q = 3$) by circles and arrows, respectively. As regards to the compromise problem, the DAG is made of Q rows (i.e., the number of sub-arrays) and $M - Q + 1$ vertexes within each row (i.e., the maximum number of elements that can be assigned to a single sub-array by considering non-null clusters). Moreover, the paths inside the solution graph have the same length [(2)] equal to $M-1$ and each path codes a trial sub-array configuration $\underline{A}$.

In order to explore the solution graph looking for the path minimizing (3), the *Border Element Method* (BEM) first proposed in [12] dealing with a tree architecture is adapted here to work with the DAG, as well. Accordingly, the so-called *border elements* [12] are now those elements of the actual configuration/path whereof at least one closest element of the path belongs to a different row of the DAG (i.e., it is assigned to a different sub-array). For sake of clarity and with reference to Fig. 2, the cluster configurations are indicated by the red edges and the border elements are denoted by the blue vertexes. Likewise to [12], it is possible to obtain a new admissible trial aggregation $\underline{A}'$ just changing the membership of a border element.

As far as the sampling of the DAG structure is concerned, the BEM is first aimed at looking for the border elements of the current path $\underline{A}^{(k)}$ belonging to the DAG and successively at changing their memberships (once a time), until a termination criterion based on a maximum number of iterations K ($k = 0, ..., K$; k being the iteration index) or on a stationary condition of the cost function value $\Psi\left\{\underline{A}^{(k)}\right\}$ is reached. For illustrative purposes, a pictorial representation

[(2)] The length of a DAG is equal to the number of edges of the longest directed path.

of the BEM-based searching is given in Fig. 2. Starting from the guess solution $\underline{A}^{(0)}$ displayed in Fig. 2(*a*), the iterative process stops after two iterations determining the final aggregation $\underline{A}^{BEM} = \underline{A}^{(2)}$ shown in Fig. 2(*c*).

2.2 ACO for Graph-Searching

Analyzing the BEM [12], it is simple to recognize that such a method, for both tree and graph-like architectures, is a deterministic technique that suffers of the usually standard drawbacks of local search algorithms. In particular, the BEM solution might be trapped in a local minimum and strongly influenced by the starting guess aggregation $\underline{A}^{(0)}$ chosen at the initialization because of the non-convexity of the problem at hand.

In order to overcome the problems related to the presence of local minima in the cost function (3), the *Ant Colony Optimizer* (ACO) is adopted here to search for the optimal path $\underline{A}^{opt}$ within the solution graph that minimizes (3). The ACO is a global optimization algorithm inspired by the foraging behavior of ant colonies looking for food sources [18]. The ants look for the shortest path between the food sources and the nest. Towards this end, each ant leaves a chemical substance, called *pheromone*, while moving in the space surrounding the nest. The amount of pheromone on a path quantifies its degree of optimality, but it decays with time (*evaporation* mechanism). These mechanisms allow one to avoid poor food sources on one hand (*pheromone release*) and on the other, to efficiently sample the whole solution space (*pheromone evaporation*).

The ACO developed by *Dorigo* [22] has been widely applied especially in distributed and discrete problems such as routing [23][24], assignment [25][26], scheduling [27][28], subset [29], but it is relatively infrequent in electromagnetics. To the best of authors' knowledge, it has been recently applied to few electromagnetic problems (e.g., antenna synthesis considering binary [30] or real implementations [31][32][33] and microwave imaging [34]). However, because of its effectiveness in facing hard combinatorial problems and since the combinatorial formulation of the optimal compromise between sum and difference patterns requires the searching of the best path within a graph, the ACO seems to be a suitable metaheuristic for the problem at hand. Towards this aim, the simplest version of the ACO, namely *Ant System* [18], is used. Unlike

[19] where some preliminary results concerned with the tree-based ACO have been reported, the proposed ACO implementation is customized to the graph architecture to properly address the synthesis of small as well as large arrays. As a matter of fact, due to the high number of vertexes needed for the storage of the solution, the approach in [19] presents some memory limitations when dealing with very large dimensional spaces. On the other hand, it must be pointed out that the ACO performances in terms of solution accuracy do not depend on the representation of the solution space, but only the feasibility and the computational indexes (i.e., the storage resources and the rate of sampling the solution space) are affected by the architecture at hand.

The proposed implementation of the ACO-based approach can be summarized as follows. Each i-th ($i = 1, ..., C$) ant codes a vector $\underline{c}_i$ of M integer values that models a trial sub-array configuration $\underline{A}_i$ (i.e., $\underline{c}_i = \underline{c}\{\underline{A}_i\}$). Every vector is initialized to the null one at each iteration (i.e., $\underline{c}_i^{(k)} = \{0, ..., 0\}$, $k = 1, ..., K$ and $i = 1, ..., C$), such that all ants start from the root of the graph (Fig. 3). Successively, the vectors are filled step-by-step while the ants are moving through each level of the graph as shown in Fig. 3. At the initialization ($k = 0$), the quantity of pheromone on each edge $\tau^{(0)}\left(e_z^r\right)$, $e_z^r = 1, ..., E$ is the same and each edge of the graph can be explored with a uniform probability $p^{(0)}\left(e_z^r\right) = 0.5$. As regards to the apex r, it is equal to $q \rightarrow q$ if the edge e_z^r connects two vertexes belonging to the same sub-array (i.e., the same row of the DAG) and to $q \rightarrow q + 1$ if it connects two vertexes assigned to different sub-arrays (i.e., different rows of the DAG). Moreover, the pedex z, $z = z_1, ..., z_{M-1}$, identifies the level of the edge within the graph. Concerning the iterative loop ($k > 0$), the probability of choosing one of the two subsequent edges (if present) at each vertex is given by

$$p^{(k)}\left(e_z^r\right) = \frac{\tau^{(k)}\left(e_z^r\right)}{\tau^{(k)}\left(e_z^{q \rightarrow q}\right) + \tau^{(k)}\left(e_z^{q \rightarrow q+1}\right)}, \quad z = z_1, ..., z_{M-1}; \; r = q \rightarrow q + [0, 1]\,. \tag{5}$$

When the whole ant colony has completed a path within the DAG, the pheromone level $\tau^{(k)}\left(e_l^r\right)$ of each edge is updated as follows

$$\tau^{(k+1)}\left(e_z^r\right) \leftarrow \tau^{(k)}\left(e_z^r\right) + \sum_{i=1}^{C} \delta_{e_z^r \underline{c}_i^{(k)}} \frac{H}{\Psi\left(\underline{A}_i^{(k)}\right)}, \quad \forall \tau^{(k)}\left(e_z^r\right) \tag{6}$$

where $\delta_{e_z^r \underline{c}_i^{(k)}} = 1$ when $e_z^r \in \underline{c}_i^{(k)}$ [$\underline{c}_i^{(k)} = \underline{c}\left\{\underline{A}_i^{(k)}\right\}$] and $\delta_{e_z^r \underline{c}_i^{(k)}} = 0$ otherwise, H being a positive constant. Successively, the evaporation procedure takes place in order to reduce and at most delete worse paths from the graph

$$\tau^{(k+1)}\left(e_z^r\right) \leftarrow (1-\rho)\,\tau^{(k+1)}\left(e_z^r\right), \quad \forall \tau^{(k+1)}\left(e_z^r\right) \tag{7}$$

$\rho \in (0\,,\,1]$ being a parameter aimed at controlling the evaporation rate. Finally, the same stopping criterion ($k = k_{end}$) used for the BEM is adopted here for the ACO-based method to allow fair comparisons.

3 Numerical Simulations and Results

Because of the novelty of the proposed approach, the first part of this section (Sect. 3.1) is devoted to the calibration of the ACO algorithm [35] when dealing with the searching of the "best compromise" solution among those admissible within the solution graph. Successively, the use of the ACO is motivated (Sect. 3.2) showing how the BEM solution suffers from the non-convexity of the aggregation problem because of the local nature of the algorithm. Finally, a set of comparative results concerned with a wide number of compromise problems are reported (Sect. 3.3) to point out potentialities and current limitations of the ACO-based approach. Since both BEM and ACO are aimed at determining the *best compromise* difference pattern close as much as possible to the optimal one, besides the EMM by *McNamara* [1] discussed in [12], no comparisons with other pattern-features optimization procedures (i.e., [6][7][8][9][15][16][17]) will be reported since these latter are devoted to satisfy different criteria and not at better matching an optimal difference pattern.

3.1 *ACO* Calibration

A key feature of the ACO algorithm is the simple implementation. As a matter fact, besides the number C of ants in the colony, it only requires the definition of two parameters to work, namely the pheromone update coefficient H and the pheromone evaporation coefficient ρ. In order to determine their optimal values for the problem at hand, an extensive set of numerical

experiments has been carried out by considering an array of $N = 40$ elements and $Q = 6$ sub-arrays as reference benchmark. In this case, the number of contiguous partitions is equal to $T^{(ess)} = \begin{pmatrix} 19 \\ 5 \end{pmatrix} = 11628$. As far as the reference excitations are concerned, those affording a Dolph-Chebyshev sum pattern with $SLL = -25\,dB$ [3] and a Zolotarev difference pattern with $SLL = -30\,dB$ [36] have been chosen. Concerning the calibration study, the values of the ACO control coefficients have been varied in the range $H \in [0 : 5]$ and $\rho \in (0 : 1]$ [18], respectively. Moreover, because of the stochastic nature of the ACO algorithm, 100 different simulations have been performed for each setting of the calibration parameters. Each simulation has been run with a number of ants equal to $C = [3,\ 5,\ 8,\ 10,\ 100,\ 1000]$ for a maximum number of $K = 1000$ iterations.

As a representative result, the average performances for each parameter configuration when $C = 3$ are reported in Fig. 4. As it can be observed, the convergence cost function value is more sensitive to the evaporation coefficient ρ and less to the value of the parameter H that controls the pheromone update. A similar conclusion holds true whatever the value of C. Concerning the optimal setup, the configuration $H = 1$ and $\rho = 0.05$ has been selected since the corresponding representative point in Fig. 4 lies in the "lowest-Ψ-value" region and the value $H = 1$ has already been identified as an optimal choice in other graph searching problems (e.g., *Traveling Salesman Problem* [23]).

As regards to the dimension of the ant colony, the analysis has been devoted to define the optimal value of C in relationship to the dimension of the solution space $T^{(ess)}$. Towards this end, C has been varied between 1 and $\frac{1}{10}T^{(ess)}$. Figure 5 shows the results of the statistical study, each cross being the average Ψ among the values reached at the end of each group of 100 simulations. For completeness, the standard deviation is shown, as well. From these results, it can be inferred that the choice $C \simeq \left[\frac{1}{125}T^{(ess)} : \frac{1}{100}T^{(ess)}\right]$ defines a good rule of thumb to reach the global solution with a percentage above 90% [(3)] . On the other hand, the minimum value of $C_{lb} = 5$ ants has been set as lower bound in order to exploit the cooperative behavior of the ACO in those problems where the previous criterion would give too small values (i.e.,

[(3)] It is worth noting that the results here reported have been obtained under the assumption of a maximum number of iterations equal to $K = 1000$. Probably, increasing the number of iterations would allow a reduction of the number of ants for obtaining the same conclusions.

$C < C_{lb}$).

3.2 *ACO*'s Hill-Climbing Behavior

In order to show how the performance of the BEM [12] are influenced from the choice of the initial solution, while the ACO is not dependent on the starting guess and therefore more robust to the local minima problem thanks to its hill-climbing properties, three samples of compromise syntheses concerned with small as well as larger arrays for different number of sub-arrays are discussed in the following.

The first experiment deals with a 20-elements array ($M = 10$) with inter-element spacing $d = \frac{\lambda}{2}$. The optimal sum and difference coefficients have been chosen to afford a Dolph-Chebyshev sum pattern with $SLL = -25\,dB$ [3] and a Zolotarev difference pattern with $SLL = -30\,dB$ [36], respectively. As regards to the compromise feed network, $Q = 3$ sub-arrays have been used.

Concerning the *Contiguous Partition Method* (CPM) presented in [12] and customized in the present work to the searching within the solution graph, the optimal weights g_m, $m = 1, \ldots, M$, are first computed as described in Sect. (2) and then sorted on a line in order to obtain the list $\underline{L} = \{l_h : l_h \leq l_{h+1},\, h = 1, ..., M-1\}$ [12], where $l_1 = min\{g_m\}$ and $l_M = max\{g_m\}$. Each element of the sorted list $\underline{L}$ is assigned to a level of the solution graph as shown in Fig. 2. Starting from a uniform sub-arraying (i.e., a sub-array configuration wherein the number of elements within each sub-array differs at most of one element when M is or not a multiple of Q), the initial sub-array vector turns out to be $\underline{A}^{(0)} = \{1\,1\,1\,2\,2\,3\,3\,3\,2\,1\}$ and $\Psi\left(\underline{A}^{(0)}\right) = 2.17 \times 10^{-2}$. Then, the iterative loop of the BEM takes place as described in Sect. 2.1. For completeness, Figure 2 shows the corresponding evolution of the BEM trial solution in the solution graph. As it can be noticed, the BEM gets stuck only after $k_{end}^{BEM} = 2$ iterations. The final grouping is $\underline{A}^{BEM} = \underline{A}^{(2)} = \{1\,1\,2\,2\,3\,3\,3\,3\,2\,1\}$ [Fig. 2(*c*)] with a convergence fitness value of $\Psi\left(\underline{A}^{BEM}\right) = 1.08 \times 10^{-2}$, while the intermediate solution $\underline{A}^{(1)} = \{1\,1\,2\,2\,2\,3\,3\,3\,2\,1\}$ [Fig. 2(*b*)] has a fitness equal to $\Psi\left(\underline{A}^{(1)}\right) = 1.48 \times 10^{-2}$. The radiation patterns generated at the various iterations and the reference pattern are reported in Fig. 6, as well.

Successively, the ACO has been applied to the same test case. Since the number of trial so-

lutions within the graph is equal to $T^{(ess)} = \begin{pmatrix} 9 \\ 2 \end{pmatrix} = 36$ and C, according to the criterion previously defined, would result lower than one, the ACO population has been set to $C = C_{lb} = 5$. Moreover, the pheromone update H and the evaporation ρ have been fixed to their optimal values. As expected, the ACO outperforms the BEM since the fitness value of the synthesized solution $\underline{A}^{ACO} = \{1\,2\,2\,3\,3\,3\,3\,3\,3\,2\}$ is equal to $\Psi\left(\underline{A}^{ACO}\right) = 8.26 \times 10^{-3}$ [vs. $\Psi\left(\underline{A}^{BEM}\right) = 1.08 \times 10^{-2}$]. To further confirm the ACO effectiveness, it is worth noting that the clustering determined by the ACO is the one having the minimum fitness among the $T^{(ess)} = 36$ admissible different clustering. On the contrary, the BEM has been able to retrieve the second best solution coded into the solution graph as shown in Fig. 7 (red line) where each cross denotes the Ψ value among the $T^{(ess)} = 36$ contiguous partitions ranked according to their cost function values. More specifically, the BEM solution is evidenced with a circle, while the minimum fitness value or global minimum of the excitation matching cost function coincides with the ACO clustering [i.e., $\Psi^{opt} = \Psi\left(\underline{A}^{ACO}\right)$]. On the other hand, it is also interesting to point out that, even though the BEM solution is the second best compromise, it has four elements over ten whose sub-array memberships are different from those of the global optimum $\underline{A}^{opt}$ recognized by the ACO-based algorithm, $\underline{A}^{ACO} = \underline{A}^{opt}$.

For completeness, Table I details the results obtained with the BEM and the ACO by reporting the final sub-array configurations and the weight values. Moreover, the synthesized difference compromises are shown in Fig. 8(*a*). Because of the excitation-matching nature of the proposed technique, let us quantify the closeness of the arising patterns with respect to the optimal/reference one by computing the *pattern matching* Δ index [12] defined as follows [(4)]:

$$\Delta = \frac{\int_0^1 \left| |AF(u)|_n^{ref} - |AF(u)|_n^{syn} \right| du}{\int_0^1 |AF(u)|_n^{ref} \, du}, \tag{8}$$

where $u = sin\theta$, $\theta \in [0, \pi/2]$, $|AF(u)|_n^{ref}$ and $|AF(u)|_n^{syn}$ are the normalized reference pattern and the synthesized one, respectively. As expected and indicated by the corresponding lower

[(4)] Such an index is the main performance indicator since both ACO and BEM are concerned with an excitation matching problem [1][12][13] and not with the minimization of a pattern parameter (e.g., the SLL) as it happens in [6][7][8][9][15][16][17].

fitness value, the ACO pattern is closer to the reference one. As a matter of fact, it is $\Delta^{ACO} = 0.2689$ vs. $\Delta^{BEM} = 0.3199$ (Tab. II). Table II also reports the values of other indexes in order to give a complete overview of the features of the obtained patterns (i.e., sidelobe level, SLL, and main lobe width, BW). Moreover, the computational issues are pointed out by the following indexes: the number of convergence iterations, k_{end}, the number of function evaluations, F_{end}, and the CPU-time t necessary to find $\underline{A}^{(k_{end})}$ on a $3.4\,GHz$ PC with $2\,GB$ of RAM. As it can be noticed, both BEM and ACO are able to find a convergence solution almost in real time since $t < 10^{-8}$. Such an event points out once again the computational efficiency of the CPM approach [13], but also the usefulness of the graph representation that enables the use of an evolutionary algorithm without excessively increasing the computational costs and memory resources.

In the second experiment, the same array geometry of the previous example has been considered, but the array has been partitioned into $Q = 8$ sub-arrays. Moreover, a Zolotarev difference pattern with $SLL = -40\,dB$ [36] has been adopted as reference target. It is worth observing that despite the higher number of sub-arrays, the dimension of the solution space is still equal to $T^{(ess)} = 36$ thanks to the symmetric nature of the binomial distribution [i.e., $T^{(ess)} = \begin{pmatrix} 9 \\ 7 \end{pmatrix} = \begin{pmatrix} 9 \\ 2 \end{pmatrix} = 36$]. Analogously to the previous example, the BEM stops after $k_{end}^{BEM} = 2$ iterations synthesizing the solution in Tab. III, but in this case other 8 solutions with lower fitness values are present in the solution graph (Fig. 7 - green line). On the other hand, the ACO has been able to reach the global optimum in Tab. III after $k_{end}^{ACO} = 2$ iterations with a total number of fitness evaluation equal to $F_{end}^{ACO} = 10$ since $C = C_{lb} = 5$. In particular, the ACO solution presents a fitness value of more than one order in magnitude below the one of the BEM [i.e., $\Psi\left(\underline{A}^{ACO}\right) = 1.13 \times 10^{-5}$ vs. $\Psi\left(\underline{A}^{BEM}\right) = 2.49 \times 10^{-4}$] and $\frac{\Delta^{BEM}}{\Delta^{ACO}} \simeq 3.76$ as it can be qualitatively observed by comparing the patterns in Fig. 8(*b*). For the sake of completeness, Table II compares the retrieved solutions in terms of performance indexes.

The last experiment of this section is concerned with a larger uniform array of 40 $\frac{\lambda}{2}$-spaced elements. A Dolph-Chebyshev sum pattern with $SLL = -25\,dB$ [3] and a Zolotarev difference pattern with $SLL = -30\,dB$ [36] have been chosen as reference patterns and the number of sub-arrays has been set to $Q = 4$. In such a case, the number of possible sub-array configurations

within the solution space is equal to $T^{(ess)} = 969$. As far as the ACO is concerned, $C = 10$ ants have been used. The two approaches have found the corresponding solutions after $k_{end}^{BEM} = 21$ and $k_{end}^{ACO} = 34$ as shown in Fig. 9 where the behavior of the cost function during the iterative searching process for both the BEM and the ACO is described. The synthesized sub-array configurations and weights are given in Tab. IV, whereas the corresponding patterns indexes are reported in Tab. II. As expected and likewise to the previous experiments, the BEM is still trapped into a local minimum and the retrieved solution turns out to be sub-optimal. However, it should be observed (Fig. 7 - blue line) that the BEM configuration is the third best contiguous partition among $T^{(ess)} = 969$ different solutions and the value of the ratio $\frac{\Delta^{BEM}}{\Delta^{ACO}} \simeq 1.11$ assesses its closeness to the optimal one. As regards to the computational issues, such a test further confirms the efficiency of the BEM (in terms of speed) in exploring the solution space being $t^{BEM} < 10^{-7}$ while $t^{ACO} = 4.5 \times 10^{-3}$. As a matter of fact, although the CPU-time required by the ACO-based approach is certainly smaller than that of standard global optimizers, it cannot be omitted that from a computational point of view the BEM results more competitive than the ACO when the ratio $\frac{M}{Q}$ gets larger and larger. Such a statement will be further analyzed in the following section.

3.3 *ACO*'s Performances and Problem Dimensions

In dealing with the optimal compromise between sum and difference patterns, different global optimization techniques have been applied to determine the most suitable partition of the array elements into sub-arrays that minimizes a suitable cost function related to some pattern features. Among them, it is worth mentioning the *Genetic Algorithm* [7], the *Differential Evolution Algorithm* [8] and its enhanced version [11], and the *Simulated Annealing* [9]. Despite the different way of tackling the problem at hand (i.e., direct optimization of element memberships and weights [7][8][11] or two-step nested approach [9] exploiting functional convexity), the dimension of the solution space to be explored for retrieving the elements aggregation is equal to $T^{(tot)} = Q^M$ since each clustered configuration can be expressed as a string of M digits in a Q-based notation system [12]. Let us now suppose to use in a standard fashion (i.e., without reformulating the problem at hand as a combinatorial one) a global optimizer and to apply the

rule deduced in Sect. (3.1) for the population size [i.e., $C^{(tot)} \simeq 10^{-2} \times T^{(tot)}$] for running a simulation in a fixed number of iterations $\hat{K}$ looking for the optimal aggregation within the set of $T^{(tot)}$ possible solutions. The total CPU time necessary to complete such a simulation turns out be $\Delta t^{(tot)} = \delta t \times \hat{K} \times C^{(tot)}$, δt being the CPU-time for one evaluation of the cost function. Moreover, it should be pointed out that there is not guarantee that the synthesized aggregation is the global optimum of the functional at hand. Then, let us refer to the combinatorial formulation of the compromise problem and map the reduced solution space of dimension $T^{(ess)}$ into the graph representation described in Sect. 2.1. By exploiting such a structure and accordingly using the proposed implementation of the ACO, the number of ants of the colony turns out to be $C^{(ess)} \simeq 10^{-2} \times T^{(ess)}$ much smaller than $C^{(tot)}$ since $T^{(ess)}$ grows at most polynomially [i.e., $T^{(ess)} = \begin{pmatrix} M-1 \\ Q-1 \end{pmatrix}$] and not exponentially as $T^{(tot)}$ [$T^{(tot)} = Q^M$]. Therefore, the iterative optimization runs for a time $\Delta t^{(ess)} = \delta t \times \hat{K} \times C^{(ess)}$, which satisfies the following condition $\Delta t^{(ess)} \ll \Delta t^{tot}$ [5] since $C^{(ess)} \ll C^{(tot)}$. Such a conclusion clearly evidences the significant reduction of the computational burden as well as the more profitable and proper use of a suitable global optimization technique within the combinatorial framework. As a matter of fact, although also in this case the convergence to the global optimum solution is not guaranteed, the probability of reaching it significantly grows compared to the standard use of global optimizers. In order to detail such an argumentation, let us assume one has at disposal a limited amount of time $\Delta t^{(tot)}$ for defining the best aggregation for the compromise problem at hand. On one hand, the ACO-based approach would have $\Delta K = \hat{K}' - \hat{K}$ more iterations for exploring the solution space, being $\hat{K}' = \frac{\Delta t^{(tot)}}{\delta t \times C^{(ess)}}$. On the other hand, it would be possible to use a larger colony of $C_1^{(ess)} = \frac{\Delta t^{(tot)}}{\delta t \times \hat{K}}$ ants for the same number of iterations $\hat{K}$ and the following conditions would hold true: $C_1^{(ess)} \gg C^{(ess)}$ and $C_1^{(ess)} \simeq T^{(ess)}$. In this latter case, the convergence of the ACO-based procedure to the optimum clustering would be assured since each ant could be assigned to explore a single and different path of the solution graph thus covering/sampling the whole solution space.

In order to assess and confirm these indications, Figures 10 and 11 summarize the performance

[5] For the sake of simplicity, δt has been assumed to be equivalent for both standard and combinatorial optimizations. However, please also consider that $\delta t^{(ess)} < \delta t^{(tot)}$ since usually $\delta t^{(tot)}$ requires the computation of a pattern feature, while $\delta t^{(ess)}$ is related to a matching operation Eq. (3).

achieved with the BEM and ACO methods. The plots refer to a representative set of simulations performed by varying the number of elements of the array aperture between $N = 20$ and $N = 500$, but maintaining a uniform inter-element distance ($d = \frac{\lambda}{2}$). In all the experiments, the sets of reference excitations have been chosen to generate a Dolph-Chebyshev sum pattern with $SLL = -25\,dB$ [3] and a Zolotarev difference pattern with $SLL = -25\,dB$ [36]. Moreover, the number of sub-arrays has been fixed to $Q = 8$. As regards to the ACO values, they are related to the average performance over a statistical set of 50 independent executions of the same simulation (i.e., with the same parametric configuration, but varying the randomness in the ACO). In particular, the plots denoted by ACO and ACO^* indicate the values obtained when the ACO algorithm has been run for $\hat{K} = 1000$ iterations with a colony of $C^{(ess)}$ and $C_1^{(ess)}$ ants, respectively. As expected, the ACO-based approach with $C_1^{(ess)}$ trial solutions for each iteration always outperforms the BEM. Unfortunately, when $T^{(ess)}$ turns out to be too large, both the computational load and the storage requirements of the ACO result quite cumbersome and once again, although with larger dimensions, verify the same drawbacks usually encountered by standard global optimizers when dealing with non-small array geometries. In such a situation, the BEM seems to be more attractive even though less robust against local minima problems.

4 Conclusions

In a recent paper, it has been shown how the excitation matching formulation of the optimal compromise problem can be recast as a combinatorial one by exploiting the knowledge of independently optimal sum and difference modes. Thanks to a tree representation of the set of admissible solutions, a local search strategy, called border element method (BEM), has been implemented to efficiently explore the reduced solution space with a large saving of computational resources. Instead, an ACO-based technique has been considered in this paper in order to avoid the occurrence of sub-optimal aggregations caused by the presence of local minima in the non-convex excitation matching functional. Towards this end, the solution space has been described through a directed acyclic graph and the Ant Colony Optimizer has been used to look for the minimum cost path of the graph fully exploiting its intrinsic characteristics very

appropriate for exploring such a kind of architectures.

From the analysis carried out within this research work and summarized in this paper, the following conclusions can be drawn:

- unlike ACO-based approach, both the dimension of the solution space and computational burden rise much more rapidly when standard global optimizers are used. In practice, these standard stochastic algorithms work effectively only with small arrays thus synthesizing array solutions having a limited angular resolution;
- being a local search technique, the BEM depends on the initial solution, but it is an excellent computational saving technique suitable for synthesizing very large arrays ($N \geq 200$) although without any guarantee of avoiding local minima solutions;
- the ACO takes on one side the advantages of global optimization approaches in facing non-convexity, while on the other and to the best of the authors' knowledge, it is the most suitable algorithm among state-of-the-art metaheuristics for path-searching in a graph-represented solution space.

Acknowledgements

A. Massa wishes to thank E. Vico for inspiring this work and C. Pedrazzani for her continuous help and patience. Moreover, the authors are very grateful to Prof. Isernia for useful discussions on sub-arraying and to F. Stringari for kindly providing some numerical results of computer simulations. Furthermore, the authors greatly appreciated the reviewing and valuable comments of the anonymous reviewers.

References

[1] D. A. McNamara, "Synthesis of sub-arrayed monopulse linear arrays through matching of independently optimum sum and difference excitations," *IEE Proc. H Microwaves Antennas Propag.*, vol. 135, no. 5, pp. 371-374, Oct. 1988.

[2] T. T. Taylor, "Design of line-source antennas for narrow beam-width and low side lobes," *Trans. IRE*, vol. AP-3, pp. 16-28, 1955.

[3] C. L. Dolph, "A current distribution for broadside arrays which optimises the relationship between beam width and sidelobe level," *Proc. IRE*, vol. 34, pp. 335-348, 1946.

[4] E. T. Bayliss, "Design of monopulse antenna difference patterns with low sidelobes," *Bell System Tech. Journal*, 47, pp. 623-640, 1968.

[5] D. A. McNamara, "Discrete $\bar{n}$-distributions for difference patterns," *Electron. Lett.*, vol. 22, no. 6, pp. 303-304, Jun. 1986.

[6] F. Ares, S. R. Rengarajan, J. A. Rodriguez, and E. Moreno, "Optimal compromise among sum and difference patterns," *J. Electromag. Waves Appl.*, vol. 10, pp.1143-1555, 1996.

[7] P. Lopez, J. A. Rodriguez, F. Ares, and E. Moreno, "Subarray weighting for difference patterns of monopulse antennas: Joint optimization of subarray configurations and weights," *IEEE Trans. Antennas Propag.*, vol. 49, no. 11, pp. 1606-1608, Nov. 2001.

[8] S. Caorsi, A. Massa, M. Pastorino, and A. Randazzo, "Optimization of the difference patterns for monopulse antennas by a hybrid real/integer-coded differential evolution method," *IEEE Trans. Antennas Propag.*, vol. 53, no. 1, pp. 372-376, Jan. 2005.

[9] M. D'Urso, T. Isernia, and E. F. Meliado', "An effective hybrid approach for the optimal synthesis of monopulse antennas," *IEEE Trans. Antennas Propag.*, vol. 55, no. 4, pp. 1059-1066, Apr. 2007.

[10] A. Massa, M. Pastorino, and A. Randazzo, "Optimization of the directivity of a monopulse antenna with a subarray weighting by a hybrid differential evolution method," *IEEE Antennas Wireless Propag. Lett.*, vol. 5, pp. 155-158, 2006.

[11] Y. Chen, S. Yang, and Z. Nie, "The application of a modified differential evolution strategy to some array pattern synthesis problems," *IEEE Trans. Antennas Propagat.*, vol. 56, no. 7, pp. 1919-1927, Jul. 2008.

[12] L. Manica, P. Rocca, A. Martini, and A. Massa, "An innovative approach based on a tree-searching algorithm for the optimal matching of independently optimum sum and difference excitations," *IEEE Trans. Antennas Propag.*, vol. 56, no. 1, pp. 58-66, Jan. 2008.

[13] P. Rocca, L. Manica, A. Martini, and A. Massa, "Synthesis of large monopulse linear arrays through a tree-based optimal excitations matching," *IEEE Antennas Wireless Propagat. Lett.*, vol. 6, pp. 436-439, 2007.

[14] P. Rocca, L. Manica, M. Pastorino, and A. Massa, "Boresight slope optimization of sub-arrayed linear arrays through the contiguous partition method," *IEEE Antennas Wireless Propag. Lett.*, in press.

[15] P. Rocca, L. Manica, and A. Massa, "Synthesis of monopulse antennas through the iterative contiguous partition method," *Electron. Lett.*, vol. 43, no. 16, pp. 854-856, Aug. 2007.

[16] P. Rocca, L. Manica, and A. Massa, "Hybrid approach for sub-arrayed monopulse antenna synthesis," *Electron. Lett.*,vol. 44, no.2, pp.75- 76, Jan. 2008.

[17] M. D'Urso, T. Isernia, "Solving some array synthesis problems by means of an effective hybrid approach," *IEEE Trans. Antennas Propagat.*, vol. 55, no. 3, pp. 750-759, Mar. 2007.

[18] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man and Cybern. B* , vol. 26, no. 1, pp. 29-41, Feb. 1996.

[19] P. Rocca, L. Manica, F. Stringari, and A. Massa, "Ant colony optimization for tree-searching based synthesis of monopulse array antenna," *Electron. Lett.*, vol. 44, no. 13, pp.783-785, Jun. 2008.

[20] L. Manica, P. Rocca, and A. Massa, "On the synthesis of sub-arrayed planar array antennas for tracking radar applications," *IEEE Antennas Wireless Propag. Lett.*, in press.

[21] W. D. Fisher, "On grouping of maximum homogeneity," *American Statistical Journal*, pp. 789-798, Dec. 1958.

[22] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28 - 39, Nov. 2006.

[23] M. Dorigo and L. M. Gambarella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53-66, Apr. 1997.

[24] M. Reimann, K. Doerner, and R. F. Hartl, "D-ants: Savings based ants divide and conquer the vehicle routing problem," *Computers & Operations Research*, vol. 31, no. 4, pp. 563-591, 2004.

[25] V. Maniezzo, "Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem," *INFORMS J. Comput.*, vol. 11, no. 4, pp. 358-369, 1999.

[26] D. Costa and A. Hertz, "Ants can colour graphs," *J. Operat. Res. Soc.*, vol. 48, pp. 295-305, 1997.

[27] D. Merkle, M. Middendorf, and H. Schmeck, "Ant colony optimization for resource constrained project scheduling," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 333-346, 2002.

[28] D. Merkle and M. Middendorf, "Ant colony optimization with global pheromone evaluation for scheduling a single machine," *Appl. Intell.*, vol. 18, no. 1, pp. 105-111, 2003.

[29] C. Blum and M. J. Blesa, "New metaheuristic approaches for the edge-weight ed k-cardinality tree problem," *Computers & Operations Research*, vol. 32, no. 6, pp. 1355-1377, 2005.

[30] C. M. Coleman, E. J. Rothwell, and J. E. Ross, "Investigation of simulated annealing, ant-colony optimization, and genetic algorithms for selfstructuring antennas," *IEEE Trans. Antennas Propag.*, vol. 52, no. 4, pp. 1007-1014, Apr. 2004.

[31] O. Quevedo-Teruel and E. Rajo-Iglesias, “Ant colony optimization for array synthesis,” in *Proc. Antennas Propag. Soc. Int. Symp.*, Jul. 2006, pp. 3301-3304.

[32] O. Quevedo-Teruel and E. Rajo-Iglesias, “Ant colony optimization in thinned array synthesis with minimum sidelobe level,” *IEEE Antennas Wireless Propag. Lett.*, vol. 5, pp. 349-352, 2006.

[33] E. Rajo-Iglesias and O. Quevedo-Teruel, “Linear Array synthesis using an ant-colony-optimization-based algorithm,” *IEEE Antennas Propag. Mag.*, vol. 49, no. 2, pp. 70-79, Apr. 2007.

[34] M. Pastorino, “Stochastic optimization methods applied to microwave imaging: A review,” *IEEE Trans. Antennas Propag.*, vol. 55, no. 3, pp. 538-548, Mar. 2007.

[35] D. H. Wolpert and W. G. Macready, “No free lunch theorems for optimization,” *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67-82, Apr. 1997.

[36] D. A. McNamara, “Direct synthesis of optimum difference patterns for discrete linear arrays using Zolotarev distribution,” *IEE Proc. H Microw. Antennas Propag.*, vol. 140, no. 6, pp 445-450, Dec. 1993.

FIGURE CAPTIONS

- **Figure 1.** Sketch of the sub-arrayed monopulse array antenna.
- **Figure 2.** Evolution of the BEM solution within the DAG when $M = 10$ and $Q = 3$ (*a*) at the initialization ($k = 0$) and at iteration (*b*) $k = 1$ and (*c*) $k = 2$.
- **Figure 3.** Evolution of the ACO solution within the DAG.
- **Figure 4.** *ACO Calibration* ($N = 40$, $Q = 6$) - Behavior of the average convergence cost function value versus the pheromone update constant, H, and the pheromone evaporation parameter, ρ.
- **Figure 5.** *ACO Calibration* ($N = 40$, $Q = 6$; $H = 1$, $\rho = 0.05$) - Behaviors of the statistic values (mean value and standard deviation) of the average convergence cost function value versus the ant colony dimension, C.
- **Figure 6.** *ACO's Hill Climbing Behavior* ($N = 20$, $Q = 3$) - BEM power pattern at different iterations of the iterative optimization ($k = 0, ..., k_{end} = 2$).
- **Figure 7.** *ACO's Hill Climbing Behavior* - Cost function values of the solutions coded in the solution DAG. The values i_{opt} and i_{BEM} indicate the solution index in correspondence to the fitness of the best solution and the solution obtained through the BEM, respectively.
- **Figure 8.** *ACO's Hill Climbing Behavior* - Optimal and compromise difference power patterns obtained with the BEM and the ACO when (*a*) $N = 20$, $Q = 3$ (Zolotarev [36], $SLL = -30\,dB$) and (*b*) $N = 20$, $Q = 8$ (Zolotarev [36], $SLL = -40\,dB$).
- **Figure 9.** *ACO's Hill Climbing Behavior* ($N = 40$, $Q = 4$) - Behavior of the cost function value $\Psi^{(k)}$ during the iterative optimization process when applying the BEM and the ACO [best solution value, $\Psi_{opt}^{(k)} = min_{h=1,...,k}\left\{min_{i=1,...,I}\left[\Psi\left(\underline{A}_i^{(h)}\right)\right]\right\}$, and average cost function value, $\Psi_{av}^{(k)} = \frac{1}{I}\sum_{i=1}^{I}\Psi\left(\underline{A}_i^{(k)}\right)$].
- **Figure 10.** *Comparative Assessment* (Zolotarev [36], $SLL = -25\,dB$, $Q = 8$) - Behavior of the average convergence cost function value versus the number of array elements,

N, when applying the BEM and the ACO with a colony of $C^{(ess)}$ (ACO) and $C_1^{(ess)}$ (ACO^*) ants.

- **Figure 11.** *Comparative Assessment* (Zolotarev [36], $SLL = -25\,dB$, $Q = 8$) - Behaviors of (*a*) the SLL and (*b*) the BW values of the synthesized compromise patterns versus the number of array elements, N, when applying the BEM and the ACO with a colony of $C^{(ess)}$ (ACO) and $C_1^{(ess)}$ (ACO^*) ants.

TABLE CAPTIONS

- **Table I.** *ACO's Hill Climbing Behavior* ($N = 20$, $Q = 3$) - Sub-array configurations and weights determined by the BEM and the ACO.
- **Table II.** *ACO's Hill Climbing Behavior* - Pattern performances and computational indexes.
- **Table III.** *ACO's Hill Climbing Behavior* ($N = 20$, $Q = 8$) - Sub-array configurations and weights computed with the BEM and the ACO.
- **Table IV.** *ACO's Hill Climbing Behavior* ($N = 40$, $Q = 4$) - Sub-array configurations and weights synthesized by means of the BEM and the ACO.

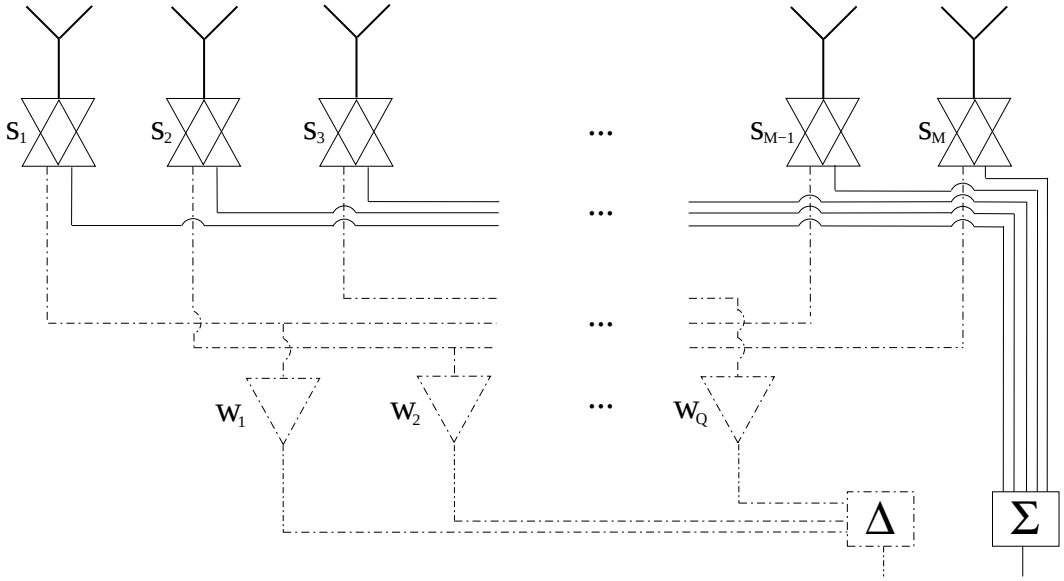


Fig. 1 - P. Rocca *et al.*, “An Improved Excitation Matching Method ...”

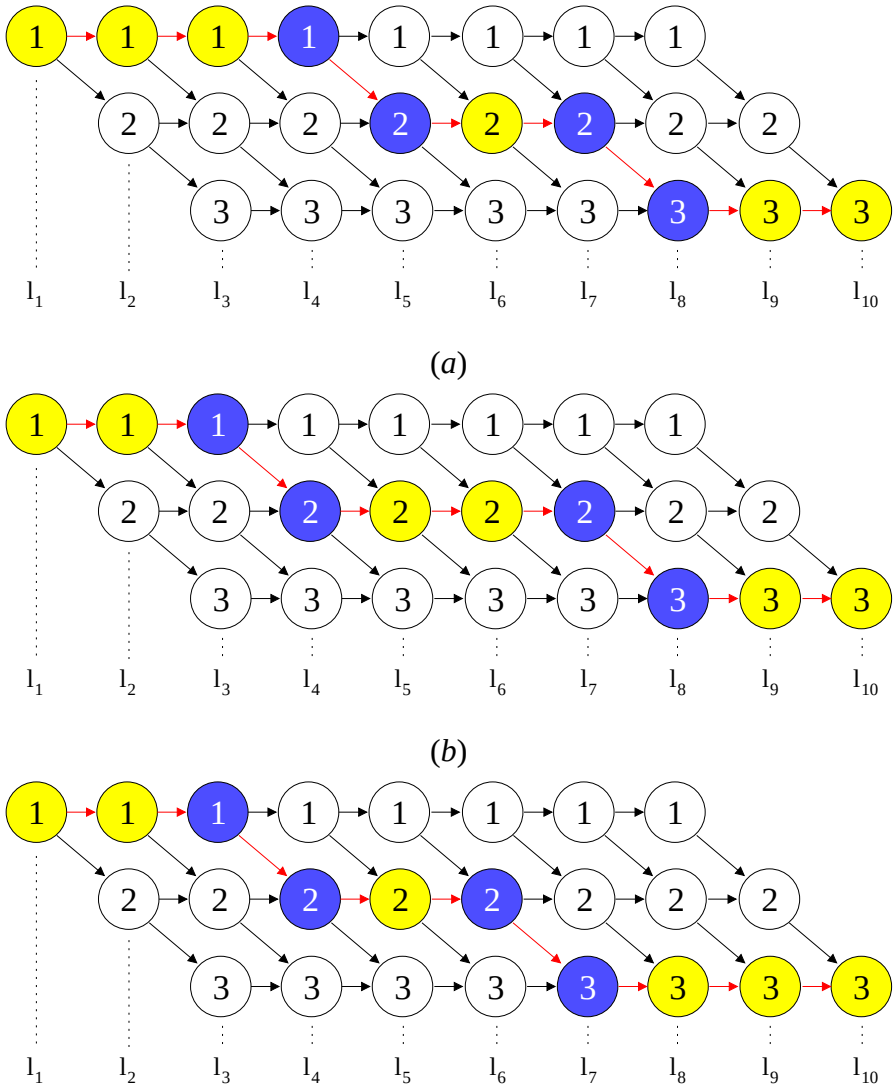


Fig. 2 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

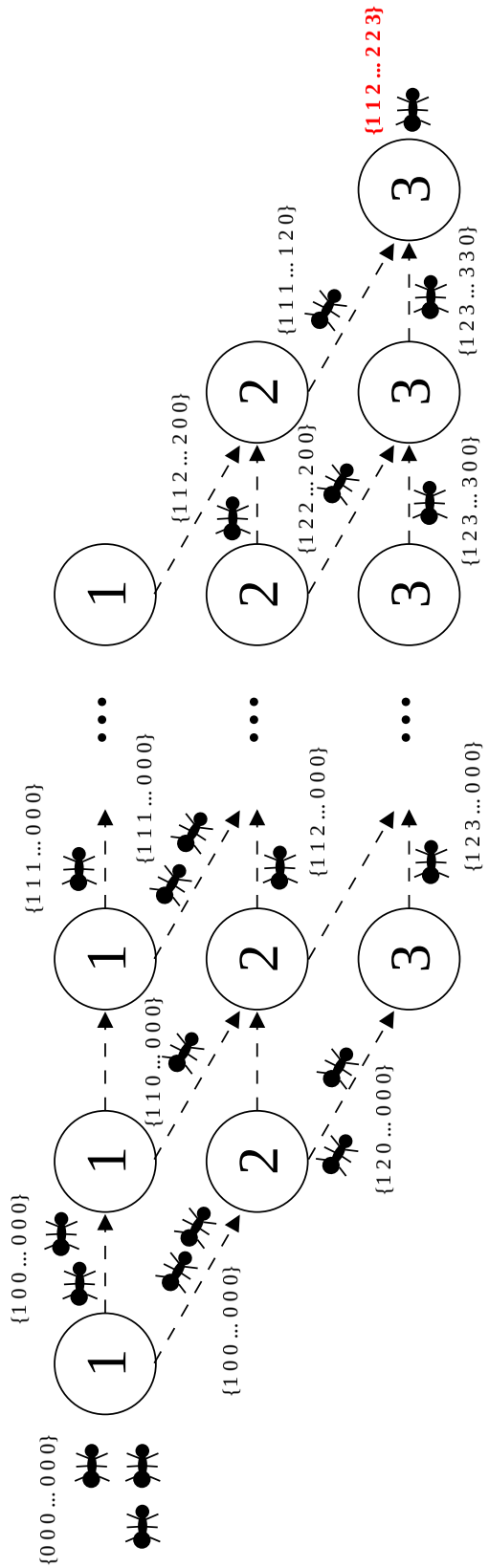


Fig. 3 - P. Rocca *et al.*, “An Improved Excitation Matching Method ...”

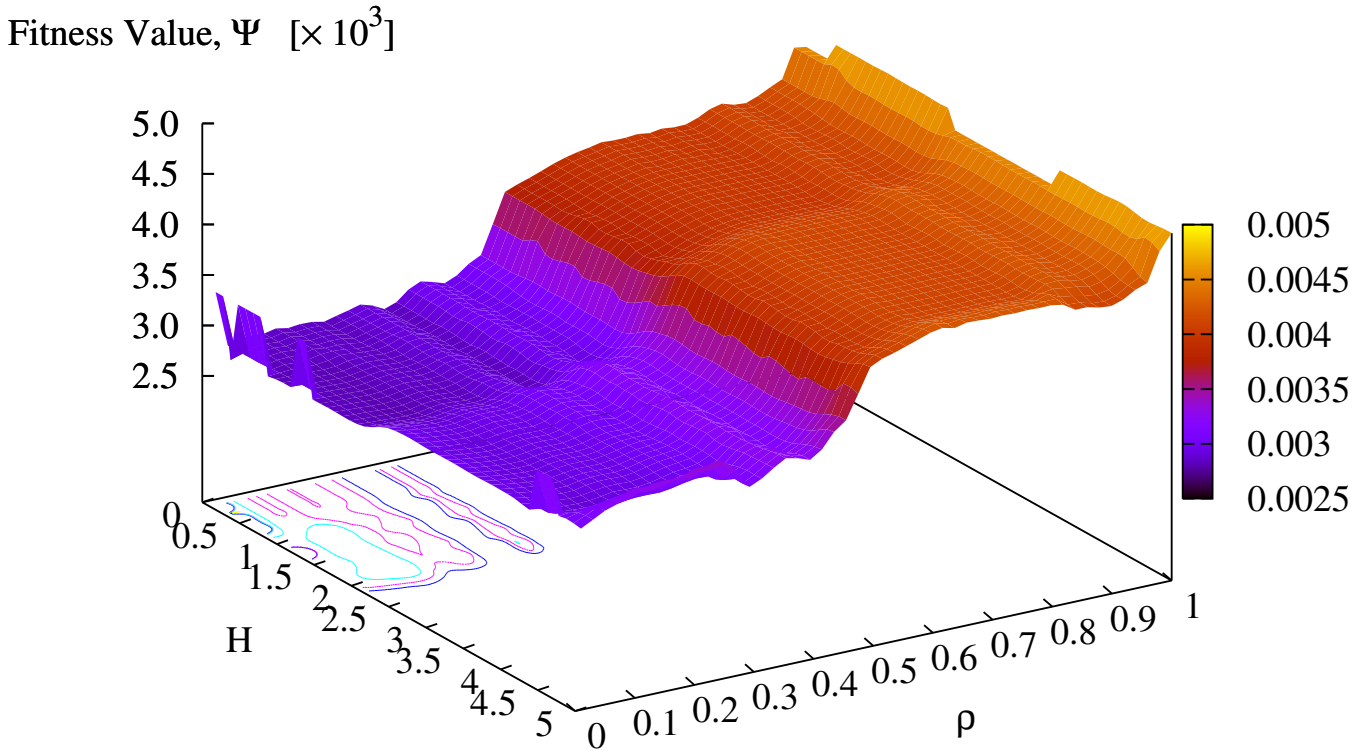


Fig. 4 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

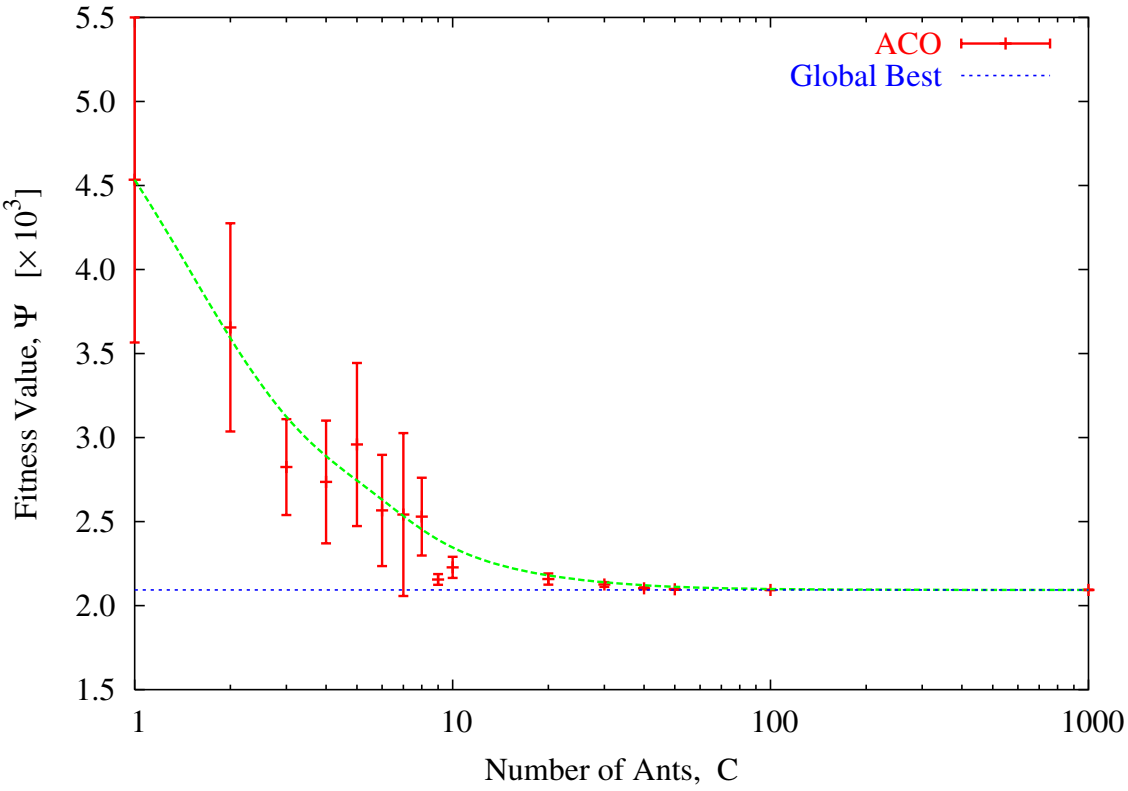


Fig. 5 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

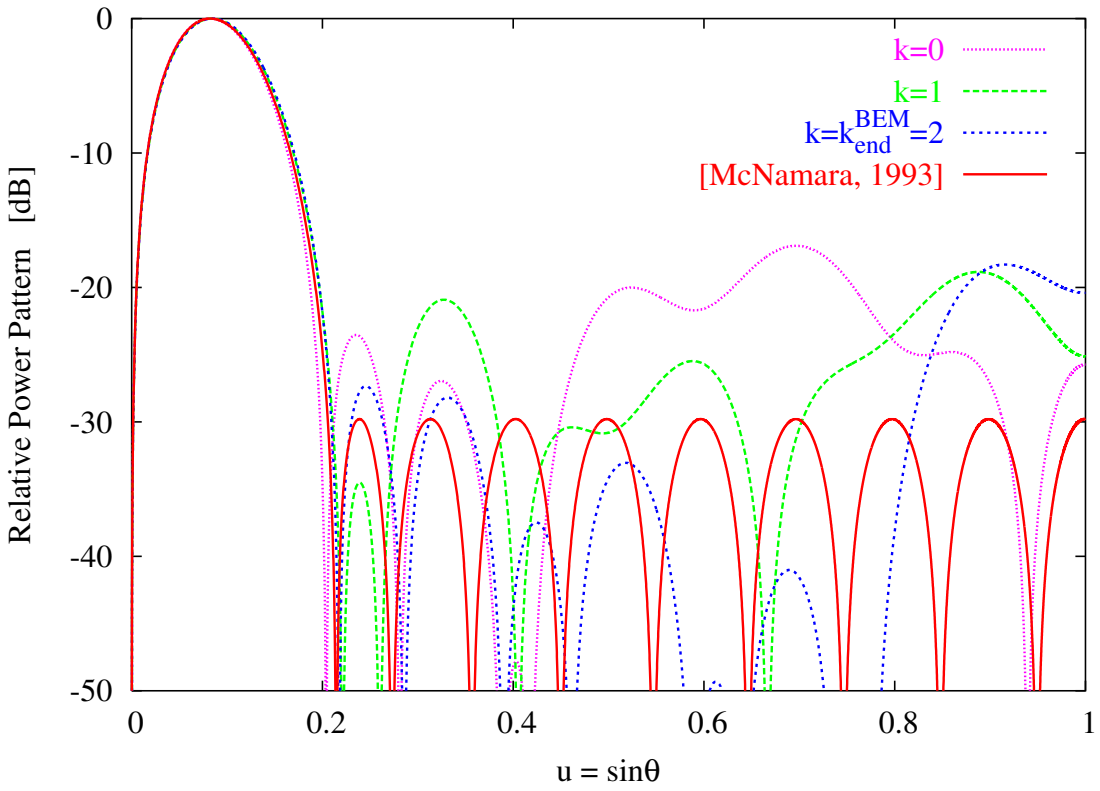


Fig. 6 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

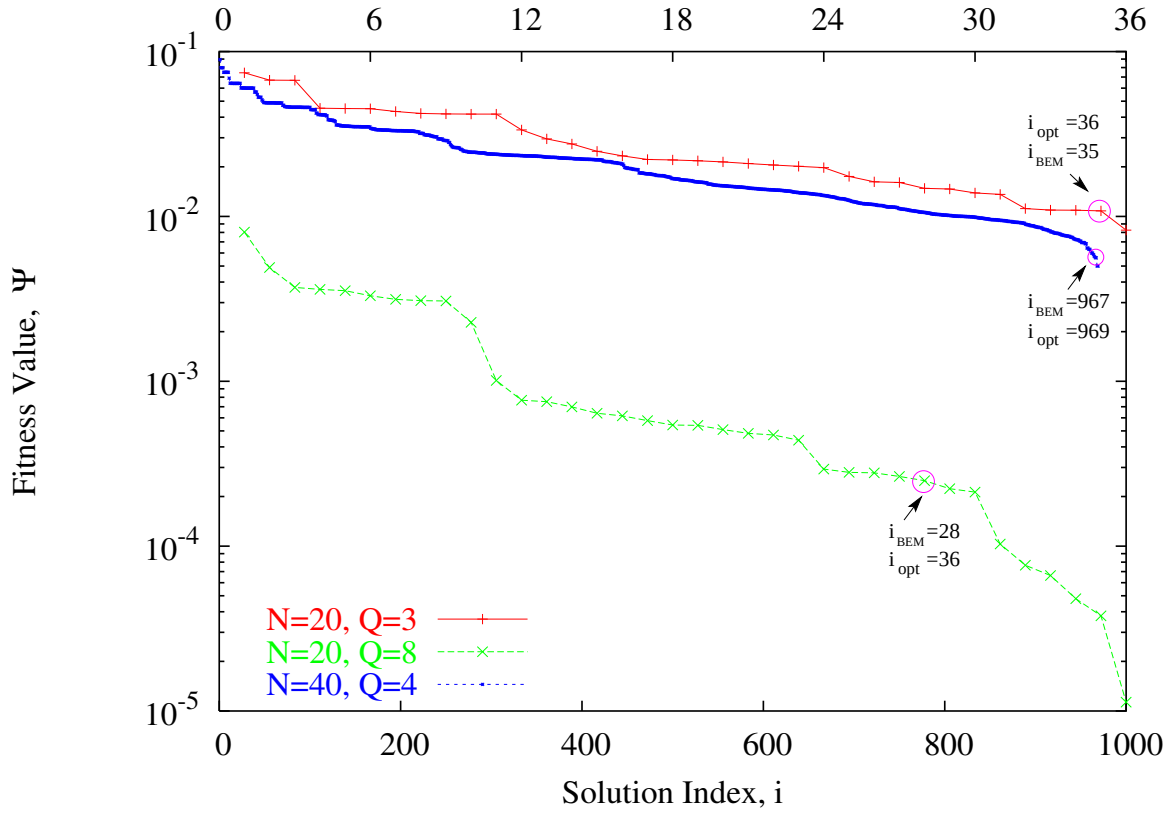
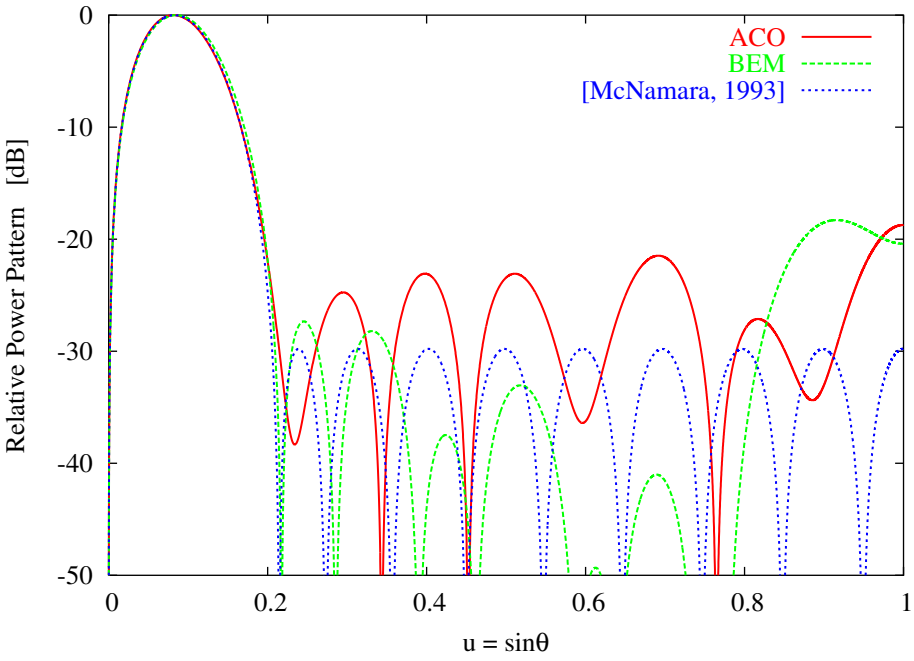
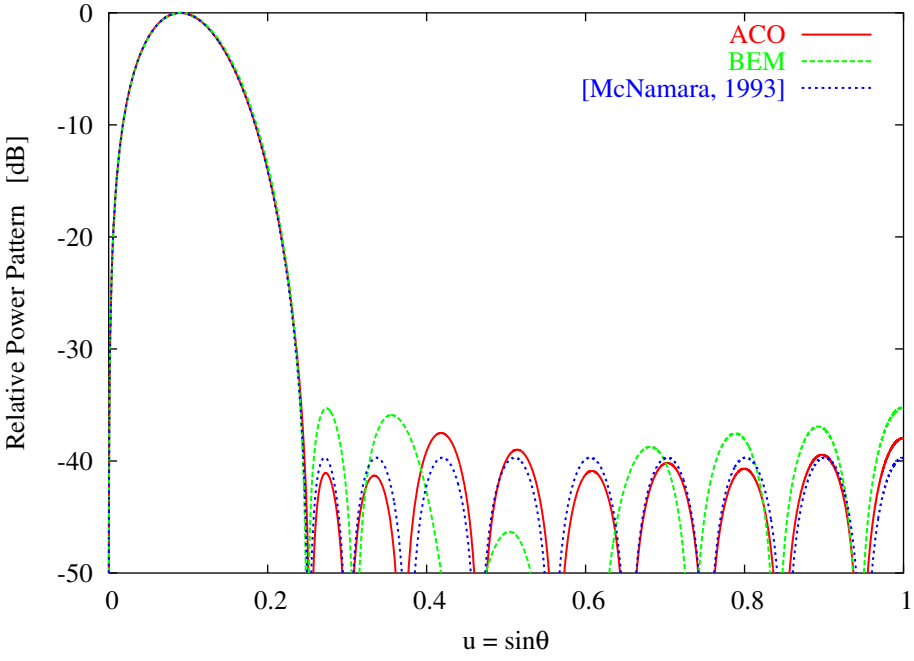


Fig. 7 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."



(*a*)



(*b*)

Fig. 8 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

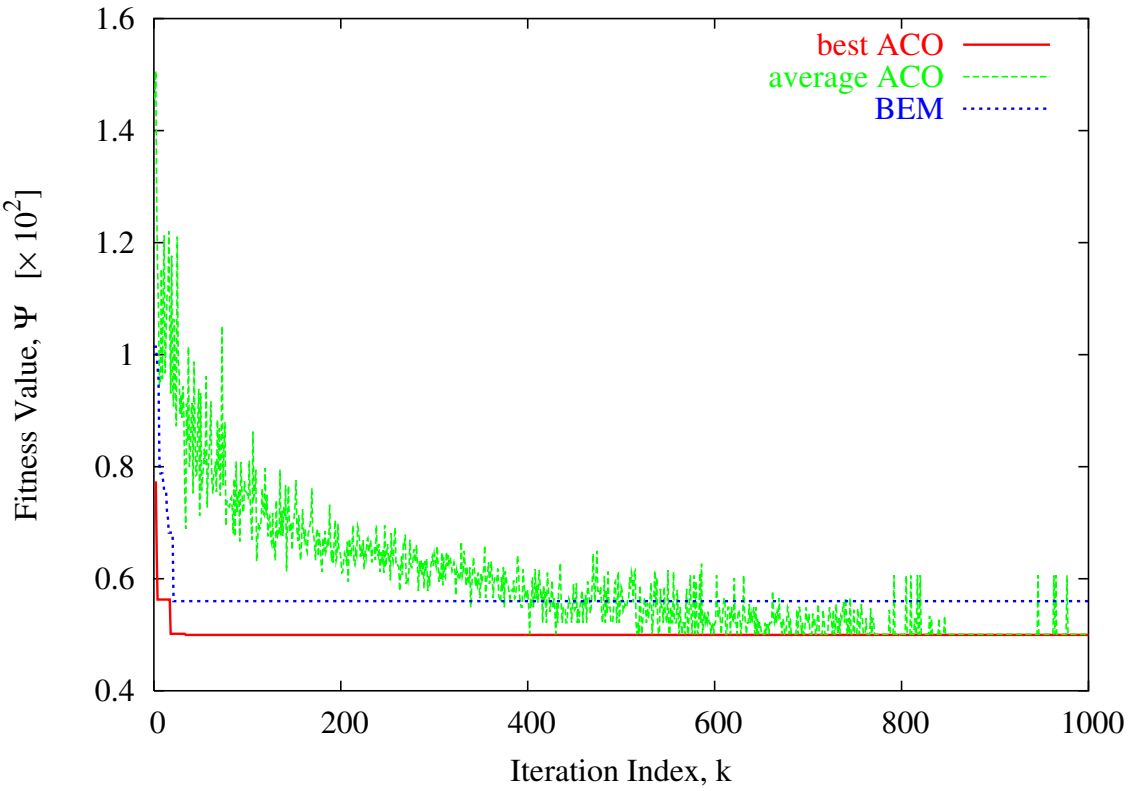


Fig. 9 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

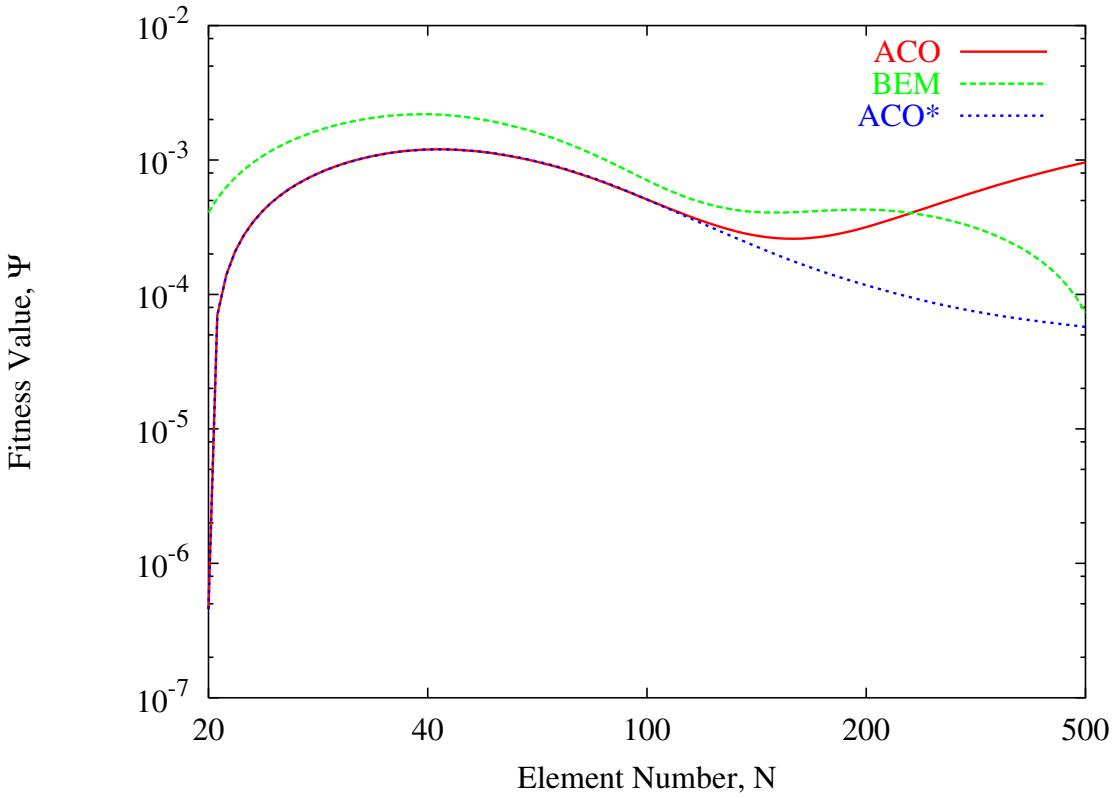


Fig. 10 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

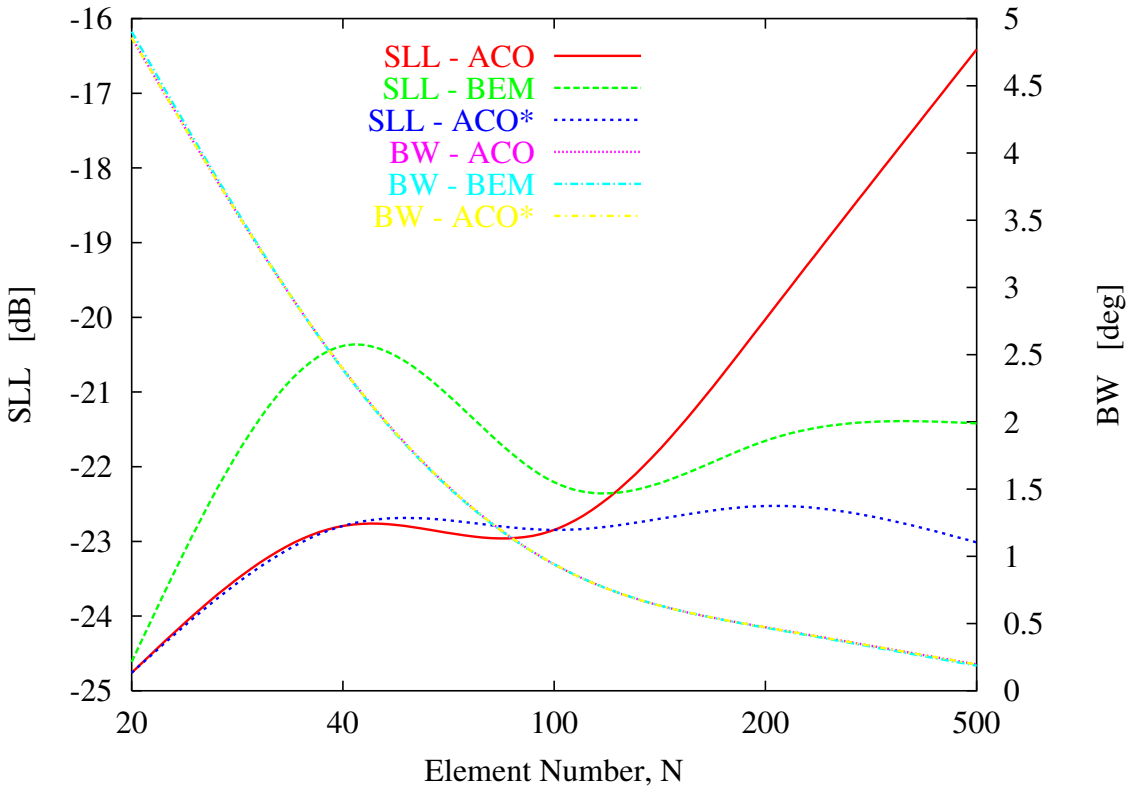


Fig. 11 - P. Rocca *et al.*, "An Improved Excitation Matching Method ..."

$M = 10$	a_m^{BEM}	$\{1\,1\,2\,2\,3\,3\,3\,3\,2\,1\}$		
	a_m^{ACO}	$\{1\,2\,2\,3\,3\,3\,3\,3\,3\,2\}$		
$Q = 3$	$w_{a_m}^{BEM}$	0.3827	0.9736	1.3363
	$w_{a_m}^{ACO}$	0.1798	0.6602	1.2549

Tab. I - P. Rocca *et al.,* "An Improved Excitation Matching Method ..."

Approach	Ψ_{opt}	Δ	$SLL\,[dB]$	$BW\,[deg]$	k_{end}	F_{end}	$t\,[sec]$	$T^{(ess)}$
$N = 2M = 20\,,\, Q = 3$								
BEM	1.08×10^{-2}	0.3199	-18.25	5.28	2	3	$< 10^{-8}$	36
ACO	8.26×10^{-3}	0.2689	-18.75	5.12	2	10	$< 10^{-8}$	36
$N = 2M = 20\,,\, Q = 8$								
BEM	2.49×10^{-4}	0.0545	-35.20	5.74	2	3	$< 10^{-8}$	36
ACO	1.13×10^{-5}	0.0145	-37.50	5.68	2	10	$< 10^{-8}$	36
$N = 2M = 40\,,\, Q = 4$								
BEM	5.60×10^{-3}	0.2886	-20.10	2.50	21	22	$< 10^{-7}$	969
ACO	4.99×10^{-3}	0.2609	-22.85	2.50	34	340	4.5×10^{-3}	969

$M = 10$	a_m^{BEM}	$\{1\,2\,3\,5\,7\,8\,6\,4\,2\,1\}$							
	a_m^{ACO}	$\{1\,3\,5\,7\,8\,8\,7\,6\,4\,2\}$							
$Q = 8$	$w_{a_m}^{BEM}$	0.2146	0.6107	0.9221	0.9825	1.1582	1.1797	1.2818	1.2864
	$w_{a_m}^{ACO}$	0.2049	0.2432	0.5937	0.7250	0.9221	0.9825	1.1650	1.2838

$M = 20$	a_m^{BEM}	$\{1\,1\,2\,2\,2\,3\,3\,3\,4\,4\,4\,4\,4\,4\,4\,4\,4\,3\,3\,2\}$			
	a_m^{ACO}	$\{1\,1\,2\,2\,3\,3\,3\,4\,4\,4\,4\,4\,4\,4\,4\,4\,4\,3\,2\}$			
$Q = 4$	$w_{a_m}^{BEM}$	0.1779	0.5658	1.0257	1.3288
	$w_{a_m}^{ACO}$	0.1779	0.5055	0.8989	1.2923

Tab. IV - P. Rocca *et al.,* "An Improved Excitation Matching Method ..."