



**HAL**  
open science

# Parametric Shape Optimization for the Conformation of Axisymmetric Reflector Antennas

Benoît Chaigne, Jean-Antoine Desideri

► **To cite this version:**

Benoît Chaigne, Jean-Antoine Desideri. Parametric Shape Optimization for the Conformation of Axisymmetric Reflector Antennas. [Research Report] 2007, pp.27. inria-00151600v1

**HAL Id: inria-00151600**

**<https://inria.hal.science/inria-00151600v1>**

Submitted on 4 Jun 2007 (v1), last revised 7 Jun 2007 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

*Parametric Shape Optimization for the  
Conformation of Axisymmetric Reflector Antennas*

Benoît Chaigne — Jean-Antoine Désidéri

N° ????

Septembre 2006

Thème NUM



*Rapport  
de recherche*



## Parametric Shape Optimization for the Conformation of Axisymmetric Reflector Antennas

Benoît Chaigne\*, Jean-Antoine Désidéri\*

Thème NUM — Systèmes numériques  
Projet Opale

Rapport de recherche n° ???? — Septembre 2006 — 27 pages

**Abstract:** A shape reconstruction problem of axisymmetric radiating structures is considered. This reads as an inverse problem. The direct problem - expression of the electric field w.r.t. a given geometry - is approximated by a simple model. For the inverse problem - find the shape whose radiation has been measured - a parametric shape functional is defined for which the *Free-Form Deformation* method is introduced. Two different strategies to solve numerically the inverse problem (minimize the functional) is discussed: the first one is deterministic and the other one semi-stochastic. It appears that the functional is highly multimodal. At last, a multilevel strategy is proposed for which we discuss both robustness and convergence.

**Key-words:** inverse problem ; shape optimization ; *Free-Form Deformation* ; Particle Swarm Optimization ; multilevel methods ; electromagnetics ; reflector antenna ; radiation diagram

\* INRIA Sophia-Antipolis, OPALE Team

## Optimisation de forme paramétrique pour la conformation d'antennes à réflecteur axisymétriques

**Résumé :** On considère un problème de reconstruction de forme de structures rayonnantes axisymétriques. Il s'exprime comme un problème inverse. Le problème direct - expression du champ électrique en fonction d'une géométrie donnée - est modélisé par une formule approchée. Pour le problème inverse - retrouver la forme dont le rayonnement a été mesuré - une fonctionnelle de forme est définie paramétriquement. À ce propos la méthode *Free-Form Deformation* est introduite. Nous comparons alors deux stratégies différentes pour résoudre numériquement le problème inverse (minimiser la fonctionnelle) : une déterministe et une semi-stochastique. On montre à cette occasion le caractère multimodal de la fonctionnelle. Enfin, à la vue des résultats, on propose une méthode hiérarchique afin d'améliorer la robustesse et la convergence.

**Mots-clés :** problème inverse ; optimisation de forme ; *Free-Form Deformation* ; optimisation par essaim de particules ; méthodes multi-niveaux ; électromagnétisme ; antenne à réflecteur ; diagramme de rayonnement

## Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Direct problem: computing the electric field</b>	<b>4</b>
2.1	Expression for the electric field, <i>Physical Optics</i> model . . . . .	4
2.2	Directive gain . . . . .	5
2.3	Numerical computation of the directive gain . . . . .	6
<b>3</b>	<b>Parametric representation of the geometry</b>	<b>7</b>
3.1	<i>Free-Form Deformation</i> principles . . . . .	9
3.2	Bernstein polynomials properties . . . . .	9
<b>4</b>	<b>Inverse problem: optimizing the shape of a reflector</b>	<b>11</b>
4.1	Shape functional . . . . .	11
4.2	Parametric functional . . . . .	11
<b>5</b>	<b>Numerical case study</b>	<b>12</b>
5.1	Case settings . . . . .	12
5.2	Gradient-based strategy . . . . .	14
5.2.1	Introduction to FFSQP . . . . .	14
5.2.2	Application to the case . . . . .	14
5.3	Semi-stochastic strategy: <i>Particle Swarm Optimization</i> . . . . .	18
5.3.1	Introduction to the algorithm . . . . .	18
5.3.2	Application to the case . . . . .	21
5.4	Towards a multilevel method with <i>PSO</i> . . . . .	23
5.4.1	Multilevel <i>PSO</i> . . . . .	23
5.4.2	Application to the case . . . . .	24
<b>6</b>	<b>Conclusion</b>	<b>26</b>

## 1 Introduction

Antenna conformation is a real-world electromagnetic issue that arises each time a reflector antenna is designed. For instance the radiation of the antenna should cover a specific geographical area while avoiding others, under international norms. Or it should keep ideal properties considering the influence of the structure that hold it.

Practically, it often requires physical intuition from conceptors. However this problem can be defined as an inverse problem yielding a shape optimization problem [5, 6]. In some special cases this problem reads as a shape reconstruction problem for which there are successful methods [6] involving multiple frequencies.

The study aims to explain why the problem is hard to solve and then suggests new methods for solving it. First, the equations for the direct problem are recalled. Then the shape parameterization is introduced, which can be critical in such problems. In an other part the objective functional of the inverse problem is defined. At last a case study is conducted where different strategies to reduce the objective function are compared.

## 2 Direct problem: computing the electric field

A reflector antenna in transmission is composed of a source of electromagnetic (EM) waves and a reflector considered as a conductor object (see Figures 1 and 2). Physically, the source will induce surface currents onto the reflector which will then act as a secondary EM waves source. The direct problem consists in determining the resulting electric field in a domain  $\Omega$ , considering a geometry and an incident wave.

Electromagnetic phenomena are modeled by Maxwell equations [1, 3]. However, a simpler model known as *physical optics*, well suited for the problem, will be considered. This model, for which the electric field becomes explicit w.r.t the geometry, allows us to avoid the heavy computations of using a Finite Element solver.

In this study axisymmetric reflectors and a single monochromatic source are considered. More precisely, the diameter of the aperture is around 30 cm, thus a reasonable frequency is  $f = 10$  GHz. The wave velocity  $c$  is equal to the speed of light in vacuum and thus the wave length is  $\lambda = \frac{c}{f} = 2.99792458 \cdot 10^{-2}$  m, that is, one tenth of the aperture.

### 2.1 Expression for the electric field, *Physical Optics* model

As said previously the Maxwell equations describe the EM field  $(\vec{E}, \vec{H})$ . The directive gain, which will be introduced in section 2.2, involves the *far field* power for which the electric field will be sufficient. Therefore one focuses on  $\vec{E}$ . Physics about EM field and antennas can be found in [1, 3].

Let  $O$  be the source localization and  $\mathcal{S}$  the reflector surface. The total electric field  $\vec{E}$  at a point  $P$  results from the summation of the incident field  $\vec{E}^i$  radiated by the source and the scattered field  $\vec{E}^s$ :

$$\vec{E}(P) = \vec{E}^i(P) + \vec{E}^s(P). \quad (1)$$

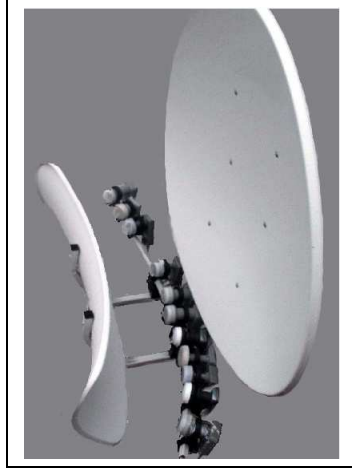


Figure 1: example of a toroidal reflector antenna with several sources [4]

If  $P$ , identified by  $\vec{R} = \overrightarrow{OP}$ , is far enough from the reflector (far field) and if the diameter of the reflector is much larger than  $\lambda$  the scattered field is given by:

$$\vec{E}^s(P) = -i\omega\mu \frac{e^{-ikR}}{4\pi R} \iint_{\mathcal{S}} \left[ \vec{I} - (\vec{I} \cdot \vec{R}) \vec{R} \right] e^{ik\rho(\vec{R} \cdot \vec{\rho})} d\mathcal{S}, \quad (2)$$

where  $\mu$  stands for the magnetic permeability,  $\omega$  the wave pulsation and  $k$  the wave number ( $k = \frac{2\pi}{\lambda}$ ).  $\vec{I}$  is the induced current on  $\mathcal{S}$  at a point  $Q$  identified in (2) by  $\vec{\rho} = \overrightarrow{OQ}$ .

The *physical optics* approach consists in considering that at each point of the surface, the scattered wave is equivalent to the scattered wave on the tangent plane, perfectly conductor. This means in particular that the diffraction effects due to singularities on the reflector (such as the border) are neglected. This model is quite accurate when the wave length is much smaller than the reflector, as assumed. Thus, formally, the surface current on  $\mathcal{S}$  is given by:

$$\vec{I} = -2\vec{n} \times \vec{H}^i, \quad (3)$$

where  $\vec{H}^i$  is the incident magnetic field.

## 2.2 Directive gain

The directive gain is the radiation intensity (power radiated by unit solid angle) normalized by the one of the corresponding isotropic source [3]. The set of directive gain for a set of directions is called radiation diagram.



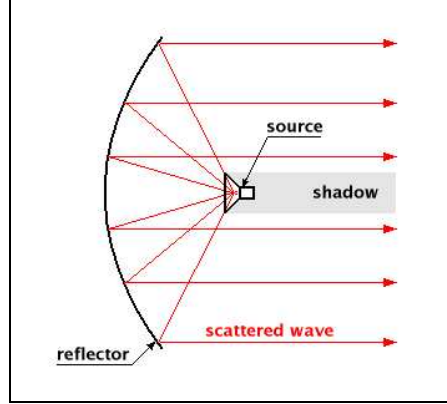


Figure 2: schematic scattering of a reflector antenna (parabola) [4]

Let  $U(\theta, \phi)$  be the radiation intensity for the  $(\theta, \phi)$  direction. In our case the power density is given by the Poynting vector:

$$\mathcal{P} = \frac{1}{2} |\vec{E} \times \vec{H}^*| = \frac{1}{2\eta} |\vec{E}|^2. \quad (4)$$

Thus  $U = R^2 \mathcal{P}$ . In far field  $\vec{E}$  is a spherical wave, which yields

$$\vec{E}(P) = \frac{e^{-ikR}}{4\pi R} \vec{E}(\theta, \varphi).$$

Consequently

$$U(\theta, \varphi) = \frac{1}{2\eta(4\pi)^2} |\vec{E}(\theta, \varphi)|^2. \quad (5)$$

Finally, to obtain the directive gain  $D$ , one must normalize with the intensity of the corresponding isotropic source which is the mean of  $U$  over all solid angles, that is:

$$U_i = \frac{\iint_{\Omega} U(\theta, \varphi) d\Omega}{4\pi}, \quad (6)$$

and hence

$$D(\theta, \varphi) = \frac{U(\theta, \varphi)}{U_i} = \frac{4\pi |\vec{E}(\theta, \varphi)|^2}{\iint_{\Omega} |\vec{E}(\theta, \varphi)|^2 d\Omega}. \quad (7)$$

### 2.3 Numerical computation of the directive gain

Practically, for the computation of the radiation diagram, the considered source will be an elementary dipole for which the radiation  $(\vec{E}^i, \vec{H}^i)$  is known analytically. It can be shown

that such a dipole is unimodal (in the sense that its Fourier serie decomposition reduces to a single mode  $m = 1$ ). Thus the knowledge of  $\vec{E}$  on the planes defined by  $\varphi = 0$  and  $\varphi = \frac{\pi}{2}$  (for  $\theta \in [0, 2\pi]$ ) is sufficient to deduce the electric field everywhere else. Moreover, it can be shown that on each plane, only a single component (in spherical coordinates) is non zero. The unknown, depending on a surface  $\mathcal{S}$ , will from now be noted

$$\vec{E}(\mathcal{S}, \theta) = \begin{pmatrix} E^\theta(\mathcal{S}, \theta) \\ E^\varphi(\mathcal{S}, \theta) \end{pmatrix} \quad (8)$$

where  $E^\theta$  and  $E^\varphi$  stand for the non zero components.

The reflector being axisymmetric,  $\mathcal{S}$  results from the revolution of a curve  $\mathcal{C}$  of  $\mathbb{R}^2$  called meridian. Equation (2) reduces then to a single integral over  $\mathcal{C}$ . Numerically,  $\mathcal{C}$  is piecewise linear and the total scattered field corresponds to the sum of fields scattered by each segment. The integrals over the segments are computed by Gauss quadratures.

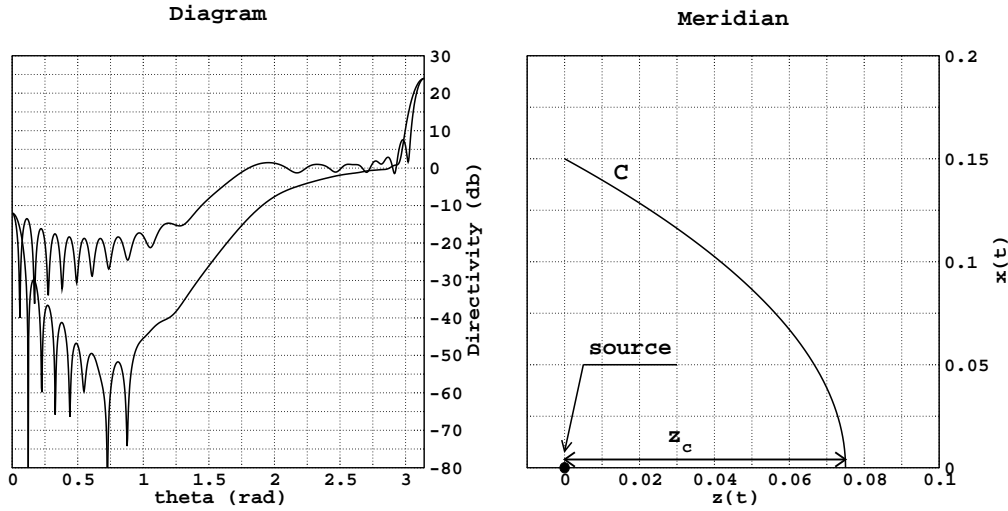


Figure 3: radiation diagram of a parabola radiated by an elementary dipole,  $f = 10$  GHz ( $\lambda = 30$  mm), aperture diameter  $d = 30$  cm =  $10\lambda$ ,  $z_c = 7.5$  cm =  $2.5\lambda$

### 3 Parametric representation of the geometry

Parameterization was said to be critical in the introduction. Indeed, the choice of parameters will induce the nature of the optimization parameters. For instance, in [5] they are the nodes of discretization of the shape. The disadvantage of using such a parameterization is that singularities may be introduced during the optimization process. Moreover, it

introduces stiffness since the number of nodes should be large so that the direct problem becomes accurate enough. Therefore here is proposed an other parameterization where the optimization variables and the discretization are independant.

Let  $\mathcal{C}$  be the meridian of  $\mathcal{S}$  represented in the half-plane as follows:

$$\mathcal{C} \begin{cases} x(s) \in \mathcal{C}^1([0, 1], \mathbb{R}^+) \\ z(s) \in \mathcal{C}^1([0, 1], \mathbb{R}) \end{cases} \quad s \in [0, 1]. \quad (9)$$

Note that  $\mathcal{C}$  should be regular enough for the *Physical Optics* model to be valid. The surface  $\mathcal{S}$  is obtained by rotation around the axis of equation  $x = 0$ . In addition the following condition is required:

$$\begin{cases} x(0) = 0 \\ z(0) = z_c \end{cases}, \quad (10)$$

where  $z_c$  represents the distance between the source (located at the origin of the system of coordinates) and the center of the reflector fixed on the  $z$ -axis (see Figures 3 and 4).

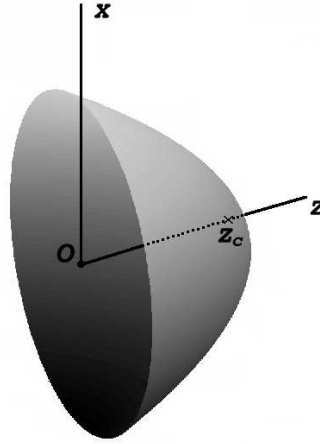


Figure 4: Paraboloid

The surface considered in the optimization will actually result from the deformation of an initial surface  $\mathcal{S}^0$ , namely:

$$\mathcal{S} = \mathcal{S}^0 + \Delta\mathcal{S}, \quad (11)$$

where  $\Delta\mathcal{S}$ , the deformation, depends on a finite number of parameters. This approach is known as *Free-Form Deformation (FFD)* method, first introduced for Computer Graphics issues [7], in order to deform 2D or 3D objects regardless of what these objects are (noted

$S^0$  here). In this way the number of parametric variables, also called design variables, is chosen according to the parameterization. In addition, the smoothness of the shape will be controlled as explained in 3.1

### 3.1 Free-Form Deformation principles

This section is devoted to explain the *FFD* technique for a 2D problem. Let  $\mathcal{C}$  be a curve of  $\mathbb{R}^2$  (non necessarily regular) contained in a subset  $D \subset \mathbb{R}^2$ . Consider  $\phi$  an homeomorphism that maps  $D$  onto the unit square:

$$\begin{aligned} \phi : D &\longrightarrow [0, 1] \times [0, 1] \\ (x, z) &\longmapsto (t_x, t_z) \end{aligned} \quad , \quad (12)$$

that is, a local system of coordinates on  $D$  is defined. Then, let  $\{b_k\}_{k=1}^n$  be a linearly independent family of  $\mathbb{R}$ -valued  $C^1$  functions defined on  $[0, 1] \times [0, 1]$ . The deformation of the domain  $D$  is then defined as a linear combination of the basis functions  $b_k$ . Note that this approach defines a global smooth transformation over the whole domain. Formally, for all points  $(x, z)$  of  $D$ , *a fortiori* of  $\mathcal{C}$ , the deformation reads:

$$\begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix} = \sum_{k=1}^n b_k(\phi(x, z)) \vec{p}_k, \quad (13)$$

where  $\vec{p}_k \in \mathbb{R}^2$ ,  $k = 1, \dots, n$ , are the coefficients of the deformation in the basis  $\{b_k\}_{k=1}^n$ . The space  $\mathbf{X} \equiv \text{span}\{b_1, b_2, \dots, b_n\}$  can be considered as the deformation space, which contains all the shapes that can be generated. Henceforth let  $\vec{p}_k$  be called the shape parameters. Since the parameterization applies on the deformation and not on the shape itself, no regularity of  $\mathcal{C}$  is required. Moreover, since the deformation is smooth, the singularities are kept.

Theoretically any (linearly independent) family  $\{b_k\}_{k=1}^n$  can be considered. Practically, in this study and historically [7], a tensorial product of Bernstein polynomials is used. This is justified by certain of their properties that will be discussed in the next section. The deformation reads then as follows:

$$\begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix} = \sum_{k_x=0}^{n_x} \sum_{k_z=0}^{n_z} B_{k_x}^{n_x}(t_x) B_{k_z}^{n_z}(t_z) \vec{p}_{k_x k_z}. \quad (14)$$

Note that the deformation takes the form of a Bézier surface although it is not a surface itself. Therefore the  $\vec{p}_{k_x k_z}$  are also called control points, by analogy. At last  $n_x$  and  $n_z$  are the degree of parameterization, which define the dimension of the deformation space.

### 3.2 Bernstein polynomials properties

The Bernstein Polynomials of degree  $n$  are defined by:

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{(n-k)} \quad k = 0, \dots, n \quad , \quad (15)$$

where  $\binom{n}{k}$  is the binomial coefficient:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}. \quad (16)$$

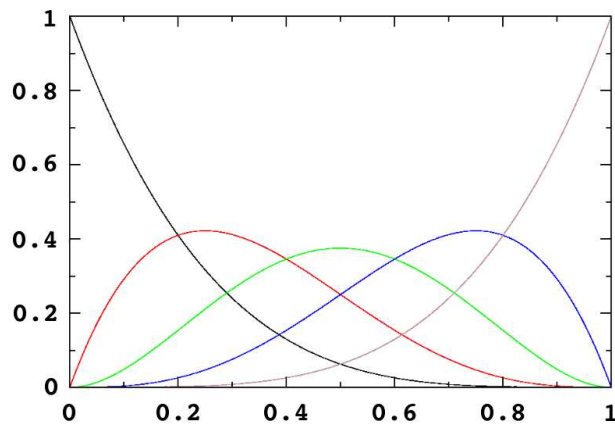


Figure 5: Bernstein polynomials of degree  $n=4$

**Proposition 3.1** *Basis: the Bernstein polynomials of degree  $n$  form a basis of  $\mathcal{P}^n$ , vector space of the polynomials of degree smaller or equal to  $n$ .*

Proposition 3.1 justifies the choice of choosing Bernstein polynomials as basis functions.

**Proposition 3.2** *Degree Elevation: let  $P(t) \in \mathcal{P}^n$  be represented in the Bernstein polynomials basis of degree  $n$  by the vector of coefficients  $p \in \mathbb{R}^{n+1}$ , hence the representation  $\tilde{p}$  of  $P(t)$  in the Bernstein polynomials basis of degree  $n+1$  is*

$$\begin{cases} \tilde{p}_0 &= p_0 \\ \tilde{p}_k &= \frac{k}{n+1}p_{k-1} + \left(1 - \frac{k}{n+1}\right)p_k, \quad 1 \leq k \leq n \\ \tilde{p}_{n+1} &= p_n \end{cases} \quad (17)$$

The degree elevation is the key property to the hierarchical method which will be introduced in section 5.4. This provides a method to express any polynomials of degree  $n$  in any polynomials spaces of higher degree in the Bernstein basis.

**Proposition 3.3** Unity Partition:

$$\forall n, \sum_{k=0}^n B_k^n(t) = 1, \forall t$$

Proposition 3.3 shows how to use the *FFD* method for a one dimension deformation and such that the deformation is constant in this dimension. For instance, let  $\vec{p}_{k_x k_z} = \vec{p}_{k_x}$ , for all  $k_z$ . Hence, according to (14), the deformation reads

$$\begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix} = \sum_{k_x=0}^{n_x} B_{k_x}^{n_x}(t_x) \vec{p}_{k_x} \underbrace{\sum_{k_z=0}^{n_z} B_{k_z}^{n_z}(t_z)}_{=1} = \sum_{k_x=0}^{n_x} B_{k_x}^{n_x}(t_x) \vec{p}_{k_x}. \quad (18)$$

## 4 Inverse problem: optimizing the shape of a reflector

In the direct problem, the radiation diagram can be computed providing a geometry and a source. Inversely, considering the geometry as the unknown, how can be determined the shape of  $\mathcal{S}$  to fit a given target diagram ?

To solve this inverse problem a shape functional that penalizes the discrepancy with the given diagram is minimized. First, the shape functional is defined and then different strategies to reduce it are discussed.

### 4.1 Shape functional

Let  $\vec{E}^t$  be the target electric field known on the planes  $\varphi = 0$  and  $\varphi = \frac{\pi}{2}$  (one reminds from section 2.3 that this is sufficient to deduce the far field everywhere else).  $J_1(\mathcal{S})$  is defined as the  $L_2$ -norm of the difference between the field given by the current shape  $\mathcal{S}$  (moving during the optimization process) and  $\vec{E}^t$ :

$$J_1(\mathcal{S}) = \frac{1}{2} \int_0^{2\pi} \|\vec{E}(\mathcal{S}, \theta) - \vec{E}^t(\theta)\|_{\mathbb{C}^2}^2 d\theta. \quad (19)$$

### 4.2 Parametric functional

Considering the parametric representation of  $\mathcal{S}$  (11) with the *FFD* technique (14), given a degree of parameterization  $(n_x, n_z)$ , the meridian  $\mathcal{C}$  defined by the so-called shape parameters  $\vec{p}_{k_x k_z}$  becomes:

$$\mathcal{C}(p) = \mathcal{C}^0 + \Delta\mathcal{C}(p) = \mathcal{C}^0 + \sum_{k_x=0}^{n_x} \sum_{k_z=0}^{n_z} B_{k_x}^{n_x}(t_x) B_{k_z}^{n_z}(t_z) \vec{p}_{k_x k_z}, \quad (20)$$

from which  $\mathcal{S}(p)$  is derived by axisymmetry. Thus a parametric functional  $j(p)$  is defined as follows:

$$j(p) = J(\mathcal{S}^0 + \Delta\mathcal{S}(p)), \quad (21)$$

where  $J$  is given by (19). Finally, the optimization problem reads then:

$$\min_{p \in \mathbb{R}^{2 \times d}} j(p), \quad (22)$$

where  $d$  stands for the dimension of the deformation space  $\mathbf{X}$ . Note that  $d = (n_x + 1)(n_z + 1)$  with the considered basis functions.

## 5 Numerical case study

As for several inverse problems, the existence of the solution nor its uniqueness, if there exists one, is ensured. Consequently, these problems are usually hard to solve numerically. Therefore, in order to be able to investigate the problem, a test-case is built such that the existence of a solution is established. Namely, the target field will come from a known geometry. This kind of problem is called a *reconstruction* problem.

Two different methods have been considered: a gradient-based, deterministic one, and a semi-stochastic one. Solve a numerical optimization problem implies the use of softwares. So, for each method, an experimentation is performed to ensure that the software is valid. Then a few tests on the problem itself are carried out.

### 5.1 Case settings

The initial shape  $\mathcal{S}^0$  of meridian  $\mathcal{C}^0$  is defined as follows:

$$\mathcal{C}^0 \begin{cases} x_0(t) = t \\ z_0(t) = z_c \end{cases} . \quad (23)$$

The target diagram corresponds to the computed diagram of  $\mathcal{S}^t$  with meridian  $\mathcal{C}^t$ :

$$\mathcal{C}^t \begin{cases} x(t) = t \\ z(t) = z_c + P(t) \end{cases} , \quad (24)$$

for some  $P(t) \in \mathcal{P}^m$  (see Figure 6). The difference between the two curves is non zero only in the  $z$  direction and is polynomial:

$$\begin{pmatrix} \Delta x \\ \Delta z \end{pmatrix} = \begin{pmatrix} 0 \\ P(t) \end{pmatrix} . \quad (25)$$

If a *FFD* deformation of degree  $(n, 0)$  is considered, in the  $z$  direction only, the corresponding deformation space is spanned by the family  $\{B_k^n\}_{k=0}^n$  of dimension  $d = n + 1$ . Since

this family is a basis of  $\mathcal{P}^n$  (see proposition 3.1), then for all  $n \geq m$ ,  $\Delta z$  belongs to the deformation space. In other words,

$$\forall n \geq m, \exists p = (p_0 \dots p_n) \in \mathbb{R}^{n+1} \quad ; \quad \Delta z(t) = \sum_{k=0}^n p_k B_k^n(t), \quad (26)$$

where  $p$  is unique for each fixed  $n$  since this is simply the representation of  $P(t)$  in a basis. So, this test-case is well defined to validate our optimization routines: a solution exists and, by construction, this solution, noted  $p^*$ , is known. This problem can be seen as a reconstruction problem more than a real optimization problem. In the concrete example in Figure 6 where  $\Delta z \in \mathcal{P}^6$  (and is exactly of degree 6), for  $n = m = 6$ ,  $p^*$  reads:

$$p^* = \begin{pmatrix} 0 \\ 0 \\ 0.003 \\ 0.009 \\ 0.01918125 \\ 0.03590625 \\ 0.06363 \end{pmatrix}. \quad (27)$$

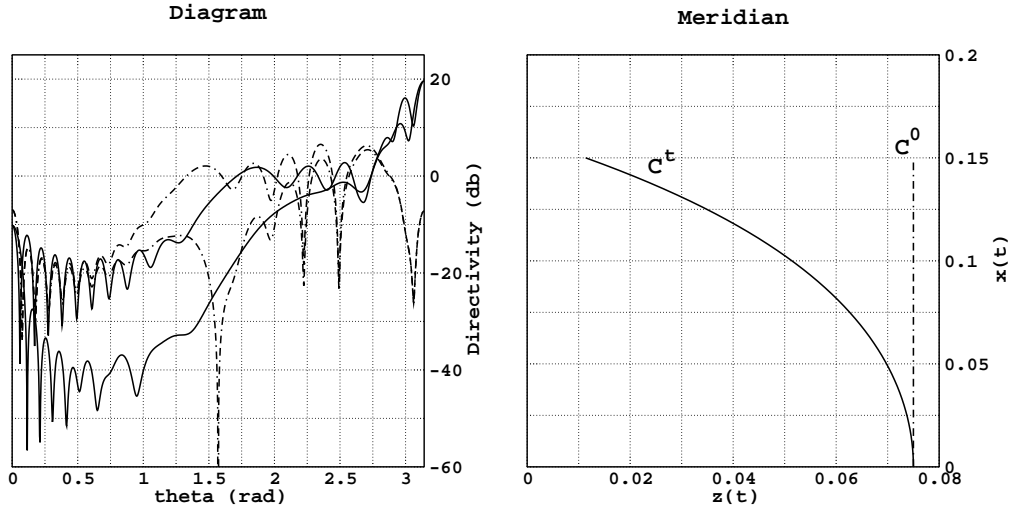


Figure 6: initial and target diagrams (left) / shapes (right),  $\Delta z \in \mathcal{P}^6$

**Remark** One can check that for all  $k > 0$ ,  $B_k^n(0) = 0$  and  $B_0^n(0) = 1$ . Thus, in order to ensure condition (10),  $p_0 = 0$ . Actually, any point on the line  $x = 0$  will not be deformed.



This implies that with degree of parameterization  $n$ , there are  $n$  degrees of freedom for each direction of deformation.

## 5.2 Gradient-based strategy

The Quasi-Newton methods are gradient-based optimization methods. They require the knowledge of the gradient while the Hessian is estimated through the iterations. This ensures quadratic convergence around the solution since the quadratic approximation of the functional becomes accurate.

### 5.2.1 Introduction to FFSQP

The FFSQP [12] routine is a powerful Quasi-Newton implementation with BFGS update [11]. It is based on Sequential Quadratic Programming (SQP) and ensures all inequality and equality constraints. This routine has been chosen for its maturity.

### 5.2.2 Application to the case

**Method validation:** First, consider a small perturbation of  $\mathcal{C}^t$  as initial shape instead of  $\mathcal{C}^0$  in Figure 6 and the highest degree,  $n = 6$ . Namely,  $\mathcal{C}^0$  is deformed with parameter  $p = p^* + \delta p$  for some non-zero  $\delta p \in \mathbb{R}^7$  (see Figure 8). Since the initial shape is close to the target one, close diagrams and consequently a fast converge are expected with such a method because the second-order Taylor expansion of the functional becomes accurate in the neighborhood of the solution. However note that such a perturbation in the shape has a non-negligible effect on the diagram: initial  $D(\mathcal{S}^0, \theta)$  and target  $D(\mathcal{S}^t, \theta)$  directivities are depicted in Figure 7. One focuses on the main directions  $\theta \in [2.5, \pi]$  where the difference mostly appears.

As expected, after fewer than 20 iterations the target diagram has been found ( $J(p) \rightarrow 0$ ), as well as the target shape (both curves are superimposed, see Figure 8).

**Case resolution:** After the above validation, the considered case is solved for different degrees:  $n = 2, 4$ , and 6. Obviously, the target shape cannot be found when  $n < 6$  because in these cases  $\Delta z \notin \mathbf{X}$ . One simply hopes to find a good approximation with fewer degrees of freedom.

It appears in Figure 9 that the shapes at convergence of the algorithm are very far from the target. The greater the degree, the closer is the diagram, however they are still very different ( $J(p) > 2$ ). Since a descent method is supposed to find the closest point  $p$  from the initial value such that  $J'(p) = 0$ , one can suppose that the algorithm has converged towards some local minima.

**Functional study:** After the previous experimentations one may think that the functional is highly sensitive to shape perturbations and that there might be many local minima. In order to check these assumptions, let us evaluate the functional around the solution for some

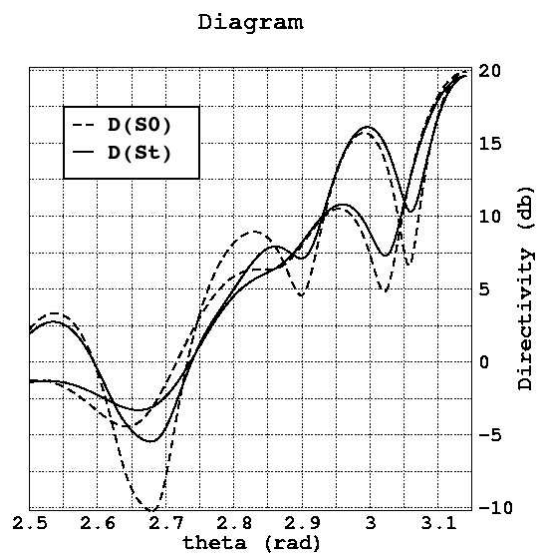


Figure 7: initial and target diagrams

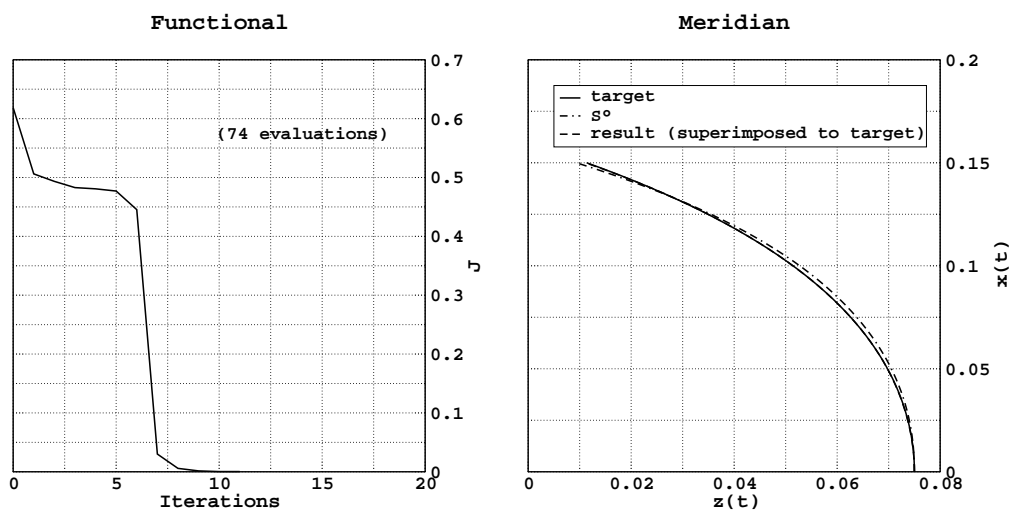


Figure 8: stability around the solution - FFSQP

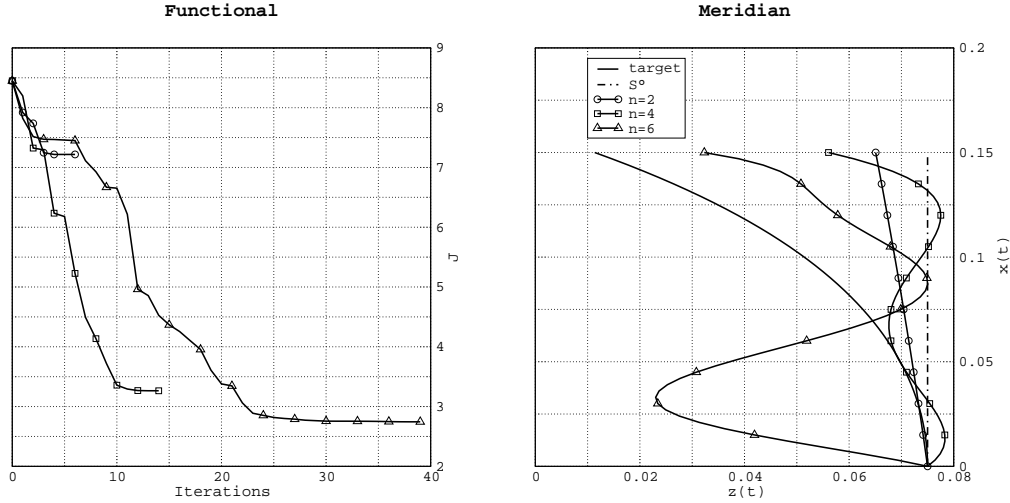


Figure 9: numerical solutions w.r.t degree - FFSQP

parameters. For a sake of simplicity a 2-parameters problem is considered. The target shape is the starting shape so that the solution yields  $p = 0$ . Then consider a FFD deformation for degree  $n = 2$  and set  $p_0 = 0$  as explained in the remark of section 5.1. Thus the functional is computed w.r.t. parameters  $p_1$  and  $p_2$  as shown in Figure 10.

Clearly the functional has many local minima and hence the solution given by a descent method depends on the initial shape. In other words, if the initial shape is close enough to the target shape, that is, if it is in the neighborhood around the solution for which the functional is locally convex, then the algorithm will converge towards the target.

In addition the Froebenius-norm of the Hessian (evaluated by Finite Differences) is depicted in 10 with a color scale. One can see that the convex curvature is, for this norm, maximum at the minimum of the functional (see also Figure 11).

It has been shown in [6] that the wave length has a “dilatation” effect on the functional. Namely, if the wave length is of the dimension of the antenna, then the functional looks unimodal for deformations that keep the shape in this dimension. This implies that one should measure the target field for one specific frequency or for multiple frequencies. In pure reconstruction problems, very good results for single and multiple frequencies can be seen in [6].

However, in optimization problems the target is the directivity, not the shape itself. In this case the scattered field at lower frequencies is unknown. Hence the “dilatation effect” of the functional cannot be used since the considered frequency cannot be changed. It was shown previously that the functional is highly multimodal w.r.t. the chosen shape parameters.

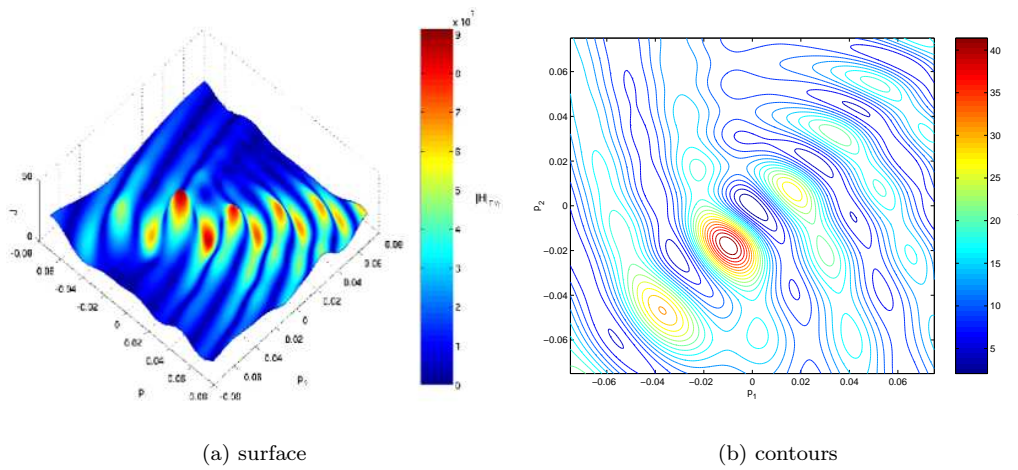


Figure 10: representation of the functional around the solution w.r.t. FFD shape parameters  $p_1$  and  $p_2$ : degree  $n = 2$ ,  $p_0 = 0$ ,  $S^0 = S^t$ , solution  $p = (0 \ 0 \ 0)^T$

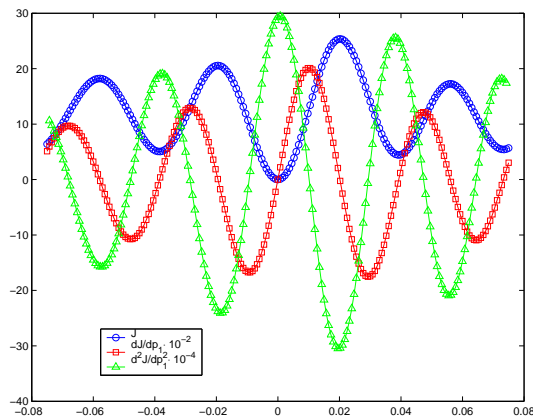


Figure 11: functional, first and second derivatives w.r.t.  $p_1$  with  $p_2 = 0$

Two configurations can be considered. if one has a good estimation of an initial shape, which can be the case for practical engineering problems, this method can be performant. An example is given in Figure 12 where the initial shape is relatively close to the target one. This configuration has been solved with degree  $n_x = 15, n_z = 1$ .  $S^0$  is a deformation of the same degree.

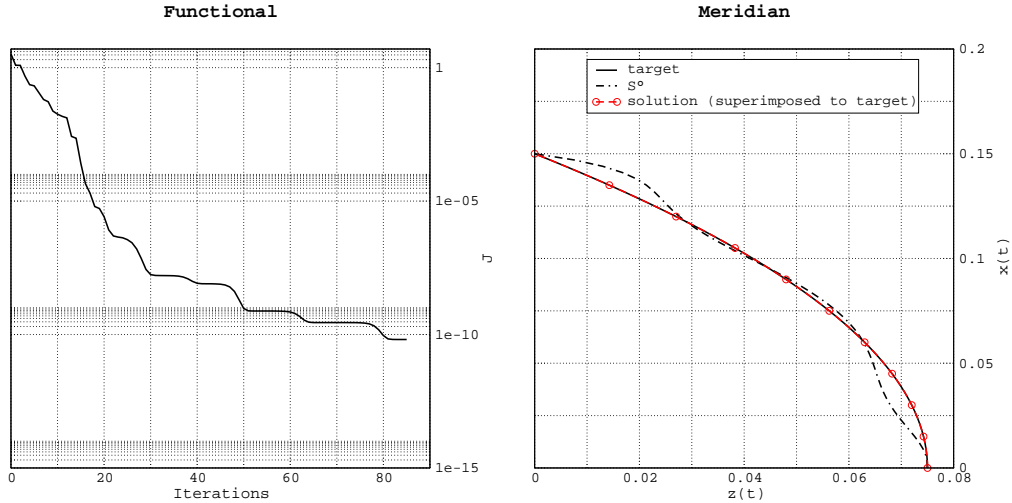


Figure 12: reconstruction of a deformed antenna

Else, if one has no *a priori* knowledge about a good initial shape or wants to check the robustness of its initial guess, evolutionary algorithms may be more suitable.

### 5.3 Semi-stochastic strategy: *Particle Swarm Optimization*

The *Particle Swarm Optimization (PSO)* method is an algorithm that was first developed to simulate bird flocks or fish schools. It has then been introduced as an optimizer [13]. Nowadays there exists several improved versions but the main characteristics are well described in [14].

#### 5.3.1 Introduction to the algorithm

The *PSO* algorithm is a semi-stochastic, population-based, algorithm. The stochastic part comes from the use of random numbers to introduce diversity in the population. In so doing one hopes to avoid local minima through a global search. In the meanwhile some best solutions (to be defined) are kept in memory in order to influence the particles to search towards some directions.

Consider a classical unconstrained optimization problem in  $\mathbb{R}^n$ :

$$\min_{p \in \mathbb{R}^n} J(p) \quad (28)$$

for some continuous objective function  $J$ . The key idea is to generate a sequence of populations where at each iteration, each particle in the swarm moves in the search space with a certain velocity. The way the velocity is defined contains all the characteristics of the algorithm. Namely, it involves:

- the previous velocity (inertia);
- the swarm best solution (global trust);
- the personal best solution for each particle (self trust);
- a random effect.

Formally, the algorithm is described in Algorithm 1 and the algorithm variables are defined in Table 1.

Table 1: variable definition for PSO

$\mathbf{P}_i = \{p_i^j\}_{j=1}^N$	population at iteration $i$
$\mathbf{V}_i = \{v_i^j\}_{j=1}^N$	velocity vectors of the population at iteration $i$
$p^* = \text{Argmin}_{p \in \{p_0^*, \dots, p_i^*\}} J(p)$	best particle so far (global trust)
$p^{j*} = \text{Argmin}_{p \in \{p_0^j, \dots, p_i^j\}} J(p)$	best position found by particle $j$ so far (self trust)
$w_i$	inertia coefficient at iteration $i$
$w_c$	inertia reduction coefficient
$c_1$	self trust coefficient
$c_2$	global trust coefficient
$r_1, r_2$	random numbers

Note that there are a few parameters to determine. Once they are fixed and once the random numbers sequences are generated, the evolution of the swarm is determined. The retained solution after the last iteration is  $p^*$ . Since it is hard to define a convergence criterion, the number of iterations is usually fixed. However, there is a mechanical convergence when the inertia is reduced ( $w_c < 1$ ). In practice, there is no need to iterate when the inertia becomes smaller than a certain value. There is no diversity enough to look for other solutions. Therefore, if the standard deviation of the functional is smaller than a given value  $\epsilon$  before iteration  $imax$ , the algorithm stops.

---

**Algorithm 1** Particle Swarm Optimization
 

---

**Require:**  $N, c_1, c_2, w_0, w_c, imax, p_0, \mathcal{C}^0, \epsilon$  (parameters to be determined)

(1) First geometry :  $p_0, \mathcal{C}^0$

Compute  $J(p_0)$

$p^* \leftarrow p_0$

(2) Initialize uniformly the population and velocities

in some subsets  $D, D' \subset \mathbb{R}^n$

(3) Loop

**for**  $i \leftarrow 0$  to  $imax$  **do**

  compute  $J(p), p \in \mathbf{P}_i$

  update  $p^*, \{p^{j*}\}_{j=1}^N$

  set random numbers  $r_1$  and  $r_2$

$\mathbf{V}_{i+1} : v_{i+1}^j \leftarrow \underbrace{w_i v_i^j}_{inertia} + \underbrace{c_1 r_1 (p^{j*} - p_i^j)}_{self\ trust} + \underbrace{c_2 r_2 (p^* - p_i^j)}_{global\ trust}$

$\mathbf{P}_{i+1} : p_{i+1}^j \leftarrow p_i^j + v_{i+1}^j$

$w_{i+1} \leftarrow w_i w_c$  {inertia reduction}

**if**  $\text{std}(J) < \epsilon$  **then**

    quit loop

**end if**

**end for**

**return**  $p^*$

---

### 5.3.2 Application to the case

As in section 5.2, the stability of the algorithm around the solution is tested. Then the results for different degrees of parameterization are compared. In order to compare solutions for the same amount of computational work, the number of particles for each degrees is fixed to the value  $N = 70$ . Moreover, the following parameters are set for all experimentations:

Parameter	Value
$w_0$	1.2
$\alpha$	0.99
$c_1$	2
$c_2$	2
$\epsilon$	$10^{-2}$

**Method validation:** Figure 13 shows that the algorithm is able to find the shape parameters that give the target diagram ( $J \rightarrow 0$ ). However, compared to the descent method, the number of functional evaluations is much larger:  $7000 \gg 11$ .

The algorithm is built so that there is a more global search at the beginning (with a large inertia) and a more local search at the end around the best solution found so far by the swarm. Since the first evaluation is close to the solution, one has to wait until the inertia becomes small enough so that  $p^*$  has enough influence on the swarm. Therefore, in this validation case, the number of evaluation is very large and there is not much improvement during the first iterations, i.e., during the global search. This can be seen in Figure 14. Clearly, the swarm begins to converge around iteration 40: the functional mean begins to decrease and a better particle is found almost at each step (see Figure 13).

**Case resolution:** Now the case is solved for the same degrees as that used in section 5.2:  $n = 2, 4$ , and 6. One can see in Figure 15 the results in terms of shape and functional.

With degree  $n = 2$  one can see that the algorithm converges towards a shape that approximates quite well the target. As noted previously, with this degree the target is not reachable. Therefore the functional converges towards some non-zero value. Note also that the numerical solution is obtained after 50 iterations (3500 functional evaluations).

With  $n = 4$  the result is even better. The shape is actually almost superimposed to the target (it appears here that the chosen deformation  $\Delta z$  is almost of degree 4, but compared to the validation case, the value of  $J$  at convergence is still greater). However the algorithm is slower to converge: 100 iterations are needed to reach the best value (7000 functional evaluations).

At last, with  $n = 6$  one can see that 150 iterations are not sufficient to converge. The functional is still decreasing but far from 0.

More generally, the greater the degree, the slower the algorithm converges. It is quite intuitive that it is more difficult to explore a much larger domain with fewer particles. In [11] this phenomenon is called the ‘‘curse of dimensionality’’.



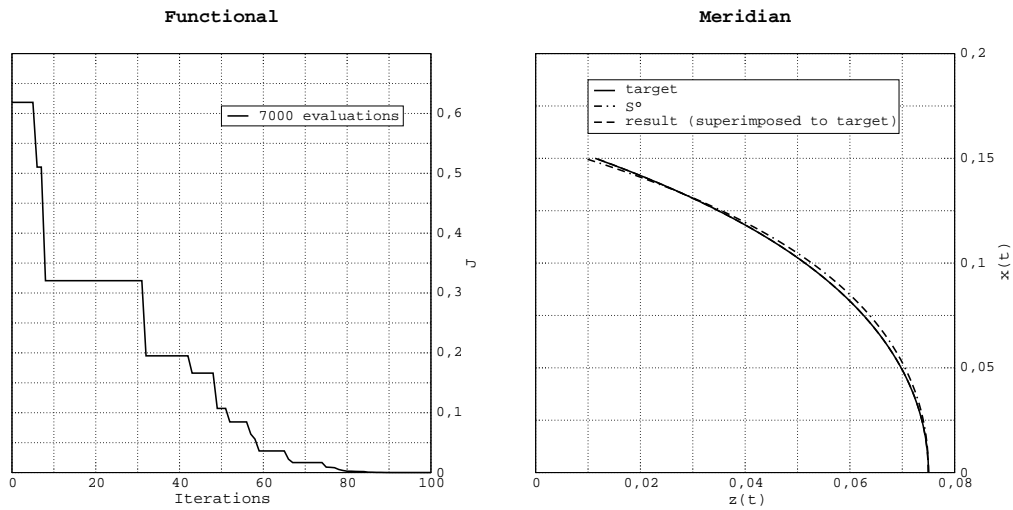


Figure 13: stability around the solution

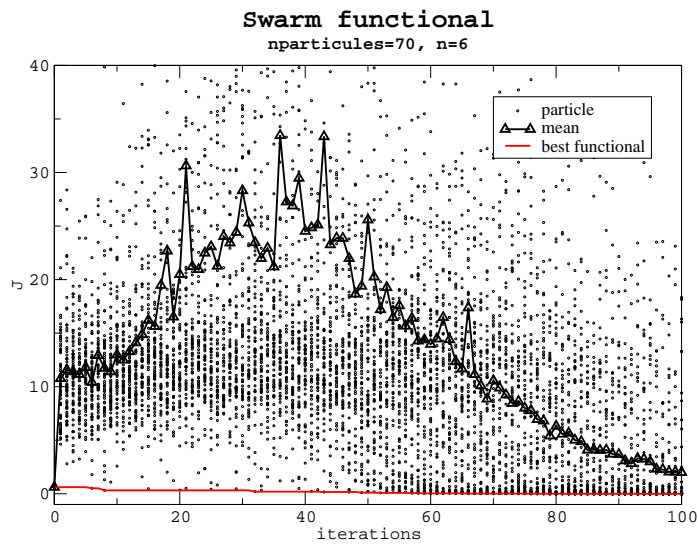
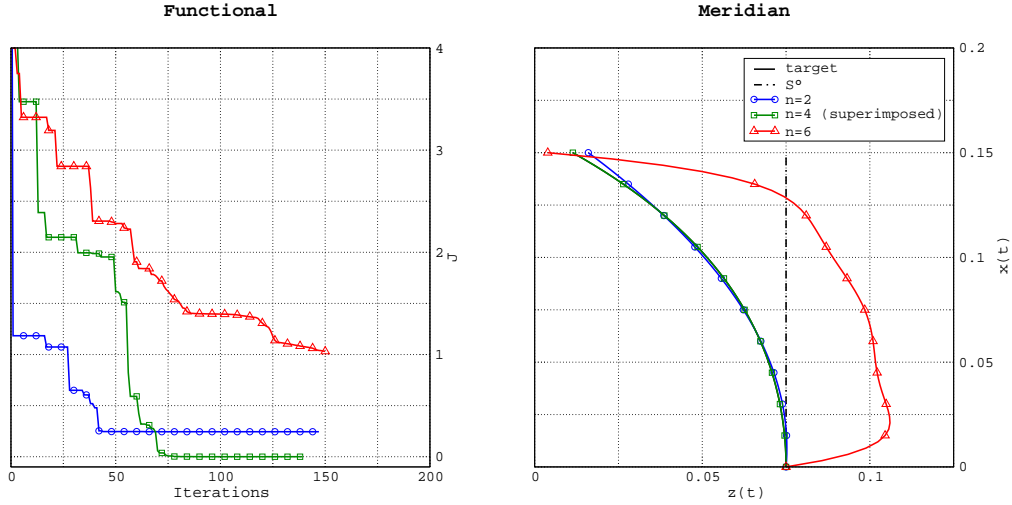


Figure 14: swarm functional values ; when no better solution has been found at another point of the search space (significantly far), the whole swarm is attracted by the best particle, thus the functional mean value converges towards the best value (here the convergence begins around the 50th iteration).


 Figure 15: numerical solutions w.r.t. degree, *PSO*

In the following section a strategy is proposed to obtain a solution with the highest accuracy (with  $n = 6$ ) for at most the same amount of computational work (at most 150 iterations with 70 particles).

## 5.4 Towards a multilevel method with *PSO*

Hierarchical approaches for shape optimization with Bézier parameterization and inspired by the multi-grid theory has been recently introduced and applied in aerodynamic design [8, 9]. From simple level increase to Full Multi-Grid, this strategy is based on embedded parameterizations of increasing degree. This previous methods have been used with a gradient-based and a simplex optimizers. Here, a semi-stochastic multilevel algorithm is considered. The simple case, with only degree elevation, is considered.

### 5.4.1 Multilevel *PSO*

Starting from a parameterization of degree  $n = 2$  the best shape parameters noted  $p^{*2}$  is kept in memory. According to proposition 3.2 the same deformation can be expressed in the basis of degree  $n = 4$  by 2 successive degree elevations. The corresponding shape parameters is noted  $p^{*4}$ . Hence one can use  $p^{*4}$  as first best solution for another few iterations of *PSO*. In addition, since one has now an approximation of the solution from the coarse parameterization, the initialization domain is reduced around it and the inertia value is kept. More precisely, let  $B_\infty(p^*, \rho) \equiv \{p \in \mathbb{R}^n ; \|p - p^*\|_\infty \leq \rho\}$ . Thus the first initialization domain is defined as  $D_0 = B(p_0, \rho_0)$  for some  $\rho_0 > 0$  and then  $\rho$  is reduced at each elevation.

And so on for all considered degrees. The new algorithm reads in Algorithm 2. This strategy intends to avoid the issues due to the large dimensionality on the finer parameterization.

---

**Algorithm 2** Multilevel Particle Swarm Optimization
 

---

**Require:**  $N, c_1, c_2, w_0, w_c, imax, p_0, \mathcal{C}^0, \rho_0, \beta$  (parameters to be determined)

**Require:**  $k_i, i = 1 \dots n$ : degree of parameterization for level  $i, k_1 < k_2 < \dots < k_n$

(1) First geometry :  $p_0$  in  $\mathbb{R}^{k_1}, \mathcal{C}^0$

Compute  $J(p_0)$

$p^{*k_1} \leftarrow p_0$

$D_0 \leftarrow B_\infty(p_0, \rho_0)$

(2) Loop over degrees

**for**  $i \leftarrow 1$  to  $n - 1$  **do**

$imax$  iterations of *PSO*

    update  $p^{*k_i}$

$p^{*k_{i+1}} \leftarrow \text{elevation}(p^{*k_i})$

$\rho_{i+1} \leftarrow \beta \rho_i$

$D_{i+1} \leftarrow B_\infty(p^{*k_{i+1}}, \rho_{i+1})$

**end for**

(3) Finest degree

$imax$  iterations of *PSO*

update  $p^{*k_n}$

$p^* \leftarrow p^{*k_n}$

**return**  $p^*$

---

#### 5.4.2 Application to the case

The following parameters are used:

Parameter	Value
$w_0$	1.2
$\alpha$	0.99
$c_1$	2
$c_2$	2
$\rho_0$	0.150
$\beta$	0.20

In addition, let  $imax = 40$  for the first two levels and  $imax = 70$  for the last one. In Figure 16 the results obtained with the multilevel algorithm and the single parameterization *PSO* are compared. It is important to reduce the initialization domain on the finer grids.

If not, one would not make use of the information of the best solution from the previous parameterization.

**Functional**

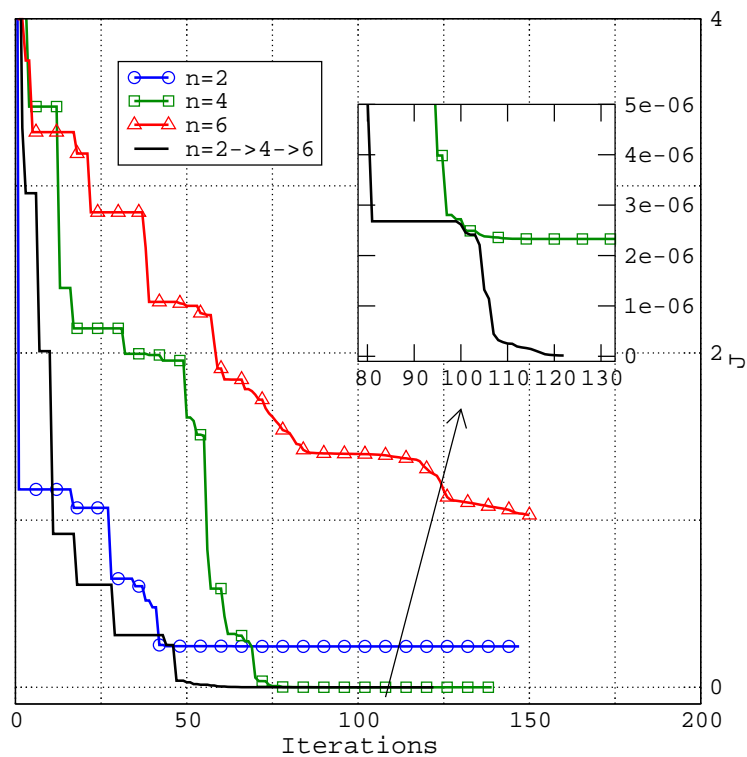


Figure 16: comparison of the functional evolution for single *PSO* with multilevel *PSO* with progressive degree elevation  $2 \rightarrow 4 \rightarrow 6$

## 6 Conclusion

A conformation of axisymmetric antennas problem has been considered. The functional reads as the  $L_2$ -norm of the difference between a measured field (or target field) and the computed field, given a geometry of reflector. A *Free-Form Deformation* method has been adopted to represent parametrically the deformation of an initial shape. The optimization problem consists in finding the minimal value of the functional (in particular, 0 for a reconstruction problem).

With such a configuration it has been shown that the problem is usually hard to solve since the functional is highly multimodal, even with 2 degrees of freedom. Thus, when there is no *a priori* knowledge about the initial shape, a deterministic method has no chance to converge towards a global solution. However, when a better initial shape is given, the diagram can be significantly improved.

Then a semi-stochastic method, the *Particle Swarm Optimization* method, has been discussed. First, as expected, it appears that such a method is more robust since it is no more sensitive to local minima. However, this method requires more evaluations of the functional. When the degrees of freedom number increase, the problem becomes almost unsolvable. Hence, only solutions with small degrees of freedom, and thus unaccurate, can be obtained.

At last a multilevel *Particle Swarm Optimization* algorithm, inspired from multilevel optimization methods used in aerodynamics, has been proposed. This latter algorithm appears to be the best method for such problems since it is both robust and accurate.

## References

- [1] P. F. COMBES, *Micro-Ondes*, Dunod (1997) vol. 2 chapters 5, 11.2, 12.3.1, 13.4.2
- [2] R. BILLS, *Problème extérieur pour les equations de Maxwell Application aux antennes de révolution*, Thèse de doctorat, Université de Nice (1982)
- [3] S. J. ORFANIDIS, *Electromagnetic Waves and Antennas*, Rutgers University (2004), <http://www.ece.rutgers.edu/~orfanidi/ewa>
- [4] WIKIPÉDIA, *Antenne* — *Wikipédia, l'encyclopédie libre*, (2006), <http://fr.wikipedia.org/w/index.php?title=Antenne&oldid=7941063>
- [5] M. LEVEQUE, *Conformation et optimisation des antennes à réflecteur*, stage réalisé à France Télécom R&D La Turbie (2000)
- [6] P. DUBOIS, *Optimisation de structures rayonnantes métalliques 3D par déformation de surfaces iso-niveaux en régime harmonique*, Thèse de doctorat, École des mines de Paris (2005)
- [7] T. SEDERBERG AND S. PARRY, *Free-Form Deformation of Solid Geometric Models*, Computer Graphics (1986)
- [8] J.-A. DÉSIDÉRI, B. ABOU EL MAJD, A. JANKA, *Nested and Self-Adaptive Bézier Parameterizations for Shape Optimization*, International Conference on Control, Partial Differential Equations and Scientific Computing, Beijing, Science Press (Sept. 13-16, 2004)
- [9] J.-A. DÉSIDÉRI, *Two-level Ideal Algorithm for Parametric Shape Optimization*, Journal of Numerical Mathematics (2006)
- [10] J. PICARD, *Paramétrisation hiérarchique pour l'optimisation d'un réflecteur*, stage réalisé à l'INRIA Sophia Antipolis (2004)
- [11] P. E. GILL, W. MURRAY, M. H. WRIGHT, *Practical Optimization*, Academic Press (1986)
- [12] J. L. ZHOU, A. L. TITS, C. T. LAWRENCE, *User's Guide for FFSQP Version 3.7: a Fortran Code for Solving Constrained Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality and Linear Constraints*, Institute for Systems Research, University of Maryland, College Park, MD 20742 (April 1997)
- [13] J. KENNEDY, R. EBERHART, *Particle Swarm Optimization*, IEEE International Conference on Neural Networks, Vol.4 pp.1942-1948 (Nov./Dec. 1995)
- [14] G. VENTER, J. SOBIESZCZANSKI-SOBIESKI, *Particle Swarm Optimization*, AIAA Journal, Vol.41 No.8 (August 2003)



---

Unité de recherche INRIA Sophia Antipolis  
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes  
4, rue Jacques Monod - 91893 ORSAY Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)  
<http://www.inria.fr>  
ISSN 0249-6399