MULTI-LEVEL GRADIENT-BASED METHODS AND PARAMETRISATION IN AERODYNAMIC SHAPE DESIGN

M. Martinelli* — **F. Beux***

* *Scuola Normale Superiore di Pisa, Piazza dei Cavalieri, 7 - 56126 Pisa (Italy)*
{max.martinelli,fbeux}@sns.it

ABSTRACT. *The present study focuses on multi-level approaches in the context of discrete gradient-based methods for aerodynamic shape design. More precisely, the minimisation is done alternatively on different control subspaces according to multigrid-like cycles, providing at each sub-level a particular gradient preconditioning. Starting from an existing multi-level gradient-based formulation associated to shape grid-points coordinates, a possible generalisation to more compact shape representations is proposed through the construction of adequate sets of embedded shape sub-parametrisations. The behaviour of the new formulation is illustrated on different 2D inverse problems for inviscid flows.*

RÉSUMÉ. *Cette étude est consacrée aux stratégies multiniveaux pour les méthodes de descente utilisant le calcul d'un gradient discret, dans le cadre de l'optimisation de formes aérodynamiques. Plus précisement, la minimisation s'effectue sur différents sous-espaces de contrôle utilisés cycliquement, imitant ainsi les stratégies multigrilles. A chaque sous-niveau correspond un spécifique préconditionnement du gradient. Partant d'une formulation basée sur l'utilisation comme variables de contrôle des coordonnées des points du maillage sur la frontière, on propose la généralisation à des représentations plus compactes de la forme, ceci grace à la construction d'adéquates sous-paramétrisations de forme emboitées. Le comportement des méthodes proposées est illustré sur différents exemples de problèmes inverses bidimensionels pour écoulements non visqueux.*

KEYWORDS: *multi-level methods, aerodynamic shape parametrisation, optimum shape design*

MOTS-CLÉS : *méthodes multiniveaux, paramétrisation de forme aérodynamique, conception optimale de forme*

1re soumission à *REMN*, le 31/12/06

1. Introduction to multi-level approaches in aerodynamic shape design

Due to the ripeness reached by computational fluid dynamics (CFD) combined with the rapid advances of computational power, research in the field of aerodynamic shape design has experienced a large development in the last years, allowing to deal with more and more complex optimisation problems. However, shape optimisation for aerodynamic applications remains a costly task since the system of governing flow equations (as, for instance, Euler or Navier-Stokes equations) should be solved, many a time during the whole procedure. Thus, even if significant progress have been done for optimisation tools and related techniques (as, for instance, the use of adjoint approaches in the context of gradient-based methods), the improvement of optimisation algorithm efficiency still appears as an important goal.

On another hand, to deal with complex engineering design optimisation, different multi-level or multi-scale approaches, in which the whole problem is decomposed in several simpler sub-problems to be solved in a predetermined sequence, have been developed (see e.g. (Migdalas *et al.*, 1997; Schwabacher *et al.*, 1998)). Each optimisation sub-problem can differ according to objective function, constraints, design space and/or optimisation algorithm allowing a better treatment of complex systems (multidisciplinary design, multiple local optima, large-scale system, multi-objective optimisation, ...). Alternatively, efficiency can be also increased using various degrees of fidelity, i.e. varying the complexity of the physical modelling and/or the accuracy of the numerical approach (see e.g. (Alexandrov *et al.*, 2001)). Note that a particular case of low fidelity model can be obtained through the use of coarse meshes, in which the flow solution is computed more easily and at a lower cost. For instance, in the field of aerodynamic design optimisation, in (Feng *et al.*, 1995), a reduced Hessian SQP algorithm is combined with a solution refinement, while in (Dadone *et al.*, 2000), a progressive optimisation is proposed in which starting from a low accurate computation of the sensitivity derivatives (using coarse mesh and partially converged flow solutions) the degree of accuracy is progressively increased during the optimisation process. A similar idea is proposed in (Pironneau *et al.*, 2002) (see also Chapter 6 of (Mohammadi *et al.*, 2001)), in which mesh refinement is combined with approximate gradients in order to speed up the convergence on the finest mesh of the descent algorithm. Methods based on multigrid principles can be the successive step; some works can be found in literature in which multigrid-like techniques have been applied to optimal control problems involving partial differential equations. For instance, the MG/Opt algorithm (Nash, 2000) recursively uses coarse resolution problems (coarse mesh) to generate search directions at a cheaper cost for finer resolution problems. For further examples, we refer, for instance, to (Gelman *et al.*, 1990; Dreyer *et al.*, 2000; Borzi, 2003; Gratton *et al.*, 2004). In the context of aerodynamic design optimisation, in (Kuruvila *et al.*, 1994) a one-shot method, in which the flow and the sensitivities are simultaneously solved, is coupled with a multigrid approach. In this formulation, the design variables corresponding to low-frequencies of the shape are updated on a coarse level (i.e., a coarse mesh) while the other design variables are updated on a finer level. Finally, in (Catalano *et al.*, 2005; Catalano *et al.*, 2007),

the progressive optimisation, introduced in (Dadone *et al.*, 2000), is coupled with a multigrid-aided finite-difference approach, in which the gradient is obtained through finite-difference sensitivities computed using only flow solutions on a coarse mesh.

Another approach, also based on multi-level concepts and ideated for gradient-like methods, has been introduced in (Beux *et al.*, 1994). In this preconditioned gradient method, the minimisation is done alternatively on different subsets of control parameters according to multigrid-like cycles. More particularly, using shape grid-point coordinates as design variables, a hierarchical parametrisation was defined considering different subsets of parameters extracted from the complete parameterisation, which can be prolongated to the higher level by linear mapping. This approach acts as a smoother and, on another hand, makes the convergence rate of the gradient-based method low dependent of the number of control parameters. The good behaviour observed in different numerical experiments (Beux, 1994; Beux *et al.*, 1994), have been also corroborated by a theoretical view point in (Guillard, 1993; Guillard *et al.*, 1995). Note that, contrary to the other approaches based on multigrid concepts, only one computational mesh is employed since the coarseness acts only on the number of design parameters. The increase of efficiency is only related to the faster convergence obtained considering less degrees of freedom and to the improvement on convergence rate typical of multigrid techniques. Different extensions of the original approach have been, successively, proposed: in (Held *et al.*, 2002) the same hierarchical parametrisation is associated with a finite-difference/one-shot formulation, in (Marco *et al.*, 1997) the generalisation to 3D case involving unstructured meshes is done through the use of agglomeration technique while an additive multi-level preconditioner has been also defined in (Koobus *et al.*, 1997; Courty *et al.*, 2006).

Another multi-level approach based on a family of embedded parametrisations has also been proposed in (Désidéri, 2003) (see also successive works, as e.g. (Abou El Majd *et al.*, 2004; Désidéri, 2007; Abou El Majd *et al.*, 2007)). However, contrary to the method introduced in (Beux *et al.*, 1994) and its different extensions, this approach is based on a polynomial representation of the shape through the use of Bézier curves, and is not specifically focused on gradient-based methods.

2. Optimum shape design problem in aerodynamics

2.1. *The Optimal shape problem in a fully discrete context*

The optimal shape problem consists in minimising a cost functional j with respect to some control variables α, which should characterise the shape. Moreover, for aerodynamic shape optimisation, j can not be expressed directly in a explicit way as a function of α since it also depends on the flow variables. Indeed, for each shape configuration, and thus, for each choice of α, a particular flow is obtained by solving the governing equations, i.e typically Euler or Navier-Stokes equations. Note that we consider an optimisation in a discrete context in which the optimisation algorithm is applied to the problem already fully discretised, i.e. with both the discrete govern-

ing equations and the discrete cost functional. Then, the unconstrained optimal shape problem can be written as follows:

$$\text{Find } \alpha_{opt} \in I\!R^p \text{ such that } j(\alpha_{opt}) = \min_{\alpha \in I\!R^p} j(\alpha) \qquad [1]$$

in which the cost functional j can be defined introducing $J : I\!R^p \times I\!R^N \rightarrow I\!R$ such that:

$$\forall \alpha \in I\!R^p \quad j(\alpha) = J(\alpha, W(\alpha)) \qquad [2]$$

where α is a discrete set of parameters (see Sec. 2.3) while W represents the values of the flow variables at each point of the computational mesh.

Furthermore, the discretised shape is fully determined by the coordinates of the grid-points localised on the shape. Thus, the control variables influence the discrete cost functional only through these coordinates, which can be introduced as intermediate variables. More precisely, let us consider $\mathcal{L} : I\!R^p \rightarrow I\!R^q$, the operator which, for each set of control parameters, furnishes the corresponding set of shape grid-points coordinates. Then, instead of [2] the cost functional can be expressed as follows:

$$\forall \alpha \in I\!R^p \quad j(\alpha) = \mathcal{I}(\mathcal{L}(\alpha)) = I\big(\mathcal{L}(\alpha), W(\mathcal{L}(\alpha))\big) \qquad [3]$$

in which $\mathcal{I} : I\!R^q \rightarrow I\!R$ and $I : I\!R^q \times I\!R^N \rightarrow I\!R$ are defined by $j = \mathcal{I} \circ \mathcal{L}$ and $J = I \circ \mathcal{L}$ respectively.

2.2. *Computation of the sensitivity derivatives*

If a gradient-based method is used as optimisation strategy, then, the computation of cost functional derivatives with respect to control variables is required. The sensitivity derivatives can be obtained by the differentiation of [2] (or [3]). This task can be highly tedious since the governing equations are also involved through the flow variables. To avoid this exact differentiation, the sensitivity derivatives can be approximated by finite differencing in which the flow solver is used only as a black box. Thus, the resulting approach is very easy to implement and can be applied in a rather general context. Nevertheless, this approach requires a careful parameter monitoring in order to obtain an accurate gradient approximation, and on the other hand, gives dramatic low computational performances as soon as not only a very low number of control variables is used. Alternatively, in the direct differentiation or flow sensitivity approach, the Gâteaux derivatives with respect to each component direction are exactly computed. But, the computational cost problem for a large number of control variables is still present, since, for one gradient computation, p (p being the number of design parameters) linearised systems of large dimension should be solved. Finally, an efficient computation of the exact discrete gradient can be achieved through an adjoint formulation (see, e.g. (Giles *et al.*, 2000; Jameson, 2003)). Indeed, the gradient evaluation requires to solve only one extra linear system (the adjoint system), and thus,

is independently of the number of design variables. Consequently, at present, this formulation seems to be the more suitable way to solve complex aerodynamic shape optimisation problem in the context of gradient-based methods. Nevertheless, to avoid the very hard task of a differentiation implementation by hand-coding, automatic differentiation tools have been also developed (see e.g. (Giering *et al.*, 2005; Hascoët *et al.*, 2005)). An alternative or additional approach, often used in presence of complex physical models and numerical discretisation (Navier-Stokes equations with a turbulent model, high-order schemes, non-structured meshes, ...), is to freeze or approximate some steps in the differentiation of the flow solver (see e.g. (Nemec *et al.*, 2002)). We refer to (Dwight *et al.*, 2006; Carpentieri *et al.*, 2007) for a study on the effect on the gradient accuracy of the various approximations of the discrete adjoint computation. A more drastic approach has been introduced in (Mohammadi, 1997), in which the adjoint computation is dropped out, on the one hand, by adding some intermediate geometrical quantities in the differentiation, and on the other hand, by neglecting the flow derivatives. This incomplete gradient formulation is based on the fact that when the objective functional is defined as a boundary integral of quantities evaluated on the shape, sensibilities with respect to the geometrical quantities give the main contribution to the gradient value. In (de' Michieli Vitturi *et al.*, 2006) this approach has been coupled with a multi-level method allowing to consider a completed gradient computation in which the flow derivatives are also (at least partially) taken into account. Moreover, efficient approaches based on finite-difference sensitivities have been also proposed; indeed, in (Catalano *et al.*, 2005) a progressive optimisation coupled with a multigrid-aided finite-difference is considered while an one-shot method coupled with a multi-level strategy is used in (Held *et al.*, 2002). Finally, for more details and references on sensitivity analysis for aerodynamic shape optimisation, we refer, for instance, to (Newman III *et al.*, 1999) and (Mohammadi *et al.*, 2001).

In the present study, an exact hand-coding discrete adjoint approach is used for a Euler stationary flow solver based on an unstructured finite-volume first-order spatial discretisation and a pseudo-unsteady approach associated with a linearised implicit algorithm. Finite-difference sensitivities through the formulation proposed in (Held *et al.*, 2002) are also considered in Sec. 6.

2.3. *Parametrisations for aerodynamic shape representation*

An important ingredient which should be also specified in the optimisation process is the representation of the shape, which is defined through the choice of the control parameters. Indeed, the shape parametrisation plays a crucial role for the shape optimisation since it directly acts on the accuracy of the final solution and on the efficiency of the particular optimisation strategy.

The use of shape grid-point coordinates as design variables appears the more natural approach since, in this case, the parameterisation is directly correlated with the explicit representation of the discrete shape. Furthermore, for optimisation algorithms based on an exact discrete gradient, the computation of the sensitivity derivatives are

simplified since, in this case, [2] and [3] coincide. Nevertheless, since the geometry is modified by moving individual grid points, non-smooth profiles are often obtained, particularly during intermediate phases of the convergence to optimum (see, e.g. (Beux *et al.*, 1992)). Moreover, the large number of variables involved in this case has a negative effect on the computational cost (slow convergence and, possibly, large number of cost functional evaluations). As a matter of fact, the multi-level strategy introduced in (Beux *et al.*, 1994) and described in Sec. 3.2 was defined exactly in order to reduce these drawbacks. Note that the lack of shape smoothness, particularly critical for shape grid-points parametrisation, can be also linked with a regularity loss of the gradient with respect to the control variables, already verified in the continuous case (see, e.g. (Courty *et al.*, 2006)). To avoid oscillations, in many works dealing with a shape grid-point parameterisation, a smoothing is applied (see e.g. (Mohammadi *et al.*, 2001; Reuther *et al.*, 1995)).

On the other hand, a classical approach for the parameterisation of the shape is to use a polynomial representation which permits a compact description of the shape with only few parameters. For instance, Bézier control points, i.e. coefficients in a basis of Bernstein polynomials, can be used as design variables. The Bézier representation has suitable properties at an algorithmic level (efficient recursive algorithms), but also, with regard to geometrical aspects. In particular, this shape representation is well adapted to deal with geometric constraints since a convex hull property is verified and the curve derivates are easily available (see, e.g. (Farin, 1990)). Furthermore, the Bézier curves act as a basic tool to define other representations as B-splines and non-uniform rational B-spline (NURBS) more suitable for high-degree polynomial and non smooth geometries respectively.

Another possible choice, frequently considered in the context aerodynamic shape design (typically, for airfoil or wing design), is to represent the shape through a linear combination of given geometric shapes. The control variables are, then, the coefficients in this basis of shape functions. In this case, few parameters are sufficient to obtain a good shape representation, but, on the other hand, the final solution is highly related to the choice of the particular basis. Thus, this representation yields a priori a smaller design space with respect to the parametrisations based on shape grid-points or polynomial control points. The basis is, in general, composed of existing geometric shapes or alternatively, a given base shape and a set of modified shapes obtained from the first one through some perturbation functions, such as the Hicks-Henne analytical functions (see (Hicks *et al.*, 1978)). Moreover, in order to improve the completeness of the design space and thus, avoid the presence of nearly linear dependent functions, some authors use orthogonal functions obtained through a Gram-Schmidt orthogonalisation (see, (Kuruvila *et al.*, 1994; Chang *et al.*, 1995; Catalano *et al.*, 2005)) or analytically (e.g., through the use of Chebychev polynomials in (Carpentieri *et al.*, 2007)).

The three kinds of parametrisations, described here, represent a large range of widely used approaches, and, have been considered in this study in the framework of

multi-level methods. For a more complete overview of possible shape parametrisations, we refer, as example, to (Samareh, 2001) or (Selmin, 2007).

3. Multi-level gradient-based approaches for shape design

3.1. *Change of Hilbert control space*

Multi-level methods in the context of optimum shape design, as initially proposed in (Beux *et al.*, 1994), are based on a change of control space. More precisely, let us consider the optimisation of a differentiable functional $j : U \to I\!R$ in a Hilbert space U. Then, instead of a direct minimisation of j in U, one can also envisage a minimisation of j in the subset $f(V) \subset U$, in which V is a second Hilbert space and f an application from V to U. It can be formulated, equivalently, as the minimisation of $\mathcal{J} = j \circ f$ in V, i.e.:

$$\text{Find } \alpha_{opt} \in V \text{ such that } \mathcal{J}(\alpha_{opt}) = \min_{\alpha \in V} [j \circ f](\alpha) \qquad [4]$$

The Fréchet derivative of $\mathcal{J}$ at $\alpha \in V$ can be expressed as follows:

$$\forall h \in V \quad \mathcal{J}'(\alpha)(h) = [j \circ f]'(\alpha)(h) = j'\left(f(\alpha)\right)\left(f'(\alpha)(h)\right)$$

Since $f'(\alpha) \in \mathcal{L}(V,U)$, i.e is a linear continuous application from V to U, the following relation can be also obtained in terms of gradient for any $h \in V$:

$$\begin{aligned} \langle \mathrm{grad}_V \mathcal{J}(\alpha), h \rangle_V &= \langle \mathrm{grad}_U j(\gamma), f'(\alpha)h \rangle_U \\ &= \langle \left(f'(\alpha)\right)^* \mathrm{grad}_U j(\gamma), h \rangle_V \end{aligned} \qquad [5]$$

where $\gamma = f(\alpha)$, $\left(f'(\alpha)\right)^* \in \mathcal{L}(U,V)$ is the adjoint of $f'(\alpha)$ and $\langle .,. \rangle_U$ and $\langle .,. \rangle_V$ are the inner products associated to U and V respectively.

Furthermore, let us consider the particular case in which f affine, and thus, it exists $b \in U$ and $P \in \mathcal{L}(V,U)$ such that $f : \alpha \to P\alpha + b$. Then, since in this case $f'(\alpha) = P$ for any $\alpha \in V$, solving the minimisation problem [4] through a gradient descent method corresponds to the following iterative algorithm:

$$\alpha_0 \in V \text{ given}, \quad \text{for } r \geq 0 \quad \alpha_{r+1} = \alpha_r - \omega_r P^* \mathrm{grad}_U\, j(f(\alpha_r))$$

Nevertheless, applying the operator f permits to go back to space U, and thus, to obtain:

$$f(\alpha_{r+1}) = P\alpha_{r+1} + b = f(\alpha_r) - \omega_r P P^* \mathrm{grad}_U\, j(f(\alpha_r))$$

Thus, considering as initial solution $\gamma_0 = f(\alpha_0)$, the following iterative algorithm is finally defined in U:

$$\text{for } r \geq 0 \quad \gamma_{r+1} = \gamma_r - \omega_r P P^* \mathrm{grad}_U\, j(\gamma_r) \qquad [6]$$

The resulting algorithm is a weak descent method in U (see (Beux *et al.*, 1994; Martinelli *et al.*, 2007b) for a proof), and can be also interpreted as a preconditioned gradient method.

Furthermore, since we are interested to the shape optimisation problem [1] in a discrete context, $U = I\!R^p$ while the second Hilbert space is, typically, $V = I\!R^{\bar{p}}$. Then, the linear operator P and its adjoint P^* are associated to a matrix $M \in I\!R^{p \times \bar{p}}$ and its transpose respectively, and thus, the algorithm can be simply rewritten as:

$$\text{for } r \geq 0 \quad \gamma_{r+1} = \gamma_r - \omega_r \, M M^T g_r \qquad [7]$$

with $g_r = \text{grad}\, j(\gamma_r) \in I\!R^p$.

3.2. *A hierarchical parametrisation based on shape grid-points*

An optimisation algorithm based on control space change as presented in Sec. 3.1 has been initially proposed in (Beux *et al.*, 1994) for the case of a linear operator, i.e. for $f = P$. In this study, the ordinates of the grid-points, localised on the shape which should be optimised, have been chosen as control variables γ. Then, a set of points, extracted from the complete set of shape grid-points, is considered as sub-parametrisation α. For this choice of shape parametrisation and sub-parametrisation, it seems natural to define the linear prolongation operator f through interpolation. It has been shown that taking an Hermitian interpolation of degree 3, an enough smooth parametrisation is then obtained with satisfying numerical results.

Moreover, instead of considering a single space V, the cost functional is minimised alternatively on different control subspaces of decreasing dimension. More precisely, a family of embedded sub-parametrisations is considered, in which for each increase of level the number of points is doubled. At a particular level l, the prolongation operator from level l to the finest one L, i.e. to the complete parametrisation, is defined by

$$P^{(l)} = P_{L-1}^{L} \circ \cdots \circ P_{l+1}^{l+2} \circ P_{l}^{l+1}$$

in which P_i^{i+1} is the cubic interpolation used for the prolongation from level i to the next one. In practice, at each optimisation iteration r corresponds a particular level l, and following [6], minimising on this coarse level l corresponds to replace the gradient g_r by the descent direction $p_r^{(l)} = P^{(l)}(P^{(l)})^* \, g_r$. This means that the gradient is projected on a coarse level and then prolongated back to the fine level, which is equivalent to the addition of a high frequency filter. Note that, only the gradient on the finest level is computed while the effects of the minimisation on a coarse level are taking into account by the preconditioner $P^{(l)}(P^{(l)})^*$. The choice of the particular subspace, at each optimisation iteration, is determinate by a strategy of level changes similar to multi-level/multigrid strategies used for the resolution of partial differential equations (as, for instance, V-cycles).

Note that the multi-level approach elaborated in (Beux *et al.*, 1994) is strongly linked to the particular type of parametrisation used. Nevertheless, the formulation

described in Sec. 3.1 is rather more general since, providing that the prolongation operator be affine, any type of design variables and sub-parametrisations can be proposed.

3.3. *Generalisation to other kinds of parametrisation*

The multi-level method, as described in the previous sections, is based on the idea of control space change. Moreover, it also need the definition of a family of sub-parametrisations and the corresponding prolongation operators. In Sec. 3.2, a set of sub-parametrisations has been found in a natural way considering shape grid-points as control parameters.

More generally, let us, now, consider an optimal shape design associated to α, a generic set of control variables. Nevertheless, as pointed out in Sec. 2, the shape grid-points coordinates can be used as intermediate variables. Then, since the cost functional can be expressed as $j = \mathcal{I} \circ \mathcal{L}$, the following relation between the gradients of j and $\mathcal{I}$ can be obtained in a similar way as done to obtain [5]:

$$\forall \alpha \in I\!R^p \quad \mathrm{grad}_\alpha j(\alpha) = \left(\mathcal{L}'(\alpha)\right)^* \mathrm{grad}_\gamma \mathcal{I}\left(\mathcal{L}(\alpha)\right)$$

where the subscript α and γ denote, here, that the gradient is in $I\!R^p$ and $I\!R^q$ respectively. Thus, the gradient descent method for the minimisation of j with respect to α corresponds to the following iterative algorithm:

$$\alpha_0 \in I\!R^p \text{ given}, \quad \text{for } r \geq 0 \quad \alpha_{r+1} = \alpha_r - \omega_r \left(\mathcal{L}'(\alpha_r)\right)^* \mathrm{grad}_\gamma \mathcal{I}(\mathcal{L}(\alpha_r))$$

After the computation of α_{r+1}, the shape grid-points coordinates should be also updated. This updating is done through $\mathcal{L}$, which lies the control variables to the shape grid-points coordinates γ, in the following way:

$$\text{for } r \geq 0, \quad \gamma_{r+1} = \mathcal{L}(\alpha_{r+1}) = \mathcal{L}\left(\alpha_r - \omega_r \left(\mathcal{L}'(\alpha_r)\right)^* \mathrm{grad}_\gamma \mathcal{I}(\gamma_r)\right) \qquad [8]$$

with $\gamma_0 = \mathcal{L}(\alpha_0)$.

Note that, if the shape parametrisation is relied to the coordinates of the shape grid-points by an affine application, and thus, in particular, it exists a matrix $M \in I\!R^{q \times p}$ such that for any $h \in I\!R^p$ we have $\left[\mathcal{L}'(\alpha_r)\right](h) = Mh$, then [8] can be rewritten as follows:

$$\gamma_{r+1} = \gamma_r - \omega_r M M^T \mathrm{grad}_\gamma \mathcal{I}(\gamma_r)$$

In this case, a descent direction is obtained considering as control variables α as well as γ. From the point of view of an optimisation with respect to γ, α acts as a sub-parametrisation, and, we exactly recover the formulation of Sec. 3.1 taking, here, $f = \mathcal{L}$, $j = \mathcal{I}$ and $\mathcal{J} = j$. Thus, the approach proposed in (Beux *et al.*, 1994) corresponds to choose as control parameters the coordinates of a subset of the shape grid-points.

On another hand, if it is possible to find an affine application from $\mathbb{R}^{\bar{p}}$ to $\mathbb{R}^p$ (associated to a matrix $D \in \mathbb{R}^{p\times\bar{p}}$), then, as done in Sec. 3.1, a preconditioned gradient method can be also defined for the optimisation of j with respect to α. It corresponds to the following iterative algorithm:

$$\alpha_0 \in \mathbb{R}^p \text{ given}, \quad \text{for } r \geq 0 \quad \alpha_{r+1} = \alpha_r - \omega_r DD^T \text{grad}_\alpha\, j(\alpha_r) \qquad [9]$$

Thus, two possible ways can be envisaged to generalize the multi-level approach: considering α as control variables as done in [9], or alternatively, considering γ as control variables if $\mathcal{L}$ is affine. In this last case, one can also combined the two kinds of preconditioners. Indeed, [9] can be expressed in terms of shape grid-points as follows:

$$\gamma_{r+1} = \mathcal{L}\left(\alpha_r - \omega_r DD^T \left(\mathcal{L}'(\alpha_r)\right)^* \text{grad}_\gamma\, \mathcal{I}(\gamma_r)\right)$$

Then, if $\mathcal{L}$ is an affine application, it can be also rewritten as:

$$\gamma_{r+1} = \gamma_r - \omega_r MD(MD)^T \text{grad}_\gamma\, \mathcal{I}(\gamma_r) \qquad [10]$$

In this case, the same algorithm can be interpreted as a preconditioned gradient method considering as control variables α (through [9]) as well as γ (through [10]). In particular, let us consider a parametrisation α with $\mathcal{L}$ affine and an adequate set of sub-levels associated to the matrices $D^{(l)}$, $l = 1, \cdots, L$. Then, [10] furnishes a practical way to define the different preconditioning matrices for the optimisation with respect to γ taking $M^{(l)} = MD^{(l)}$.

Note that, in order to really define a multi-level strategy, one should also introduce a notion of coarseness for the sub-levels. In particular, we should be able to define a family of sub-parametrisations with a decreasing number of parameters. Furthermore, a coarser level should correspond to a representation of the shape with lower frequencies. Finally, it will be also interesting to define, as done for the original formulation (Beux *et al.*, 1994), a family of embedded parametrisations in which the passage from a given level to the finest one, can be done progressively considering successively the different intermediate sub-levels.

4. Examples of alternative multi-level approaches

In the framework of the formulation defined in Sec. 3.3, we propose, here, to elaborate alternative multi-level approaches based on some classical shape parametrisations. More precisely, parameters, which can be related by linear or affine application to the set of shape grid points, are individualized, and then, used to define an adequate family of sub-parametrisations. Note that, as done in (Beux *et al.*, 1994), we consider the case in which the design control acts only on the ordinates of the shape grid points whereas the abscissas $x_0^\Gamma, \cdots, x_m^\Gamma$ are defined by the knowledge of the initial mesh and frozen during the optimisation process. In the context of aerodynamic shape optimisation, this approach is often chosen, and does not appear so restrictive as long as only slender bodies are considered.

4.1. *Formulation based on Bézier control points*

4.1.1. *Shape representation through Bézier curves*

A Bézier curve of degree n can be defined as follows:

$$S(t) = (x(t), y(t))^T = \sum_{q=0}^{n} B_q^n(t) S_q \quad \text{with } t \in [0,1] \qquad [11]$$

where $S_q = (\mathrm{x}_q, \mathrm{y}_q)^T$ is the q-th Bézier control point while $B_q^n(t)$ corresponds to the q-th Bernstein polynomial of degree n.

Thus, for a given set of parameters $(t_k)_{k=0,m}$ with $t_0 = 0 < t_1 < \cdots < t_m = 1$, the ordinates of the Bézier control points are directly related through a linear application with the ordinates of the shape grid-points, $y_0^\Gamma, \cdots, y_m^\Gamma$. Indeed, applying [11] with $y(t_k) = y_k^\Gamma$, we obtain:

$$\left\{ \begin{array}{l} y_0^\Gamma = y(0) = \mathrm{y}_0 \\ y_k^\Gamma = y(t_k) = \displaystyle\sum_{q=1}^{n-1} B_q^n(t_k) \mathrm{y}_q + s_n(t_k) \quad \text{for } k = 1, m-1 \\ y_m^\Gamma = y(1) = \mathrm{y}_n \end{array} \right. \qquad [12]$$

in which $s_n(t_k) = B_0^n(t_k)\mathrm{y}_0 + B_n^n(t_k)\mathrm{y}_n$.

Nevertheless, geometrical constraints are often imposed at the extremities of the shape which should be optimised. For instance, the shape extremities are in general fixed; it corresponds, here, to freeze y_0^Γ and y_m^Γ. Then, $\alpha = \left(\mathrm{y}_1, \cdots, \mathrm{y}_{n-1}\right)^T$ are related to $\gamma = (y_1^\Gamma, \cdots, y_{m-1}^\Gamma)^T$ by the following affine operator [1]:

$$\begin{array}{rccl} f: & I\!R^{n-1} & \longrightarrow & I\!R^{m-1} \\ & \alpha & \longmapsto & \gamma = M\alpha + b \qquad b \in I\!R^{m-1}, M \in I\!R^{m-1 \times n-1} \end{array}$$

where

$$\left\{ \begin{array}{l} M_{kq} = B_q^n(t_k), \ \text{ for } q = 1, n-1 \ \text{ and } \ k = 1, m-1 \\ \\ b_k = s_n(t_k) = (1-t_k)^n y_m^\Gamma + (t_k)^n y_0^\Gamma, \ \text{ for } k = 1, m-1 \end{array} \right.$$

Thus, according to Sec. 3.1, it can be envisaged to define a strategy similar to the original multi-level approach (Beux *et al.*, 1994). Indeed, if the ordinates of the shape grid-points are taken as control variables, each sub-parametrisation is chosen as a set of Bézier control points instead of a subset of boundary grid-points. Then, the following descent direction is obtained at iteration r and level l:

$$d_r^{(l)} = M^{(l)} (M^{(l)})^T g_r \ \text{ with } \ M_{ij}^{(l)} = B_j^{n_l}(t_i^{(l)}). \qquad [13]$$

1. note that we choose, here, to follow the notations introduced in Sec. 3.1 of instead of the ones used in Sec. 2.1 and 3.3, i.e. to call the affine operator f instead of $\mathcal{L}$.

To fully describe the different sub-parametrisations, at each sub-level $X^{(l)} = (x_0^{(l)}, \cdots, x_{n_l}^{(l)})^T$ with $n_l > n_{l-1}$ and $T^{(l)} = (t_0^{(l)}, \cdots, t_m^{(l)})^T$ should be defined. Furthermore, this definition should be done consistently with [11], i.e.:

$$x_k^\Gamma = x(t_k^{(l)}) = \sum_{q=0}^{n_l} B_q^{n_l}(t_k^{(l)}) \mathrm{x}_q^{(l)} \quad \text{for } k = 0, \cdots, m \qquad [14]$$

in which the abscissas of shape grid-points ($\{x_k^\Gamma\}_{k=0,m}$) are given.

Note that the definition of the different sub-parametrisations is not dependent of the particular optimisation iteration r, and thus, should be done as a preprocessing.

4.1.2. *Sub-parametrisations based on the degree-elevation property*

In order to define an adequate parametrisation at level l, the classical degree-elevation property of the Bézier curves (see, e.g. (Farin, 1990))), which allows to increase the degree and the number of control points, is used here. Note that the good features of the degree-elevation property have been pointed out, and, already, employed to construct an embedded parametrisation in (Désidéri, 2003) .

Given a Bézier curve of degree s associated to the $s+1$ control points $S_q = (\mathrm{x}_q, \mathrm{y}_q)^T$, the same geometrical curve can be also understood as a Bézier curve of degree $s+1$ considering a new set of $s+2$ control points $\bar{S}_q = (\bar{\mathrm{x}}_q, \bar{\mathrm{y}}_q)^T$ obtained from S_q as follows (with the convention $S_{-1} = S_{s+1} = (0,0)^T$):

$$\bar{S}_q = \frac{q}{s+1} S_{q-1} + \left(1 - \frac{q}{s+1}\right) S_q \quad \text{for } q = 0, \cdots, s+1 \qquad [15]$$

An interesting feature is that the distribution of the parameters t over the Bézier curve does not change by degree elevation, and thus:

$$\sum_{q=0}^{s} B_q^s(t)\, \mathrm{x}_q = \sum_{q=0}^{s+1} B_q^{s+1}(t)\, \bar{\mathrm{x}}_q \qquad \forall t \in [0,1] \qquad [16]$$

Consequently, if the parametrisation on the coarsest level has been yet defined with $X^{(0)}$ and $T^{(0)}$ consistent, i.e. with [14] verified for $l = 0$, then, thanks to [16], keeping the parameters t_k unchanged on all the levels, i.e. $T^{(l)} = T^{(0)}$ for all $l > 0$, the consistency is preserved by applying successively the degree-elevation algorithm starting from $X^{(0)}$. More precisely, $X^{(l)}$ is obtained from $X^{(l-1)}$ by applying $n_l - n_{l-1}$ times the degree-elevation algorithm, and thus, we obtain a family of embedded sub-parametrisations with a progressive increase of the number of control points.

Note that, the conditions at the endpoints $\mathrm{x}_0^{(l)} = x_0^\Gamma$ and $\mathrm{x}_n^{(l)} = x_m^\Gamma$ are automatically verified at each level if these relations occur for $l = 0$. Furthermore, additional geometrical constraints are often imposed on the shape: for instance, a vertical tangent at the origin is a standard constraint at the leading edge for airfoil profiles. With a Bézier curve, the derivatives at the endpoints can be easily managed, and in particular,

a vertical tangent at the origin can be enforced by the condition $x_0 = x_1$. It is easy to see that the degree elevation also preserves this condition.

Using the present procedure, the different sub-levels are fully determined by the construction of a consistent coarsest sub-level. For instance, if the $(x_k^\Gamma)_{k=0,m}$ are uniformly distributed, a consistent coarsest level can be easily obtained by simply choosing an uniform distribution for both $X^{(0)}$ and $T^{(0)}$. More generally, let us consider the case in which a vertical tangent should be imposed at the origin with a generic distribution of the shape grid-points abscissas. Then, $\tilde{X}_0 = (x_0^\Gamma, x_0^\Gamma, x_m^\Gamma)^T$ gives the simplest initial set of abscissas for Bézier control points. $\tilde{X}_0$ is associated to the following distribution of the parameters t_k:

$$t_k = \sqrt{\frac{x_k^\Gamma - x_0^\Gamma}{x_m^\Gamma - x_0^\Gamma}} \quad \text{for } k = 0, \cdots, m. \qquad [17]$$

A consistent coarsest level is then obtained considering $T^{(0)}$ defined by [17] and $X^{(0)}$ being any $\tilde{X}_l$ obtained from $\tilde{X}_0$ by applying successively the degree-elevation process.

4.2. *Formulation based on shape functions basis*

4.2.1. *Shape representation through analytical shape functions*

As pointed out in Sec. 2.3, another classical choice of parametrisation for aerodynamic shape design is to consider the curve coefficients in some basis of shape functions. Then, given $(x_k^\Gamma)_{k=0,m}$, the shape grid-points abscissas and the particular basis $(f_q)_{q=1,n}$, the basis coefficients $\alpha^{(n)} = (\alpha_1, \cdots, \alpha_n)^T$ can be easily related to the ordinates of the shape grid-points γ by:

$$y(x_k^\Gamma) = \sum_{q=1}^{n} \alpha_q f_q(x_k^\Gamma) \quad \forall k = 1, \cdots, m-1 \qquad [18]$$

Then, thanks to the linearity between $\alpha^{(n)}$ and γ, it is still possible to define a multi-level strategy as done previously for the Bézier points, i.e. considering γ as control variables. Here, a sub-level is simply obtained considering only a subset of the basis shape functions. More precisely, taking the first l coefficients, i.e., $\alpha^{(l)} = (\alpha_1, \cdots, \alpha_l)^T$, the preconditioning matrix $M^{(l)}(M^{(l)})^T$ to apply in [7] is determinate by:

$$M_{ij}^{(l)} = f_j(x_i^\Gamma) \quad \text{for } j = 1, l \text{ and } i = 1, m-1. \qquad [19]$$

4.2.2. *Sub-parametrisations based on orthonormal shape functions*

As pointed out in Sec. 3.3, the number of freedom degrees considered is not enough to well define the notion of level coarseness. More specifically, to define a coarser level by taking less shape functions as done previously, does not make sense

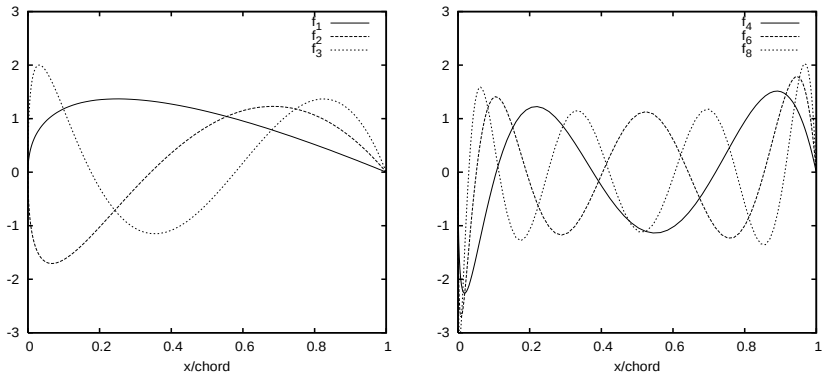


Figure 1. *Orthonormal shape functions: the 1-2-3 (left) and the 4-6-8 (right) basis functions.*

if a hierarchy between the different shape functions can not be established. This difficulty can be avoided by considering orthonormal shape functions since these functions are increasingly oscillatory (see Fig. 1), and consequently, each shape function can be arranged with respect to its degree of high-frequency. The interest of using orthonormal functions associated to multigrid approach has been pointed out in (Kuruvila *et al.*, 1994) in which four orthogonal functions have been defined starting from a NACA 0012 airfoil (this set of shape functions has been successively extended to ten by (Chang *et al.*, 1995) in order to represent a supercritical wing). More recently, in (Catalano *et al.*, 2005) a family of orthonormal shape functions based on Bézier curves has been defined and associated to a multigrid-aided finite-difference method.

Let us, here, consider the following family of functions which just corresponds to consider the functions defined by (Kuruvila *et al.*, 1994) for any degree:

$$g_1(x) = \sqrt{x} - x \ \text{ and } \ g_q = x^{q-1}(1-x) \quad \text{for } q \geq 2$$

Then, the orthonormal shape functions (with respect to $L^2\big([0,1]\big)$ scalar product) $(f_q)_{q=1,n}$ are obtained by applying a classical Gram-Schmidt procedure. Note that, here, the orthonormalisation is not done numerically but analytically through symbolic calculus. In this way, we have access possibly to the exact derivatives, which can be useful, for instance, for imposing some geometrical constraints.

5. Reinterpretation of the new multi-level approaches

5.1. *Parametrisation based on Bézier control points*

In the present formulation, even if each sub-parametrisation corresponds to the ordinates of a set of Bézier control points, these control points do not explicitly appear

in the definition of the descent direction since only the parameters $(t_k)_{k=0,m}$ are directly involved in [13]. Thus, the knowledge of the particular position of the Bézier control points is not required in the practical algorithm implementation. Nevertheless, an explicitly dependence on these control points can be, also, reintroduced in the formulation.

Let us consider a particular level l, with $0 \leq l < L$, L corresponding to the finest level of Bézier control points.

$$\text{for } k = 1, m-1 \quad y_k^\Gamma = \sum_{q=1}^{n_l-1} B_q^{n_l}(t_k)\, \mathrm{y}_q^{(l)} + b_k^{(l)} = \sum_{q=1}^{n_L-1} B_q^{n_L}(t_k) \mathrm{y}_q^{(L)} + b_k^{(L)}$$

The last equality is due to the fact that $(\mathrm{y}_q^{(L)})_{q=0,n_L}$ is obtained from $(\mathrm{y}_q^{(l)})_{q=0,n_l}$ by a degree-elevation process, i.e. using iteratively the relation [16] for the ordinates. Thus, the prolongation operator at level l, i.e. $f^{(l)} : \alpha \longmapsto \gamma = M^{(l)}\alpha + b^{(l)}$, verifies

$$f^{(l)} = f^{(L)} \circ d_l^L$$

in which $f^{(L)} : \beta \longmapsto \gamma = M^{(L)}\beta + b^{(L)}$ is the operator relating the finest level of Bézier control points to the shape grid-points while d_l^L corresponds to the prolongation operator from level l to level L.

Note that the degree-elevation process is linear, and thus, excluding the endpoints, the application d_s, which furnishes s internal control points from $s-1$ ones, is affine. More precisely, d_s is defined by:

$$d_s : \begin{array}{ccc} \mathbb{R}^{s-1} & \longrightarrow & \mathbb{R}^s \\ \alpha & \longmapsto & D_s\alpha + \lambda_s \end{array} \quad \text{with} \quad \left\{ \begin{array}{l} \lambda_s = \dfrac{1}{s+1}(y_0^\Gamma, 0, \cdots, 0, y_m^\Gamma)^T \\ \\ (D_s)_{ij} = \dfrac{1}{s+1}\big((s+1-i)\delta_{ij} + i\delta_{(i-1)j}\big) \end{array} \right.$$

Consequently, the prolongation operator from level l to the finest level L of Bézier control points can be expressed as $d_l^L = d_{n_L-1} \circ \cdots \circ d_{n_l}$, and thus, is an affine application associated to the matrix $\mathcal{D}_l^L = D_{n_L-1} \cdots D_{n_l+1} D_{n_l}$. In conclusion, at a particular level l, one can directly relate the current set of Bézier control points with the shape grid-points through $f^{(l)}$, or alternatively, apply successively the operators d_l^L and $f^{(L)}$. The two ways are equivalent from a theoretical view point, even if they can differ in the implementation. Thus, the following algorithm can be used instead of eqs. [7] and [13]:

$$\gamma_{r+1} = \gamma_r - \omega_r\, M^{(L)} \mathcal{D}_l^L (\mathcal{D}_l^L)^T (M^{(L)})^T\, g_r \qquad [20]$$

The algorithm [20] exactly corresponds to [10] of Sec. 3.3, in which $M = M^{(L)}$ and $D^{(l)} = \mathcal{D}_l^L$. Thus, the approach described in Sec. 4.1 can be also interpreted as a multi-level gradient-based method in which the Bézier control points (on the finest level L) are taken as control variables. Then, [9] corresponds, in this specific case, to the following iterative algorithm (expressed, here, at iteration $r+1$ and sub-level l):

$$\beta_{r+1} = \beta_r - \omega_r\, \mathcal{D}_l^L (\mathcal{D}_l^L)^T\, \mathcal{G}_r \qquad [21]$$

where $\beta = \alpha^{(L)} = (y_1, \cdots, y_{n_L-1})^T$ and $\mathcal{G}_r = (M^{(L)})^T g_r$ is the functional gradient with respect to β at iteration r.

Note that this pointed out a strong analogy with the approach proposed in (Désidéri, 2003) since, in the both algorithms, the degree-elevation property is directly used to prolongate from a coarse level to the complete set of design variables, which are Bézier control points (a more detailed comparison between the two approaches is proposed in Sec. 7).

5.2. *Parametrisation based on shape functions*

The degree-elevation property, which has been used as prolongation operator for the Bézier-based embedded parametrisation, allows to consider the same geometrical curve as an element of a larger space, An analogous operation can be done here; indeed, the curve with coordinates $(\alpha_1, \cdots, \alpha_l)^T$ in the basis $(f_1, \cdots, f_l)^T$ is exactly equal to the curve with coordinates $(\alpha_1, \cdots, \alpha_l, 0, \cdots, 0)^T$ in the basis $(f_1, \cdots, f_n)^T$. This simple way to prolongate an element of $\mathbb{R}^l$ in $\mathbb{R}^n$ corresponds to consider the linear prolongation operator, which is associated with the rectangular matrix $N_l^n \in \mathbb{R}^{n \times l}$ defined by

$$\left(N_l^n\right)_{ij} = \delta_{ij} \quad \text{for } i = 1, n \text{ and } j = 1, l$$

Then, taking $\beta = \alpha^{(n)}$ as control variables, a multi-level preconditioned gradient method can be defined as follows:

$$\beta_{r+1} = \beta_r - \omega_r \, N_l^n \left(N_l^n\right)^T \text{grad}_\beta j(\beta_r)$$

and the corresponding updating of the shape grid-points is expressed as:

$$\gamma_{r+1} = \gamma_r - \omega_r \, M^{(n)} N_l^n \left(M^{(n)} N_l^n\right)^T \text{grad}_\gamma j(\gamma_r)$$

in which $M^{(n)}$ is defined by [19] with $l = n$.

Thus, as for Bézier-based parametrisation, a direct correlation between the different sub-level has been established allowing the definition of an algorithm as [10].

6. Computation of an approximate gradient

As previously pointed out, an adjoint approach seems to be the more suitable way to compute the sensitivities. However, an approximate gradient, computed using finite-differences sensitivities, remains interesting as far as the exact differentiation is a too complex task or some problems of non-differentiability are present. Note that a multi-level method, as described in this study, is *a priori* independent of the way in which the discrete gradient is computed. Nevertheless, in the case of an approximate

gradient, the computational cost can be noticeably reduced by coupling with a multi-level approach in which the finite differences are applied on the coarser levels (see (Beux *et al.*, 1994)). This multi-level/finite-differences formulation has been used in (Held *et al.*, 2002) coupled with a one-shot approach while, in (de' Michieli Vitturi *et al.*, 2006), a multi-level adjoint-free gradient formulation is proposed in which finite differences are used to approximate the flow sensitivities.

Sections 3.3 and 5 show that there are two possible ways to prolongate from a particular level l to γ, the shape grid-points ordinates. Indeed, it can be directly applied $f^{(l)}$ or, alternatively, used successively the affine operators d_l^L and $f^{(L)}$, i.e. in terms of functional, it signifies that $\mathcal{J}^{(l)} = j \circ f^{(l)}$ can be also expressed as $\mathcal{J}^{(L)} \circ d_l^L$. Consequently, there are also two possible ways to compute its Gâteaux derivative. More precisely, for all h in $I\!R^{n_l-1}$, we have:

$$\left[\mathcal{J}^{(l)}\right]'(\alpha)(h) = j'(\gamma)\left(M^{(l)}h\right) = \lim_{\theta\to 0^+} \frac{j\left(\gamma + \theta M^{(l)}h\right) - j(\gamma)}{\theta} \qquad [22]$$

and

$$\left[\mathcal{J}^{(l)}\right]'(\alpha)(h) = \left(\mathcal{J}^{(L)}\right)'(\beta)\left(\mathcal{D}_l^L h\right) = \lim_{\theta\to 0^+} \frac{\mathcal{J}^{(L)}\left(\beta + \theta\mathcal{D}_l^L h\right) - \mathcal{J}^{(L)}(\beta)}{\theta} \qquad [23]$$

with $\gamma = M^{(l)}\alpha \in I\!R^{m-1}$ and $\beta = \mathcal{D}_l^L\alpha \in I\!R^{n_L-1}$ respectively.

Consequently, two different algorithms can be proposed in the context of an approximate gradient. The first one is obtained considering the ordinates of the shape grid-points as design variables, and can be expressed as follows:

$$\gamma_{r+1} = \gamma_r - \omega_r M^{(l)}\,\tilde{g}_r^{(l)}(\epsilon) \qquad [24]$$

in which $\tilde{g}_r^{(l)}(\epsilon)$ is an approximation of $\left(M^{(l)}\right)^T g_r$ obtained using [22], and thus, is defined by:

$$\forall i = 1, \cdots, n_l - 1 \qquad \left(\tilde{g}_r^{(l)}(\epsilon)\right)_i = \frac{1}{\epsilon}\left[j\left(\gamma_r + \epsilon M^{(l)}e_i\right) - j(\gamma_r)\right]$$

in which e_i is the i-th element of the canonical basis of $I\!R^{n_l-1}$ and ϵ a small given parameter.

On another hand, from [21] the following algorithm is obtained if the ordinates of the Bézier control points[2] are directly used as design variables:

$$\beta_{r+1} = \beta_r - \omega_r\mathcal{D}_l^L\,\tilde{\mathcal{G}}_r^{(l)}(\epsilon) \qquad [25]$$

2. note that the notations are, here, coherent with the ones used in Sec. 3.1 and successively in Sec. 5.1. Obviously, the results of this section are also true for the case of a parametrisation based on shape functions as described in sections 4.2 and 5.2.

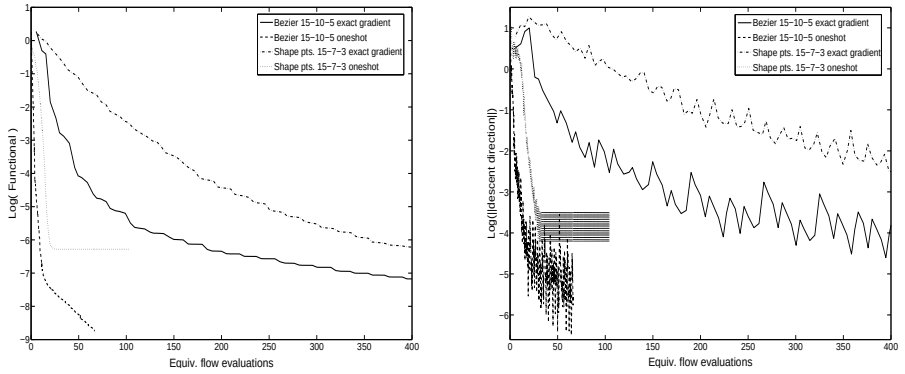


Figure 2. *Convergence histories for $j(\gamma_r)$ (left) and $d_r^{(l)}$ (right) (shape grid-points and Bézier-based parametrisations): comparison between the exact adjoint gradient computation and the gradient approximated by divided differences associated with a one-shot approach.*

in which $\tilde{\mathcal{G}}_r^{(l)}(\epsilon)$ approximates $\left(\mathcal{D}_l^L\right)^T \mathcal{G}_r$ using [23], and thus:

$$\forall i = 1, \cdots, n_l - 1 \qquad \left(\tilde{\mathcal{G}}_r^{(l)}(\epsilon)\right)_i = \frac{1}{\epsilon}\left[\mathcal{J}^{(L)}\left(\beta_r + \epsilon \mathcal{D}_l^L e_i\right) - \mathcal{J}^{(L)}(\beta_r)\right]$$

Since $M^{(l)} = M^{(L)}\mathcal{D}_l^L$, applying the affine operator $f^{(L)}$ to [25] one can easily recover algorithm [24] in which $\gamma_r = f^{(L)}(\beta_r)$. Thus, according to what happens for an exact gradient computation, the two algorithms are equivalent, and moreover, in the both cases, at level l, only $n_l - 1$ evaluations of the cost function are needed.

In order to illustrate the interest of the multi-level approximate gradient formulation, some results presented in (Martinelli *et al.*, 2007a) are reported in Fig. 2. For an inverse problem similar to the one presented in Sec. 8.2.1, V-cycle three-levels approaches are applied both for the adjoint method and the finite-differences one associated with a one-shot approach (as described in (Held *et al.*, 2002)). The good behaviour of the one-shot approach without adjoint, already obtained in (Held *et al.*, 2002), is also confirmed for the Bézier-based parametrisation. Moreover, the same improvement obtained by using the Bézier parametrisation instead of the shape grid-points is observed with the multi-level/finite-differences one-shot approach.

7. Reinterpretation of the approach proposed in (Désidéri, 2003)

As previously pointed out, the present Bézier-based multi-level approach has strong similarities with the approach proposed by (Désidéri, 2003). This approach has been defined independently of the specific optimisation algorithm, and thus, is *a priori* gradient-free. Nevertheless, in order to try to better see the analogies, let us con-

sider a two-level algorithm, in which a steepest-descent-like method is used both for the finest level and for the sub-level (see (Désidéri, 2007; Abou El Majd *et al.*, 2007)). It consists in alternate a relaxation phase on the upper-level of Bézier parametrisation (few shape optimisation iterations) with a coarse-level correction phase, in which the shape perturbation is parametrised on a coarse Bézier sub-parametrisation. Then, the cycle c of the algorithm can be rewritten with the notations used in the present study as:

1) Upper level: given $\beta_0^{(c)} \in I\!R^{n_L-1}$ from the previous cycle, iterate:

$$\text{for } r = 1, r_L \qquad \beta_r^{(c)} = \beta_{r-1}^{(c)} - \tilde{\omega}_r \text{grad}_\beta(\beta_{r-1}^{(c)}) \tag{26}$$

2) Solve the following minimisation problem on the coarser level:

$$\text{Find } \bar{\delta}_\alpha^{(c)} = Arg \min_{\delta_\alpha} \mathcal{I}^{(c)}(\delta_\alpha) \tag{27}$$

in which $\mathcal{I}^{(c)}(\delta_\alpha) = \mathcal{J}^{(L)}\left(\beta_{r_L}^{(c)} + \tilde{d}_l^L(\delta_\alpha)\right)$ for $\delta_\alpha \in I\!R^{n_l-1}$.

3) A new cycle can be, then, computed starting from $\beta_0^{(c+1)} = \beta_{r_L}^{(c)} + \tilde{d}_l^L\left(\bar{\delta}_\alpha^{(c)}\right)$ on the upper level.

$\tilde{d}_l^L$ is the prolongation operator from level l to L based on degree-elevation property, which depends on the particular choice of $T^{(l)}$ and $X^{(l)}$. Furthermore, since we are working, here, on shape perturbations, the conditions at the endpoints are $(\delta_\alpha)_0 = 0$ and $(\delta_\alpha)_{n_l} = 0$, and consequently, $\tilde{d}_l^L$ is linear, i.e. $\tilde{d}_l^L(\delta_\alpha) = \tilde{\mathcal{D}}_l^L \delta_\alpha$.

To solve [27] and obtain an approximation of $\bar{\delta}_\alpha^{(c)}$, s_l ($s_l \gg r_L$) iterations of a steepest-descent method are performed starting from a given initial solution $(\delta_\alpha)_0$:

$$s = 1, s_l \quad (\delta_\alpha)_s = (\delta_\alpha)_{s-1} - \tilde{\omega}_{s-1} \text{grad}_{\delta_\alpha} \mathcal{I}^{(c)}\left((\delta_\alpha)_{s-1}\right)$$

Furthermore, let us define $\varphi_{r_L}^{(c)}$ by $\varphi_{r_L}^{(c)}(\delta_\alpha) = \beta_{r_L}^{(c)} + \tilde{\mathcal{D}}_l^L \delta_\alpha$, since $\mathcal{I}^{(c)} = \mathcal{J}^{(L)} \circ \varphi_{r_L}^{(c)}$ with $\varphi_{r_L}^{(c)}$ affine, we have:

$$\text{grad}_{\delta_\alpha} \mathcal{I}^{(c)}(\delta_\alpha) = (\tilde{\mathcal{D}}_l^L)^T \text{grad}_\beta \mathcal{J}^{(L)}\left(\varphi_{r_L}^{(c)}(\delta_\alpha)\right).$$

Consequently, the Bézier control points on the finer level are obtained at the end of the cycle by:

$$\begin{aligned} \beta_0^{(c+1)} &= \beta_{r_L}^{(c)} + \tilde{\mathcal{D}}_l^L \left(\delta_\alpha\right)_{s_l} \\ &= \beta_{r_L}^{(c)} - \tilde{\mathcal{D}}_l^L \left(\tilde{\mathcal{D}}_l^L\right)^T \sum_{i=0}^{s_l-1} \tilde{\omega}_i \text{grad}_\beta \mathcal{J}^{(L)}\left(\beta_{r_L}^{(c)} + \tilde{\mathcal{D}}_l^L (\delta_\alpha)_i\right) - \tilde{\mathcal{D}}_l^L \left(\delta_\alpha\right)_0 \end{aligned}$$

Let us consider, now, the approach defined in sections 4.1 and 5.1, associated with a two-level algorithm with r_l and s_l iterations for the upper and coarser levels

respectively. From [21], the following expression is obtained for the Bézier control points on the finest level are obtained at the end of the cycle c:

$$\bar{\beta}_0^{(c+1)} = \beta_{[r_0(c)+s_l]} = \beta_{r_0(c)} - \mathcal{D}_l^L \left(\mathcal{D}_l^L\right)^T \sum_{i=0}^{s_l-1} \omega_i \, \mathcal{G}_{[r_0(c)+i]}$$

with $r_0(c) = (c-1)(r_L + s_l) + r_L$.

Proposition *With an adequate choice of the parameters which characterize the two-level algorithms, the two strategies coincide at the end of each cycle, i.e.*

$$\beta_0^{(c+1)} = \bar{\beta}_0^{(c+1)} \quad \textit{for each cycle } \; c$$

Proof Let us consider that the same choice of family of embedded sub-parametrisations, i.e. the same choice of $T^{(l)}$ and $X^{(l)}$ at each sub-level, is done, and thus, in particular $\tilde{\mathcal{D}}_l^L = \mathcal{D}_l^L$.

– If the two algorithms start from the same shape configuration, then $\beta_0^{(1)} = \bar{\beta}_0^{(1)}$.

– Let us suppose, now, that $\beta_0^{(c)} = \bar{\beta}_0^{(c)}$ for $c \geq 1$.

If the same 1D search strategy (i.e., the same scalar parameter ω) is employed, the iterative algorithms on the upper level, i.e. [21] and [26], coincide, and thus, we have that $\beta_{r_0(c)} = \beta_{r_L}^{(c)}$. Moreover, choosing $(\delta_\alpha)_0 = 0$ and supposing that $\tilde{\omega}_i = \omega_{[r_0(c)+i]}$ at each iteration on the lower level, we obtain that

$$\beta_{[r_0(c)+i]} = \varphi_{r_L}^{(c)}\left((\delta_\alpha)_i\right) \quad \text{for } \; i = 0, s_l$$

and consequently

$$\beta_0^{(c+1)} = \bar{\beta}_0^{(c+1)}$$

■

8. Numerical experiments

8.1. *Parametrisation and shape representations*

Since in a multi-level approach, different levels of coarseness are involved, it should be interesting to evaluate the capability of shape representation of each level. For instance, let us investigate on the accuracy in which a cambered RAE2822 profile can be represented by the parametrisations previously defined. At a level l, the Bézier curvefit, at the discrete least-squares sense, is obtained minimising with respect to $\alpha^{(l)} = \left(\mathrm{y}_1, \cdots, \mathrm{y}_{n_l-1}\right)^T$ the Euclidean norm in $\mathbb{R}^{m-1}$ of the residual $Res\left(\alpha^{(l)}\right) = \left[M^{(l)}\alpha^{(l)} - b^{(l)}\right] - \gamma^{target}$. On another hand, for the parametrisation defined in Sec. 4.2.2, since we consider an orthonormal basis, the reconstruction of a

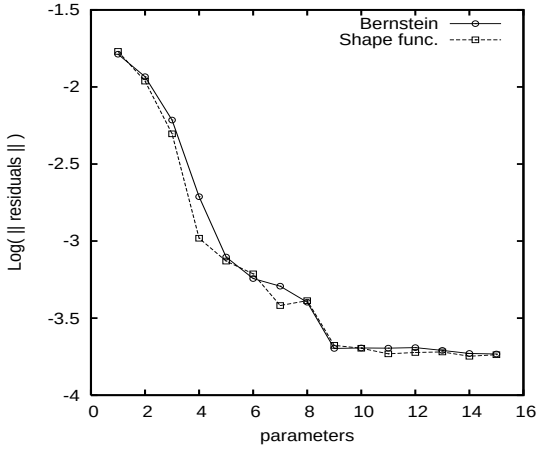


Figure 3. *Curvefit error (*$\log \|Res(\left(\alpha^{(l)}\right))\|_\infty$*) with respect to the number of parameters for the two kinds of parametrisations.*

given shape f, known at $x_0^\Gamma, \cdots, x_m^\Gamma$, on a sub-level l is obtained as $\sum_{i=1}^{l} \langle f, f_i \rangle f_i$ in which $\langle ., . \rangle$ is an adequate discrete approximation of the $L^2\left([x_0^\Gamma, x_m^\Gamma]\right)$ scalar product.

The curvefit error with respect to the number of parameters[3] is plotted in Fig. 3. A very similar behaviour is observed for the two kinds of parametrisation. A rather good representation of the RAE2822 profile is already furnished with few degrees of freedom (as, also, shown in Fig. 4 with the plot of profiles for small number of parameters) whereas for more than 8 parameters a plateau is obtained with an error value of about $2\,10^{-4}$. Obviously, the capability of shape representation is strongly correlated to the choice of the particular configuration (e.g. for the case of a symmetric NACA0012 profile, a Bézier curvefit error minor than 10^{-8} is obtained). Note that, an increase of the number of Bézier control points amplifies the irregularity of the control polygon (see Fig. 5). Consequently, a very small variation in the solution space (shape) can correspond to a large variation in the design space (Bézier control points). This depends on the particular shape (with the NACA0012, very regular control polygons are obtained), but also, on the particular construction of the sub-parametrisations, i.e. on the choice of $T^{(l)}$ and $X^{(l)}$. This problem has been already pointed out in (Tang *et al.*, 2002; Abou El Majd *et al.*, 2004), in which a procedure of parametrisation adaptation is also proposed which acts dynamically during the optimisation procedure.

3. Two different curves are considered (for a Bézier curve as well as for a linear combination of orthonormal shape functions): one for the upper side and one for the lower one. Thus, the number of variables really involved is two times more. In particular, 30 parameters are globally used on the finest level of parametrisation.

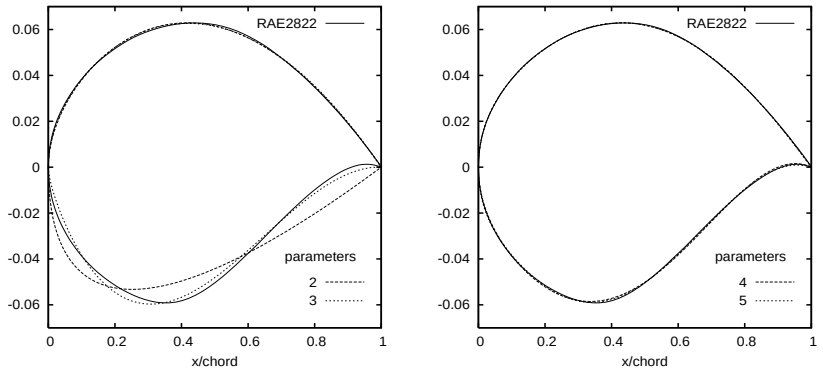


Figure 4. *Curvefits of RAE2822 profile by orthonormal shape functions: using 2 and 3 (left) and 4 and 5 parameters (right) respectively.*

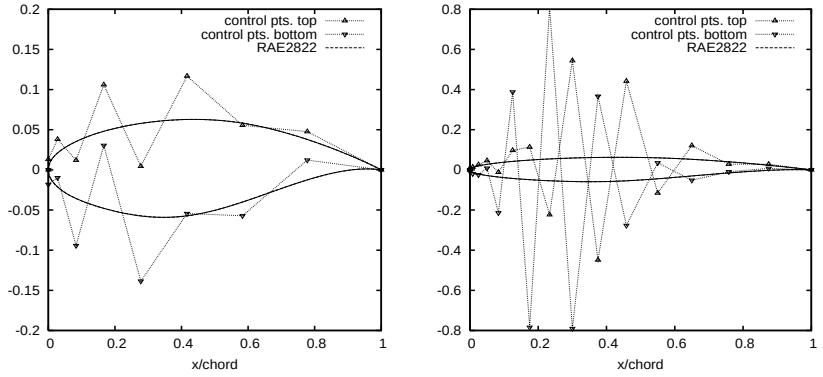


Figure 5. *RAE2822 curvefits with Bézier control points (8 (left) and 15 (right) parameters): profiles and corresponding control polygons.*

8.2. *Numerical experiments on 2D inverse problems*

8.2.1. *Test-case 1: a 2D nozzle inverse problem*

The first test-case, already used for the multi-level approach associated to shape grid-point coordinates parametrisation (Beux, 1994; Beux *et al.*, 1992; Beux *et al.*, 1994; Held *et al.*, 2002), is a 2D convergent-divergent nozzle inverse problem for inviscid subsonic flows (the flow is modelled, here, by the Euler equations). Here, the particular inverse problem is characterised by an initial constant-section nozzle and a target sine shape. For the considered mesh of 1900 nodes (see Fig. 6), 63 shape grid-points are available. Since, the abscissas of the shape grid-points points

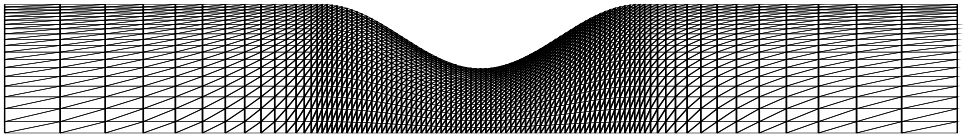
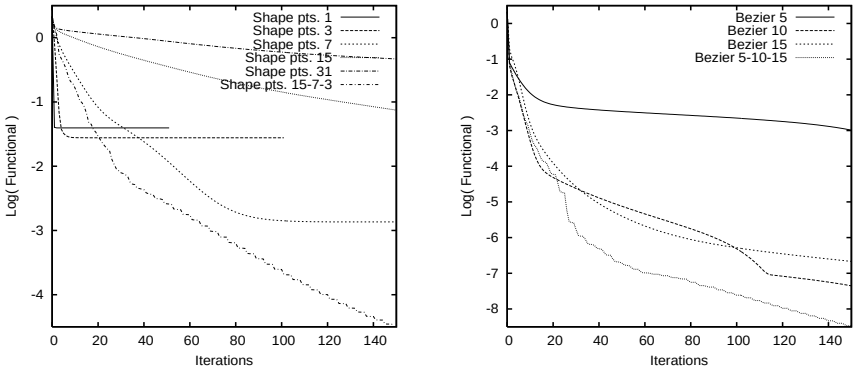


Figure 6. *Test-case 1: computational mesh (target configuration)*



(a) (b)

Figure 7. *Test-case 1: Convergence history for one-level and V-cycle three-level strategies (a) shape grid-points (b) Bézier-based parametrisation.*

are uniformly distributed, an uniform distribution is also taken for the abscissas of the Bézier control points at each level. Fig. 7a shows the convergence history obtained using the original shape grid-points parametrisation. It clearly shows the effect of each level, indeed, the lower is the number of control parameters involved, the lower is the accuracy of the solution, but also, the faster is the convergence to the solution. Note that, the one-level method with 31 parameters already corresponds to a preconditioned gradient method since 63 degrees of freedom are available for the present mesh. Then, advantaging of the speed-up on the coarser levels, the multi-level approach largely improves the convergence rate to reach an accurate solution given by the higher levels. Concerning the new set of sub-parametrisation based on Bézier control points (see Fig. 7b), even the one-level approach with 10 or 15 parameters gives interesting results. The corresponding multi-level strategy yields ulterior improvements in the final part of the convergence history. Nevertheless, interpreted as an optimisation with respect to the Bézier control points, this multi-levelling appears less impressive. Indeed, it seems that the principal gain is due to the use of a Bézier-based parametrisation more than the change of control subspaces. This can be explained by the lack of convergence speed-up of the coarser level as shown in Fig. 7b.

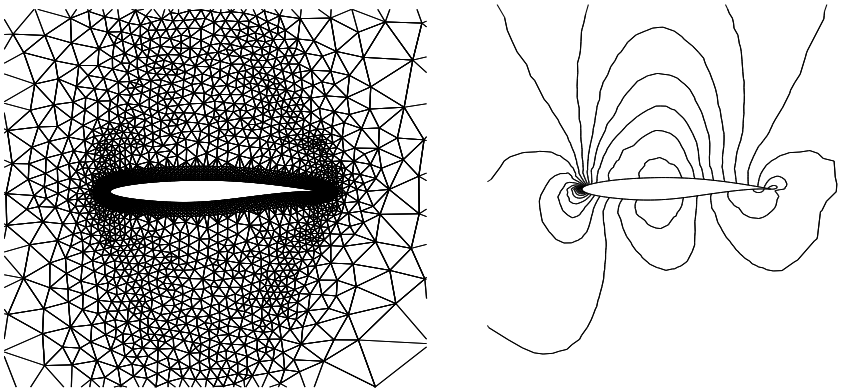
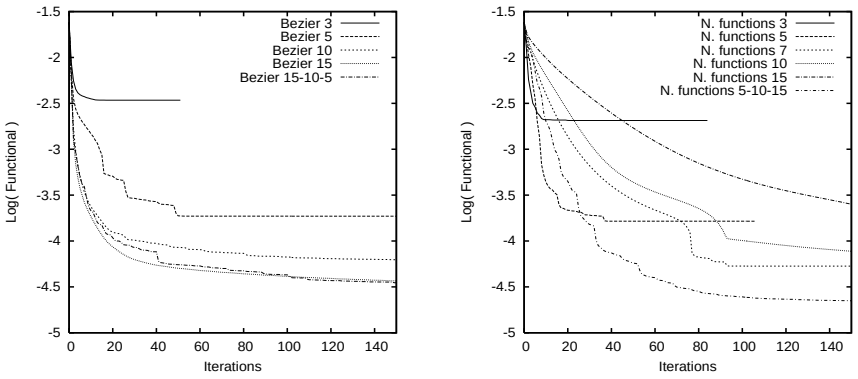


Figure 8. *Target configuration for test-case 2: (left) computational mesh and (right) Mach contours plot (equidistant iso-contours from* 0.1 *to* 1.1 *with* $\Delta M = 0.05$*).*

8.2.2. *Test-case 2: an airfoil inverse problem*

In this second test-case, starting from a symmetric NACA0012 airfoil, the cambered RAE2822 should be rebuilt. The initial flow conditions are characterised by a far-field Mach number of 0.734 and an angle of incidence of 2.79^o. Furthermore, as in the previous test-case, only inviscid flows are considered. A mixed unstructured/structured mesh of 3282 nodes has been generated on a circular computational domain centred on the airfoil (see Fig. 8). The definition of the Bézier-based sub-parametrisations have been done, here, imposing a vertical tangent at the leading edge, and thus, following the procedure proposed in Sec. 4.1.2.

The direct use of the parametrisation based on shape grid-points seems more difficult for this optimisation problem, in particular, near the training edge where the upper and lower profiles may be crossed over. This problem can be solved by imposing geometrical constraints, considering only coarser sub-levels in order to increase the smoothing or/and modifying the criterion for the choice of the descent step ω_r. Nevertheless, we choose, here, to consider only the other sub-parametrisations. Fig. 9a shows the convergence behaviour for one-level strategies as well as a V-cycle multi-level strategy on three sub-levels (5, 10 and 15 parameters) in the case of a Bézier-based parametrisation. The behaviour observed with the previous test-case is magnified, here, since the finer is the level, the more accurate is the solution but without any lost in convergence speed. Consequently, a very good behaviour is obtained with the one-level approach with 15 parameters, but on another hand, the multi-level algorithm does not improve the convergence rate. Fig. 9b shows the convergence behaviour obtained with the parametrisation based on the orthonormal shape functions. As expected from the results obtained in Sec. 8.1, the same kind of behaviour is obtained between the two kinds of parametrisation in terms of solution accuracy, for the



(a) (b)

Figure 9. *Test-case 2: Convergence history for one-level and V-cycle three-level strategies (a) Bézier-based parametrisation (b) parametrisation based on orthonormal shape functions.*

one-level approaches. However, since, here, the increase of number of parameters corresponds to a decrease of terms of convergence, the multi-level strategy yields an important improvement of the convergence behaviour.

9. Conclusion

In the present study, the description of multi-level gradient-based methods for aerodynamic shape design is addressed. Starting from an existing formulation (Beux *et al.*, 1994) based on an embedded parametrisation of shape grid-points and on interpolation operators, a possible generalisation to other kinds of parametrisations is described. This extension requires the elaboration of an adequate family of sub-parametrisations, possibly embedded, associated to affine prolongation operators. Two particular examples are then presented, and since the shape grid-points are yet used as control variables, the resulting approaches can be interpreted as multi-level strategies as defined in (Beux *et al.*, 1994), in which a particular prolongation operator (i.e. with a particular preconditioning) is applied. However, it can also be reinterpreted directly as multi-level approaches with respect to the new family of shape parametrisation. In the first example, the sub-levels are defined through the use of Bézier control points, and starting from a consistent coarsest level, the degree-elevation property of Bézier curves is applied to successively define the different finer levels. In this context, even if, in practice, the Bézier control points do not need to be explicitly computed, the proposed algorithm can be also interpreted as a descent method for Bézier control points as control variables. Thus, the present descent method seems very close to the approach proposed in (Désidéri, 2003), even if, this last one has been defined independently of the specific optimisation algorithm. Indeed, we prove that, under

specific conditions, the two approaches can be equivalent. In this way, it has been built up an explicit link between two kinds of multi-levelling: on the one hand, the preconditioned gradient-based method initially defined in (Beux *et al.*, 1994), and on the other hand, the multi-level algorithm for parametric shape optimisation presented in (Désidéri, 2003) and its successive extensions. In the second example, the definition of the set of sub-parametrisations is based on the use of an orthonormal basis of shape functions as shape representation. As for the case of Bézier-based parametrisation, a descent direction is obtained considering as control parameters the ordinates of the shape grid-points as well as the finest sub-parametrisation.

The numerical experiments shows that the new families of sub-parametrisations have suitable effects, if there are understood as an alternative gradient preconditioning for the optimisation with respect to the shape grid-points. Nevertheless, to extend the range of interest of this kind of methods, it should be interpreted as a descent method in which the control variables are taken through the new set of parameters. Note that, in this case, we start with a less inefficient non preconditioned algorithm[4] since the lack of shape smoothness, typical of shape grid-points parametrisation, has been avoided. Nevertheless, for the parametrisation based on orthonormal shape functions, the multi-level strategy still yields an interesting speed-up of the convergence. Concerning the Bézier-based parametrisation, the results are more disappointing since the multi-levelling seems poorly efficient. This is due, here, to a good convergence behaviour on the finest levels while the coarsest levels do not yield any additional speed-up, and thus, the basic conditions are not present to apply effectively the multi-level/multigrid principles. Thus, such additional investigation should be performed in order to better understand the present behaviour, which is also inconsistent with the results obtained by J.-A. Désidéri and collaborators. If more attractive results can be obtained for Bézier-based parametrisation, since the Bézier curves act as a basic tool for polynomial shape representation, one can also envisage to extend the formulation to more complex shape representation as B-splines (which also have properties of degree-elevation), and also, to 3D case through, for instance, tensorial Bézier parametrisation. In the both examples of parametrisations presented in this study, the shape parameters are related by linear or affine application to the set of shape grid-points. Even if it appears the simplest way to behave, alternative approaches can be also envisaged since the algorithm should not be obligatory a descent direction with respect to the shape grid-points. Finally, note that, even if the multi-levelling has been used, in the present study, only associated with a steepest descent approach, other gradient-based methods can be also considered as, for instance, BFGS-type formulation. In this case, the use of an additive multi-level preconditioner, which can be defined as soon as a set of embedded sub-parametrisations is available, seems more suitable (see (Courty *et al.*, 2006)).

4. indeed, with respect to the shape grid-points, it can be viewed as a method already preconditioned but associated to a one-level strategy.

Acknowledgements

The authors thank J.-A. Désidéri and R. Duvigneau for helpful discussions.

10. References

Abou El Majd B., Désidéri J.-A., Duvigneau R., " Multilevel strategies for parametric shape optimization in Aerodynamics", *REMN*, 2007.

Abou El Majd B., Désidéri J.-A., Janka A., " Nested and self-adaptive Bézier parameterization for shape optimization", *International Conference on Control, Partial Differential Equations and Scientific Computing*, Beijing, China, 13-16 Sept, 2004.

Alexandrov N., Lewis R., Gumbert C., Green L., Newman P., " Approximation and model management in aerodynamic optimisation with variable-fidelity models", *Journal of Aircraft*, vol. 38, n° 6, p. 1093-110, 2001.

Beux F., " Shape optimization of an Euler flow in a nozzle", *Notes on Numerical Fluid Mechanics, 55, Périaux et al. eds*, Vieweg Publishing, p. 115-131, 1994.

Beux F., Dervieux A., " Exact-gradient shape optimization of a 2-D Euler flow", *Finite Elements in Analysis and Design*, vol. 12, n° 3-4, p. 281-302, 1992.

Beux F., Dervieux A., " A hierarchical approach for shape optimization", *Engineering Computations*, vol. 11, n° 6, p. 25-38, 1994.

Borzi A., Multigrid methods for optimality systems, Habilitation thesis, Univ. Graz, 2003.

Carpentieri G., van Tooren M., Koren B., " Adjoint-based aerodynamic shape optimization on unstructured meshes", *Journal of Computational Physics*, 2007. in press.

Catalano L., Dadone A., Daloiso V., " Progressive optimization on unstructured grids using multigrid-aided finite-difference sensitivities", *International Journal for Numerical Methods in Fluids*, vol. 47, n° 10-11, p. 1383-1391, 2005.

Catalano L., Dadone A., Daloiso V., " Turbine cascade design viam ultigrid-aided finite-difference progressive sensitivities", *REMN*, 2007.

Chang I., Torres F., Tung C., Geometric analysis of wing sections, Technical Memorandum n° 110346, NASA, April, 1995.

Courty F., Dervieux A., " Multilevel functional preconditioning for shape optimisation", *International Journal of Computational Fluids Dynamics*, vol. 20, n° 7, p. 481-490, 2006.

Dadone A., Grossman B., " Progressive optimization of inverse fluid dynamic design problems", *Computers and Fluids*, vol. 29, n° 1, p. 1-32, 2000.

de' Michieli Vitturi M., Beux F., " A discrete gradient-based approach for aerodynamic shape optimisation in turbulent viscous flow", *Finite Elements in Analysis and Design*, vol. 43, n° 1, p. 68-80, 2006.

Désidéri J.-A., " Hierarchical optimum-shape algorithms using embedded Bézier parameterizations", *Numerical Methods for Scientific Computing, Variational Problems and Applications*, E. Heikkola et al. eds, CIMNE, Barcelona, 2003.

Désidéri J.-A., " Two-level ideal algorithm for parametric shape optimization", *J. Numer. Math.*, 2007. to appear.

Dreyer T., Maar B., Schulz V., " Multigrid optimization in applications", *J. Comp. Appl.*, vol. 120, p. 67-84, 2000.

Dwight R., Brézillon J., " Effects of various approximations of the discrete adjoint on gradient-based optimization", *AIAA paper 2006-0690*, 2006.

Farin G., *Curves and surfaces for computed-aided geometric design - A practical guide*, W. Reinboldt and D. Siewiorek eds., Academic Press, Boston, 1990.

Feng D., Pulliam T., Aerodynamic design optimization via reduced Hessian SQP with solution refining, Tech. Report n° 95-24, Research Institute for Advanced Computer Science (RIACS), NASA Ames Research Center, 1995.

Gelman E., Mandel J., " On multilevel iterative methods for optimization problems", *Mathematical Programming*, vol. 48, n° 1, p. 1-17, 1990.

Giering R., Kaminski T., Slawig T., " Generating efficient derivative code with TAF: adjoint and tangent linear Euler flow around an airfoil", *Future generation computer systems*, vol. 21, n° 8, p. 1345-1355, 2005.

Giles M., Pierce N., " An introduction to the adjoint approach to design", *Flow, Turbulence and Combustion*, vol. 65, p. 393-415, 2000.

Gratton S., Sartenaer A., Toint P., Recursive trust-region methods for multilevel nonlinear optimization (part I): global convergence and complexity, Tech. Report n° 04/06, Dep. Math., Uni. Namur, 2004.

Guillard H., Convergence analysis of a multi-level relaxation method, rapport de recherche n° 1884, INRIA, 1993.

Guillard H., Marco N., " Some aspects of multigrid methods on non-structured meshes", *Proceedings of the Conference of Copper Mountain on Multigrid Methods, NASA*, April, 1995.

Hascoët L., Greborio R.-M., Pascual V., *Computing Adjoints by Automatic Differentiation with TAPENADE*, B. Sportisse and F.-X. LeDimet eds., Springer, 2005. (see also http://www-sop.inria.fr/tropics/tapenade.html).

Held C., Dervieux A., " One-Shot airfoil optimisation without adjoint", *Computers and Fluids*, vol. 31, n° 8, p. 1015-1049, 2002.

Hicks R., Henne P., " Wing design by numerical optimisation", *Journal of Aircraft*, vol. 15, n° 7, p. 407-413, 1978.

Jameson A., *Aerodynamic shape optimisation usign the adjoint method*, Von Karmann Institute course, 2003.

Koobus B., Marco N., Dervieux A., " An additive multilevel preconditioning method", *Journal of Scientific Computing*, vol. 12, n° 3, p. 233-251, 1997.

Kuruvila G., Ta'asan S., Salas M., *Airfoil optimization by the one-shot method.*, Von Karmann Institute Lecture Series, 1994.

Marco N., Dervieux A., " Multilevel parametrization for aerodynamical optimization of 3D shapes", *Finite Elements in Analysis and Design*, vol. 26, p. 259-277, 1997.

Martinelli M., Beux F., " Multilevel gradient-based methods in aerodynamic shape design", *submitted to ESAIM: Proceedings*, 2007a.

Martinelli M., Beux F., " Multilevel gradient method with Bézier parametrisation for aerodynamic shape optimisation", *Applied and Industrial Mathematics in Italy - II. Series on Advances in Mathematics for Applied Sciences*, World Scientific Pub, 2007b. in press.

Migdalas A., Pardalos P., Värbrand P. (eds), *Multilevel optimization: algorithms and applications*, Kluwer Academic, 1997.

Mohammadi B., " A new optimal shape design procedure for inviscid and viscous turbulent flows", *International Journal for Numerical Methods in Fluids*, vol. 25, p. 183-203, 1997.

Mohammadi B., Pironneau O., *Applied Shape Optimization for Fluids, Numerical Mathematics and Scientific Computation*, Oxford University press, 2001.

Nash S., " A multigrid approach to discretized optimization problems", *Journal of Optimization Methods and Software*, vol. 14, p. 99-116, 2000.

Nemec N., Zingg D., " Newton-Krylov algorithm for aerodynamic design using the Navier-Stokes equations", *AIAA Journal*, vol. 37, p. 1146-1154, 2002.

Newman III J., Taylor III A., Barnwell R., Newman P., Hou G.-W., " Overview of Sensitivity Analysis and Shape Optimization for Complex Aerodynamic Configurations", *Journal of Aircraft*, vol. 36, n° 1, p. 87-96, 1999.

Pironneau O., Polak E., " Consistent approximations and approximate functions and gradients in optimal control", *J. SIAM control and optimization*, vol. 41, p. 487-510, 2002.

Reuther J., Jameson A., " Aerodynamic shape optimization of wing and wing-body configurations using control theory", *AIAA Paper, 95-0123*, 1995.

Samareh J., " Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization", *AIAA Journal*, vol. 39, n° 5, p. 877-884, 2001.

Schwabacher M., Gelsey A., " Multilevel simulation and numerical optimization of complex engineering designs", *Journal of Aircraft*, vol. 35, n° 3, p. 387-397, 1998.

Selmin V., " Geometry modelling and industrial parameterisation issues", *REMN*, 2007.

Tang Z., Désidéri J.-A., Towards self-adaptive parameterization of Bézier curves for airfoil aerodynamic design, Rapport de recherche n° 4572, INRIA, 2002.