UNIVERSITY
OF TRENTO

DIPARTIMENTO DI INGEGNERIA E SCIENZA DELL'INFORMAZIONE

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

A FAST GRAPH-SEARCHING ALGORITHM ENABLING THE EFFICIENT SYNTHESIS OF SUB-ARRAYED PLANAR MONOPULSE ANTENNAS

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

January 2011

Technical Report # DISI-11-033

A Fast Graph-Searching Algorithm Enabling the Efficient Synthesis of Sub-Arrayed Planar Monopulse Antennas

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

ELEDIA Research Group

Department of Information and Communication Technology,

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

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

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

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

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

A Fast Graph-Searching Algorithm Enabling the Efficient Synthesis of Sub-Arrayed Planar Monopulse Antennas

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

Abstract

In this paper, an innovative approach in its different implementations for the synthesis of compromise sum and difference patterns of monopulse planar arrays is presented. The synthesis method is based on a sub-arraying technique aimed at generating the compromise patterns through an optimal excitation matching procedure. By exploiting some properties of the solution space, the synthesis problem is reformulated as a combinatorial one to allow a considerable saving of computational resources. Thanks to a graph-based representation of the solution space, the use of an efficient path-searching algorithm is enabled to speed-up the convergence to the compromise solution. In the numerical validation, a set of representative examples concerned with both pattern matching problems and pattern-feature optimization are reported in order to assess the effectiveness and flexibility of the proposed approach. Comparisons with previously published results and solutions obtained by a hybrid version of the approach customized to deal with the optimization of the sidelobe level (SLL) are reported and discussed, as well.

Key words: Planar Arrays, Monopulse Antennas, Sum and Difference Modes, Direct Acyclic Graph.

1 Introduction

A monopulse tracker [1] is a device aimed at detecting the position of a target by using the information collected from an antenna that generates *sum* and *difference* patterns. These beams can be synthesized by means of a reflector antenna with two (tracking on a plane) or three (3D tracking) feeds, or by using linear or planar array antennas, respectively. The latter solution is usually preferred since array antennas are easy to built and they do not require mechanical positioning systems to steer the beam pattern. Moreover, array structures can also be easily installed on mobile vehicles (e.g., aircrafts). Unlike linear structures, a planar array allows the generation of a sum and two spatially-orthogonal difference patterns [2] [i.e., the *azimuth difference mode* ($H-mode$) and the *elevation difference mode* ($E-mode$)] useful to give a complete description of the trajectory of a target in terms of range, azimuth, and elevation.

In order to synthesize independent optimal sum and difference patterns, Taylor [3] and Bayliss [4] developed analytical techniques to compute the corresponding excitation coefficients by sampling suitable continuous distributions. However, these optimal solutions require three independent feeding networks. Hence, high manufacturing costs usually arise and electromagnetic interferences unavoidably take place because of the large number of elements in planar monopulse arrays.

In order to overcome these drawbacks, the sub-arraying technique [5] is a suitable compromise solution aimed at optimizing pre-specified sub-array layouts. To deal with such an optimal compromise problem, global optimization approaches [6][7][8] as well as hybrid techniques have been considered [9][10]. However, since in optimization-based techniques the dimension of the solution space grows exponentially with the number of array elements, few examples concerned with planar arrays have been dealt with. To the best of the authors' knowledge, the compromise synthesis of planar arrays has been recently faced only in [9], where the sub-array aggregation has been *a-priori* fixed and a Simulated Annealing (SA) optimizer has been used to determine only the sub-array gains.

In [11], an innovative method for the optimal compromise among sum and difference patterns of linear arrays has been proposed. The optimization problem has been recast as a combinatorial one to significantly reduce the dimension of the solution space and to allow a fast synthesis process. The sub-optimal difference pattern has been computed by means of an iterative searching

algorithm looking for the best solution that belongs to a complete set coded in a non-complete "linear" binary tree. Thanks to its computational efficiency [12], such a technique appears to be a good candidate to deal also with two-dimensional ($2D$) arrays avoiding the computational drawbacks of stochastic optimization methodologies. However, the extension of the range of applicability of [11] to planar monopulse arrays is not a trivial task. As a matter of fact, some fundamental issues have to be carefully addressed. Let us consider that the computational efficiency of the approach in [11][12] comes from the careful customization to linear arrays of the synthesis strategy. Moreover, unlike linear structures, the three-dimensional ($3D$) tracking of planar arrays needs of two difference patterns (i.e., the difference $E-mode$ and the $H-mode$), instead of a single one. Furthermore, although the approach for linear geometries allows a significant reduction of the dimension of the solution space, the memory requirements when dealing with planar arrays are not negligible due to the large number of radiating elements. Consequently, the use of an innovative direct acyclic graph algorithm able to reduce the space of the admissible solutions and increase the efficiency is considered in order to profitably cope with $2D$ synthesis problems.

The key-points of the proposed approach preliminary presented in its simpler version in [14][15] are summarized in the following. By exploiting the properties of the solution of the planar compromise problem, the "solution tree" of the linear case has been collapsed into a more compact structure, namely the *direct acyclic graph* (DAG) [13], to describe the whole solution space. Such a representation enables the excitation matching synthesis of planar arrays with large numbers of elements [16] thanks to the significant reduction of both the computational time and the CPU memory requirements. Moreover, the DAG allows the implementation and an effective use of a fast graph-searching algorithm to look for the optimal planar compromise.

The paper is organized as follows. In Sect. 2, the problem is mathematically formulated by summarizing the synthesis procedure (Sub-Sect. 2.1) as well as the graph-based searching algorithm (Sub-Sect. 2.2) aimed at exploiting the DAG architecture to efficiently sample the solution space. Selected results from a wide set of numerical experiments are reported in Sect. 3 to carefully illustrate the behavior of the proposed method and its different implementations as well as to assess its effectiveness. For completeness, a comparative study with previously published results from state-of-the-art techniques is proposed (Sub-Sect. 3.2), as well. Further-

more, a customized hybrid version of the approach is implemented and tested (Sub-Sect. 3.3) to effectively deal with the SLL optimization problem in planar array. Finally, some conclusions are drawn (Sect. 4).

2 Mathematical Formulation

Let us consider a planar array lying on the $xy-plane$ whose elements are located on a rectangular grid with inter-element spacing $d_x = d_y = d$. The coordinates of each array element are given by $x_m = \left[m - \frac{sgn(m)}{2}\right] \times d$, $m = \pm 1, ..., \pm N_x$ and $y_n = \left[n - \frac{sgn(n)}{2}\right] \times d$, $n = \pm 1, ..., \pm N_y^m$, and the array factor turns out to be [17]:

$$AF(\theta, \phi) = \sum_{m=-N_x}^{N_x} \sum_{n=-N_y^m}^{N_y^m} I_{mn} e^{j(k_x x_m + k_y y_n)} \tag{1}$$

where I_{mn} is an excitation coefficient and N_y^m is an integer function of the row index m depending on to the array boundary. Moreover, $k_x = \frac{2\pi}{\lambda} \sin\theta \cos\phi$ and $k_y = \frac{2\pi}{\lambda} \sin\theta \sin\phi$.
Dealing with monopulse systems, the optimal/reference sum mode is generated by setting the excitations I_{mn} to a set of real excitations $\mathbf{A} = \left\{\alpha_{mn} = \alpha_{(-m)n} = \alpha_{m(-n)} = \alpha_{(-m)(-n)};\ m = 1, \ldots, N_x;\ n = 1, \ldots, N_y^m\right\}$ characterized by a central and quadrantal symmetry. As regards to the optimal/reference difference patterns [i.e., the $E-mode$ and the $H-mode$ difference patterns], they are still determined by real excitations $\mathbf{B^{\Theta}} = \Big\{\beta_{mn}^{\Theta} = \beta_{(-m)n}^{\Theta} = -\beta_{m(-n)}^{\Theta} = -\beta_{(-m)(-n)}^{\Theta};$ $m = 1, \ldots, N_x;\ n = 1, \ldots, N_y^m\Big\}$, $\Theta = E, H$, but with quadrantal anti-symmetric distributions. Usually, the resulting patterns are characterized by narrow beamwidths and low side lobe levels (SLLs). Furthermore, these coefficient distributions assure pattern features attractive for tracking purposes (e.g., high directivity and maximum normalized slope along the boresight direction, i.e., a high angular sensitivity). Unfortunately, the use of three independent feeding networks is generally impracticable and it is mandatory to find a suitable trade-off between the optimality of the synthesized patterns and the device feasibility. Towards this purpose, the sub-arraying strategy has been introduced by McNamara [5]. The original problem has been reformulated into a compromise one: "*for a given optimal sum (difference) mode, to define the sub-array configuration and the corresponding sub-array gains, such that, the synthesized dif-*

ference (sum) modes are as close as possible to the optimal/reference ones". Accordingly, the grouping operation yields to a sub-array configuration of the difference $\Theta - mode$ described by aggregation vector $\mathbf{c}^{\Theta}$

$$\mathbf{c}^{\mathbf{\Theta}} = \left\{ c_{mn}^{\Theta};\, m = 1, ..., N_x;\, n = 1, ..., N_y^m \right\} \tag{2}$$

where $c_{mn}^{\Theta} \in [1, Q]$ is the sub-array index of the element located at the m-th row and n-th column within the array architecture. Then, the compromise (*sub-arrayed*) difference excitations are given by

$$\mathbf{B}_{\mathbf{sa}}^{\mathbf{\Theta}} = \left\{ b_{mn}^{\Theta} = \alpha_{mn} \delta \left(c_{mn}^{\Theta},\, q \right) w_q^{\Theta};\, m = 1, ..., N_x;\, n = 1, ..., N_y^m;\, q = 1, ..., Q \right\} \tag{3}$$

where w_q^{Θ} is the weight associated to the q-th sub-array and $\delta \left(c_{mn}^{\Theta},\, q \right) = 1$ if $c_{mn}^{\Theta} = q$ and $\delta \left(c_{mn}^{\Theta},\, q \right) = 0$, otherwise. Hence, the synthesis problem turns out to be equivalent to the definition of the configuration $\mathbf{c}_{opt}^{\mathbf{\Theta}}$ and the corresponding set of weights $\mathbf{w}_{opt}^{\mathbf{\Theta}}$ such that the compromise sub-arrayed difference patterns, generated by $\mathbf{B}_{\mathbf{sa}}^{\mathbf{\Theta}}$, are as close as possible to the optimal/reference ones.

2.1 Synthesis Procedure

In order to find the solution that better approximates the optimal difference patterns, McNamara in [5] introduced the following metric/residual

$$R \left(\mathbf{c}^{\mathbf{\Theta}},\, \mathbf{w}^{\mathbf{\Theta}} \right) = \sum_{m=1}^{N_x} \sum_{n=1}^{N_y^m} \left| \beta_{mn}^{\Theta} - b_{mn}^{\Theta} \left(\mathbf{c}^{\mathbf{\Theta}},\, \mathbf{w}^{\mathbf{\Theta}} \right) \right|^2 \tag{4}$$

to be minimized with respect to the clustering $\mathbf{c}^{\mathbf{\Theta}}$ and related sub-array weights $\mathbf{w}^{\mathbf{\Theta}}$ in order to solve the *compromise problem*. Equation (4) quantifies the "distance" between the optimal independent excitations, β_{mn}^{Θ}, and the actual ones, b_{mn}^{Θ}, which are function of $\mathbf{c}^{\mathbf{\Theta}}$ and $\mathbf{w}^{\mathbf{\Theta}}$. Starting from such a formulation and similarly to [11], the following cost function is defined

$$\Psi \left(\mathbf{c}^{\mathbf{\Theta}} \right) = \frac{1}{J} \sum_{m=1}^{N_x} \sum_{n=1}^{N_y^m} \alpha_{mn}^2 \left| \left[\gamma_{mn}^{\Theta} - \sum_{q=1}^{Q} g_{mnq} \left(\mathbf{c}^{\mathbf{\Theta}} \right) \right] \right|^2 \tag{5}$$

where J is the number of elements lying on the aperture, $J = \sum_{j=1}^{N_x} N_y^j$, and γ_{mn}^{Θ}

$$\gamma_{mn}^{\Theta} = \frac{\beta_{mn}^{\Theta}}{\alpha_{mn}} \quad (m = 1, ..., N_x;\ n = 1, ..., N_y^m) \tag{6}$$

is the so-called *reference gain* [i.e., the gain to be assigned to the (m, n)-th element of the sum array in order to afford the optimal difference pattern]. Moreover, the *estimated gains*, $g_{mnq}^{\Theta} = g_{mnq}\left(\mathbf{c^{\Theta}}\right)$, are obtained by simply exploiting the grouping theory described by Fisher in [18]. More in detail, the value of g_{mnq}^{Θ} is the weighted (with weights α_{mn}^2) arithmetic mean of the reference gains of those elements belonging to the same q-th sub-array:

$$g_{mnq}^{\Theta} = \frac{\sum_{m=1}^{N_x} \sum_{n=1}^{N_y^m} \alpha_{mn}^2 \delta\left(c_{mn}^{\Theta},\, q\right) \gamma_{mn}^{\Theta}}{\sum_{m=1}^{N_x} \sum_{n=1}^{N_y^m} \alpha_{mn}^2 \delta\left(c_{mn}^{\Theta},\, q\right)}, \quad m = 1, ..., N_x;\ \ n = 1, \ldots, N_y^m;\ \ q = 1, ..., Q, \tag{7}$$

and the sub-array weights are given by

$$w_q^{\Theta} = \delta\left(c_{mn}^{\Theta},\, q\right) g_{mnq}^{\Theta} \quad m = 1, ..., N_x;\ \ n = 1, \ldots, N_y^m;\ \ q = 1, ..., Q. \tag{8}$$

Now, the compromise problem can be reformulated in the following *dual form*: "*for a given optimal/reference sum mode, to define the sub-array configuration (I) that minimize* (5) (i.e., $\mathbf{c}_{opt}^{\mathbf{\Theta}} = arg\left\{min_{c^{\Theta}}\left[\Psi\left(\mathbf{c^{\Theta}}\right)\right]\right\}$) *or, in an equivalent way, (II) where each cluster is composed by array elements whose reference gains have minimum variance*". Fisher in [18] proved that a *contiguous partition*[(1)] of the array elements is the optimal compromise solution of such a dual problem. To determine the set of admissible solutions, the array elements are sorted in an ordered list $\mathbf{L}$ according to their *reference gains* $\gamma_{mn}^{\Theta} \in \mathbb{R}$. The list $\mathbf{L} = \{l_j;\ j = 1, ..., J\}$, being j the list index and where $l_i \leq l_{i+1}$ $(i = 1, ..., J-1)$, $l_1 = min_{mn}\left\{\gamma_{mn}^{\Theta}\right\}$, $l_J = max_{mn}\left\{\gamma_{mn}^{\Theta}\right\}$, is then iteratively partitioned in Q parts to define each time a *contiguous partition* (i.e., a sub-array configuration characterized by Q convex sets[(2)] of $\mathbf{L}$). The number of admissible sub-array con-

[(1)] A grouping of array elements is a contiguous partition if the generic (m_2, n_2)-th array element belongs to the q-th sub-array only when two elements, namely the (m_1, n_1)-th element and the (m_3, n_3)-th one, belong to the same sub-array and the condition $\gamma_{m_1 n_1}^{\Theta} < \gamma_{m_2 n_2}^{\Theta} < \gamma_{m_3 n_3}^{\Theta}$ holds true.

[(2)] A set S in a vector space over $\mathbb{R}$ is called a *convex set* if the line segment joining any pair of points of S lies entirely in S.

figurations (i.e., the number of *contiguous partitions*) candidate to minimize the cost function (5) is equal to $U^{(ess)} = \begin{pmatrix} J-1 \\ Q-1 \end{pmatrix}$ instead of $U = Q^J$ as for stochastic optimization techniques (e.g., [6][7][10]) with the strong reduction highlighted by Fig. 1(*a*). These admissible aggregations define the solution space $\Re^{(ess)}$ to which the optimal sub-array configuration $\mathbf{c}_{opt}^{\mathbf{\Theta}}$ belongs.

2.2 Iterative Graph-Searching Algorithm

Likewise to [11], $\Re^{(ess)}$ could be formally represented by means of a non-complete binary tree of depth J[(3)], wherein each path identifies an admissible sub-array configuration (i.e., a *contiguous partition* of the array elements). However, although the introduction of the solution tree for the planar case, as well, could allow a simple representation of the solutions space, non-negligible computational problems would still remain since a large amount of memory $M^{(IBT)}$ [see Fig. 1(*b*) where an indication of the storage resources is given] would be necessary to store and to explore such an architecture. Therefore, a more compact data structure, indicated as *direct acyclic graph* (DAG) [13] and shown in Fig. 2(*a*), is built starting from the observation that some "sub-trees" are recursively shared in the non-complete planar solution tree [Fig. 2(*b*)] when defined analogously to the linear case. Generally speaking, the DAG is an oriented graph without cycles. In this case, the DAG of Fig. 2(*a*) is a rooted DAG since it has a node (the *root*) with no arcs pointing into it. Moreover, it is a binary DAG whose nodes have a maximum of two arcs leaving them [13]. With reference to Fig. 2(*a*), the main characteristics of the *direct acyclic graph* as well as its advantages over the non-complete binary tree [11] are[(4)]:

- Unlike the binary tree, the DAG is a non-redundant and more compact structure made up of Q rows and J columns, where the q-th row corresponds to the q-th sub-array ($q = 1, ..., Q$) and the j-th column ($j = 1, ..., J$) refers to the l_{j-th} element of the sorted list $\mathbf{L}$;
- The total number of nodes (called *vertexes*) required for the storage of the whole DAG

[(3)] In graph theory, a tree is a graph defined as a nonempty finite set of vertexes or nodes. Two nodes are connected by exactly one path. In our case, the tree is a binary tree since it is either empty or each of its nodes has not more than two sub-trees. Furthermore, some nodes in a non-complete binary tree has either no children, or only one left/right child [13].

[(4)] The interested reader is referred to [15] for a more in depth description of the graph structure.

is equal to $V^{(DAG)} = Q \times (J - Q + 1)$[5] and $V^{(DAG)} \ll V^{(IBT)}$ [Fig. 1(*b*)], $V^{(IBT)}$ and $V^{(CBT)}$ being the number of nodes that would be necessary for the storage of the tree when using the non-complete and the complete binary tree, respectively;

- In the DAG, a trial solution/path is denoted by $\psi(V, E)$ [e.g., the red line in Fig. 2(*a*)] since it might be described by a set of $V = J$ vertexes and through $E = J - 1$ arcs among the vertexes of the path. In particular, the generic v-th vertex ($v = 1, ..., V$) is represented by a black circle in Fig. 2(*a*). It identifies the sub-array membership of the array element whose *reference gain* γ_{mn} is given by l_j. For example, the first vertex on the left of Fig. 2(*a*) indicates that the array element, whose reference gain is equal to l_1, belongs to the $q = 1$-st sub-array. The e-th ($e = 1, ..., E$) edge describes a link between two vertexes of a path within the graph. It is denoted by a black arrow in Fig. 2(*a*).

By virtue of the above considerations, (*I*) the non-negligible memory saving [$M^{(DAG)}$ vs. $M^{(IBT)}$ - Fig. 1(*b*)], (*II*) the easier and more compact [$V^{(DAG)}$ vs. $V^{(IBT)}$ - Fig. 1(*b*)] representation of the solution space are worth of notice. Furthermore, the DAG still guarantees that (*III*) the elements grouped in the same sub-array have close γ_{mn}^{Θ} values and (*IV*) the solution of the compromise problem can be recast as the search of a path inside the DAG.

To this purpose, let us observe that only some elements of the list $\mathbf{L}$, called "*border elements*" and identified by the l_j indexes whose adjacent list values l_{j-1} or/and l_{j+1} belong to a different sub-array, are candidate to change their sub-array membership without violating the sorting condition of the admissible aggregations. Therefore, the tree-searching procedure of the linear case is suitably modified [14] and extended to look for the optimal sub-array configuration $\mathbf{c}_{opt}^{\mathbf{\Theta}}$ that minimizes $\Psi\left(\mathbf{c}^{\mathbf{\Theta}}\right)$ (5) among the solutions available into the DAG. The procedure, which follows the guidelines of the pseudo-code in Fig. 3, starts with the definition of an initial path $\psi_0 = \psi(V_0, E_0)$ randomly-chosen among the paths of the DAG and setting the initial aggregation as follows:

$$\mathbf{c}_0^{\mathbf{\Theta}} = \left\{ c_1^{\Theta} = 1;\, c_j^{\Theta} = ran\,[1,\, Q] \,:\, c_{j-1}^{\Theta} \leq c_j^{\Theta} \leq c_{j+1}^{\Theta},\, j = 2, ..., J-1;\, c_J^{\Theta} = Q \right\} . \quad (9)$$

[5] Since each row of the DAG has $V = J - Q + 1$ vertexes, which is the maximum number of elements that can be grouped into a sub-array, and the DAG is composed by Q rows.

Successively, a sequence of trial solutions, $\left\{\mathbf{c}_t^{\mathbf{\Theta}};\, t = 1, ..., T_{max}\right\}$, is generated by iteratively updating the trial path ψ_t (i.e., $\psi_t \leftarrow \psi_{t+1}$, $t = 1, ..., T_{max} - 1$). The new solutions are obtained by changing the memberships of the border elements of the DAG as detailed in [15]. The searching procedure is stopped when a condition based on either a maximum number of iterations T_{max} ($t > T_{max}$) or the stationariness of the cost function holds true, T_{window} and τ (Fig. 3) being a fixed number of iterations and a fixed numerical threshold, respectively. The solution obtained at the end of the iterative searching procedure (i.e., the path ψ_{opt}) is assumed as the optimal path that unequivocally identifies the best sub-array configurations $\mathbf{c}_{opt}^{\mathbf{\Theta}}$ and corresponding weights $\mathbf{w}_{opt}^{\mathbf{\Theta}}$.

3 Numerical Simulations and Results

In order to assess the effectiveness of the proposed method, a set of numerical experiments has been performed and some representative results will be shown in the following sections. The first section will be devoted to illustrate in a detailed fashion the behavior of the proposed method [indicated in the following as *Border Element Method* (BEM)] as well as its reliability to synthesize a difference pattern as close as possible to the reference one. The others sections will be aimed at comparing the BEM with state-of-the-art techniques in dealing with benchmark test cases in order to complete the preliminary validation presented in [14][15][19] and further confirm, in a more exhaustive and complete fashion, the underlying proof-of-concept. Moreover, the third section will be devoted to present a hybrid version of the approach for the direct sidelobe control of planar arrays and a set of representative results will be shown, as well.

3.1 Sub-Arrayed Planar Array Synthesis

To describe the behavior of the BEM in dealing with planar sub-arrayed monopulse antennas with large numbers of elements, let us consider a test case concerned with a planar geometry of $N = 4 \times J = 1264$ elements $\frac{\lambda}{2}$-spaced and distributed on a circular aperture of radius equal to $r = 10\lambda$. It is worth to point out that in these situations, the use of stochastic optimization-based techniques (e.g., [6][7][10]) involves a high computational burden that limit/prevent their application. The sum pattern excitations $\mathbf{A}$ have been fixed to afford a Taylor pattern [3][20]

with $SLL = -35\,dB$ and $\overline{n} = 6$ [Fig. 4(*a*)]. Moreover, the reference $H-mode$ excitations $\mathbf{B^H}$ have been chosen equal to those of a Bayliss difference pattern [4][20] with $SLL = -35\,dB$ and $\overline{n} = 5$ [Fig. 4(*b*)]. Because of the symmetry of the array geometry, the optimal $E-mode$ coefficients are related to the $H-mode$ ones as follows

$$\mathbf{B^E} = \left\{\beta^E_{mn} = -\beta^H_{mn};\; m = 1, ..., N_x;\; n = 1, ..., N^m_y\right\} \tag{10}$$

and, analogously, the same relationship holds true for the compromise difference excitations

$$\mathbf{B^E_{sa}} = \left\{b^E_{mn} = -b^H_{mn};\; m = 1, ..., N_x;\; n = 1, ..., N^m_y\right\}. \tag{11}$$

As regards to the compromise feeding network, $Q = 4$ partitions have been considered.

For illustrative purposes, let us analyze some steps of the BEM application. Once the *reference gains* have been computed by applying (6), they are sorted in a list $\mathbf{L}$ as indicated by the red line in Fig. 5. At each iteration ($t = 1, ..., T_{opt}$), a path ψ_t within the DAG is selected and a trial sub-array configuration $\mathbf{c}^{\mathbf{\Theta}}_t$ is obtained. As shown in Fig. 5, such an operation is equivalent to subdivide the list in $Q = 4$ subsets by selecting $Q-1$ cut points. Starting from a randomly-selected partition ($t = 0$ - Fig. 5), the path within the DAG is iteratively updated by changing the sub-array membership of the *border elements*, which results in a modification of the partition points in the list as shown in Fig. 5. The evolution of the sub-array memberships of the array elements is shown in Fig. 6 where the sub-array configurations determined at $t = 0$, $t = 10$, $t = 30$, and $t = T_{opt} = 48$ are shown. It is worth to note that, starting from a random and unbalanced sub-array configuration [Fig. 6(*a*) - $N^{(q)}_{T_{opt}}\Big|_{q=1} = 212$, $N^{(q)}_{T_{opt}}\Big|_{q=2} = 584$, $N^{(q)}_{T_{opt}}\Big|_{q=3} = 284$, $N^{(q)}_{T_{opt}}\Big|_{q=4} = 184$], the array elements tend to be homogeneously distributed to each sub-array as shown in Fig. 6(*d*) ($N^{(q)}_{T_{opt}}\Big|_{q=1} = 308$, $N^{(q)}_{T_{opt}}\Big|_{q=2} = 312$, $N^{(q)}_{T_{opt}}\Big|_{q=3} = 352$, $N^{(q)}_{T_{opt}}\Big|_{q=4} = 292$). Moreover, the convergence values assigned to the sub-array gains are: $w_1 = 0.2371$, $w_2 = 0.6838$, $w_3 = 1.0848$, and $w_4 = 1.5027$. The corresponding difference patterns in the $(u,\, v)$-plane (Fig. 7) and along the $\phi = 0^o$ plane (Fig. 8) confirm that the BEM is able to effectively sample the solution-graph, thus performing an effective compromise synthesis. As a matter of fact, starting from an initial pattern characterized by high SLLs as pointed out by the plot at $t = 0$ in Fig. 8, the final compromise solution [Fig. 7(*d*)] presents SLLs that do

not exceed the value of $SLL_{max} = -27\,dB$ in the whole angular region. Moreover, the main lobes of the synthesized pattern are close to those of the optimal difference pattern [see Fig. 8 ($v < 0.12$) and the central region of Fig. 7(*d*) compared to that in Fig. 4(*b*)] by guaranteeing the same accuracy in terms of angular resolution despite the large average number of array elements per sub-array ($av_{q=1,\ldots,Q}\left\{N^{(q)}\right\} = \frac{J}{Q} = 79$). To quantitatively appreciate the fitting between the compromise pattern and the optimal one, let us observe the behaviors and the convergence values of both the cost function Ψ_t (5) and the *matching index* Δ (defined as [11], but extended to the planar case) shown in Fig. 9.

As far as the computational issues are concerned, it should be firstly noticed that the dimension of the solution space is reduced from $U = 1.7822 \times 10^{190}$ (i.e., the dimension of the solution space when using stochastic optimization approaches) to $U^{(ess)} = 5.1598 \times 10^{6}$. Moreover, the convergence compromise solution ψ_{opt} and the corresponding aggregation vector $\mathbf{c}_{opt}^{\mathbf{\Theta}}$ are determined just after $T_{opt} = 48$ iterations with a CPU-time equal to $39.44\,[sec]$ (on a $3.4\,GHz$ PC with $2\,GB$ of RAM).

3.2 Comparative Assessment - SLL Control Procedure

To the best of the authors' knowledge, the synthesis of monopulse planar antennas with sub-arraying techniques has not been addressed with state-of-the-art excitation matching techniques, probably because of the arising theoretical and computational problems (i.e., ill-conditioning and matrix storage resources), and it has been recently carried out in terms of pattern-feature optimization only by Ares *et al.* in [9], where a SA-based procedure has been used to define the sub-array weights of an *a-priori* fixed sub-array configuration $\mathbf{c}^{\Theta}$. More in detail, the weight vector $\mathbf{w}_{SA}^{\Theta}$ has been calculated by minimizing the cost function, evaluated at many azimuthal patterns (ϕ-cuts), which penalizes a maximum side lobe level SLL exceeding a specified tolerable maximum level SLL_{req}. Consequently, since the BEM as well as other optimal excitations matching procedures do not allow a direct and individual control of the grating lobes of the synthesized patterns [10], the proposed approach has been integrated into an iterative loop to perform the SLL minimization [or the optimization of other beam pattern features as the beamwidth (BW) or directivity, etc...] through the matching with a reference pattern (e.g., a pattern optimum in the Dolph-Chebyshev sense [4]) as profitably used for the synthesis of

monopulse linear arrays [21]. The flowchart of such a procedure, indicated as *Iterative-*BEM ($IBEM$), is given in Fig. 10. In particular, the reference excitations $\mathbf{B}^{\Theta(k)}$ (k being the index of the iterative loop) are recursively chosen until the compromise pattern synthesized with the BEM satisfies the user-defined constraints.

As far as the comparison at hand is concerned, the *side lobe ratio* (SLR)

$$SLR\left(\phi\right) = \frac{SLL\left(\phi\right)}{max_{\theta}\left[AF\left(\theta,\phi\right)\right]},\ \ 0 \leq \theta < \frac{\pi}{2} \tag{12}$$

shown in Fig. 9 of [9] has been chosen as the pattern feature to be minimized [i.e., $SLR^{H(k)}\left(\phi\right) \leq SLR_{req}\left(\phi\right)$, $SLR_{req}\left(\phi\right)$ being the user-defined threshold] and the same benchmark investigated by Ares in [9] with $Q = 3$ sub-arrays has been taken into account. Other Q values have been also considered, but no comparisons with other state-of-the-art methods are reported here since not available in the published literature. The obtained results are just shown in Fig. 11 for validation purposes in the framework of an asymptotic assessment aimed at pointing out, as expected, the convergence of the SLR behavior to that of the reference pattern when Q grows. The planar array consists of $N = 300$ equally-spaced ($d = \frac{\lambda}{2}$) elements arranged on a rectangular grid and belonging to a circular aperture $r = 4.85\,\lambda$ in radius. The sum mode has been set to a circular Taylor pattern [3][20] with $SLL = -35\,dB$ and $\overline{n} = 6$.

As far as the application of the $IBEM$ is concerned, the first ($k = 1$) reference excitations set $\mathbf{B}_{opt}^{H(1)}$ has been chosen equal to that of a circular Bayliss pattern [4][20] with $SLL_{ref}^{H(1)} = -25\,dB$ and $\overline{n} = 6$. At the end of the first loop of BEM iterations ($t^{(1)} = 1, ..., T_{opt}^{(1)}$, $T_{opt}^{(1)} = 14$ - Fig. 12), the path $\psi_{opt}^{H(1)}$ in the DAG shown in Fig. 13 has been identified, but the arising SLR plot (Fig. 14) does not favorably compares with that in [9] since the BEM has not been able to efficiently match the reference pattern.

Successively ($k > 1$), the reference pattern has been updated to iterate the $IBEM$ process. In particular, the reference set of excitations has been set to that generating a $\overline{n} = 6$ Bayliss pattern with $SLL_{ref}^{H(2)} = -30\,dB$ ($k = 2$) and with $SLL_{ref}^{H(3)} = -35\,dB$ ($k = K_{opt} = 3$), respectively. The plot of the cost function throughout the minimization process is reported in Fig. 12. As it can be noticed, although different and more restrictive targets (i.e., $SLL_{ref}^{H(k)} < SLL_{ref}^{H(k-1)}$) have been used, the value of $\Psi_{opt}^{H(k)}$ still decreases until the convergence when the path $\psi_{opt}^{H(3)}$

in Fig. 13 [coding the aggregation in Fig. 15(*c*)] and the corresponding pattern shown in Fig. 16(*c*) have been synthesized. At the end of the process, the control of the $H-mode$ in ϕ range $[0^o,\, 80^o]$ appears to be more satisfactory than that in [9] as pointed out by the plots in Fig. 14 since $SLR_{IBEM}\left(\phi\right) \leq SLR_{SA}\left(\phi\right)\, \forall\, \phi \in [0^o,\, 80^o]$ and $av_{\phi}\left\{SLR_{IBEM}\left(\phi\right)\right\} = -25.87\ dB$ vs. $av_{\phi}\left\{SLR_{SA}\left(\phi\right)\right\} = -22.52\ dB$. The final values of the sub-array gains $\mathbf{w}_{opt}^{H(3)}$, computed according to (8) starting from the element distribution $\mathbf{c}_{opt}^{H(3)}$ in Fig. 16(*c*), are given in Tab. I.

As regards to the iterative process and with reference to Fig. 12, it is worthwhile to notice that, as expected and unlike the matching index Δ, the values of the beam-pattern indexes (i.e., SLL, SLR, and BW) do not monotonically decrease because of the excitation matching nature of the BEM, although in its iterative version, which allows only an indirect control of the pattern features.

For completeness, the computational efficiency of the $IBEM$ is further pointed out by the following indications. The dimension of the solution space decreases from $U = 6.0826 \times 10^{35}$ to $U^{(ess)} = 2701$ [Fig. 1(*a*)], whereas the CPU-time required to complete the outer iterative loop is equal to $2.64\,[sec]$. Moreover, it should be pointed out that the use of the DAG instead of the non-complete binary tree (IBT) allows a memory saving of about $\frac{M^{(IBT)}}{M^{(DAG)}} \approx 10^{19}$ ($\frac{M^{(CBT)}}{M^{(IBT)}} \approx 10^{2}$).

3.3 Hybrid Formulation - The $Hybrid-IBEM$ Approach

Inspired by the investigations on the synthesis of difference patterns carried out in [22], it has been shown in [10] that the definition of the sub-array weights for compromise monopulse array antennas can be formulated as the solution of a convex programming (CP) problem once the sub-array configuration is given. By exploiting such a property, a hybrid approach has been first proposed in [23] to deal with the synthesis of monopulse linear arrays. In the following, a hybrid version of the $IBEM$ (i.e., the $Hybrid-IBEM$) is customized to the synthesis of planar arrays in order to extend the range of applicability of the planar BEM from excitation matching to pattern optimization allowing, unlike the $IBEM$, a direct control of the pattern features (i.e., SLL, BW, etc...).

Similarly to [23], the hybrid approach consists of a two-step procedure where at the first step the sub-array configuration is computed according to the $IBEM$ (i.e., $\mathbf{c}_{Hybrid-IBEM}^{\mathbf{\Theta}} = \mathbf{c}_{opt}^{\mathbf{\Theta}}$).

Successively, the weights $\mathbf{w}^{\mathbf{\Theta}}_{Hybrid-IBEM}$, $\Theta = E,\ H$, of the sub-arrayed difference network are computed by means of a standard CP procedure minimizing the following cost function

$$\Psi^{CP}\left(\mathbf{w}^{\mathbf{\Theta}}\right) = \min_{\left\{w_q^{\Theta};\, q=1,\ldots,Q\right\}} \left. \frac{\partial \left\{ \sum_{m=-N_x}^{N_x} \sum_{n=-N_y^m}^{N_y^m} \left[\Re\left(b_{mn}^{\Theta}\right) \cos \Upsilon\left(\theta,\phi\right) - \Im\left(b_{mn}^{\Theta}\right) \sin \Upsilon\left(\theta,\phi\right) \right] \right\}}{\partial \chi} \right|_{\substack{\theta=0 \\ \phi=0}} \tag{13}$$

χ being either θ or ϕ and $\Upsilon\left(\theta,\phi\right) = k_x x_m + k_y y_n$, subject to

$$\left. \frac{\partial \left\{ \sum_{m=-N_x}^{N_x} \sum_{n=-N_y^m}^{N_y^m} \left[\Re\left(b_{mn}^{\Theta}\right) \sin \Upsilon\left(\theta,\phi\right) + \Im\left(b_{mn}^{\Theta}\right) \cos \Upsilon\left(\theta,\phi\right) \right] \right\}}{\partial \chi} \right|_{\substack{\theta=0 \\ \phi=0}} = 0 \tag{14}$$

and

$$AF\left(\theta,\phi\right)|_{\substack{\theta=0 \\ \phi=0}} = \sum_{m=-N_x}^{N_x} \sum_{n=-N_y^m}^{N_y^m} b_{mn}^{\Theta} = 0 \tag{15}$$

and to $\left|AF\left(\theta,\phi\right)\right|^2 \leq \mathcal{M}\left(\theta,\phi\right)$ where $\mathcal{M}\left(\theta,\phi\right)$ is a function descriptive of a user-defined mask on the synthesized difference power pattern. In Eq. (13), $\Re\left(\cdot\right)$ and $\Im\left(\cdot\right)$ denote the real and imaginary part, respectively. At the initialization of the CP procedure, the guess solution is set to the values of the sub-array weights obtained at the end of the $IBEM$, $\mathbf{w}^{\mathbf{\Theta},(\mathrm{o})} = \mathbf{w}^{\mathbf{\Theta}}_{opt}$ [Eq. (8)].

In order to show the SLL/BW control allowed by the hybrid approach, Figure 17 summarizes the results from a comparative study between the $IBEM$ and its hybrid version in terms of maximum SLL [Fig. 17(*a*)] and corresponding BW computed on the principal plane [i.e., the $\phi = 0^o$ cut of Fig. 16(*c*)] [Fig. 17(*b*)] dealing with the same array configuration of Sect. 3.2. To better and more exhaustively analyze the potentialities of the proposed hybrid approach, the number of sub-arrays has been varied in the range $Q \in [2, 8]$ and the synthesized sub-arrays configurations and weights are shown in Fig. 15. For completeness, the corresponding patterns are also given [Fig. 16]. As it can be observed (Figs. 16-17), the solutions from the $Hybrid - IBEM$ outperform those of the $IBEM$ in terms of pattern indexes even though with heavier computational costs. As far as the computational issues are concerned, the dimension of the solution space $U^{(DAG)}$ and the storage resources $M^{(DAG)}$ are given in Fig. 1, whereas the CPU-time and number of iterations T_{CP} required to get the final solution for the $Hybrid - IBEM$ and $IBEM$ are reported in Tab. II to point out the trade-off between pattern efficiency and

computational burden.

4 Conclusions

In this paper, an efficient approach for the synthesis of sub-arrayed monopulse planar antennas has been presented. Starting from the guidelines of an effective procedure previously developed to deal with linear geometries, some innovative features have been introduced to extend the capability of the approach as well as its efficiency, thus enabling the synthesis of planar monopulse arrays. As a matter of fact, by exploiting some features of the solution, a simple and compact representation of the space of admissible solutions has been defined, which allows a considerable reduction of the problem complexity as well as a significant saving in terms of storage resources and CPU-time to synthesize the compromise solution.

The effectiveness of the proposed excitation matching technique in sampling the solution space has been assessed through some experiments concerned with high-dimension synthesis problems computationally unfeasible for stochastic optimization procedures. Furthermore, for comparison purposes and to deal with user-defined requirements besides matching a reference pattern, the BEM has been integrated in a recursive process that proved to improve the performance of optimization techniques in dealing with the benchmarks available in the related literature and concerned with SLL control.

As regards to the SLL control, the convexity of the problem with respect to a part of the unknowns has been exploited by defining a two-step hybrid approach based on the $IBEM$. Although the heavier computational burden affecting the second step of the hybrid method (i.e., the CP procedure), the obtained results point out the improvements coming from the exploitation of the effectiveness of the $IBEM$ in defining the sub-array aggregation and the convexity of the problem with respect to the sub-array weights.

References

[1] I. M. Skolnik, *Radar Handbook*. McGraw-Hill, 1990.

[2] P. W. Hannan, "Optimum Feeds for All Three Modes of a Monopulse Antenna I: Theory," *IRE Trans. Antennas Propagat.*, vol. 9, no. 5, pp. 444-454, Sept. 1961.

[3] T. T. Taylor, “Design of a circular aperture for narrow beamwidth and low sidelobes,” *IRE Trans. Antennas Propagat.*, vol. AP-8, pp. 17-22, 1960.

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

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

[6] 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 Propagat.*, vol. 49, no. 11, pp. 1606-1608, 2001.

[7] 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 Propagat.*, vol. 53, no. 1, pp. 372-376, 2005.

[8] 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.

[9] F. Ares, J. A. Rodriguez, E. Moreno, and S. R. Rengarajan, “Optimal compromise among sum and difference patterns,” *J. Electromag. Waves Appl.*, vol. 10, pp. 1543-1555, 1996.

[10] M. D’Urso, T. Isernia, and E. T. Meliadò, “An Effective Hybrid Approach for the Optimal Synthesis of Monopulse Antennas, ” *IEEE Trans. Antennas Propagat.*, vol. 55, no. 4, pp. 1059-1066, Apr. 2007.

[11] 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 Propagat.*, vol. 56, no. 1, pp. 58-66, Jan. 2008.

[12] 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.

[13] D. B. West, *Introduction to Graph Theory*. Englewood Cliffs, NJ: Prentice-Hall, 2000.

[14] L. Manica, P. Rocca, and A. Massa, "Synthesis of sub-arrayed monopulse planar arrays by means of an innovative excitation matching method" *Proc. 2007 IEEE Radar Conf.*, Roma, Italy, May 26-29, 2008.

[15] P. Rocca, L. Manica, and A. Massa, "An effective matching method for the synthesis of optimal compromise between sum and difference patterns in planar arrays," *Progress in Electromagnetics Research B*, PIER-B 3, pp. 115-130, 2008.

[16] L. Manica, P. Rocca, and A. Massa, "On the synthesis of sub-arrayed planar array antennas for tracking radar applications," *IEEE Antennas Wireless Propagat. Lett.*, 2008. In press.

[17] R. E. Hodges and Y. Rahmat-Samii, "On sampling contiguous aperture distributions for discrete planar arrays," *IEEE Trans. Antennas Propagat.*, vol. 44, no. 11, pp 1499-1508, Nov. 1996.

[18] W. D. Fisher, "On grouping for maximum homogeneity," *American Statistical Journal*, 789-798, 1958.

[19] P. Rocca, L. Manica, A. Martini, and A. Massa, "Computationally-effective optimal excitation matching for the synthesis of large monopulse arrays," *Proc. 2007 IEEE AP-S International Symp.*, Honolulu, Hawaii, USA, pp. 3153-3156, June 10-15, 2007.

[20] R. S. Elliott, *Antenna Theory and Design*. Wiley-Interscience IEEE Press, 2003.

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

[22] O. Bucci, M. D'Urso, and T. Isernia, "Optimal synthesis of difference patterns subject to arbitrary sidelobe bounds by using arbitrary array antennas," *IEE Proc. Microwaves Antennas Propagat.*, vol. 152, no. 3, pp. 129-137, Jun. 2005.

[23] 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.

FIGURE CAPTIONS

- **Figure 1.** *Sub-Arrayed Planar Array Synthesis* ($d = \frac{\lambda}{2}$, $r = 4.85\lambda$). *Computational Analysis* - (*a*) Dimension of the solution space U and (*b*) memory resources, M, and number of vertexes, V, for the storage of the representations of the solution space versus Q in correspondence with $N = 300$ and $N = 40$ (CBT →Complete Binary Tree, IBT →Non-Complete Binary Tree, and DAG →Direct Acyclic Graph).

- **Figure 2.** Pictorial representation of (*a*) the *Direct Acyclic Graph* (DAG) and (*b*) its corresponding planar tree ($J = 6$ and $Q = 3$).

- **Figure 3.** Pseudo-code of the iterative graph-searching algorithm (BEM).

- **Figure 4.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Relative power distribution of the reference (*a*) sum Taylor pattern ($SLL = -35\,dB$, $\overline{n} = 6$) and of the (*a*) $H - mode$ difference Bayliss pattern ($SLL = -35\,dB$, $\overline{n} = 6$), respectively.

- **Figure 5.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Illustrative description of the BEM. Evolution of the list partition ($t = 0, 10, 30, T_{opt}$, $T_{opt} = 48$).

- **Figure 6.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Evolution of the sub-array memberships of the array elements. Sub-array configurations synthesized by the BEM at (*a*) $t = 0$, (*b*) $t = 10$, (*c*) $t = 30$, and (*d*) $t = T_{opt} = 48$.

- **Figure 7.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Relative power distribution at (*a*) $t = 0$, (*b*) $t = 10$, (*c*) $t = 30$, and (*d*) $t = T_{opt} = 48$.

- **Figure 8.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Azimuthal ($\phi = 0^o$) relative power distribution at (*a*) $t = 0$, (*b*) $t = 10$, (*c*) $t = 30$, and (*d*) $t = T_{opt} = 48$.

- **Figure 9.** *Sub-Arrayed Planar Array Synthesis* ($N = 1264$, $d = \frac{\lambda}{2}$, $r = 10\lambda$, $Q = 4$) - Behavior of the cost function Ψ and of the pattern matching Δ versus the iteration index t.

- **Figure 10.** Flow chart of the $IBEM$.

- **Figure 11.** *Asymptotic Validation* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$) - Plots of the synthesized SLR values in the range $\phi \in [0^o,\ 80^o]$ for different values of Q. Reference Bayliss pattern [4]: $\overline{n} = 6$ and $SLL^{)}_{ref} = -35\,dB$.

- **Figure 12.** *Comparative Assessment* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Behaviors of the cost function $\Psi_t^{H(k)}$, SLL, SLR, and beamwidth BW versus the iteration index (Reference Bayliss pattern $\overline{n} = 6$ [4]: $SLL_{ref}^{H(1)} = -25\,dB$, $SLL_{ref}^{H(2)} = -30\,dB$, and $SLL_{ref}^{H(3)} = -35\,dB$).

- **Figure 13.** *Comparative Assessment* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Evolution of the solution path $\psi_{opt}^{H(k)}$ synthesized with the $IBEM$ within the DAG mapping the solution space.

- **Figure 14.** *Comparative Assessment* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Plots of the synthesized SLR values in the range $\phi \in [0^o,\ 80^o]$. Reference Bayliss pattern $\overline{n} = 6$ [4]: $SLL_{ref}^{H(1)} = -25\,dB$, $SLL_{ref}^{H(2)} = -30\,dB$, and $SLL_{ref}^{H(3)} = -35\,dB$.

- **Figure 15.** *Hybrid Formulation* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Sub-array configurations (*left column*) and array element weights (*right column*) synthesized with the $IBEM$ and the $Hybrid-IBEM$ for different values of Q [$Q = 2$ (*first row*), $Q = 3$ (*second row*), $Q = 5$ (*third row*), and $Q = 8$ (*fourth row*)].

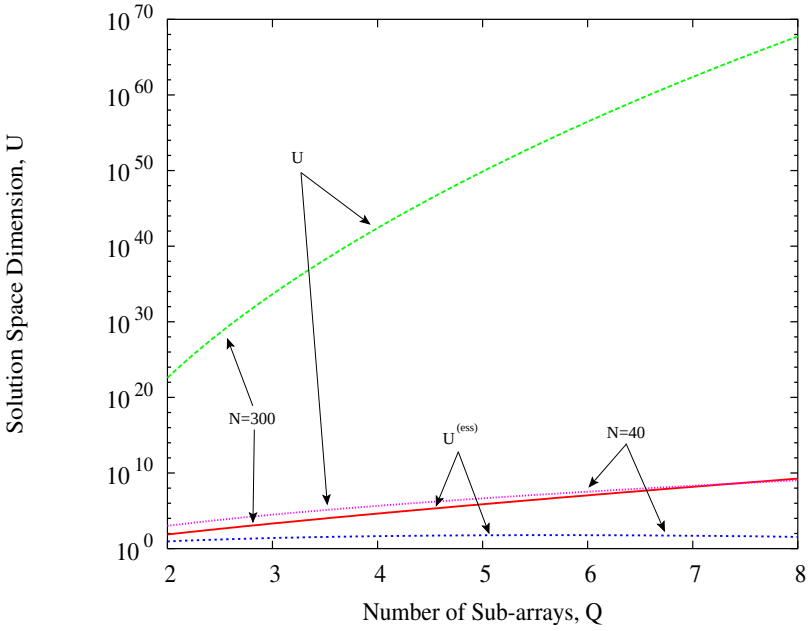
- **Figure 16.** *Hybrid Formulation* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Beam patterns synthesized with the $IBEM$ (*left column*) and the $Hybrid - IBEM$ (*right column*) for different values of Q [$Q = 2$ (*first row*), $Q = 3$ (*second row*), $Q = 5$ (*third row*), and $Q = 8$ (*fourth row*)].

- **Figure 17.** *Hybrid Formulation* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$) - Behavior of the (*a*) SLL and of the (*b*) BW for the compromise patterns synthesized by means of the $IBEM$ and the $Hybrid - IBEM$ when $Q \in [2, 8]$.

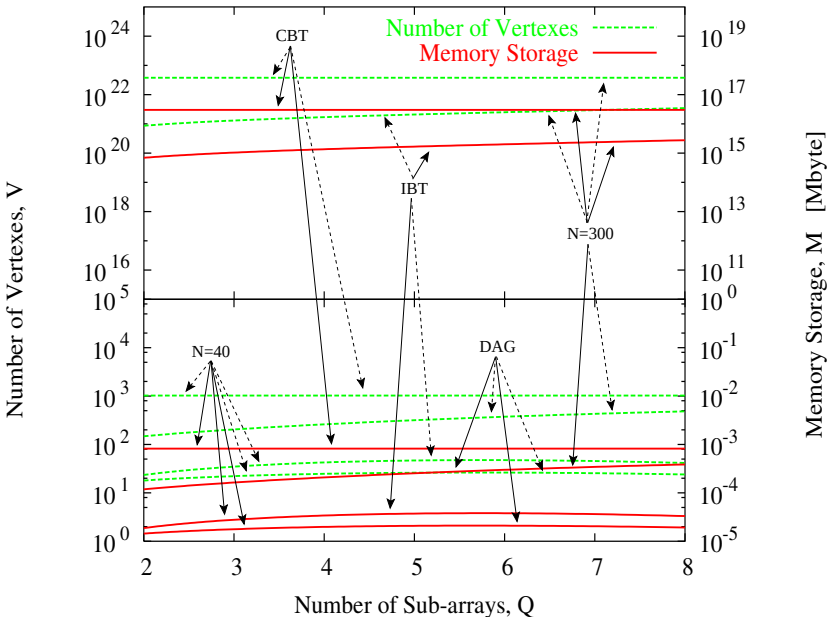
TABLE CAPTIONS

- **Table I.** *Comparative Assessment* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$, $Q = 3$) - Sub-array weights ($IBEM$).

- **Table II.** *Hybrid Formulation* ($N = 300$, $d = \frac{\lambda}{2}$, $r = 4.85\lambda$) - Computational indexes for the solution obtained with the $IBEM$ and the $Hybrid - IBEM$.

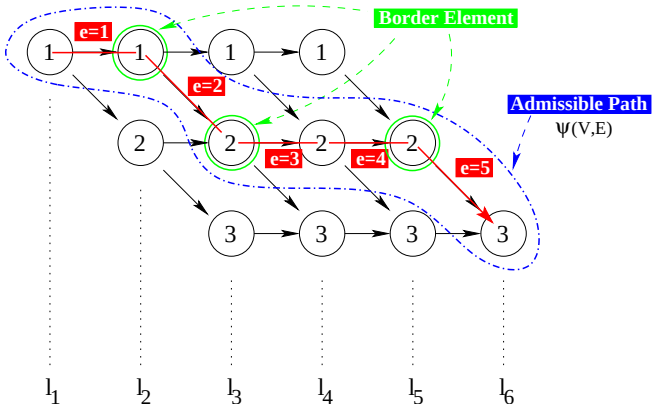


(a)

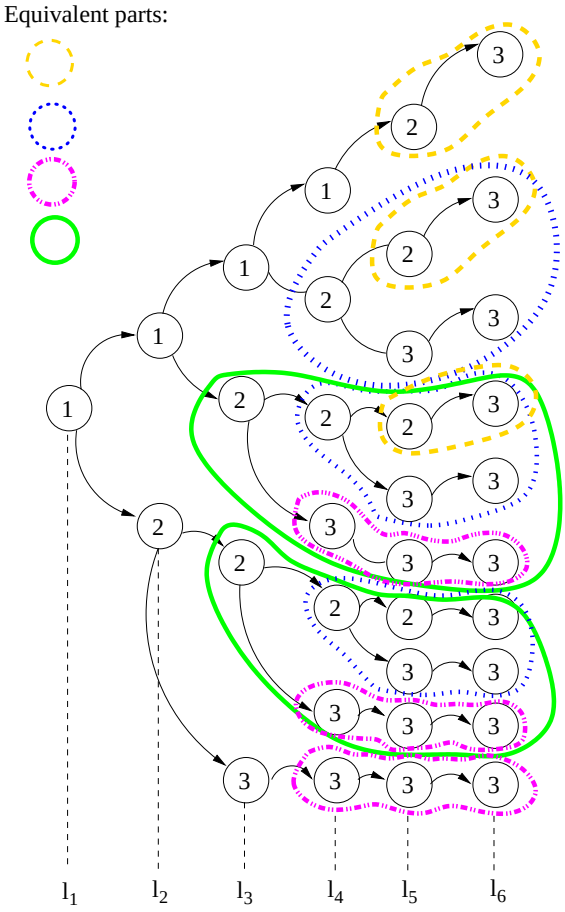


(b)

Fig. 1 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."



(a)

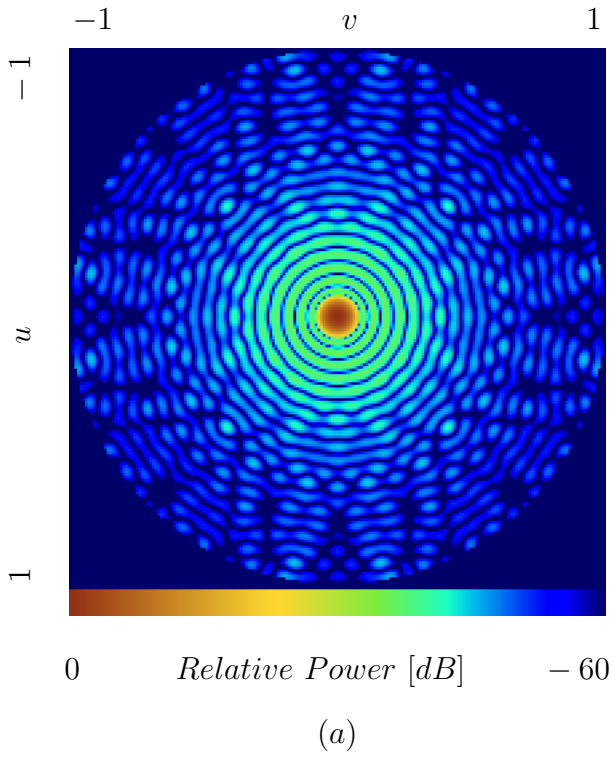


(b)

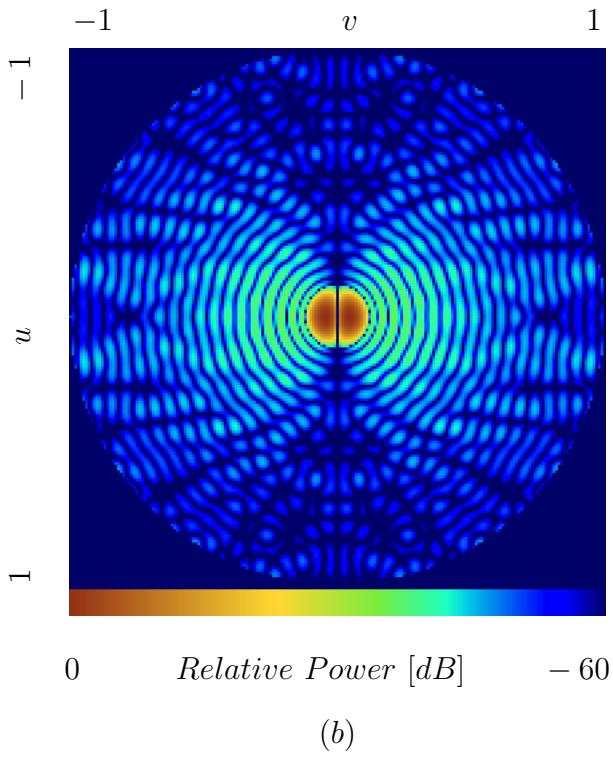
Fig. 2 - L. Manica *et al.,* "A Fast Graph-Searching Algorithm ..."

```
Compute γ_j : j = 1, ..., J;
Sort γ_j : j = 1, ..., J;
Randomly Initialize ψ_0;
Compute Cost of ψ_0: Ψ_0 = Ψ(c_0^Θ)
until (t < T_max)

    Update t: t ← t + 1;
    Set c_t^Θ: c_t^Θ = c_{t-1}^Θ;
    for (j = 2, ..., J − 1) do
        if (j == border element) then
            if (c_j^Θ = c_{j-1}^Θ) then
                s_j = c_j^Θ + 1;
            else if (c_{j(t)}^Θ = c_{j(t)+1}^Θ) then
                s_j = c_{j(t)}^Θ − 1;
            endif
            Update ψ_f: c_f^Θ = {c_1^Θ = 1, ..., c_{j-1}^Θ, s_j, c_{j+1}^Θ, ..., c_J^Θ = Q};
        endif
        Compute Cost of ψ_f: Ψ_f = Ψ(c_f^Θ)
        if (Ψ_f < Ψ_t) then
            Update t: t ← t + 1;
            Update Ψ_t^opt: Ψ_t^opt = Ψ_f;
            Update ψ_t: ψ_t = ψ_f;
            Update c_t^Θ: c_t^Θ = c_f^Θ;
            if ((t = T_max) or ( |T_window Ψ_{t-1} − Σ_{h=1}^{T_window} Ψ_h| / Ψ_t^opt ≤ τ)) then
                Set T_opt: T_opt = t;
                Set ψ_opt: c_opt^Θ = c_t^Θ;
                Compute w_opt^Θ (8);
                stop
            endif
        endif
    endfor

enduntil
```

Fig. 3 - L. Manica *et al.,* “A Fast Graph-Searching Algorithm ...”



(a)



(b)

Fig. 4 - L. Manica *et al.,* "A Fast Graph-Searching Algorithm ..."

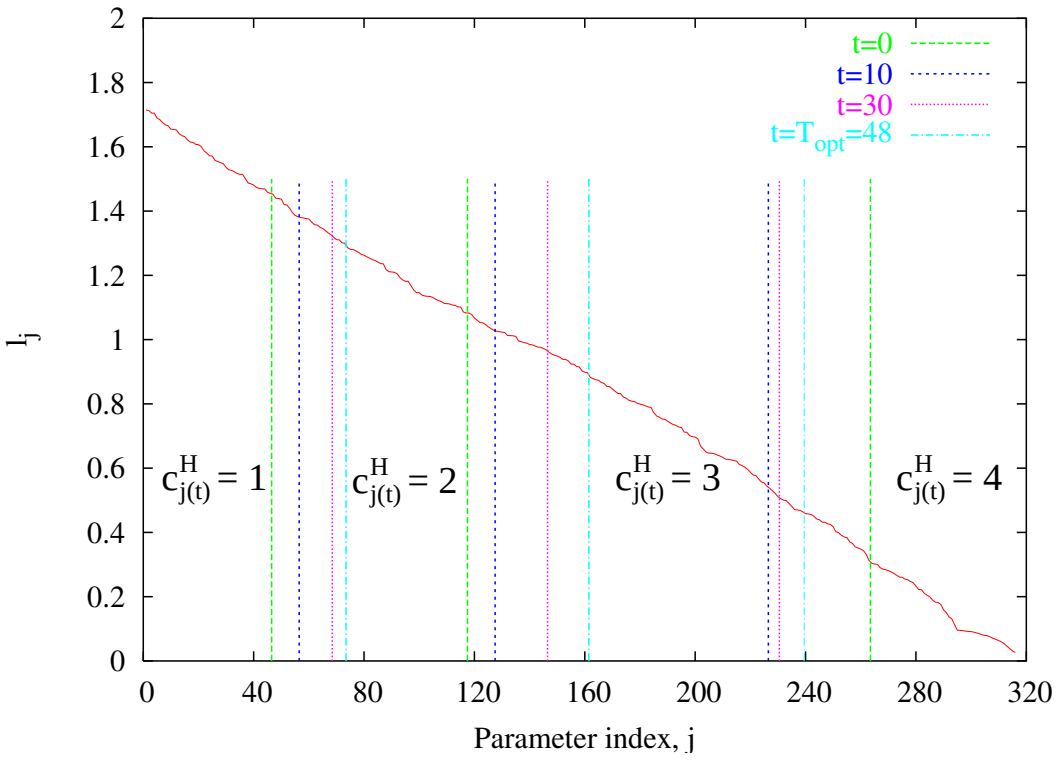


Fig. 5 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

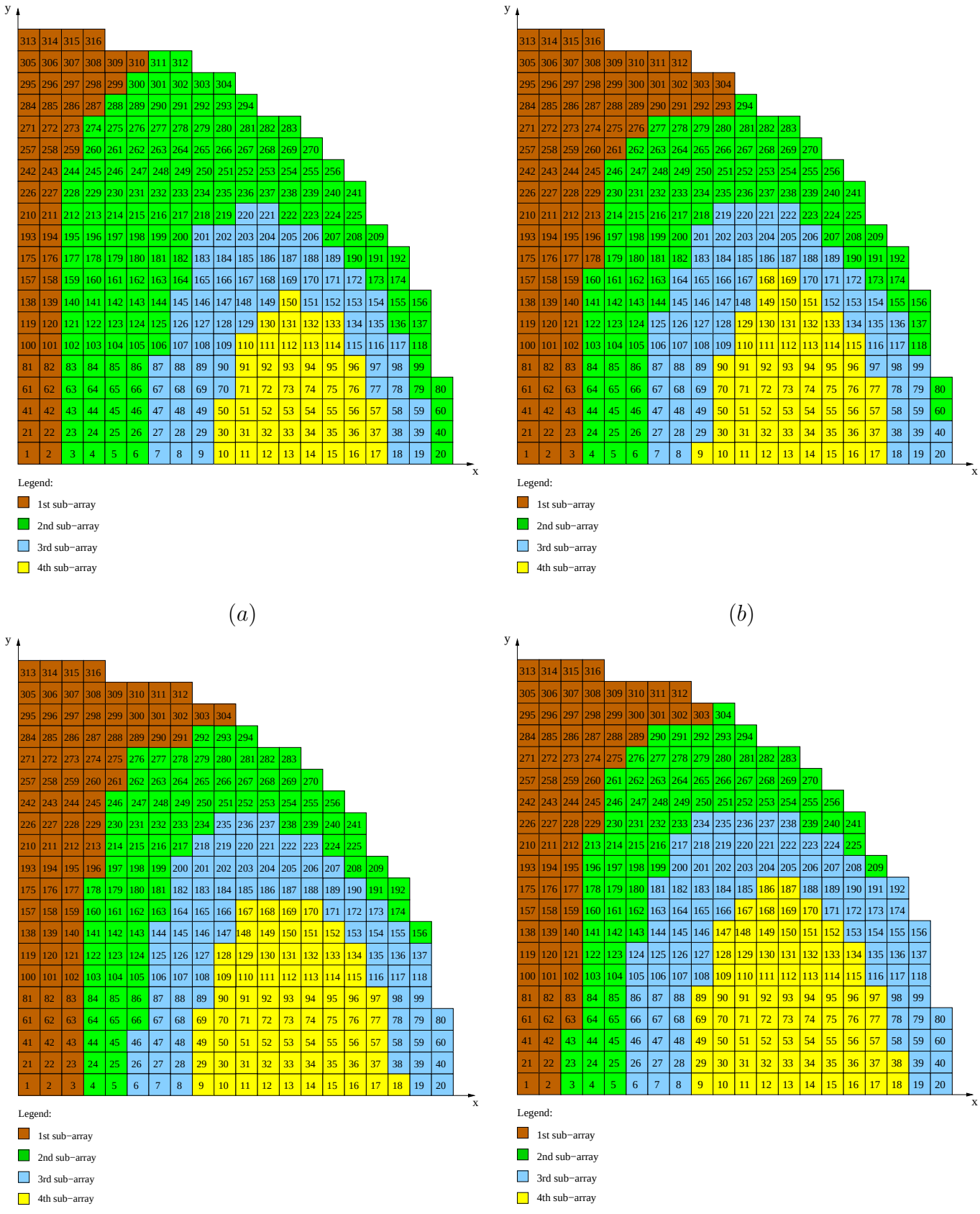


Fig. 6 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

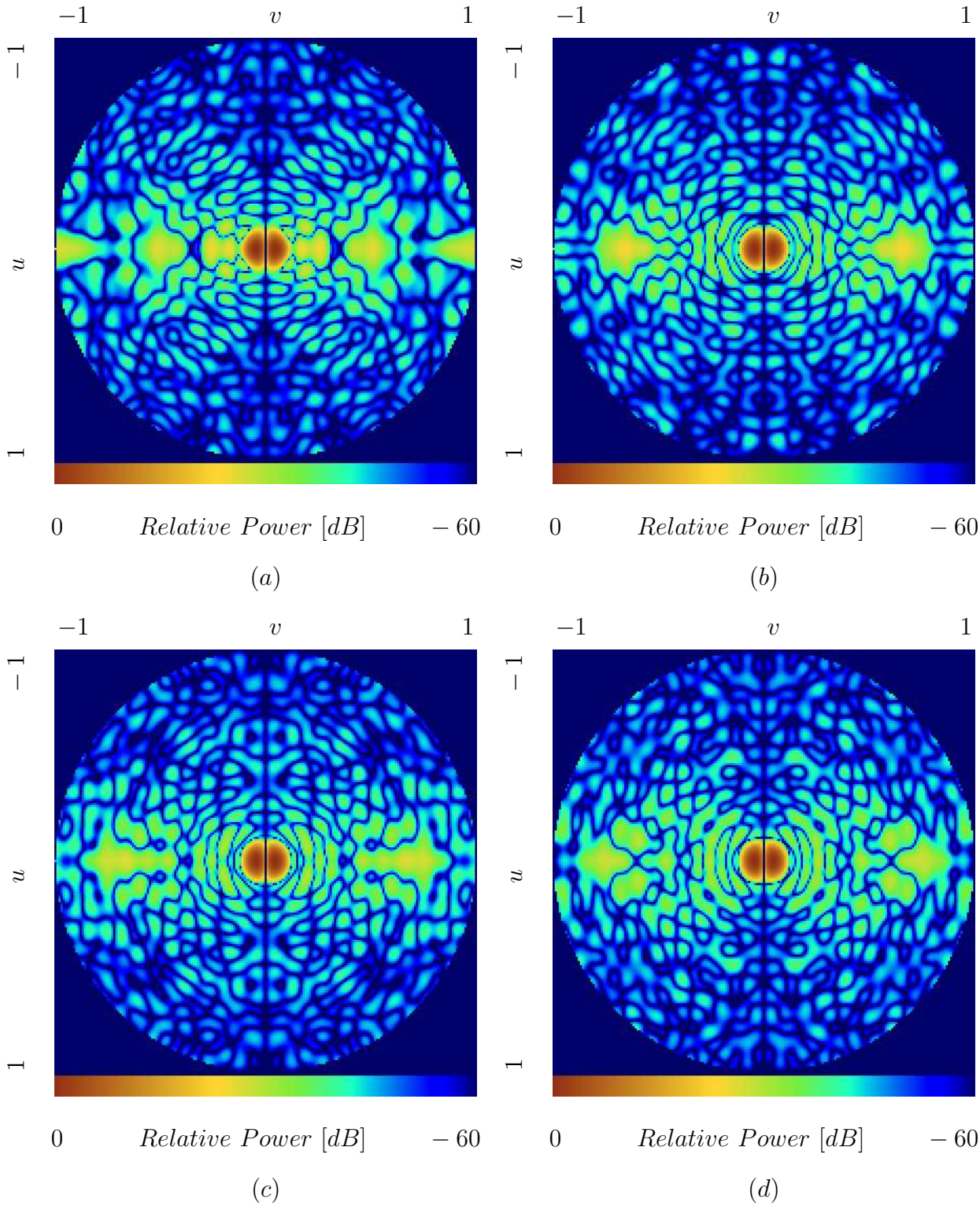


Fig. 7 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

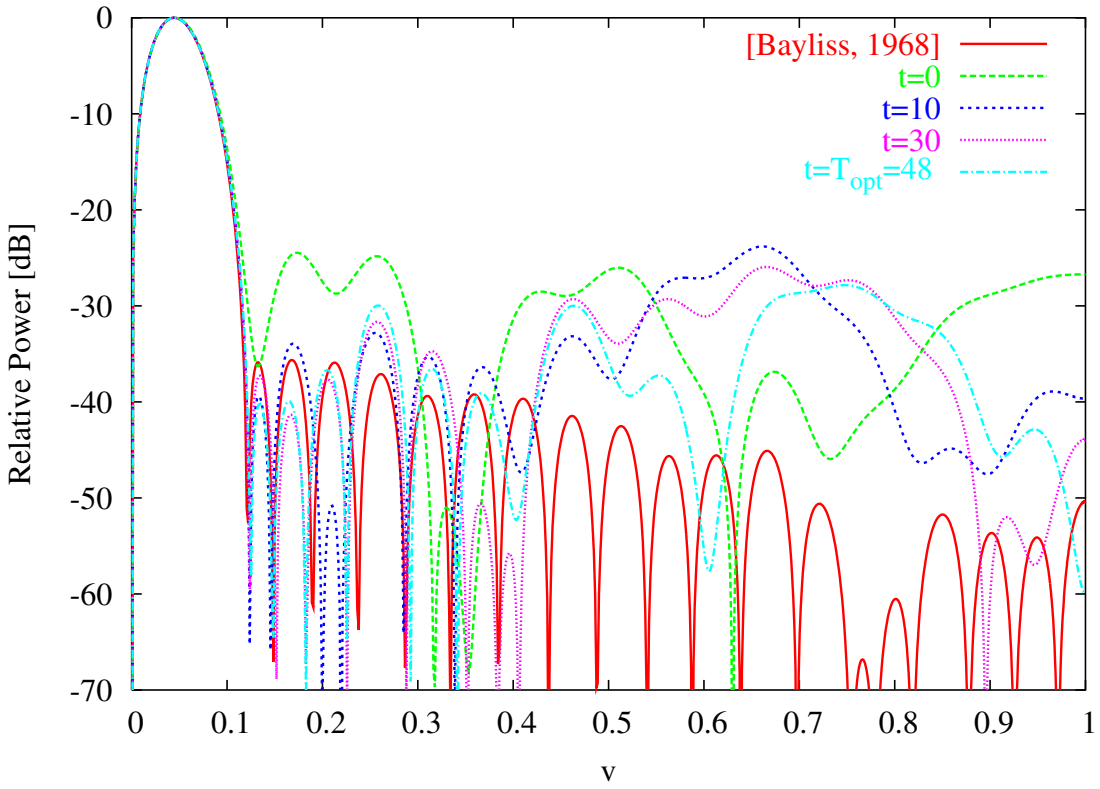


Fig. 8 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

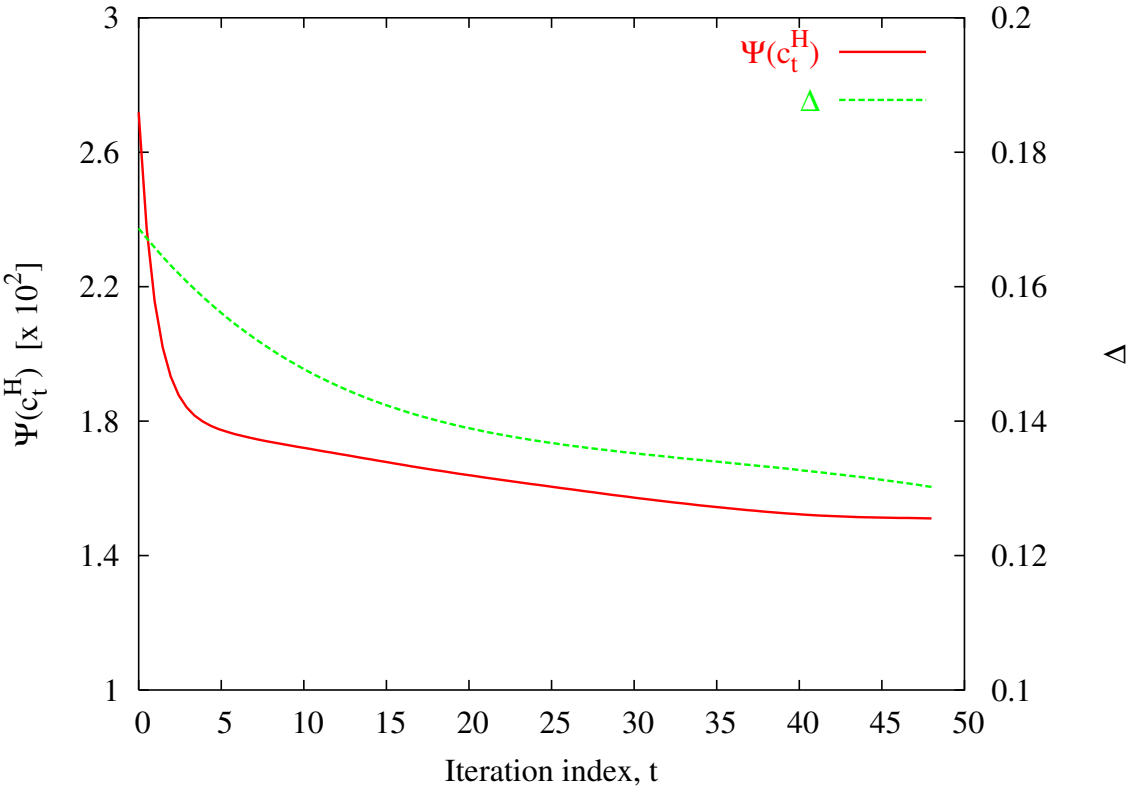


Fig. 9 - L. Manica *et al.,* "A Fast Graph-Searching Algorithm ..."

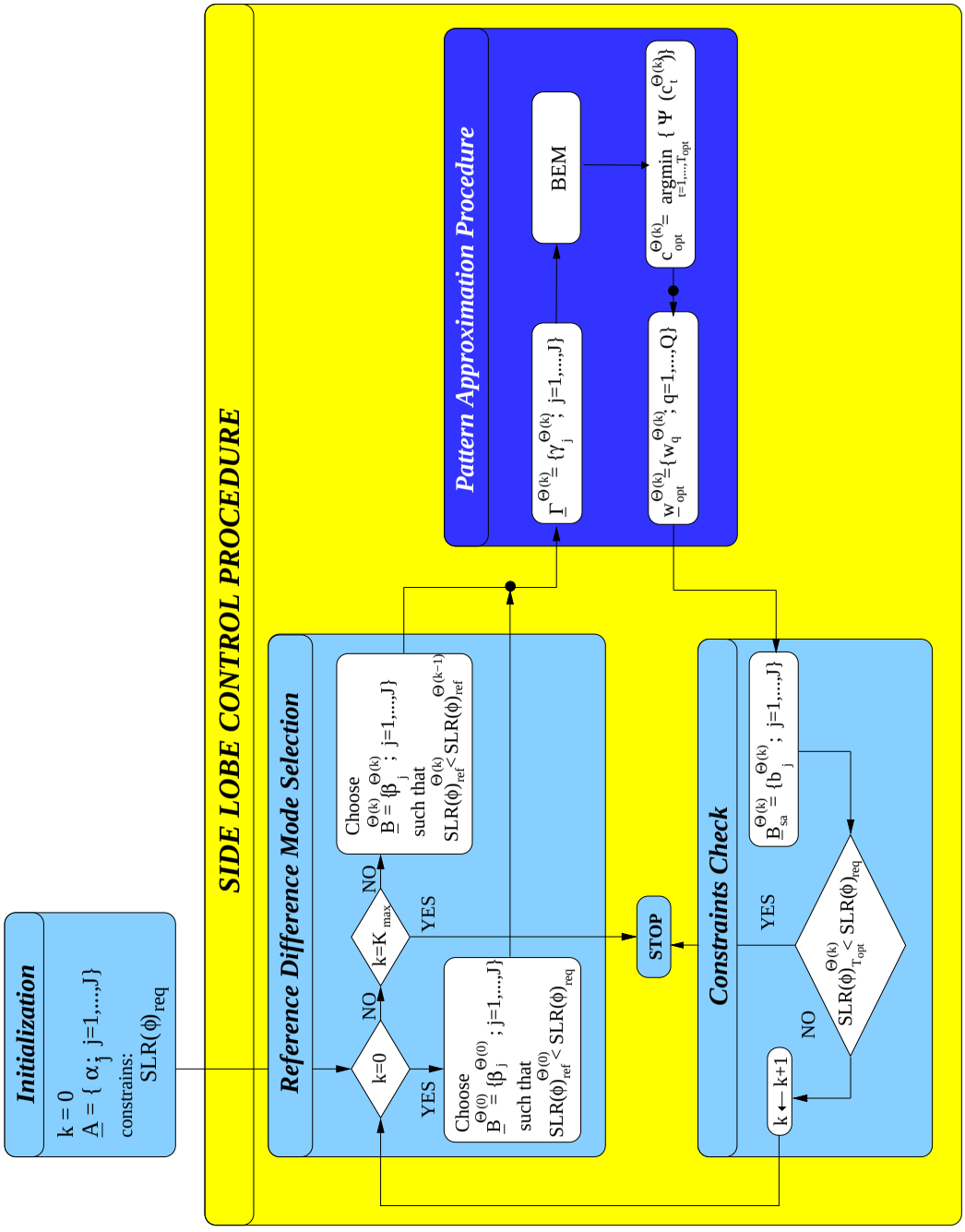


Fig. 10 - L. Manica *et al.,* "A Fast Graph-Searching Algorithm ..."

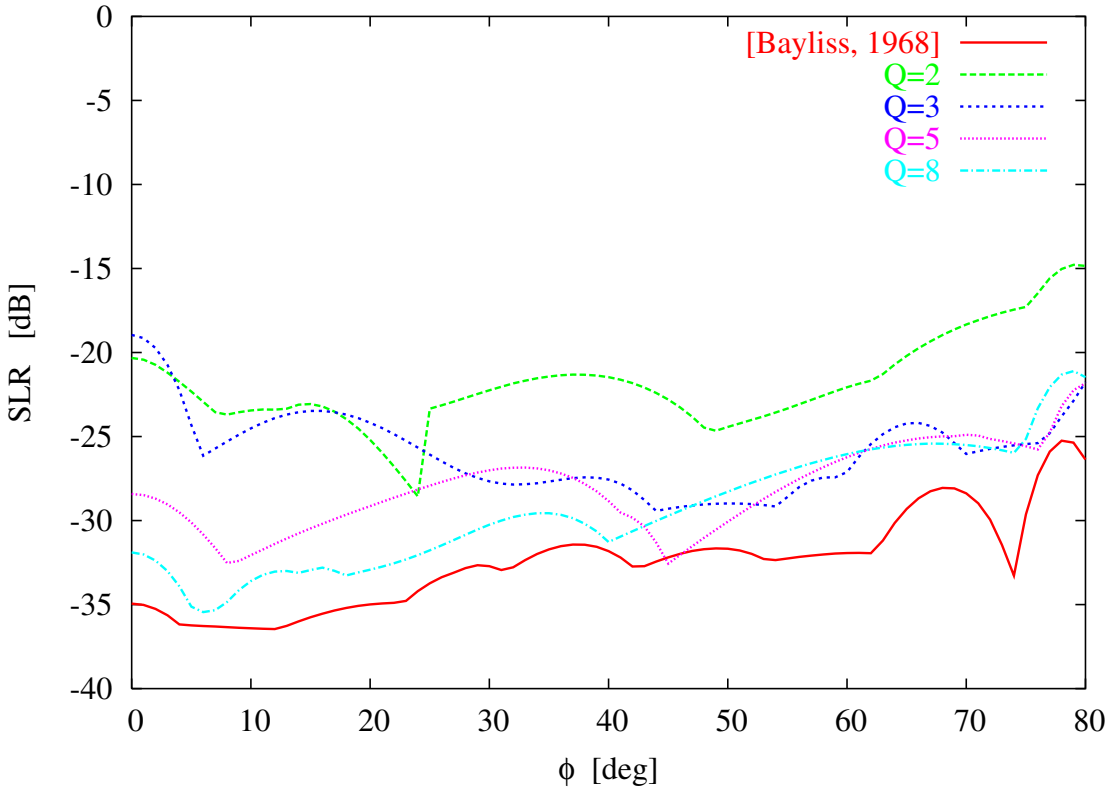


Fig. 11 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

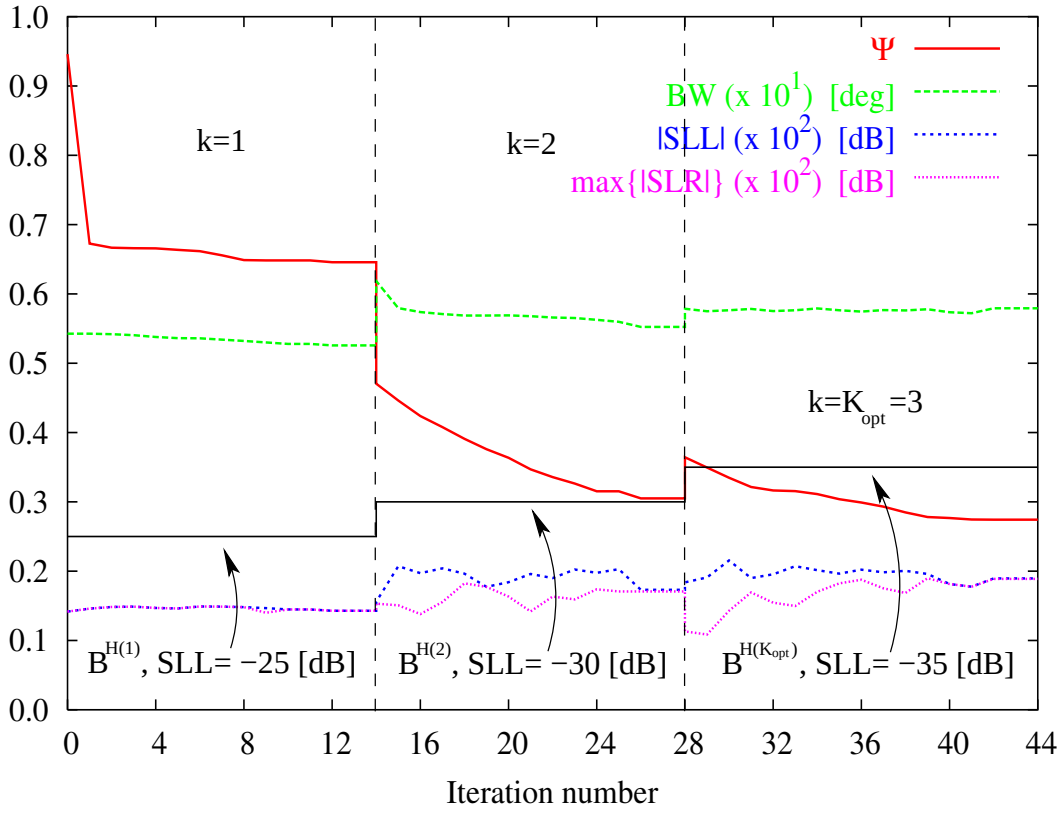


Fig. 12 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

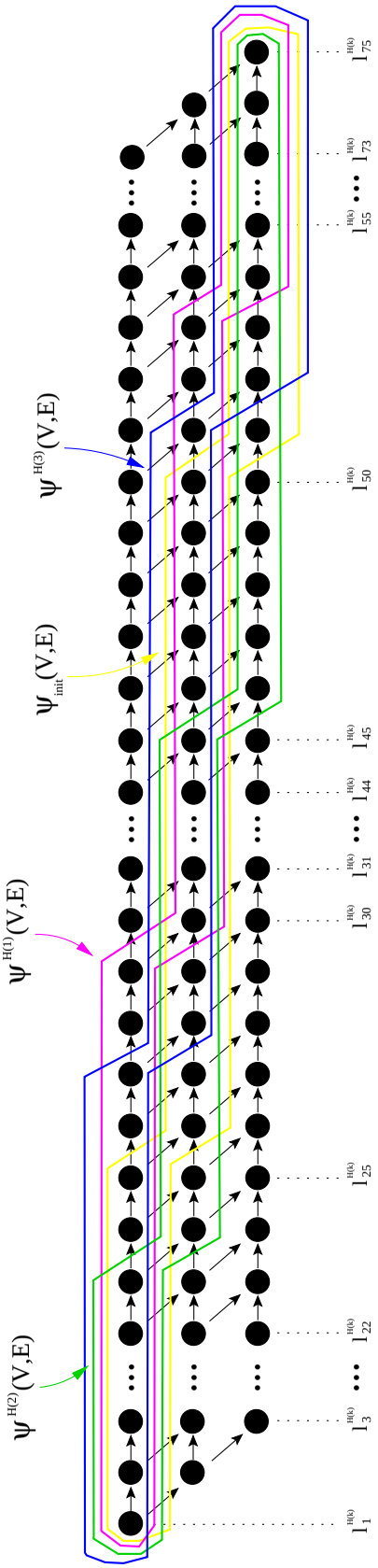


Fig. 13 - L. Manica *et al.*, “A Fast Graph-Searching Algorithm ...”

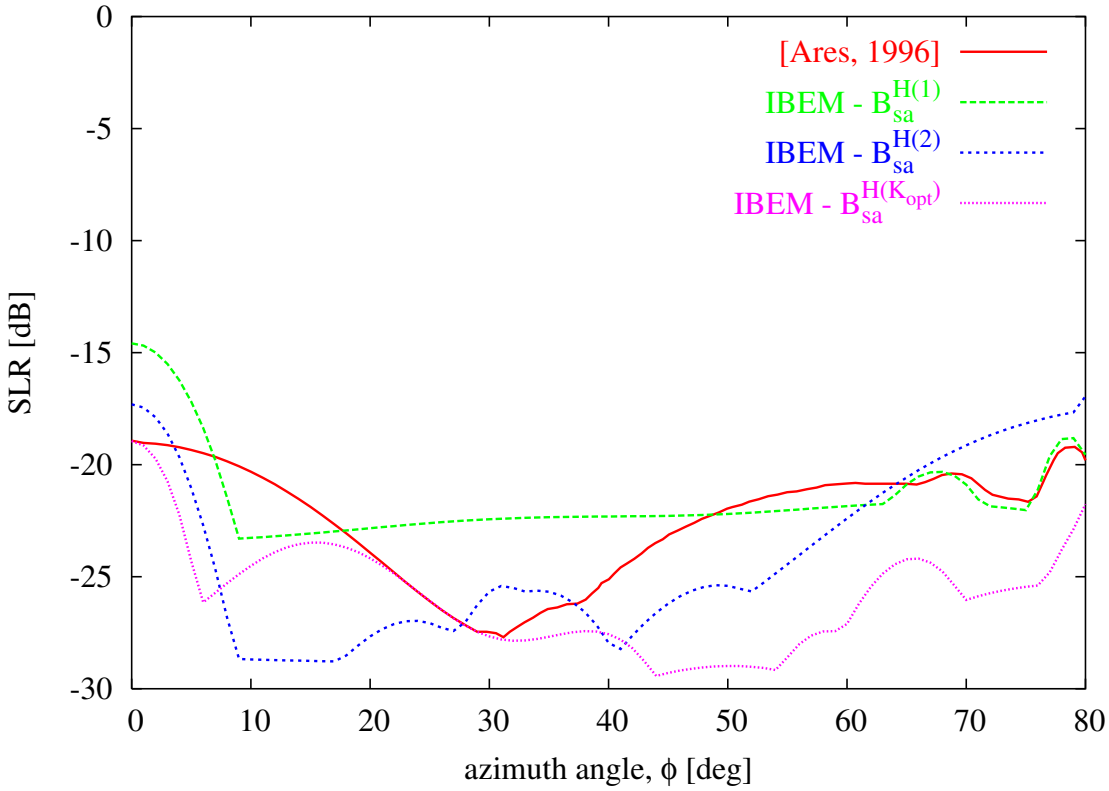
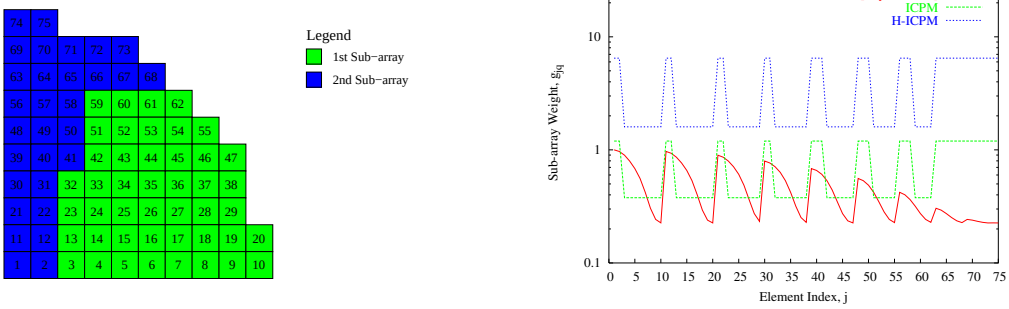
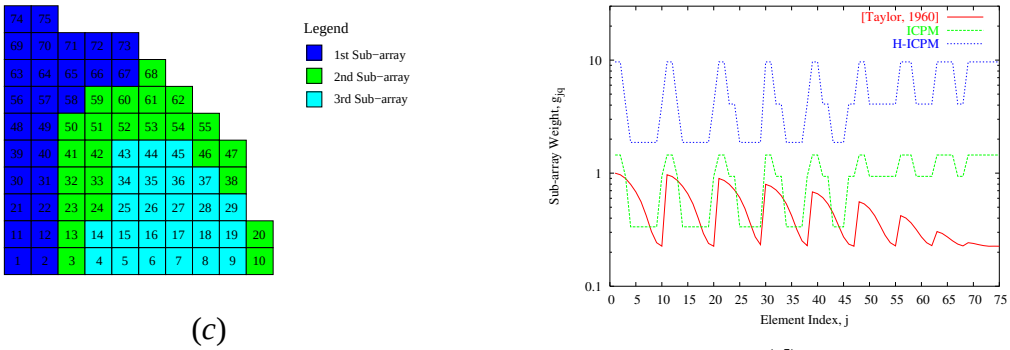


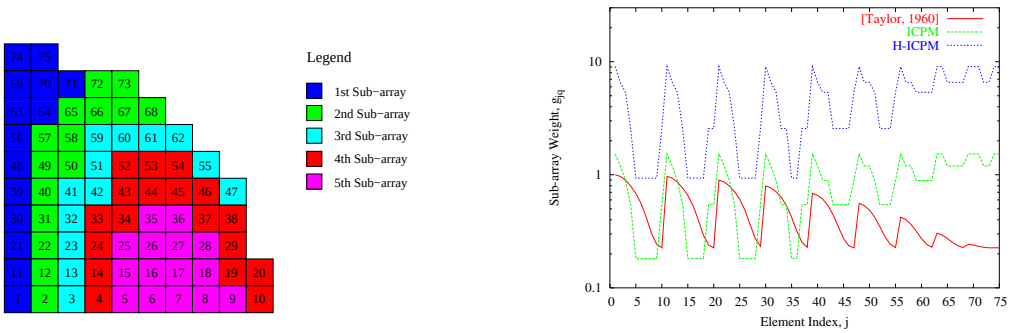
Fig. 14 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."



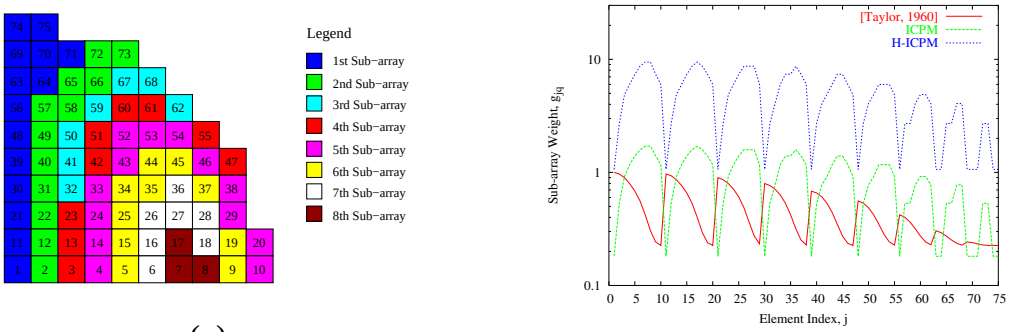
(*a*) (*b*)



(*c*) (*d*)



(*e*) (*f*)



(*g*) (*h*)

Fig. 15 - L. Manica *et al.*, “A Fast Graph-Searching Algorithm ...”

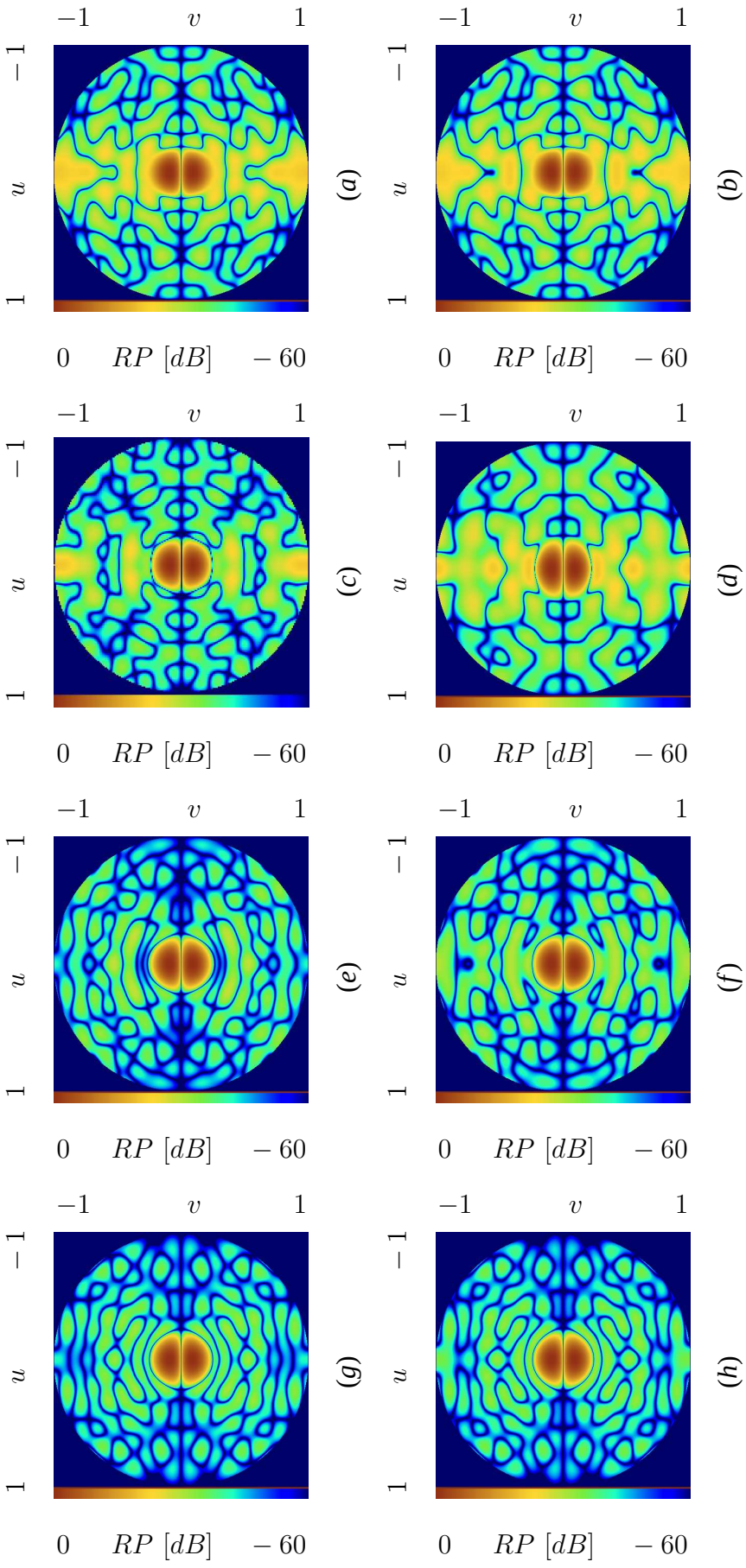
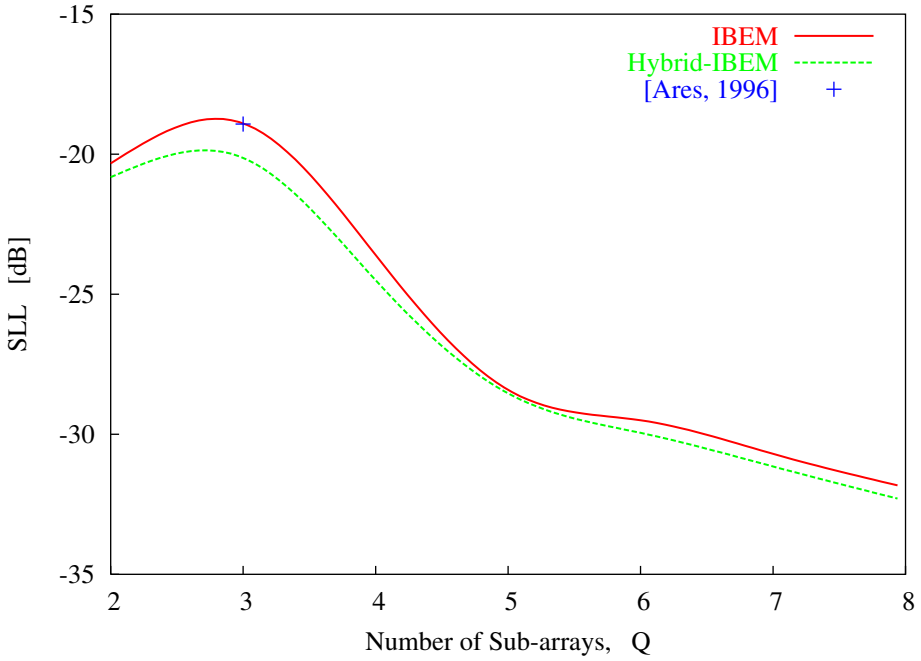
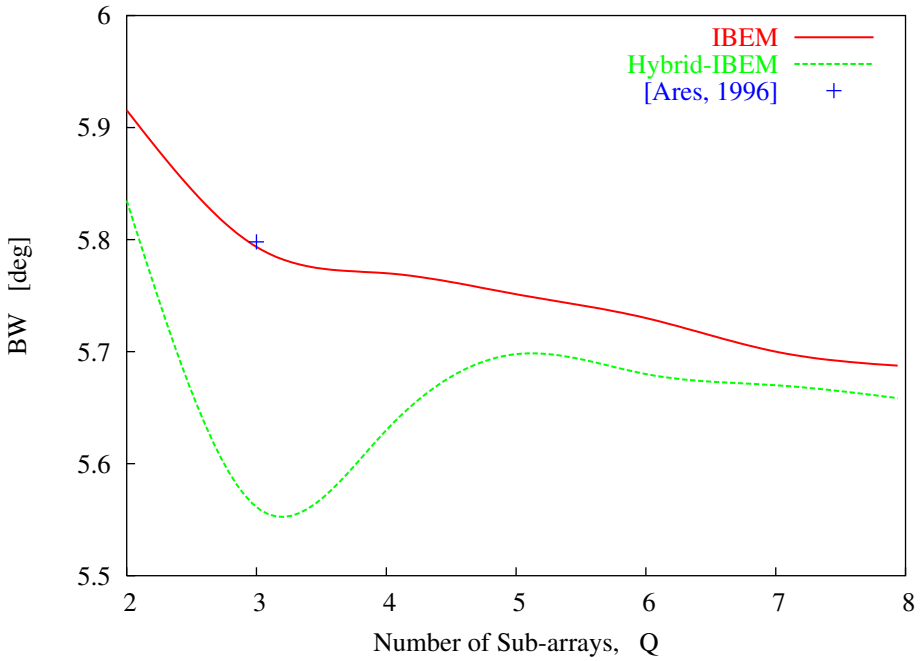


Fig. 16 - L. Manica *et al.*, “A Fast Graph-Searching Algorithm ...”



(a)



(b)

Fig. 17 - L. Manica *et al.*, "A Fast Graph-Searching Algorithm ..."

$\mathbf{w}_{\mathbf{opt}}^{\mathbf{H(k)}}$			
k	w_1	w_2	w_3
1	0.4668	1.3435	2.1736
2	0.3337	0.9763	1.6091
$K_{opt} = 3$	0.3355	0.9381	1.4469

Tab. I - L. Manica *et al.,* “A Fast Graph-Searching Algorithm ...”

	$CPU - Time\ [sec]$				T_{CP}			
Q	2	3	5	8	2	3	5	8
$IBEM$	2.30	2.64	3.12	7.23	37	45	57	120
$Hybrid - IBEM$	7554.68	8678.15	9623.57	7314.06	2114	2415	2675	2113