Efficient Crosspolar Optimization of Shaped-Beam Dual-Polarized Reflectarrays Using Full-Wave Analysis for the Antenna Element Characterization

Daniel R. Prado, Manuel Arrebola,
Rafael R. Boix,
and Fernando Las-Heras

Marcos R. Pino, Rafael Florencio,
José A. Encinar,

Abstract— A method for the optimization of the crosspo-
lar component of dual-polarized reflectarrays using full-wave
analysis at the element level is described and demonstrated.
The reflectarray full-wave analysis is based on local periodicity
(LP) and integrated within the optimization process in order
to accurately characterize the crosspolar far field. The proposed
method is based on the generalized Intersection Approach frame-
work using the Levenberg–Marquardt Algorithm as backward
projector, and the employed full-wave analysis is based on the
Method of Moments assuming Local Periodicity (MoM-LP).
Several strategies to accelerate the computations are exploited,
such as the parallelization of all the algorithm building blocks.
To minimize the impact of MoM-LP in the optimization process,
a strategy to reduce the number of MoM-LP calls is described,
further accelerating the algorithm. Moreover, the convergence
is improved by working with the squared field amplitude,
alleviating the trap problem of local optimizers. This method
allows to optimize the crosspolar component in the whole visible
region or only in the coverage zone to facilitate the convergence,
reducing computing time and memory usage. Two test cases are
provided to validate the technique, one with an isoflux pattern
for global Earth coverage and another with European coverage
for direct broadcast satellite application.

Index Terms— Crosspolar optimization, direct broadcast
satellite (DBS), full-wave analysis, generalized Intersection
Approach, isoflux, Levenberg–Marquardt algorithm (LMA),
reflectarray, shaped beam antenna.

I. INTRODUCTION

SINCE the first reflectarray antenna prototype based on
waveguide elements introduced in 1963 [1] and its popu-
larization in the 1980s due to the development of low-profile
printed antennas [2], reflectarrays have demonstrated to be a
feasible alternative to parabolic antennas, allowing to design
high gain antennas with similar efficiency. Printed reflectarrays
have also demonstrated their capabilities to produce con-
toured beams, for instance, for direct broadcast satellite (DBS)
applications [3], where a single-layer reflectarray was used.
However, single-layer reflectarrays present bandwidth limita-
tions, although they can be overcome by other types of broad-
band elements, such as several layers of stacked rectangular
patches [4], several parallel dipoles [5], [6], or other types [7].
Stacked rectangular patches were used to design reflectarrays
with enhanced bandwidth for DBS applications [8], [9] with
very stringent requirements.

The most common technique for the design of shaped-
beam reflectarrays consists in a phase-only synthesis (POS),
which can be carried out with several algorithms, such as
Intersection Approach [10], [11], Levenberg–Marquardt [12],
or particle swarm optimization [13], among others. Then, the
geometry of each cell is adjusted in order to fit the required
phase shift previously synthesized. This technique has proved
to be very effective to design reflectarrays that fulfill very
stringent requirements [8], [9], but since it is a POS, it has
no control over the crosspolar far field during the synthesis
process.

Some techniques for lowering the crosspolarization of
reflectarrays have been described, which rely on positioning
the elements in certain configurations [14] or by adjusting
the elements dimensions forcing a null in the amplitude and
a change of sign in the phase of the crosspolar reflection
coefficients [15]. However, these techniques present limitations
and cannot be used to synthesize shaped-beam reflectarrays.
In [16], an extension of the Intersection Approach algorithm
of [11] was developed to include crosspolar requirements in
the synthesis while keeping the efficiency of the algorithm
by using the fast Fourier transform (FFT) in both projectors.

The result was a distribution of reflection coefficient matrices. The reflectarray geometry should be adjusted in such a way that the element behavior matched the required synthesized matrix, which can be difficult to accomplish. In [17], a similar approach to [16] was used, adjusting the reflectarray element geometry to match the reflection coefficient matrix by applying certain geometrical transformations to the unit cell described in [7]. Another technique for the crosspolar optimization of reflectarrays has been explained in [18]–[20] where the Intersection Approach is used along with the Broyden–Fletcher–Goldfarb–Shanno (BFGS) optimization algorithm. However, only single-polarized reflectarrays are considered with very few elements (225 in total), since the algorithm was slow. Finally, in [21]–[23], a general optimization technique for printed reflectarrays is presented and used, which employs a gradient minimax algorithm for the direct optimization of the reflectarray geometry. It allows to synthesize both copolar and crosspolar components using several degrees of freedom. Nevertheless, it uses a lookup table of scattering parameters to speed up computations during the optimization process, which needs to be generated for each substrate and unit cell. However, as the number of optimizing variables is increased, the time required to generate the table rapidly increases as well. Furthermore, to create the table some variables, which are not needed during the optimization process (i.e., they are not optimized) are considered, such as the pair of incident angles, since for a given reflectarray geometry, varying the incident angle can modify the reflection coefficients [2]. Also, because interpolation is used to calculate the scattering parameters from the lookup table, accuracy decreases. This fact is important, since the crosspolar behavior is highly nonlinear and its absolute values are very low with regard to those of the copolar component.

In this paper, a new alternative technique for the crosspolar optimization of dual-polarized reflectarrays is presented. The reflectarray is modeled using a full-wave analysis based on local periodicity (LP) and the spectral domain method of moments LP (MoM-LP) [24]–[26] to analyze the antenna element. The full-wave analysis is integrated within the optimization process in order to accurately characterize the crosspolar far field, which is being optimized. The technique is based on the generalized Intersection Approach (generalized IA) framework of [27] and it uses the Levenberg–Marquardt Algorithm (LMA) as its backward projector. Several strategies are employed that allow the use of MoM-LP directly in the optimization process within acceptable computing time. These strategies comprise the parallelization of all the algorithm building blocks in order to scale well with the number of available processors, taking advantage of modern CPU units and workstations; and also strategies to minimize the number of MoM-LP calls. In order to improve the algorithm convergence, the squared field amplitude is used, which allows to obtain good results in less iterations in comparison when working with the complex field, as has been recently demonstrated in [28] where both approaches are compared. The algorithm can handle several thousands of optimizing variables, and they can be managed along with the number of far-field points to control computing time and memory usage. Two test cases
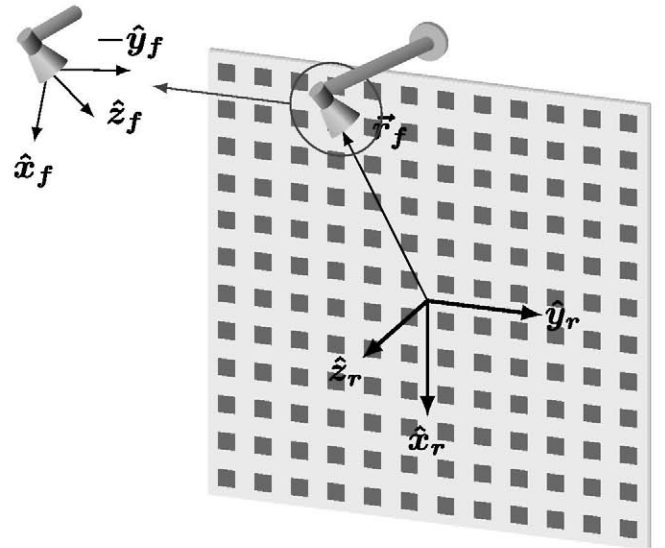

Fig. 1. Geometry of a single-offset printed reflectarray.

are provided (an isoflux pattern for global Earth coverage and a European coverage for DBS application), which show the capabilities of the developed technique, maintaining the copolar pattern shape within specifications while lowering the crosspolar levels for both polarizations at the same time.

This paper is organized as follows. Section II provides a brief overview of the reflectarray analysis and the computation of the far fields. Section III details the algorithm employed in the crosspolar optimization along with some techniques to improve its performance. Section IV is devoted to the two test cases to validate the presented algorithm. Section V discusses the results obtained in Section IV. Finally, Section VI contains the conclusions.

II. ANALYSIS OF THE REFLECTARRAY

A. Field in the Aperture

The scheme of an offset printed reflectarray is shown in Fig. 1. The reflectarray is illuminated by a feed, which generates an incident electric field, $\vec{E}_{\text{inc}}$, on the surface of the reflectarray. This tangential field can be expressed for both polarizations in compact notation as

$$\vec{E}_{\text{inc}}^{X/Y}(x, y) = E_{\text{inc},x}^{X/Y}(x, y)\,\hat{x} + E_{\text{inc},y}^{X/Y}(x, y)\,\hat{y} \qquad (1)$$

where the superscripts indicate the feed polarization and the subscripts the component of the field with regard to the reflectarray coordinate system (e.g., E_x^Y would be the $\hat{x}$-component of the projected field over the reflectarray surface, in the reflectarray coordinate system, when the feed radiates in Y polarization, in the feed coordinate system). Note that each polarization has, in general, two components of incident field, the desired and the crosspolar. The crosspolar component of the incident field appears, since the feed is not ideal and also due to the projection of the field radiated by the feed onto the reflectarray surface. In a similar way, the reflected field on the reflectarray surface can be written as

$$\vec{E}_{\text{ref}}^{X/Y}(x, y) = E_{\text{ref},x}^{X/Y}(x, y)\,\hat{x} + E_{\text{ref},y}^{X/Y}(x, y)\,\hat{y}. \qquad (2)$$

The relation between the incident and reflected fields at each element (m,n) of the reflectarray is given by a matrix of reflection coefficients that characterize that element

$$\vec{E}_{\text{ref}}^{X/Y}(x_m, y_n) = \mathbf{R}^{mn} \cdot \vec{E}_{\text{inc}}^{X/Y}(x_m, y_n) \tag{3}$$

where

$$\mathbf{R}^{mn} = \begin{pmatrix} \rho_{xx}^{mn} & \rho_{xy}^{mn} \\ \rho_{yx}^{mn} & \rho_{yy}^{mn} \end{pmatrix} \tag{4}$$

and (x_m, y_n) are the coordinates of the (m,n)th element.

The components of $\mathbf{R}^{mn}$ are complex numbers and fully characterize the behavior of the cell. This method of analysis considers three sources of crosspolarization: the feed, the projection of the incident field onto the surface of the reflectarray, and the elements through the use of $\mathbf{R}^{mn}$. The reflection coefficient matrix for each element is computed using MoM assuming LP (i.e., it embeds each element in an infinite array comprised of the same element) [24]–[26] and it approximately considers the losses due to the substrate, mutual coupling between elements, the field reradiated by the element (metallizations), and reflected by the substrate and ground plane. Out of the three crosspolarization sources, the $\mathbf{R}^{mn}$ matrix is the most important for the crosspolar optimization, since both the feed and incident field projection contributions remain fixed during the optimization, while the $\mathbf{R}^{mn}$ matrix changes for each element at each iteration of the algorithm. Hence, a correct characterization of $\mathbf{R}^{mn}$ using a full-wave technique based on LP, such as MoM-LP, during the optimization process is required for a correct prediction of the crosspolar far field. The LP assumption can be improved by an extended LP, which also considers the real surrounding elements [29], [30], although the extra computational burden introduced by this technique might not be suitable for optimization purposes [29]. In any case, in this paper, MoM-LP [24]–[26] will be used to compute $\mathbf{R}^{mn}$. This technique [25] has already been demonstrated by full-wave simulations of the whole antenna [31] and measurements [32], showing good agreement both in copolar and crosspolar patterns.

On the other hand, the tangential reflected magnetic field is computed at each reflectarray element assuming a locally incident plane wave coming from the feed, using the following expression:

$$\vec{H}_{\text{ref}}^{X/Y}(x_m, y_n) = \frac{\vec{k}_{\text{ref}} \times \vec{E}_{\text{ref}}^{X/Y}(x_m, y_n)}{\omega \mu_0} \tag{5}$$

with $\omega = 2\pi f_0$ and the components of $\vec{k}_{\text{ref}}$

$$k_x = -k_0 \sin\theta^{mn} \cos\varphi^{mn}$$
$$k_y = -k_0 \sin\theta^{mn} \sin\varphi^{mn}$$
$$k_z = +k_0 \cos\theta^{mn} \tag{6}$$

where $(\theta^{mn}, \varphi^{mn})$ is the incident angle of the plane wave for the (m,n)th element. $\vec{k}_{\text{ref}}$ corresponds to the reflected propagative wave in the specular reflection direction, in the absence of grating lobes.

In order to compute the cross product in (5), all three components of the electric field are needed, but (3) only

provides two, namely, E_x and E_y. The E_z-component can be obtained through the plane wave relation

$$\vec{k}_{\text{ref}} \cdot \vec{E}_{\text{ref}} = 0. \tag{7}$$

Solving (7), it immediately follows that:

$$E_{\text{ref},z} = \frac{-k_{\text{ref},x} E_{\text{ref},x} - k_{\text{ref},y} E_{\text{ref},y}}{k_{\text{ref},z}} \tag{8}$$

which allows to solve (5), obtaining the reflected tangential magnetic field on the reflectarray surface.

B. Far-Field Patterns

Once the tangential electric field has been obtained using (3), the radiated far fields can be efficiently computed using the FFT algorithm [2]. It has been demonstrated [32], [33] that the use of the First Principle of Equivalence allows a correct characterization of both the copolar and crosspolar components, in contrast to the Second Principle, which might not give accurate predictions for the crosspolar far field. The radiated far field by a planar aperture using the First Principle of Equivalence is [34]

$$E_\theta^{X/Y} = \frac{jk_0 e^{-jk_0 r}}{4\pi r} \left[P_x^{X/Y} \cos\varphi + P_y^{X/Y} \sin\varphi \right.$$
$$\left. - \eta \cos\theta \left(Q_x^{X/Y} \sin\varphi - Q_y^{X/Y} \cos\varphi \right) \right]$$

$$E_\varphi^{X/Y} = -\frac{jk_0 e^{-jk_0 r}}{4\pi r} \left[\eta \left(Q_x^{X/Y} \cos\varphi + Q_y^{X/Y} \sin\varphi \right) \right.$$
$$\left. + \cos\theta \left(P_x^{X/Y} \sin\varphi - P_y^{X/Y} \cos\varphi \right) \right] \tag{9}$$

where $\eta = \mu_0 c_0$ is the vacuum impedance and P_x, P_y, Q_x, and Q_y are the spectrum functions, which can be calculated as the Fourier transforms of the tangential electric and magnetic fields in the aperture as

$$P_{x/y}^{X/Y}(u,v) = \iint_S E_{\text{ref},x/y}^{X/Y}(x,y) e^{jk_0(ux+vy)} dx \, dy$$

$$Q_{x/y}^{X/Y}(u,v) = \iint_S H_{\text{ref},x/y}^{X/Y}(x,y) e^{jk_0(ux+vy)} dx \, dy \tag{10}$$

with $u = \sin\theta \cos\varphi$, $v = \sin\theta \sin\varphi$ and S the reflectarray surface, which is already discretized into a regular grid of periodicity $a \times b$. Please note that with the compact notation in (10), there are two spectrum functions per polarization, making a total of eight considering both fields. Considering the tangential fields constant within each element (m,n), the spectrum functions take the form

$$P_{x/y}^{X/Y}(u,v) = K \cdot \sum_m \sum_n E_{\text{ref},x/y}^{X/Y}(x_m, y_n) e^{jk_0(ux_m+vy_n)}$$

$$Q_{x/y}^{X/Y}(u,v) = K \cdot \sum_m \sum_n H_{\text{ref},x/y}^{X/Y}(x_m, y_n) e^{jk_0(ux_m+vy_n)} \tag{11}$$

being K a constant

$$K = a \, b \, \text{sinc}\left(\frac{k_0 u a}{2}\right) \text{sinc}\left(\frac{k_0 v b}{2}\right) \tag{12}$$

that accounts for the amplitude element pattern (when the array factor approximation is used, $K = 1$ [35]). The double sum

in (11) can be written as a 2-D inverse discrete Fourier transform (IDFT2), which can be efficiently computed by a 2-D Inverse Fast Fourier Transform [2]. In that case, the spectrum functions are calculated as

$$P_{x/y}^{X/Y}(u,v) = N \cdot K \cdot \text{IDFT2}\big[E_{\text{ref},x/y}^{X/Y}(x_m, y_n)\big]$$

$$Q_{x/y}^{X/Y}(u,v) = N \cdot K \cdot \text{IDFT2}\big[H_{\text{ref},x/y}^{X/Y}(x_m, y_n)\big] \quad (13)$$

where N is the total number of reflectarray elements.

Once the far field has been obtained in spherical coordinates using (9), the copolar (E_{cp}) and crosspolar (E_{xp}) radiation patterns given by Ludwig's third definition [36] are obtained for both polarizations. The gain (G_{cp} and G_{xp}) is estimated by computing the total power radiated by the feed [2] (input power), which is usually modeled with a $\cos^q \theta$ function [37]. The input power remains constant through the synthesis. This method of analysis considers the spillover and illumination efficiencies as well as substrate losses [2].

The far-field pattern specifications can be given in form of templates with top and bottom bounds [2], [10], [12], $T_{\text{max}}(u,v)$ and $T_{\text{min}}(u,v)$, which can be defined in the whole UV grid or in the area of interest. These templates should be provided for copolar and crosspolar patterns for both polarizations and are usually given in gain [2]. In this regard, the phase variation in the far field is not of concern, even within the main beam, since the developed technique is for broadcast applications only. The control of the radiated far-field phase could be of interest in some synthesis processes, since the spatial dispersion produces different delays in receptors situated at different points within the shaped region. However, this fact does not affect broadcast applications, and thus, far-field phase variation over the coverage zone is not critical.

III. CROSSPOLAR SYNTHESIS BASED ON IA-LMA

In this section, a description of the generalized IA algorithm using the LMA as backward projector is provided. It directly optimizes the geometry of the elements using MoM-LP as analysis tool. For this reason, a special effort is made in order to accelerate the algorithm, implementing several strategies in the LMA. They are aimed to reduce the impact of MoM-LP operations through minimizing its use and parallelizing all the building blocks of the algorithm.

The generalized IA considers two sets, the set of the radiation patterns that can be obtained with the reflectarray optics (set $\mathcal{R}$) and the set of radiation patterns that comply with the specifications (set $\mathcal{M}$). At each iteration i of the algorithm, the following operation is performed:

$$\vec{E}_{i+1} = \mathcal{B}[\mathcal{F}(\vec{E}_i)] \quad (14)$$

where $\vec{E}$ is the radiated far field by the antenna. $\mathcal{F}$ is the forward projector, which projects the far field radiated by the antenna (belonging to the $\mathcal{R}$ set) onto the set of fields that comply with the specifications ($\mathcal{M}$ set). Conversely, $\mathcal{B}$ is the backward projector, which projects the far field that fulfills the specifications onto the set of far fields that can be radiated by the reflectarray. The goal of (14) is to find a radiation pattern that belongs simultaneously to both sets or, if that is

not possible, to find a radiation pattern whose distance to the set of patterns that fulfill the requirements is minimal. Also, the generalized IA is a local optimizer and as such the starting point is of the utmost importance, since it will determine the correct converge of the algorithm [27].

The potential local minima reached by local optimizers are commonly known as traps. In the generalized IA framework, there are two main sources of traps [27]: the number of degrees of freedom, which in this case will be the number of optimizing variables; and the nonconvexity of the involved sets. The latter problem can be addressed by working with the squared field amplitude instead of the field itself. The gain is also suitable, since it is proportional to the squared field amplitude

$$G \propto |\vec{E}|^2 \quad (15)$$

and the specifications are usually given in gain. The proportionality factor between the gain and squared field amplitude remains constant through the synthesis, so it does not change the degree of nonlinearity of the optimization problem. Also, as stated before, the reflectarray is analyzed, considering approximately the losses by means of a full-wave analysis of the unit cell and also the illumination efficiency and spillover [2]. Using the gain or squared field amplitude improves the convergence of the algorithm, since it convexifies the set $\mathcal{M}$ of radiation patterns that comply with the specifications (this set is denoted as $\mathbf{Y_C}$ in [27], where $\mathbf{Y_C}$ is defined as convex). However, this leads to a redefinition of the backward projector in [11] and [16], since the FFT cannot be used anymore to recover the field at the aperture. This causes the algorithm to lose the efficiency derived from the FFT being the most time-consuming operation. Now, to recover the field at the aperture, an optimization algorithm is needed, which in the case of the present work is the LMA.

For the first problem, the solution is to reduce the number of variables in the first steps of the synthesis process. This has been implemented in the reflectarray POS case with a fictitious reduction of variables modifying the illumination taper of the feed [11]. The idea behind the fictitious reduction of variables is that an array with a very low illumination at the edges is equivalent to using a smaller array, since the outer array elements barely contribute to the far fields, thus effectively reducing the number of variables, although, in reality, the number of variables is the same (hence the *fictitious* reduction) [11]. Nevertheless, this technique is not used here, since the reflectarray geometry is optimized with the LMA, and the reduction of the number of variables is not fictitious anymore, since one can choose which variables will be optimized and which will not.

In order to alleviate the notation, the compact notation for both polarizations will be avoided and generic gains and templates will be used when possible.

A. Forward Projection

The forward projector is divided in two steps, the first one being optional. In the first step, the templates are normalized to the value of the gain in a given direction. Then, the fields

are trimmed using those normalized templates, obtaining a far field, which complies with the specifications, but cannot, in general, be radiated by a passive reflectarray.

The requirements of the copolar radiation pattern are usually given in gain, with a maximum and minimum value. If these templates remain unaltered (i.e., the first step is skipped), the synthesis is carried out in *fixed gain*. This method can be useful to further refine a previous synthesis. Alternatively, they can be normalized to the gain in a given (u_0, v_0) direction, which belongs to the maximum gain region. It is then said that the synthesis is carried out in *float gain* [11]. This method is more useful when a synthesis is started from scratch (e.g., when doing POS), since it adapts automatically the templates to the maximum gain that can be obtained with the reflectarray. In this case, a normalizing constant is defined as

$$C_n = \frac{G_{cp}(u_0, v_0)}{T_{av}} \tag{16}$$

where T_{av} is the average value of the maximum and minimum copolar templates in the (u_0, v_0) direction

$$T_{av} = \frac{T_{cp,max}(u_0, v_0) + T_{cp,min}(u_0, v_0)}{2}. \tag{17}$$

C_n is defined for both polarizations independently to adapt the template to the current gain of each polarization (compact notation for both polarization was avoided to alleviate the notation, but C_n^X and C_n^Y are obtained using the gains and templates from both polarizations). Then, it is applied to both copolar and crosspolar templates as

$$T_{cp}^n(u, v) = T_{cp}(u, v) \cdot C_n$$
$$T_{xp}^n(u, v) = T_{xp}(u, v) \cdot C_n. \tag{18}$$

The normalization in (18) is applied to the minimum and maximum templates of both polarizations. This normalization process sets the middle point of the templates at (u_0, v_0) at the same level of the gain in that direction.

Once the templates have been normalized, both components of the radiated field should accomplish the following condition in both polarizations:

$$T_{cp,min}^n(u, v) \leq G_{cp}(u, v) \leq T_{cp,max}^n(u, v)$$
$$T_{xp,min}^n(u, v) \leq G_{xp}(u, v) \leq T_{xp,max}^n(u, v). \tag{19}$$

This condition can be fulfilled by the P_r operator, defined for all (u, v) as follows for a generic gain G:

$$P_r(G) = \begin{cases} T_{max}^n(u, v), & T_{max}^n(u, v) < G(u, v) \\ T_{min}^n(u, v), & G(u, v) < T_{min}^n(u, v) \\ G(u, v), & \text{otherwise.} \end{cases} \tag{20}$$

The result of this operation is the gain G'

$$G'(u, v) = P_r(G(u, v)). \tag{21}$$

For the sake of simplicity, a generic G gain component and normalized template T^n were used in (20) and (21), but they should be applied to the copolar and crosspolar components of both polarizations with their respective normalized templates. Also, the forward projector only works with the gain (which is proportional to the squared field amplitude), and the far-field

phase does not play any role, being able to take any value. In contrast to [11] and [16], where the far-field phase remained unchanged, now there is no such constraint, improving the convergence of the algorithm.

B. Backward Projection

Since the forward projection works with the squared field amplitude (gain) instead of the field itself, the FFT to recover the field at the aperture cannot be used, since for the use of the FFT, the complex field (amplitude and phase) is needed to perform the inverse operation of (13), as in [10], [11], and [16]. In any case, since the goal is to directly optimize the geometry of the reflectarray to improve the crosspolar far field, an optimization algorithm will be used. Now, the $\mathcal{B}$ operator is defined as the minimization of the distance of an element $m \in \mathcal{M}$ to the set $\mathcal{R}$

$$\text{dist}(m, \mathcal{R}). \tag{22}$$

For the minimization of the distance in (22), at this stage, we have the element m as the trimmed gain $G'(u, v)$ and the current reflectarray geometry, which generates a gain pattern that belongs to the $\mathcal{R}$ set, $G(u, v)$. As a distance definition, the Euclidean norm for square-integrable functions can be used, which can be easily implemented by the weighted Euclidean metric

$$d = \text{dist}^2(G'(u, v), G(u, v))$$
$$= \iint_\Omega w(u, v)(G'(u, v) - G(u, v))^2 du\, dv \tag{23}$$

where $w(u, v)$ is a weighting function and Ω is the area belonging to the visible region $(u^2 + v^2 \leq 1)$ where the far fields are optimized. This area can be the whole visible region or located around a more limited zone, and can be used as a way of controlling memory and computing time resources allocated in the synthesis (i.e., a bigger Ω area means more allocated computer memory and slower computations). Since the radiated fields are already discretized in the UV grid, the integral in (23) can be approximated by a sum for the UV points, which lie in Ω

$$d = \sum_{u,v} w(u, v)(G'(u, v) - G(u, v))^2 \Delta u\, \Delta v$$
$$= \sum_{u,v} [C(u, v)(G'(u, v) - G(u, v))]^2 \tag{24}$$

where

$$C(u, v) = \sqrt{w(u, v)\Delta u\, \Delta v} \tag{25}$$

and Δu and Δv are the steps in the discretized UV grid in u and v, respectively.

On the other hand, the LMA minimizes cost functions of the form [12]

$$F(x) = \sum_{i=1}^{M} (r_i(x))^2 \tag{26}$$

where $r(x)$ is known as residual, which is discretized in M points. Comparing (24) with (26), it is clear that

$$r(u, v) = C(u, v)(G'(u, v) - G(u, v)) \tag{27}$$

so the LMA naturally minimizes the distance between the two gain patterns.

The generic gains G and G' represent the copolar and crosspolar components for both polarizations, which are all optimized at the same time. Each far-field component is discretized into M points, hence having a total of $4M$ points in the UV grid. Also, since the relative value of the crosspolar component is significantly lower than the copolar gain, in order to properly reduce its starting value, it might be convenient to scale the crosspolar residual by means of the weighting function contained in $C(u, v)$. Finally, it is not necessary to attain a minimum (generally local) of (22) [27], only to decrease the distance at each iteration of the generalized IA, so very few iterations of the LMA are needed.

C. Minimization Algorithm for the Geometry Optimization

As stated in Section III-B, due to the similarity between (24) and (26), the LMA is a natural choice to minimize the distance in (22), and in fact will be the minimizing algorithm used here. However, other gradient-based algorithms are also suitable, such as steepest descent [38], Gauss–Newton [39], or self-scaled BFGS [40] by setting up an appropriate cost function [28]. The strategies devised to accelerate the computations of the LMA described below are also suitable for those algorithms, and in particular, also for the self-scaled BFGS, which has been used before in the context of reflectarray optimization [18]–[20], [28]. One advantage of the BFGS over the LMA is the fact that the BFGS does not perform a matrix multiplication (only matrix-vector multiplications) and it can also directly work with the inverse of the Hessian, so it does not need to solve an equation system. These two points allow the BFGS to save time with those operations. However, it presents the disadvantage that it must perform further cost function evaluations using the full-wave analysis tool in a line search to choose a proper step length [39], which wastes the time saved by the matrix multiplication and equation system solver. Moreover, the BFGS updating formula is considered the most effective quasi-Newton formula [39] when working in a standalone mode (i.e., not as part of a more general framework as in the present case). For the first iteration, the identity matrix is usually employed, which makes the first step equivalent to that of the steepest descent [39]. However, since only a few iterations of the gradient-based algorithm are necessary per iteration of the generalized IA [27], the Hessian approximation is continuously restarted and the convergence improvement of the BFGS due to the updating formula is lost. This is not a problem for the LMA, which uses always the same Hessian approximation and whose convergence is controlled by a real positive number. In any case, in this context is the generalized IA the algorithm controlling convergence [27] and the choice between the LMA and BFGS may have little impact in this fact.

The LMA described in [12] is used here. It includes a number of strategies in order to improve the computing times of the algorithm. In short, the computation of the Jacobian matrix is parallelized, computing one column per available thread and minimizing the error of the finite difference by an appropriate choosing of its step (which depends on the machine epsilon [39]) to evaluate derivatives; the matrix-vector and matrix multiplications are performed by the Intel Math Kernel Library (MKL) library [41], which takes advantage of highly optimized, fully parallelized algorithms, and low-level hardware operations in order to improve their performance and computing time; a Cholesky factorization-based solver is used, which is the fastest exact solver for this type of problems (also from the MKL library). Also, the result of a big matrix multiplication involving the Jacobian is symmetric, so only a triangular part of it is computed, with a further reduction of computation time. However, in [12], the LMA was used to perform POS, and since now, the crosspolar far field is optimized by directly analyzing the reflectarray geometry with MoM-LP, more efficient strategies are included in the cost function and the Jacobian matrix evaluation (which are the building blocks calling the MoM-LP routine).

1) Cost Function: For the POS case, the most time-consuming operation when computing the far fields from the tangential field of the reflectarray is the FFT. However, now that the reflectarray is analyzed with MoM-LP, the evaluation of the tangential field is slower, since an MoM-LP call is required per reflectarray element. However, since the evaluation of a reflectarray element is independent of the rest (since the analysis assumes LP), this operation can be easily parallelized using OpenMP [42]. This way, the evaluation of the tangential field was sped up, and can take advantage of modern CPU units, scaling well with the number of available processors. This parallelization of the cost function routine only applies when it is called the first time at the beginning of each LMA iteration, and not from the Jacobian evaluation routine, where another kind of parallelization is used.

2) Jacobian Evaluation: Despite the improvement in the Jacobian evaluation described in [12], it is still quite slow when using MoM-LP in the analysis of the reflectarray. In order to reduce the number of calls to the MoM-LP routine, first, a one-sided-difference is used in the evaluation of each column of the Jacobian, instead of the central difference. This reduces the calls to the cost function (and hence the calls to the MoM-LP routine) by half.

Still, when calling the cost function in order to evaluate the lateral difference, all elements of the reflectarray are processed with MoM-LP. Assuming that all reflectarray elements are optimized by the LMA, the Jacobian matrix has sN columns (where N is the number of reflectarray elements and s is the number of optimizing variables per reflectarray element), and considering MoM-LP as the dominant operation, the overall time cost of evaluating the full matrix would be

$$\mathcal{O}\left(O_{\text{MoM-LP}} \cdot s \cdot N \cdot N\right) \tag{28}$$

with

$$O_{\text{MoM-LP}} = \mathcal{O}\left(k_1 Z^2 + k_2 Z^3\right) \tag{29}$$

where Z is the number of unknowns in MoM-LP, $k_1 Z^2$ is the term for filling the MoM-LP matrix, and $k_2 Z^3$ for the inversion. In the present case, five basis functions per dipole are used (see Section IV for a description of the unit cell),

thus having a 40 × 40 MoM-LP matrix, which is very fast to invert. However, in this case, the bottleneck is the matrix filling, which is much slower than the inversion [25]. If L is the number of available processors, since the evaluation of the Jacobian matrix is massively parallelizable due to the fact that each column can be independently computed [12], computing times derived from (28) can be divided up to L times, considering a 100% parallelization efficiency, although in reality gains will be smaller.

The time cost in (28) is not suitable for practical implementation. If analyzing one element with MoM-LP takes an average time of 0.1 s and the reflectarray has $N = 900$ elements with $s = 6$ optimization variables per element, the time to evaluate the Jacobian matrix would be of the order of 135 h (5.6 days) for $L = 1$ and 13.5 h for $L = 10$. If the reflectarray has roughly 7000 elements as in [9], the time would escalate to 340 and 34 days for $L = 1$ and $L = 10$, respectively. The computing times shown as described earlier are only a coarse approximation of the time cost to evaluate the Jacobian matrix, and illustrate the necessity of improving (28).

In light of these results, it is obvious that the times involved in the Jacobian matrix evaluation are of no practical use. However, they can be greatly reduced by further minimizing the number of MoM-LP calls. Each column of the Jacobian is obtained by a single call to the cost function, which returns a vector with all the components of the column. Furthermore, each column is calculated by deriving the cost function with respect to just one variable, resulting in just one element being modified. Hence, there is no need to recompute $\mathbf{R}^{mn}$ for all elements using MoM-LP, but just one element per Jacobian column, reusing the rest of the reflection coefficients from the first call to the cost function at the beginning of each iteration of the LMA. This can be easily implemented by branching the cost function routine, detecting when it is used to evaluate the Jacobian matrix, and then computing only one element with MoM-LP. With this improvement, the new time cost of evaluating the Jacobian matrix is

$$\mathcal{O}(O_{\text{MoM-LP}} \cdot s \cdot N). \tag{30}$$

Using the same examples as before, the new improved computing times would be 9 and 70 min for $L = 1$, which represent an important reduction from the previous 5.6 and 340 days, respectively.

With these two optimizations in the computation of the cost function and the Jacobian matrix, apart from others introduced in [12], the LMA is able to directly optimize the geometry of large reflectarrays using MoM-LP for the crosspolar optimization in a reasonable amount of time, since the algorithm will scale well with the number of available processors, taking advantage of farm servers.

Finally, the implemented algorithm is able to handle thousands of optimizing variables obtaining good results due to its improved convergence properties derived from the use of the squared field amplitude and the elimination of the far-field phase constraint (which now can take any value), provided a suitable starting point. Also, in order to control computing times and memory usage, the number of optimizing variables can be reduced as well as the number of points in the
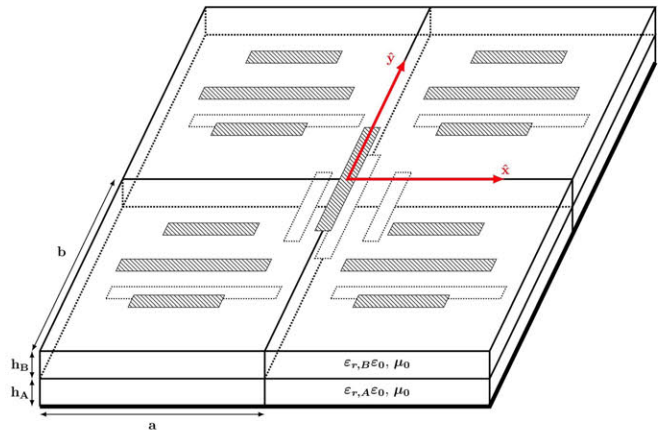

Fig. 2. Employed reflectarray unit cell based on parallel and coplanar dipoles in two layers of metallization.

UV grid where the optimization is performed, carrying it out, for instance, only around the coverage area instead of in the whole visible region.

IV. VALIDATION

In order to validate the algorithm described in Section III-C2, two shaped patterns for satellite applications are optimized in order to lower their crosspolar levels while maintaining the copolar pattern within specifications. The patterns are an isoflux pattern for global Earth coverage [43] and a European coverage for DBS application [44]. In both cases, a POS is done using the IA [11], and then, a design is obtained following [2], using the reflectarray element shown in Fig. 2. The unit cell is comprised of two layers of metallizations with four parallel and coplanar dipoles for each polarization, and is described in detail in [32].

These two designs are used as starting point for the crosspolar optimization. The copolar template is the same as the one used in the POS, and the crosspolar template is defined as a certain constant value below the maximum copolar level, for the whole region where the optimization is carried out.

A. Isoflux Pattern

The first test case is an isoflux pattern for global Earth coverage. Reference [43] details how to obtain a parametrized isoflux pattern, which in this case has been particularized for $\beta = 20°$, a geostationary orbit, side lobe level 19 dB below the maximum copolar level, and 0.35 dB of allowable ripple in the coverage area.

The reflectarray is circular, with 1020 total elements placed in a rectangular grid with 36 elements along the diameter in both reflectarray axes. The feed horn is modeled as a $\cos^q \theta$ function [37] with a q-factor of 14.8, which provides an illumination taper of -12 dB at the reflectarray edges. The feed horn is placed at $\vec{r}_f = (40, 0, 195)$ mm with regard to the reflectarray center (see Fig. 1). The working frequency is 30 GHz and the period of the unit cell is 5 mm × 5 mm (0.5λ). The substrate for both layers of the unit cell is the same,
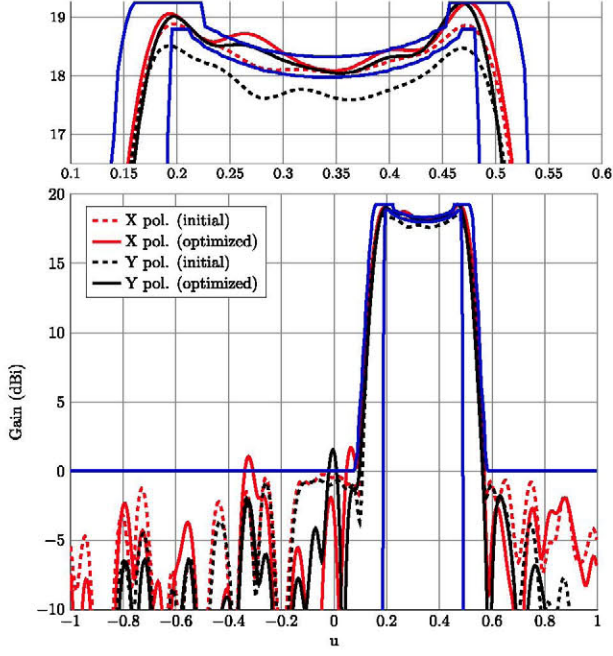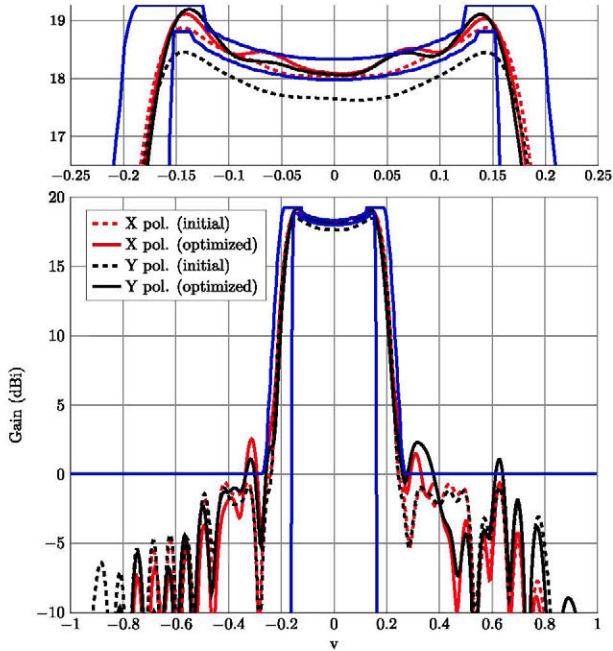
Fig. 3. Main cuts for the copolar pattern for both polarizations before and after the crosspolar optimization. The copolar gain for Y polarization is increased during the optimization. (a) Cut along u for $v = 0$. (b) Cut along v for $u = 0.34$.

with a height of $0.787\,\mathrm{mm}$ and a complex relative permittivity $\epsilon_r = 2.33 - j3.029 \cdot 10^{-3}$.

The optimization is carried out in the whole visible region, with a resolution of the far fields of 256×256 points for the FFT, having a total of $51\,543$ points in the UV grid. All of the reflectarray elements are optimized at the same time, considering all dipoles as independent optimizing variables ($s = 8$), so the total number of optimizing variables
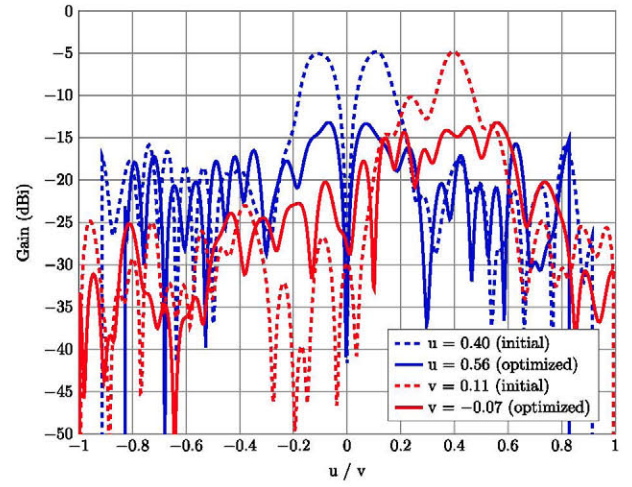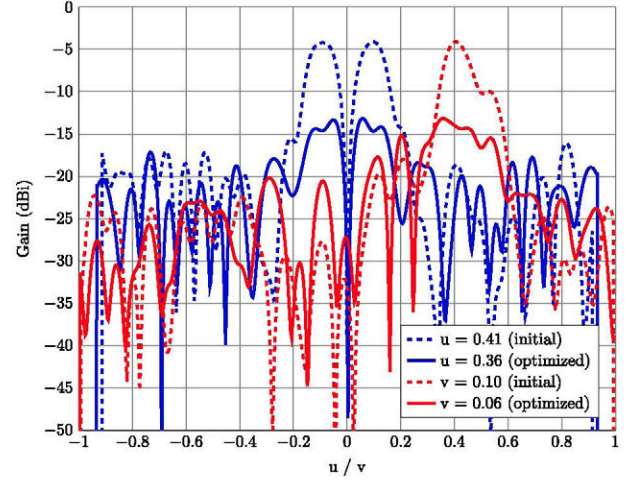
Fig. 4. Cuts along u and v of the crosspolar far field before and after the optimization. The cuts are taken at the point where the crosspolar level is maximum for each pattern. (a) X polarization. (b) Y polarization.

is 8160. The optimization is carried out in fixed gain, setting the required gain to $18.35\,\mathrm{dB}$ in the center of the coverage zone. The crosspolar template is set $35\,\mathrm{dB}$ below the maximum copolar gain template, and the crosspolar component is scaled by a factor of 100 in natural units with the weighting function. The LMA is set to perform three iterations per iteration of the generalized IA.

The results obtained after the optimization are compared with the starting point in Figs. 3 and 4. On the one hand, Fig. 3 shows the results for the copolar pattern. There were two goals: to maintain the copolar specification (isoflux shape) while reducing the crosspolar component, and to increase the gain for Y polarization while maintaining the X polarization gain. As it can be seen, the starting point has around $0.4\,\mathrm{dB}$ less in gain for Y polarization than the other in the coverage area. However, after the optimization, the gains for both polarizations become similar while mostly preserving the copolar shape within specifications.

On the other hand, Fig. 4 shows some cuts in (u, v) of the crosspolar patterns. The cuts correspond to the points were the
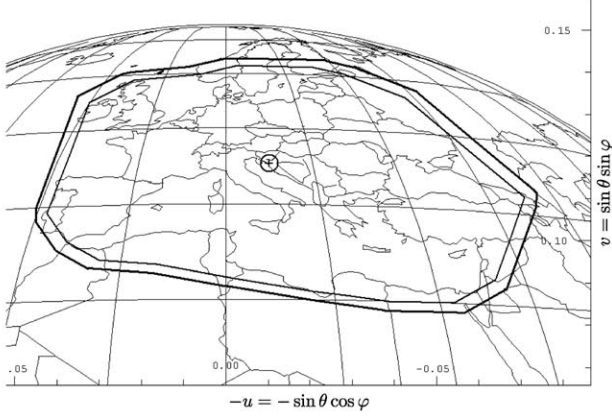
Fig. 5. European coverage. (u, v) are in the antenna coordinate system.

(a)

(b)

Fig. 6. Starting European coverage radiation pattern in gain (dBi) for Y polarization after POS. (u, v) are in the reflectarray coordinate system. (a) Copolar. (b) Crosspolar.

crosspolar is maximum, for both polarizations, even though they are not the same for the patterns before and after the optimization. For the X polarization, the initial maximum crosspolar level is -4.81 dBi, and is reduced by 8.36 dB to -13.17 dBi after the optimization. For the Y polarization, initially there is a maximum level of -4.12 dBi for the crosspolar component and is reduced an amount of 9.05 dB to -13.17 dBi. This important crosspolar reduction is achieved while keeping the gain of the copolar pattern (and slightly increasing it in the case of Y polarization) and its shape. The tradeoff is a slight increase in the side lobes, as seen in Fig. 3 and a redistribution of the lobes in the crosspolar component, which is now more uniform toward higher levels in the whole UV grid.

B. European DBS Coverage

The second test case corresponds to a European DBS coverage shaped-beam, as shown in Fig. 5 [44]. The working frequency is 11.85 GHz, and the satellite is placed in a geostationary orbit in position 10° E longitude, 0° latitude. The minimum gain specified in the coverage area is 28 dBi, which has been enlarged to consider typical pointing errors (0.1° in roll, 0.1° in pitch, and 0.5° in yaw). The coverage shown in Fig. 5 is specified in the antenna coordinate system, and before carrying out the synthesis, it needs to be transformed to the reflectarray coordinate system defined in Fig. 1 [2].

The reflectarray is square and formed by 5180 elements in a regular grid of 74×70 cells. The feed horn is modeled as a $\cos^q \theta$ function [37] with a q-factor of 23, which provides an illumination taper of -17.9 dB at the reflectarray edges. The feed horn is placed at $\vec{r}_f = (358, 0, 1070)$ mm with regard to the reflectarray center. The period of the unit cell is 14 mm $\times$ 14 mm, which is 0.55λ at the working frequency. Now, the substrate for the bottom layer has a height of 2.363 mm and a complex relative permittivity $\epsilon_r = 2.55 - j2.295 \cdot 10^{-3}$, while the top layer has a height of 1.524 mm and a complex relative permittivity $\epsilon_r = 2.17 - j1.953 \cdot 10^{-3}$.

The optimization is carried out in a region defined by $u \in [0.1, 0.45]$ and $v \in [-0.15, 0.15]$ around the coverage area, with a resolution of the far fields of 512×512 points for the FFT, having a total of 4047 points in the UV grid.
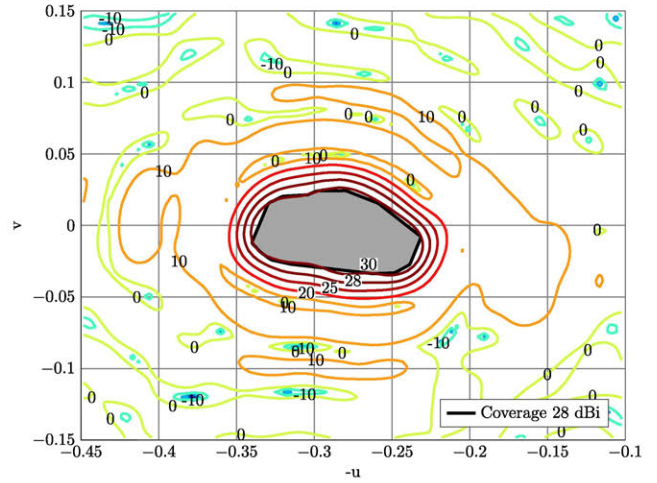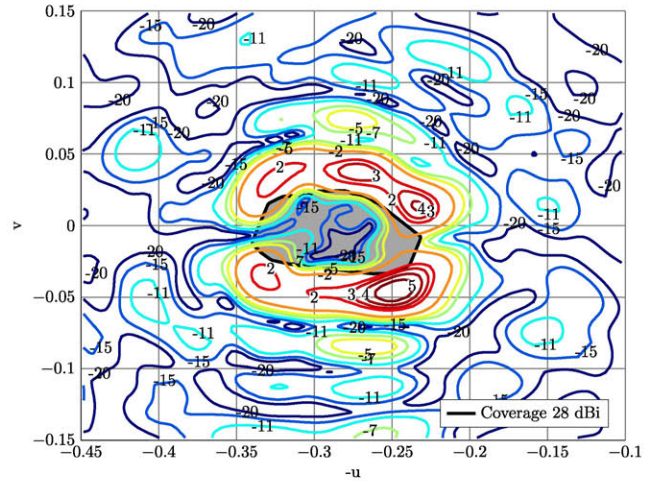
All of the reflectarray elements are optimized at the same time, considering six dipoles as independent optimizing variables ($s = 6$) maintaining the unit cell symmetry, thus having a total number of optimizing variables of 31 080. The optimization is carried out in fixed gain, setting the required gain to 31.90 dB in the direction $(u_0, v_0) = (0.28, 0)$. The crosspolar template is set 35 dB below the maximum copolar gain template, and the crosspolar component is scaled by a factor of 10^5 in natural units with the weighting function. The LMA is set to perform three iterations per iteration of the generalized IA.

Although the design and crosspolar optimization have been carried out for both linear polarizations at the same time, only the results for Y polarization will be shown graphically, since the radiation patterns are similar for both polarizations before the optimization, and the Y polarization results after the optimization are slightly worse than for the X polarization. However, numerical values will be provided for both polarizations. Fig. 6 shows the radiation pattern obtained for the design after the POS for Y polarization. The gain requirements are
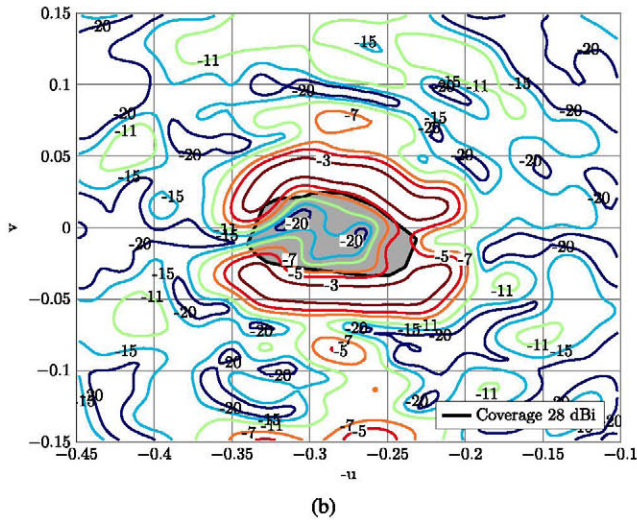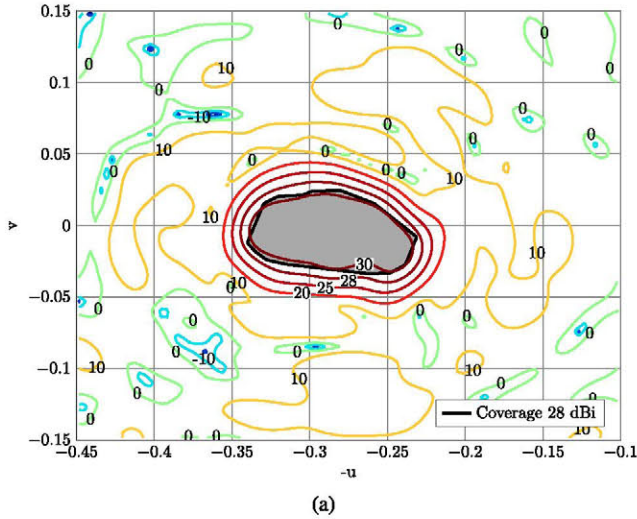
(a)

(b)

Fig. 7. Optimized European coverage radiation pattern in gain (dBi) for Y polarization. (u, v) are in the reflectarray coordinate system. (a) Copolar. (b) Crosspolar.

fulfilled in the whole extended coverage region (see Fig. 5), having a maximum copolar gain of 32.74 dBi and a maximum crosspolar gain of 5.98 dBi. For the X polarization, those values are 32.60 and −0.88 dBi for the maximum copolar and crosspolar gain, respectively.

Fig. 7 shows the optimized radiation pattern for Y polarization. Now, the maximum copolar gain is 31.47 dBi while the maximum crosspolar gain is −2.20 dBi. Paying attention to the 30-dB contour, the pattern is slightly worsened, but the specifications comply for a gain of 28 dBi in the whole coverage region, while the crosspolar pattern has been substantially reduced (an amount of 8.18 dB). For X polarization, the coverage zone is barely affected, and presents a maximum copolar gain of 32.13 dBi and a maximum crosspolar gain of −2.20 dBi (a reduction of 1.32 dB).

A better parameter to analyze the crosspolar component of the radiation pattern is the crosspolar discrimination (XPD), which is defined, for the coverage area, as the difference, point by point, of the copolar and crosspolar components in dBi. Fig. 8 shows the XPD before and after the optimization for
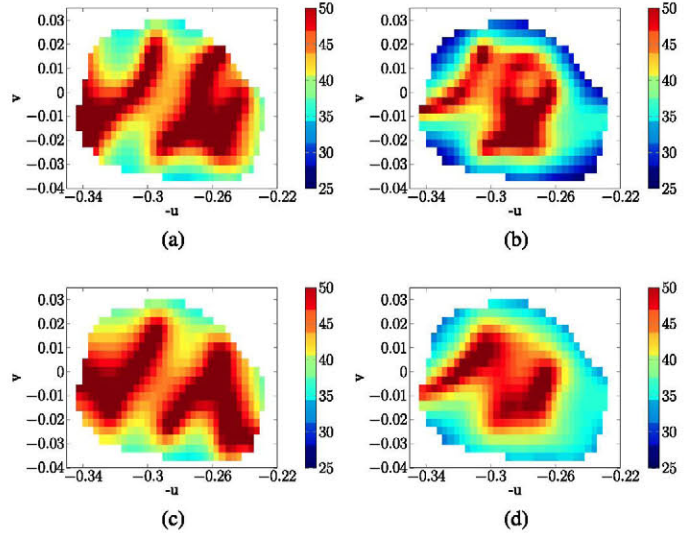

(a)

(b)

(c)

(d)

Fig. 8. XPD before and after the optimization for the European coverage shaped beam. (a) Polarization X before ($\mathrm{XPD_{min}} = 33.46$ dB). (b) Polarization Y before ($\mathrm{XPD_{min}} = 25.00$ dB). (c) Polarization X after ($\mathrm{XPD_{min}} = 33.94$ dB). (d) Polarization Y after ($\mathrm{XPD_{min}} = 30.76$ dB).

both polarizations. Due to the initial low value for the crosspolar component, the $\mathrm{XPD_{min}}$ for X polarization is 33.46 dB, and it barely improves after the optimization, obtaining a value of 33.94 dB (0.48 dB higher). However, for Y polarization, the initial value of $\mathrm{XPD_{min}}$ was 25 dB and it improved to 30.76 dB after the optimization (5.76 dB higher), despite defining the crosspolar template as a constant value in a bigger region than the coverage zone and not directly optimizing the XPD parameter.

Another parameter, stricter than XPD, is the crosspolar isolation (XPI), which is defined, for the coverage area, as the difference between the minimum copolar gain and the maximum crosspolar gain, both in dBi. For the case at hand, the initial XPI is 32.06 and 23.88 dB for X and Y polarizations, respectively. After the optimization, the new values for XPI are 32.62 dB (an improvement of 0.56 dB) and 29.73 dB (an improvement of 5.85 dB), for X and Y polarizations, respectively. As it happened to the XPD parameter, the XPI is greatly improved for Y polarization, but not for X polarization, since the starting point was already good.

V. DISCUSSION ON THE ACHIEVED IMPROVEMENTS

A. Crosspolar Reduction

Previous works [8], [9] show that, for large reflectarrays for DBS applications, the crosspolar pattern is close to fulfill requirements or they are fulfilled by little margin. Any further improvement in the crosspolar pattern would allow for manufacturing tolerances and nonidealities of the working environment, since the fulfillment margin would increase.

Crosspolar reduction varies depending on the radiation pattern and starting point in the optimization. From the two examples, in the isoflux case, the crosspolar maximum value is reduced more than 8 dB in both polarizations, while for the large reflectarray, for DBS application, the improvement is

more modest regarding the XPD and XPI. However, comparing maximum values of the crosspolar pattern before and after the optimization, for Y polarization, is reduced more than 8 dB. This is achieved at the expense of slightly affecting the copolar pattern for the 30-dB contour, but copolar requirements are still met for both polarizations in the whole coverage area (28-dB contour). Nevertheless, as a flat crosspolar template was employed, with a constant weighting function for all UV points where the far field was optimized, no focus was imposed in further minimizing crosspolarization within the coverage area. A way of further improving the XPD and XPI parameters would be to conveniently set a proper template and/or weighting function for the coverage area.

The accuracy of the obtained results strongly depends on the full-wave analysis tool based on LP, which is used by the technique. In the present case, the employed technique [25] has been validated with measurements of prototypes [32] and full-wave simulations of the whole antenna [31], showing good agreement in both cases. Hence, simulations of the obtained reflectarray layout after the crosspolar optimization should be reliable. However, since the developed technique for the crosspolar reduction does not factor in violations in the LP assumption, care should be taken regarding the size variation of the reflectarray elements, which should be more or less smooth to prevent large variations in the mutual coupling between elements with respect to the one simulated assuming LP.

Since the LP is an approximation, and considering other issues, including manufacturing errors, measurement misalignments, and so on, the crosspolar improvement obtained by the algorithm would certainly diminish. However, as shown in [32], the manufacturing errors produce variations in the crosspolar value in the order of 1 dB, which is much less than the reduction in crosspolar achieved by the optimization process. In sum, the reduction in the crosspolar pattern obtained by simulations should correspond, with a good approximation, to what can be obtained from a demonstrator fabricated and measured as long as some conditions are met, including a smooth variation of the reflectarray elements and reduction of diffraction effects by lowering the illumination level at the edges (since diffraction is not considered in the simulations).

B. Computing Times and Memory Usage

The two largest data structures in the algorithm are the Jacobian and Hessian approximation matrices. Each column of the Jacobian accommodates four far fields, each one of them with the same UV points as the rest. The total number of Jacobian columns is equal to the number of optimizing variables. On the other hand, the Hessian matrix is square, with the same number of columns as the Jacobian. Considering the total number of optimizing variables and UV points for each far field in the two examples, the sizes of both matrices are 12.5 GB and 500 MB for the Jacobian and Hessian approximation, respectively, in the case of the isoflux pattern optimization; 3.75 and 7.20 GB for the Jacobian and Hessian approximation, respectively, in the case of the European coverage pattern optimization.

Both optimizations were carried out in two different workstations to evaluate the scalability properties of the algorithm. The first workstation has two Intel Xeon X5560, each with four cores (handling eight threads, with 16 in total), at 2.8 GHz. The other workstation has two Intel Xeon E5-2650v3, each with ten cores (handling 20 threads, with 40 in total), at 2.3 GHz.

For the isoflux optimization, the LMA took an average time of 383 s per iteration in the first machine with 16 threads, and 116 s per iteration in the second machine with 40 threads. For the European coverage optimization, those times were 1143 and 344 s. As it can be seen, despite using a full-wave analysis based on LP during the optimization process, the computing time per iteration is reasonable for large reflectarrays, and due to the good scalability properties of the algorithm, it improves when better hardware is used, namely, more number of CPU units with higher clock frequency.

As a final remark, the algorithm implemented in this paper uses multicore CPU acceleration based on OpenMP [42] and the Intel MKL libraries [41]. Recent work has shown that the use of graphics processing units acceleration can substantially increase parallel performance of algorithms for antenna synthesis [28] with regard to CPU acceleration, and even to improve the performance per watt [45].

VI. CONCLUSION

A new technique for the optimization of the crosspolar component of dual-polarized reflectarrays has been described, which analyzes the reflectarray using a full-wave MoM-LP during the optimization process in order to correctly characterize the crosspolar component, which is being optimized. The algorithm is based on the generalized Intersection Approach framework and it uses the LMA as backward projector. Since it directly employs MoM-LP as the analysis tool, several strategies have been introduced to the algorithm in order to speed up computations. All the building blocks have been parallelized and strategies to reduce the number of calls to the MoM-LP routine have been devised, saving important amounts of time per algorithm iteration. Also, the convergence of the algorithm is improved by working with the squared field amplitude (or equivalently, the gain), which alleviates the problem of traps by convexifying one of the sets and also allowing the far-field phase to take any value in the backward projector. Thanks to the optimizations and improved convergence, the algorithm is able to handle thousands of optimizing variables and still attain good results in the optimization of the crosspolar component of the far field, as it has been demonstrated in the two test cases presented. Since, for very large reflectarrays, the number of variables and UV points in which the far fields are discretized can grow quickly, it is possible to optimize the far field in a small region of the UV grid using less optimizing variables, reducing the degrees of freedom, computing times, and memory usage.

REFERENCES

[1] D. Berry, R. Malech, and W. Kennedy, "The reflectarray antenna," *IEEE Trans. Antennas Propag.*, vol. 11, no. 6, pp. 645–651, Nov. 1963.
[2] J. Huang and J. A. Encinar, *Reflectarray Antennas*. Hoboken, NJ, USA: Wiley, 2008.

[3] D. M. Pozar, S. D. Targonski, and R. Pokuls, "A shaped-beam microstrip patch reflectarray," *IEEE Trans. Antennas Propag.*, vol. 47, no. 7, pp. 1167–1173, Jul. 1999.

[4] J. A. Encinar and J. A. Zornoza, "Broadband design of three-layer printed reflectarrays," *IEEE Trans. Antennas Propag.*, vol. 51, no. 7, pp. 1662–1664, Jul. 2003.

[5] J. H. Yoon, Y. J. Yoon, W. S. Lee, and J. H. So, "Broadband microstrip reflectarray with five parallel dipole elements," *IEEE Antennas Wireless Propag. Lett.*, vol. 14, pp. 1109–1112, Jan. 2015.

[6] R. Florencio, J. A. Encinar, R. R. Boix, and G. Perez-Palomino, "Dual-polarisation reflectarray made of cells with two orthogonal sets of parallel dipoles for bandwidth and cross-polarisation improvement," *IET Microw., Antennas Propag.*, vol. 8, no. 15, pp. 1389–1397, 2014.

[7] L. Moustafa, R. Gillard, F. Peris, R. Loison, H. Legay, and E. Girard, "The Phoenix cell: A new reflectarray cell with large bandwidth and rebirth capabilities," *IEEE Antennas Wireless Propag. Lett.*, vol. 10, pp. 71–74, 2011.

[8] J. A. Encinar *et al.*, "Dual-polarization dual-coverage reflectarray for space applications," *IEEE Trans. Antennas Propag.*, vol. 54, no. 10, pp. 2827–2837, Oct. 2006.

[9] J. A. Encinar, M. Arrebola, L. F. de la Fuente, and G. Toso, "A transmit-receive reflectarray antenna for direct broadcast satellite applications," *IEEE Trans. Antennas Propag.*, vol. 59, no. 9, pp. 3255–3264, Sep. 2011.

[10] O. M. Bucci, G. Franceschetti, G. Mazzarella, and G. Panariello, "Intersection approach to array pattern synthesis," *IEE Proc. H-Microw., Antennas Propag.*, vol. 137, no. 6, pp. 349–357, Dec. 1990.

[11] J. A. Zornoza and J. A. Encinar, "Efficient phase-only synthesis of contoured-beam patterns for very large reflectarrays," *Int. J. RF Microw. Comput.-Aided Eng.*, vol. 14, no. 5, pp. 415–423, Sep. 2004.

[12] D. R. Prado, J. Álvarez, M. Arrebola, M. R. Pino, R. G. Ayestarán, and F. Las-Heras, "Efficient, accurate and scalable reflectarray phase-only synthesis based on the Levenberg–Marquardt algorithm," *Appl. Comput. Electromagn. Soc. J.*, vol. 30, no. 12, pp. 1246–1255, Dec. 2015.

[13] P. Nayeri, F. Yang, and A. Z. Elsherbeni, "Design of single-feed reflectarray antennas with asymmetric multiple beams using the particle swarm optimization method," *IEEE Trans. Antennas Propag.*, vol. 61, no. 9, pp. 4598–4605, Sep. 2013.

[14] H. Hasani, M. Kamyab, and A. Mirkamali, "Low cross-polarization reflectarray antenna," *IEEE Trans. Antennas Propag.*, vol. 59, no. 5, pp. 1752–1756, May 2011.

[15] C. Tienda, J. A. Encinar, M. Arrebola, M. Barba, and E. Carrasco, "Design, manufacturing and test of a dual-reflectarray antenna with improved bandwidth and reduced cross-polarization," *IEEE Trans. Antennas Propag.*, vol. 61, no. 3, pp. 1180–1190, Mar. 2013.

[16] D. R. Prado, M. Arrebola, M. R. Pino, and F. Las-Heras, "Complex reflection coefficient synthesis applied to dual-polarized reflectarrays with cross-polar requirements," *IEEE Trans. Antennas Propag.*, vol. 63, no. 9, pp. 3897–3907, Sep. 2015.

[17] H. Legay, D. Bresciani, E. Labiole, R. Chiniard, and R. Gillard, "A multi facets composite panel reflectarray antenna for a space contoured beam antenna in Ku band," *Prog. Electromagn. Res. B*, vol. 54, pp. 1–26, Aug. 2013.

[18] O. M. Bucci, A. Capozzoli, G. D'Elia, and S. Musto, "A new approach to the power pattern synthesis of reflectarrays," in *Proc. URSI Int. Symp. Electromagn. Theory (EMTS)*, Pisa, Italy, May 2004, pp. 1053–1055.

[19] O. M. Bucci, A. Capozzoli, G. D'Elia, and S. Russo, "Power pattern synthesis of reflectarrays: Comparison between two approaches," in *Proc. 15th Riunione Nazionale Elettromagnetismo (CD-ROM)*, Cagliari, Italy, Sep. 2004, pp. 1–4.

[20] O. M. Bucci, A. Capozzoli, G. D'Elia, and S. Russo, "An advanced technique for reflectarray power pattern synthesis and its experimental validation," in *Proc. Int. Symp. Antennas Propag. (ISAP)*, Seoul, South Korea, Aug. 2005, pp. 561–564.

[21] M. Zhou, S. B. Sørensen, O. S. Kim, E. Jørgensen, P. Meincke, and O. Breinbjerg, "Direct optimization of printed reflectarrays for contoured beam satellite antenna applications," *IEEE Trans. Antennas Propag.*, vol. 61, no. 4, pp. 1995–2004, Apr. 2013.

[22] M. Zhou *et al.*, "The generalized direct optimization technique for printed reflectarrays," *IEEE Trans. Antennas Propag.*, vol. 62, no. 4, pp. 1690–1700, Apr. 2014.

[23] M. Zhou, O. Borries, and E. Jørgensen, "Design and optimization of a single-layer planar transmit-receive contoured beam reflectarray with enhanced performance," *IEEE Trans. Antennas Propag.*, vol. 63, no. 4, pp. 1247–1254, Apr. 2015.

[24] C. Wan and J. A. Encinar, "Efficient computation of generalized scattering matrix for analyzing multilayered periodic structures," *IEEE Trans. Antennas Propag.*, vol. 43, no. 11, pp. 1233–1242, Nov. 1995.

[25] R. Florencio, R. R. Boix, and J. A. Encinar, "Enhanced MoM analysis of the scattering by periodic strip gratings in multilayered substrates," *IEEE Trans. Antennas Propag.*, vol. 61, no. 10, pp. 5088–5099, Oct. 2013.

[26] D. M. Pozar and T. A. Metzler, "Analysis of a reflectarray antenna using microstrip patches of variable size," *Electron. Lett.*, vol. 29, no. 8, pp. 657–658, Apr. 1993.

[27] O. M. Bucci, G. D'Elia, G. Mazzarella, and G. Panariello, "Antenna pattern synthesis: A new general approach," *Proc. IEEE*, vol. 82, no. 3, pp. 358–371, Mar. 1994.

[28] A. Capozzoli, C. Curcio, A. Liseno, and G. Toso, "Fast, phase-only synthesis of aperiodic reflectarrays using NUFFTs and CUDA," *Prog. Electromagn. Res.*, vol. 156, pp. 83–103, 2016.

[29] M. Zhou, S. B. Sørensen, E. Jørgensen, P. Meincke, O. S. Kim, and O. Breinbjerg, "Analysis of printed reflectarrays using extended local periodicity," in *Proc. 5th Eur. Conf. Antennas Propag. (EUCAP)*, Rome, Italy, Apr. 2011, pp. 1408–1412.

[30] A. Capozzoli, C. Curcio, A. Liseno, M. Migliorelli, and G. Toso, "Analysis of periodic reflectarray using FEKO," in *Proc. 29th Int. Rev. Prog. Appl. Comput. Electromagn.*, Monterey, CA, USA, Mar. 2013, pp. 387–392.

[31] R. Florencio, J. A. Encinar, R. R. Boix, G. Pérez-Palomino, and G. Toso, "Cross-polar reduction in reflectarray antennas by means of element rotation," in *Proc. 10th Eur. Conf. Antennas Propag. (EuCAP)*, Davos, Switzerland, Apr. 2016, pp. 1–5.

[32] R. Florencio, J. A. Encinar, R. R. Boix, V. Losada, and G. Toso, "Reflectarray antennas for dual polarization and broadband telecom satellite applications," *IEEE Trans. Antennas Propag.*, vol. 63, no. 4, pp. 1234–1246, Apr. 2015.

[33] M. Zhou, S. B. Sørensen, E. Jørgensen, P. Meincke, O. S. Kim, and O. Breinbjerg, "An accurate technique for calculation of radiation from printed reflectarrays," *IEEE Antennas Wireless Propag. Lett.*, vol. 10, pp. 1081–1084, 2011.

[34] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, 3rd ed. Hoboken, NJ, USA: Wiley, 2012.

[35] D. R. Prado, M. Arrebola, M. R. Pino, and F. Las-Heras, "An efficient calculation of the far field radiated by non-uniformly sampled planar fields complying Nyquist theorem," *IEEE Trans. Antennas Propag.*, vol. 63, no. 2, pp. 862–865, Feb. 2015.

[36] A. Ludwig, "The definition of cross polarization," *IEEE Trans. Antennas Propag.*, vol. 21, no. 1, pp. 116–119, Jan. 1973.

[37] Y.-T. Lo and S.-W. Lee, Eds., *Antenna Handbook*, vol. 1. New York, NY, USA: Van Nostrand, 1993, ch. 1, pp. 28–29.

[38] J. Perini, "Note on antenna pattern synthesis using numerical iterative methods," *IEEE Trans. Antennas Propag.*, vol. 19, no. 2, pp. 284–286, Mar. 1971.

[39] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.

[40] J. Nocedal and Y.-X. Yuan, "Analysis of a self-scaling quasi-Newton method," *Math. Prog.*, vol. 61, no. 1, pp. 19–37, Aug. 1993.

[41] *Intel Math Kernel Library Reference Manual*, Intel Corp., Santa Clara, CA, USA, Aug. 2008.

[42] M. Sato, "OpenMP: Parallel programming API for shared memory multiprocessors and on-chip multiprocessors," in *Proc. 15th Int. Symp. Syst. Synth.*, Kyoto, Japan, Oct. 2002, pp. 109–111.

[43] D. R. Prado, A. Campa, M. Arrebola, M. R. Pino, J. A. Encinar, and F. Las-Heras, "Design, manufacture, and measurement of a low-cost reflectarray for global earth coverage," *IEEE Antennas Wireless Propag. Lett.*, vol. 15, pp. 1418–1421, 2016.

[44] J. A. Encinar, M. Arrebola, M. Dejus, and C. Jouve, "Design of a 1-metre reflectarray for DBS application with 15% bandwidth," in *Proc. 1st Eur. Conf. Antennas Propag. (EuCAP)*, Nice, France, Nov. 2006, pp. 1–5.

[45] T. Dong, V. Dobrev, T. Kolev, R. Rieben, S. Tomov, and J. Dongarra, "A step towards energy efficient computing: Redesigning a hydrodynamic application on CPU-GPU," in *Proc. IEEE 28th Int. Parallel Distrib. Process. Symp.*, Phoenix, AZ, USA, May 2014, pp. 972–981.