INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 20, 915–934 (1995)

OPTIMUM AERODYNAMIC SHAPE DESIGN INCLUDING MESH ADAPTIVITY

GABRIEL BUGEDA AND EUGENIO OÑATE

Escola Tècnica Superior d'Enginyers de Camins, Canals i Ports, Universitat Politècnica de Catalunya C/Gran Capità s/n, Campus Nord UPC, Mòdul C1, E-08034 Barcelona, Spain

SUMMARY

This paper presents a methodology for solving shape optimization problems in the context of fluid flow problems including adaptive remeshing. The method is based on the computation of the sensitivities of the geometrical design parameters, the mesh, the flow variables and the error estimator to project the refinement parameters from one design to the next. This sensitivity analysis is described for the incompressible potential equations and the Euler equations. The efficiency of the proposed method is checked by means of two 2D inverse problems.

KEY WORDS optimum design; mesh adaptivity; adaptive remeshing

INTRODUCTION

The increasing complexity of fluid flow problems to be analysed with the finite element method makes necessary the use of meshes with an adequate sizing of the elements adapted to the flow features. This is necessary to capture the flow complexities (shock waves and boundary layers), but it increases considerably the computational cost of the analysis.

The high cost of the analysis of a complex flow can be reduced if an adaptive remeshing strategy is employed. In this case small elements are used only in the zones where the flow is complex, whereas bigger elements are used in the rest of the domain. A general scheme of such a strategy is shown in Figure 1. The basic requirements of this strategy are the following:

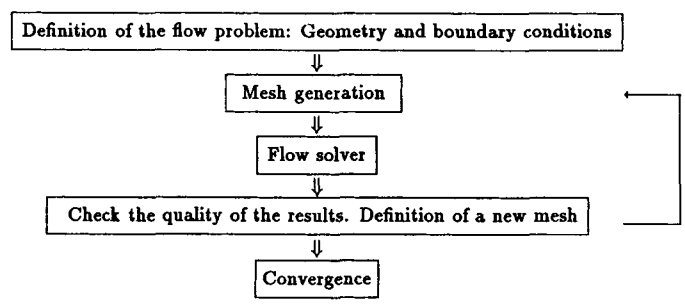

Figure 1. Adaptive remeshing scheme.

CCC 0271–2091/95/090915–20

(1) a non-structured mesh generator able to control the sizes of the elements anywhere;
(2) an *a posteriori* error estimator or indicator to assess where smaller (or larger) elements are needed;
(3) an optimality criterion for the definition of the characteristics (element sizing) of an optimal mesh for a given CPU cost, or a given precision.

The adaptive remeshing procedures are typically based on a series of successive analyses of the flow problem using meshes with increasing quality. This is possible because the characteristics of the flow problem are the same during the successive analyses. This procedure is much more complex if these characteristics are continuously changing as in the case of an optimization problem. Here a series of different designs are obtained and the characteristics of the flow can change from one design to the next. An 'optimal' mesh for the analysis of one design can become inadequate for the next one. This can be crucial if the location of flow discontinuities (i.e. shocks, separation points, etc.) changes from one design to the next.

Traditionally, optimization problems are solved iteratively following the scheme shown in Figure 2. An important development in this direction is described in Reference 1. In this the same mesh is properly adapted for the analysis of each geometry. There is no control on the quality of the results, which can lead to convergence problems and also to bad final design results. Not only the quality of the analysis but also the quality of the sensitivity analysis depends on the quality of the meshes.

To insert an adaptive remeshing loop inside the optimum design iterative process is an obvious possibility that has already been attempted.[2] Unfortunately, in this case the total CPU cost grows proportionally to the number of optimization iterations multiplied by the number of adaptive remeshings for each design.

The definition of a good mesh in an adaptive mesh enrichment process requires results from a previous analysis using a given mesh. In the iterative resolution of an optimum design problem various geometries are obtained and analysed. But the fluid flow characteristics are different for each one. The authors[3] have recently proposed a methodology where the information required for each adaptive remeshing is taken from the analysis of the previous design. Both the mesh parameters and the error estimator are 'projected' to the next enhanced design. This allows us to define *a priori* refined meshes which ensure improved quality of the results in the analysis of each design geometry. The general scheme of this methodology is shown in Figure 3. The main features of this methodology are discussed in the following sections and some of the main expressions are detailed for the cases of incompressible potential and Euler flows.

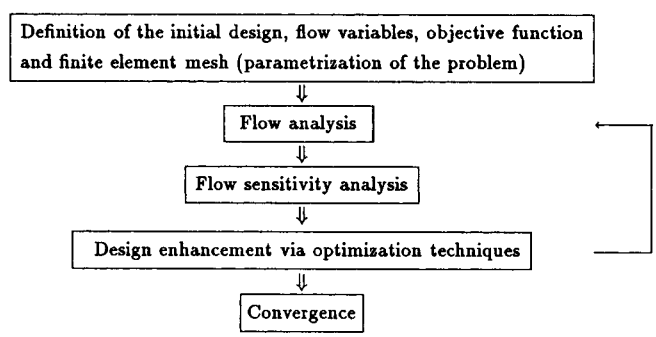

Figure 2. Classical optimization approach.

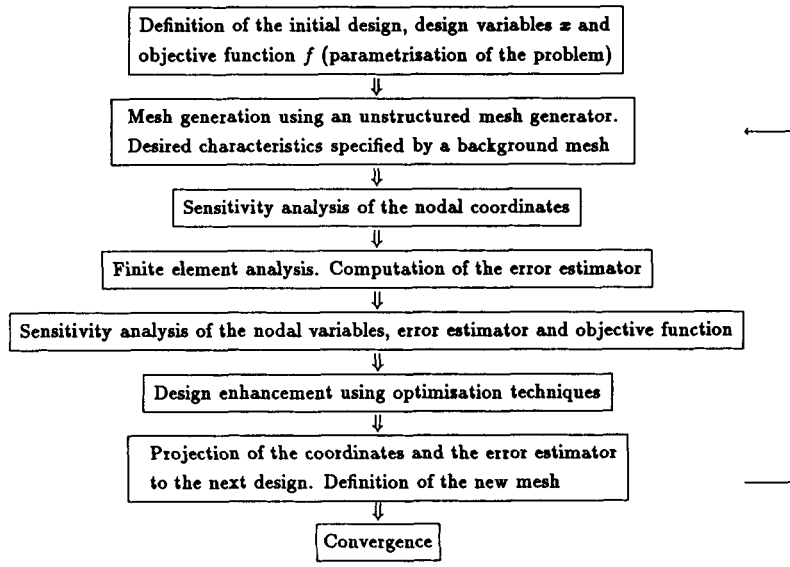

Figure 3. Proposed optimum shape design methodology.

PARAMETRIZATION OF THE PROBLEM

Each design geometry is represented by using 'definition points' which specify some interpolation curves. The curves used here are parametric B- splines. The general expression of a closed B-spline for q points is[4]

$$\mathbf{r}(t) = \sum_{l=0}^{q} r_l N_{4,l+1}(t), \tag{1}$$

where $\mathbf{r}(t)$ is the position vector depending on a parametric variable t. The co-ordinates of the definition points are recovered using $t = 0,1,2, \ldots$ (see Figure 4). The curve is expressed as a linear combination of $q + 1$ normalized fourth-order (cubic) B-splines. The degree of continuity of a cubic

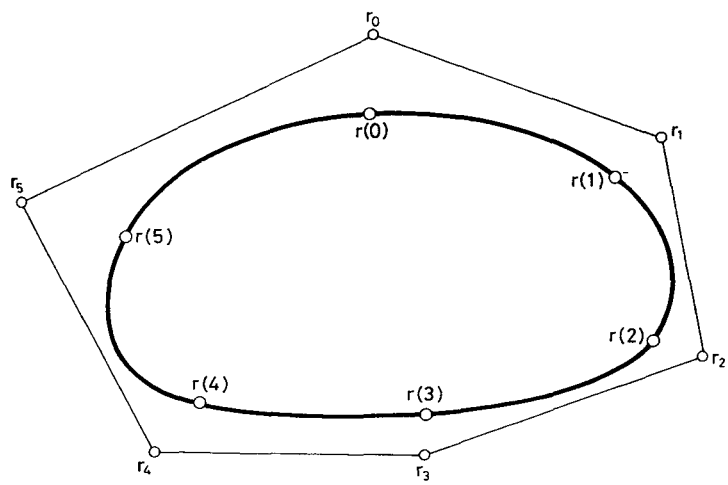

Figure 4. Definition points and polygon definition points of a B-spline.

B-spline is C^2. The coefficients $\mathbf{r}_i$ are the co-ordinates of the so-called polygon definition points and are found by using the co-ordinates of the definition points and some additional conditions about slopes and curvatures to build up a linear system of equations

$$\mathbf{V} = \mathbf{NR}, \tag{2}$$

where $\mathbf{V}$ is a vector containing the imposed conditions at the definition points (co-ordinates, slopes, curvatures, etc.), $\mathbf{N}$ is a matrix containing some terms corresponding to the values of the polynomials $N_{4,l+1}(t)$ that define each B-spline evaluated at the definition points, and the vector $\mathbf{R}$ contains the coefficients $\mathbf{r}_i$ to be computed. Details of this process can be found in Reference 4.

The first- and second-order sensitivities of $\mathbf{R}$ along a direction $\mathbf{s}$ in the variable space are given by

$$\frac{\partial \mathbf{R}}{\partial \mathbf{s}} = \mathbf{N}^{-1}\left(\frac{\partial \mathbf{V}}{\partial \mathbf{s}} - \frac{\partial \mathbf{N}}{\partial \mathbf{s}}\mathbf{R}\right), \qquad \frac{\partial^2 \mathbf{R}}{\partial \mathbf{s}^2} = \mathbf{N}^{-1}\left(\frac{\partial^2 \mathbf{V}}{\partial \mathbf{s}^2} - \frac{\partial^2 \mathbf{N}}{\partial \mathbf{s}^2}\mathbf{R} - 2\frac{\partial \mathbf{N}}{\partial \mathbf{s}}\frac{\partial \mathbf{R}}{\partial \mathbf{s}}\right). \tag{3}$$

The derivatives of $\mathbf{V}$ with respect to the co-ordinates of the definition points chosen as design variables can be easily computed. The vectors $\partial\mathbf{R}/\partial\mathbf{s}$ and $\partial^2\mathbf{R}/\partial\mathbf{s}^2$ will contain the terms $\partial\mathbf{r}_i/\partial\mathbf{s}$ and $\partial^2\mathbf{r}_i/\partial\mathbf{s}^2$ respectively.[3]

Finally, the sensitivities of the co-ordinates of any point on the interpolation curve corresponding to a constant value of t are obtained by

$$\frac{\partial \mathbf{r}(t)}{\partial \mathbf{s}} = \sum_{l=0}^{q} \frac{\partial \mathbf{r}_i}{\partial \mathbf{s}} N_{4,l+1}(t), \qquad \frac{\partial^2 \mathbf{r}(t)}{\partial \mathbf{s}^2} = \sum_{l=0}^{q} \frac{\partial^2 \mathbf{r}_i}{\partial \mathbf{s}^2} N_{4,l+1}(t). \tag{4}$$

The last expression is used to compute the sensitivities of the nodal points placed at the boundary of the finite element mesh.

MESH GENERATION AND SENSITIVITY ANALYSIS

In order to control the sizes of the elements, a non-structured mesh generation algorithm should be chosen. In this work the well-known advancing front method has been chosen.[5] The characteristics of the desired mesh are specified via a background mesh over which nodal values of the size parameter δ are defined and interpolated via the shape functions. The background mesh for the first design has to be specified by the user. For subsequent designs the background mesh is taken to coincide with the mesh projected into this design from the previous one. This projection process will be described later.

Once the sensitivities of the co-ordinates of each boundary node are known, it is also possible to compute the sensitivities of the co-ordinates of each internal nodal point (mesh sensitivities). These sensitivities are necessary to assess how the mesh evolves when the design variables change.

There are many different ways to define the evolution of the mesh in terms of the design variables. It is possible to consider a simple analogous elastic medium defining the mesh movement. This is the basis of the so-called 'spring analogy' where each element side is regarded as a spring connecting two nodes. The force induced by each spring is proportional to its length and the boundary nodes are considered as fixed. The solution of the equilibrium problem in the spring analogy is simple but expensive and it involves solving a linear system of equations with two degrees of freedom per node.

In this work the spring analogy problem has been solved iteratively using a simple Laplacian smoothing approach. This technique is frequently used to improve the quality of non-structured meshes. It consists of the iterative modification of the nodal co-ordinates of each interior node by placing it as the center of gravity of adjacent nodes. The expression of the new nodal position vector $\mathbf{r}_i$ for each iteration is given by

$$\mathbf{r}_i = \frac{\sum_{j=1}^{m_i} \mathbf{r}_j}{m_i}, \tag{5}$$

where $\mathbf{r}_j$ are the position vectors of the m_i nodes connected with the ith node.

The solution of the spring analogy problem with a prescribed error tolerance requires us to check the quality of the solution after each smoothing cycle. Taking into account that the described iterative process is only a way to obtain mesh sensitivities rather than the solution of the equilibrium problem itself, rigorous convergence conditions are not needed. For this reason the number of smoothing cycles to be applied can be fixed *a priori*. This allows us to substantially decrease the CPU time of the mesh sensitivity analysis compared with that required for the full resolution of the spring analogy equilibrium equations. In the examples presented later we have checked that 50 'smoothing' iterations are enough to ensure a good quality of results.

The first- and higher-order mesh sensitivities along any directions of the design variable space are obtained by differentiating equation (5) with respect to $\mathbf{s}$ for each cycle, i.e.

$$\frac{\partial \mathbf{r}_i}{\partial \mathbf{s}} = \frac{\sum_{j=1}^{m_i} \partial \mathbf{r}_j / \partial \mathbf{s}}{m_i}, \qquad \frac{\partial^2 \mathbf{r}_i}{\partial \mathbf{s}^2} = \frac{\sum_j^{m_i} \partial^2 \mathbf{r}_j / \partial \mathbf{s}^2}{m_i}. \tag{6}$$

The last equations provide the sensitivity analysis of the co-ordinates of all the internal nodes. There is a loop over all the internal nodes corresponding to each smoothing cycle. The sensitivities of each node are obtained from the sensitivities of the adjacent nodes. Before the first cycle only the boundary nodes have non-null sensitivities, obtained from equations (4).

FLOW ANALYSIS, ERROR ESTIMATOR AND SENSITIVITY, ANALYSIS FOR THE INCOMPRESSIBLE POTENTIAL FLOW MODEL

Flow analysis and error estimation

The basic equations for the analysis of the flow around a profile using an incompressible potential model with lifting involve a 'continuous' potential Φ_0 and a 'non-continuous' potential Φ_1 which are combined in order to accomplish the Kutta–Joukowski condition.[6]

Let us consider a domain Ω where the flow problem is defined. A cut Σ between the training edge *Te* and the boundary of Ω is defined (see Figure 5). The upper and lower sides of the cut are denoted Σ^+ and Σ^- respectively.

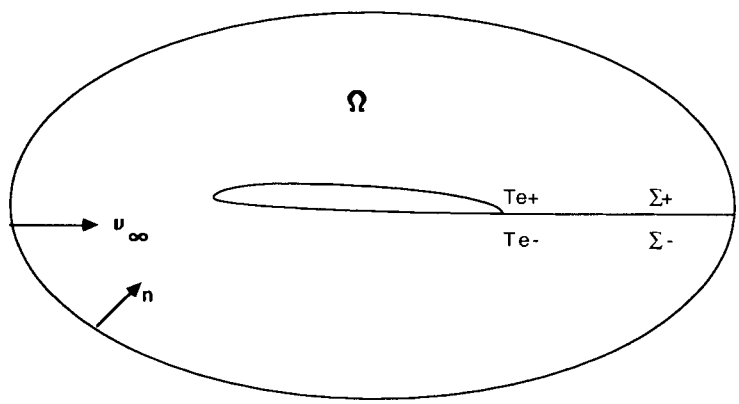

Figure 5. Definition of the domain and boundary conditions for the potential flow problem.

The corresponding equations for the 'continuous' and 'non-continuous' potentials are[6]

$$\begin{cases} \Delta\Phi_0 = 0, \\ \Phi_0|_{\Sigma^+} - \Phi_0|_{\Sigma^-} = 0, \\ \left.\dfrac{\partial\Phi_0}{\partial n^+}\right|_{\Sigma^+} + \left.\dfrac{\partial\Phi_n}{\partial n^-}\right|_{\Sigma^-} = 0, \\ \left.\dfrac{\partial\Phi_n}{\partial n}\right|_{\partial\Omega} = \mathbf{v}_\infty \mathbf{n}, \\ \Phi_0(Te^-) = 0, \end{cases} \qquad \begin{cases} \Delta\Phi_1 = 0, \\ \Phi_1|_{\Sigma^+} - \Phi_1|_{\Sigma^-} = 1, \\ \left.\dfrac{\partial\Phi_1}{\partial n^+}\right|_{\Sigma^+} + \left.\dfrac{\partial\Phi_1}{\partial n^-}\right|_{\Sigma^-} = 0, \\ \left.\dfrac{\partial\Phi_1}{\partial n}\right|_{\partial\Omega} = 0, \\ \Phi_0(Te^-) = 0. \end{cases} \tag{7}$$

The velocities at each point corresponding to each potential field are obtained from their gradients as $\mathbf{v}_0 = \nabla\Phi_0$ and $\mathbf{v}_1 = \lambda\nabla\Phi_1$.

The two potentials are combined to get the final solution $\Phi = \Phi_0 + \lambda\Phi_1$, where λ is obtained from the application of the Kutta–Joukowski condition[6] which ensures the continuity of the upper and lower velocities at the trailing edge. In this work the Kutta–Joukowski condition has been imposed by making equal the modulis of the velocities at both sides of the training edge ($[\mathbf{v}(Te^-)]^2 = [\mathbf{v}(Te^+)]^2$). This condition can be written in terms of the velocities corresponding to each potential:

$$[\mathbf{v}_0(Te^-) + \lambda\mathbf{v}_1(Te^-)]^2 = [\mathbf{v}_0(Te^+) + \lambda\mathbf{v}_1(Te^+)]^2. \tag{8}$$

The last expression allows us to obtain λ by solving a simple second-order polynomial. It will be obtained in terms of the velocities corresponding to each potential field at both sides of the training edge.

If more than one aerofoil is involved, a non-continuous potential is defined for each geometry and the Kutta–Joukowski condition is applied to each training edge. In that case there is a difference λ-factor corresponding to each aerofoil and expression (8) is replaced by a non-linear system of equations.

A finite element approximation can be used to discretize equations (7). The equations corresponding to each potential $\Phi_j, j = 0,1$, can be discretized separately, i.e. $\Phi_j \approx \hat{\Phi}_j = \sum_i N_i a_i = \mathbf{Na}$, leading to a linear system of equations in the form[6]

$$\mathbf{Ka} = \mathbf{f} \quad \text{with} \begin{cases} \mathbf{K} = \sum_e \mathbf{K}_e, \\ \mathbf{K}_e = \int_{\Omega_e} \mathbf{B}^{\mathrm{T}}\mathbf{B}\, d\Omega, \\ \mathbf{f} = \int_{\Gamma_e} \mathbf{N}\mathbf{v}^{\mathrm{T}}\mathbf{n}\, d\Gamma. \end{cases} \tag{9}$$

The matrix $\mathbf{B}$ contains the Cartesian derivatives of the shape functions N_i and can be used to obtain the velocities corresponding to the approximated potential as $\hat{\mathbf{v}} = \mathbf{B}\hat{\mathbf{\Phi}}$.

For incompressible potential problems the 'energy' of the exact solution can be defined as

$$\|U\| = \left(\int_\Omega \mathbf{v}^{\mathrm{T}}\mathbf{v}\, d\Omega\right)^{1/2}. \tag{10}$$

The error of this 'energy' can be estimated using the error estimator developed by Zienkiewicz and Zhu[7] for structural problems. The extension to potential flow problems requires the definition of the global error norm

$$\|e\| = \left(\int_\Omega (\mathbf{v} - \hat{\mathbf{v}})^{\mathrm{T}}(\mathbf{v} - \hat{\mathbf{v}})\, d\Omega\right)^{1/2}, \tag{11}$$

where $\mathbf{v}$ are the 'exact' velocities, are the velocities obtained from the finite element solution and Ω is the flow analysis domain.

Since the exact velocities are usually not known, they are approximated by $\mathbf{v} \approx \bar{\mathbf{v}}^* = \mathbf{N}\bar{\mathbf{v}}^*$, where $\bar{\mathbf{v}}^*$ are nodal values obtained by simple nodal averaging of the finite element values, local or global least squares smoothing or other appropriate projection methods.[7] A simple approach is to use global nodal smoothing with a 'mass' matrix, giving the nodal smoothed values $\mathbf{v}^*$ as

$$\mathbf{v}^* = \mathbf{M}^{-1} \int_\Omega \mathbf{N}_v \hat{\mathbf{v}} \, d\Omega, \tag{12}$$

where $\mathbf{N}_v$ are the chosen velocity- interpolating functions giving a smooth nodal velocity field[7] and $M_{ij} = \int_\Omega N_{v_i} N_{v_j} \, d\Omega$. Equation (12) can obviously be applied to solve independently for each individual velocity component.

The 'energy' of the exact solution is estimated as

$$\|U\| \approx \left(\int_\Omega \mathbf{v}^{*\mathrm{T}} \mathbf{v}^* \, d\Omega + \int_\Omega (\mathbf{v}^* - \hat{\mathbf{v}}) \, d\Omega \right)^{1/2}. \tag{13}$$

Both $\|e\|^2$ and $\|U\|^2$ can be evaluated as sums of their respective element contributions.

Mesh adaptivity

For the complete definition of the characteristics of a new mesh in the remeshing procedure it is necessary to use a mesh optimality criterion. In this work a mesh is considered as optimal when the error density is equally distributed across the volume, i.e. when $\|e\|_e^2/\Omega_e = \|e\|^2/\Omega$ is satisfied. The justification for this optimality criterion can be found in References 8–10.

The combination of the optimality criterion and the error estimation allows us to define the new element sizes. First it is necessary to define the limit of the allowable global error percentage γ as

$$\gamma = 100 \frac{\|e\|}{\|\Phi\|} \approx 100 \frac{\|e\|}{\sqrt{\left(\|e\|^2 + \|\hat{\Phi}\|^2\right)}}. \tag{14}$$

The desired error level for each element is

$$\|e\|_e^{\mathrm{d}} = \frac{\gamma}{100} \sqrt{\left((\|\hat{\Phi}\|^2 + \|e\|^2) \frac{\Omega_e}{\Omega} \right)}. \tag{15}$$

The new element sizes $\bar{h}_e$ can be computed in terms of the old ones h_e using the expression

$$\bar{h}_e = \frac{h_e}{\xi_e^{1/p}}, \tag{16}$$

where $\xi_e = \|e\|_e / \|e\|_e^d$ and p is the order of the shape function polynomials. For further details see References 8–10.

Sensitivity analysis of the objective function and the error estimator

The exact sensitivity analysis of all the integral expressions involved in the finite element discretization of the incompressible potential flow model can be obtained by direct derivation of equations (9). This provides the sensitivities of all magnitudes in terms of the mesh sensitivities previously obtained (details of this process are described in References 3). The sensitivities of an integral expression are computed after its transformation into the isoparametric domain whose shape does not depend on the design variables. The jacobian of this transformation, $|\mathbf{J}|$, can be expressed in

terms of the nodal co-ordinates, so that it can also be differentiated in order to know the integral sensitivities. Using the techniques developed in Reference 3, the sensitivities of the element stiffness matrix can be obtained as

$$\frac{\partial \mathbf{K}_e}{\partial \mathbf{s}} = \int_{\Omega_\xi} \left(\frac{\partial \mathbf{B}^{\mathrm{T}}}{\partial \mathbf{s}} \mathbf{B}|\mathbf{J}| + \mathbf{B}^{\mathrm{T}} \frac{\partial \mathbf{B}}{\partial \mathbf{s}} |\mathbf{J}| + \mathbf{B}^{\mathrm{T}}\mathbf{B} \frac{\partial |\mathbf{J}|}{\partial \mathbf{s}} \right) \mathrm{d}\xi_1 \, \mathrm{d}\xi_2, \tag{17}$$

where the sensitivity of the Jacobian is

$$\frac{\partial |\mathbf{J}|}{\partial \mathbf{s}} = |\mathbf{J}| \mathrm{tr}\left(\mathbf{J}^{-1} \frac{\partial \mathbf{J}}{\partial \mathbf{s}} \right). \tag{18}$$

In equation (17) the matrix $\mathbf{B}$ depends on the nodal co-ordinates, so that $\partial \mathbf{B}/\partial \mathbf{s}$ can be obtained from the mesh sensitivities.

This technique allow us to obtain first- and higher-order sensitivities of the element stiffness matrix $\mathbf{K}$ and of any other integral expression involved in the analysis. The detailed expressions for the first- and higher- order sensitivity analysis can be found in Reference 3.

Equation (17) allows us to obtain the sensitivities of each nodal variable $\mathbf{a}$ as

$$\frac{\partial \mathbf{a}}{\partial \mathbf{s}} = \mathbf{K}^{-1} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{s}} - \frac{\partial \mathbf{K}}{\partial \mathbf{s}} \mathbf{a} \right), \qquad \frac{\partial^2 \mathbf{a}}{\partial \mathbf{s}^2} = \mathbf{K}^{-1} \left(\frac{\partial^2 \mathbf{f}}{\partial \mathbf{s}^2} - \frac{\partial^2 \mathbf{K}}{\partial \mathbf{s}^2} \mathbf{a} - 2 \frac{\partial \mathbf{K}}{\partial \mathbf{s}} \frac{\partial \mathbf{a}}{\partial \mathbf{s}} \right). \tag{19}$$

Equation (19) show that the inverse of the stiffness matrix is needed for the sensitivity computations. If a direct solver is used, this matrix has already been factorized and each new sensitivity analysis involves only a new back-substitution process. Moreover, it is not necessary to assemble the sensitivities of the stiffness matrix, because they always appear multiplying a vector and these products can be computed in an element-by-element manner. The corresponding expressions can be separately applied to each potential field ($\hat{\Phi}_0$, $\hat{\Phi}_1$, ...).

For a single aerofoil the sensitivity analysis of the total potential $\hat{\Phi} = \hat{\Phi}_0 + \lambda \hat{\Phi}_1$ involves a combination of the sensitivities of $\hat{\Phi}_0$, $\hat{\Phi}_1$, and λ

$$\frac{\partial \hat{\Phi}}{\partial \mathbf{s}} = \frac{\partial \hat{\Phi}_0}{\partial \mathbf{s}} + \lambda \frac{\partial \hat{\Phi}_1}{\partial \mathbf{s}} + \frac{\partial \lambda}{\partial \mathbf{s}} \hat{\Phi}_1, \qquad \frac{\partial^2 \hat{\Phi}}{\partial \mathbf{s}^2} = \frac{\partial^2 \hat{\Phi}_0}{\partial \mathbf{s}^2} + \lambda \frac{\partial^2 \hat{\Phi}_1}{\partial \mathbf{s}^2} + 2 \frac{\partial \lambda}{\partial \mathbf{s}} \frac{\partial^2 \hat{\Phi}_1}{\partial \mathbf{s}^2} + \frac{\partial^2 \lambda}{\partial \mathbf{s}^2} \hat{\Phi}_1. \tag{20}$$

The terms $\partial \lambda / \partial \mathbf{s}$ and $\partial^2 \lambda / \partial \mathbf{s}^2$ can be obtained from the derivation of equation (8) with respect to $\mathbf{s}$. These will be expressed in terms of the velocities at both sides of the trailing edge and their sensitivities.

The sensitivities of the velocities corresponding to each potential field can be obtained via the equations

$$\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{s}} = \frac{\partial \mathbf{B}}{\partial \mathbf{s}} \hat{\Phi} + \mathbf{B} \frac{\partial \hat{\Phi}}{\partial \mathbf{s}}, \qquad \frac{\partial^2 \boldsymbol{\Phi}}{\partial \mathbf{s}^2} = \frac{\partial^2 \mathbf{B}}{\partial \mathbf{s}^2} \hat{\Phi} + 2 \frac{\partial \mathbf{B}}{\partial \mathbf{s}} \frac{\partial \hat{\Phi}}{\partial \mathbf{s}} + \mathbf{B} \frac{\partial^2 \hat{\Phi}}{\partial \mathbf{s}^2}. \tag{21}$$

Obviously these computations are all performed at the Gauss point level. The same operation can be performed to compute the sensitivities of the velocities at the nodal points ($\bar{\mathbf{v}}^*$) in terms of the velocities at the integration points by direct differentiation of expression (12) and using the corresponding tools for the differentiation of the integral expressions.[7]

Very similar equations to (20) and (21) hold the cases with more than one aerofoil. Here there are as many λ-parameters as different components where the Kutta–Joukowski condition is imposed. These parameters are obtained by solving a non-linear system of equations, each one corresponding to the Kutta–Joukowski condition applied over one aerofoil. To obtain the sensitivities of the λ-parameters, it is necessary to derivate this system of equations with respect to the velocities at the trailing edge of each aerofoil.

The computation of the pressure coefficient C_p and its sensitivities in terms of the velocities for the case of an incompressible flow is trivial using the expression

$$C_p = 1 - \left(\frac{|\mathbf{v}|}{|\mathbf{v}_\infty|}\right)^2. \tag{22}$$

The sensitivities of the objective function can be obtained by its direct derivation with respect to the design variables. These sensitivities can finally be expressed in terms of the sensitivities of the nodal velocities.

The sensitivities of the error estimator and the 'energy' of the solution can be obtained in terms of the sensitivities of the velocities by appropriate derivation of the integral expressions (11) and (13) using the same technique as described above.[3]

FLOW ANALYSIS, ERROR ESTIMATOR AND SENSITIVITY ANALYSIS FOR THE EULER EQUATIONS

Flow analysis

We will consider here for simplicity the solution of the Euler equations neglecting the temperature effects. These equations can be written in conservative form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x_i} + \mathbf{Q} = 0, \tag{23}$$

where for 2D problems

$$\mathbf{U} = [\rho,\ U_1,\ U_2]^{\mathrm{T}}, \qquad \mathbf{F}_i = [\rho u_i,\ U_1 u_i + p\delta_{1i},\ U_2 u_i + p\delta_{2i}]^{\mathrm{T}}, \qquad \mathbf{Q} = [0,\ -\rho f_1,\ -\rho f_2]^{\mathrm{T}}. \tag{24}$$

In the above $U_i = \rho u_i$, where ρ is the density and u_i is the flow velocity along the global ith axis, f_i are source terms due to gravitational acceleration and p is the pressure related to the density by the standard constitutive law $p = RT\rho$, where R is the universal gas constant and T is the temperature. Note that isothermal conditions have been assumed in (23) and (24).

The time–space solution of equation (23) and (24) can be found in a variety of ways. A simple option is provided by the well-known explicit Taylor–Galerkin procedure,[7] giving the final discretized systems of equations in the form

$$\mathbf{M}\Delta\mathbf{a} = -\Delta t[(\mathbf{V} + \mathbf{H})\mathbf{a} + \mathbf{f}]^n, \tag{25}$$

where $\Lambda\mathbf{a}$ are the unknown incremental variables, $\mathbf{a}$ is the vector of nodal variables and

$$\begin{aligned} \mathbf{M} &= \int_\Omega \mathbf{N}^{\mathrm{T}}\mathbf{N}\,\mathrm{d}\Omega, \qquad \mathbf{V} = \int_\Omega \mathbf{N}^{\mathrm{T}}\mathbf{A}_i\frac{\partial \mathbf{N}}{\partial x_i}\,\mathrm{d}\Omega, \\ \mathbf{H} &= \frac{\Delta t}{2}\int_\Omega \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_i}\mathbf{A}_i\mathbf{A}_j\frac{\partial \mathbf{N}}{\partial x_j}\,\mathrm{d}\Omega, \qquad \mathbf{f} = \int_\Omega \left(\mathbf{N}^{\mathrm{T}} + \frac{\Delta t}{2}\mathbf{A}_i\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_i}\right)\mathbf{Q}\,\mathrm{d}\Omega + \mathrm{BT}. \end{aligned} \tag{26}$$

Here BT denotes boundary terms and

$$\mathbf{A}_i(\mathbf{U}) = \frac{\partial \mathbf{F}_i}{\partial \mathbf{U}}. \tag{27}$$

The matrix $\mathbf{H}$ in (25) plays the role of a balancing diffusion helping to stabilize the numerical solution. Full details of this algorithm can be found in Reference 7. The discretized form (25) is usually completed by adding an '*ad hoc*' shock-capturing term in the form of a crosswind diffusion which

guarantees monotonicy of the solution. Here many alternatives are possible and some of the most popular options can be found in Reference 7.

Once the steady state has been reached, equation (25) reduces to (neglecting the shock-capturing term)

$$\mathbf{W}(\mathbf{d},\ \mathbf{a}) = (\mathbf{V} + \mathbf{H})\mathbf{a} + \mathbf{f} = 0, \tag{28}$$

where $\mathbf{d}$ are the appropriate design parameters. Note that the value of Δt in terms $\mathbf{H}$ and $\mathbf{f}$ refers now to the last time increment of the steady state solution.

Equation (28) can be used as the starting point to compute the sensitivities of the nodal variables with respect to the chosen design parameters, as will be shown in the next subsection.

Sensitivity analysis

In order to compute the sensitivities of the objective function with respect to the design parameters it is necessary to compute first the sensitivities of the nodal variables of the problem.

In the following we will assume that vector $\mathbf{W}$ of (28) accomplishes the necessary requirements for the application of the implicit function theorem. In this case the sensitivity of the nodal variables $\mathbf{a}$ along a direction $\mathbf{s}$ in the design variable space can be computed by

$$\begin{aligned}\frac{\partial \mathbf{a}}{\partial \mathbf{s}} &= -\left(\frac{\partial \mathbf{W}}{\partial \mathbf{a}}\right)^{-1}\frac{\partial \mathbf{W}}{\partial \mathbf{s}} = -\left(\frac{\partial[(\mathbf{V}+\mathbf{H})\mathbf{a}+\mathbf{f}]}{\partial \mathbf{a}}\right)^{-1}\frac{\partial[(\mathbf{V}+\mathbf{H})\mathbf{a}+\mathbf{f}]}{\partial \mathbf{s}} \\ &= -\left[\mathbf{V}+\mathbf{H}+\left(\frac{\partial \mathbf{V}}{\partial \mathbf{a}}+\frac{\partial \mathbf{H}}{\partial \mathbf{a}}\right)\mathbf{a}+\frac{\partial \mathbf{f}}{\partial \mathbf{a}}\right]^{-1}\left[\left(\frac{\partial \mathbf{V}}{\partial \mathbf{s}}+\frac{\partial \mathbf{H}}{\partial \mathbf{s}}\right)\mathbf{a}+\frac{\partial \mathbf{f}}{\partial \mathbf{s}}\right].\end{aligned} \tag{29}$$

The computation of the partial derivatives of $\mathbf{V}$, $\mathbf{H}$ and $\mathbf{f}$ with respect to $\mathbf{a}$ can be obtained using the expressions

$$\begin{aligned}&\frac{\partial \mathbf{V}}{\partial \mathbf{a}} = \int_\Omega \mathbf{N}^{\mathrm{T}}\frac{\partial \mathbf{A}_i}{\partial \mathbf{a}}\frac{\partial \mathbf{N}}{\partial x_i}\,\mathrm{d}\Omega, \qquad \frac{\partial \mathbf{H}}{\partial \mathbf{a}} = \frac{\Delta t}{2}\int_\Omega \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_i}\left(\frac{\partial \mathbf{A}_i}{\partial \mathbf{a}}\mathbf{A}_j + \mathbf{A}_i\frac{\partial \mathbf{A}_j}{\partial \mathbf{a}}\right)\frac{\partial \mathbf{N}}{\partial x_j}\,\mathrm{d}\Omega, \\ &\frac{\partial \mathbf{f}}{\partial \mathbf{a}} = \int_\Omega\left(\mathbf{N}^{\mathrm{T}} + \frac{\Delta t}{2}\frac{\partial \mathbf{A}_i}{\partial \mathbf{a}}\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_i}\right)\mathbf{Q}\,\mathrm{d}\Omega + \int_\Omega\left(\mathbf{N}^{\mathrm{T}} + \frac{\Delta t}{2}\mathbf{A}_i\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_i}\right)\frac{\partial \mathbf{Q}}{\partial \mathbf{a}}\,\mathrm{d}\Omega + \frac{\partial(\mathrm{B.T.})}{\partial \mathbf{a}}.\end{aligned} \tag{30}$$

The partial derivatives of $\mathbf{A}_i$ with respect to $\mathbf{a}$ can be obtained from its definition in (27).

For the computation of the partial derivatives of $\mathbf{V}$, $\mathbf{H}$ and $\mathbf{f}$ with respect to $\mathbf{s}$ it has to be noted from (26) that the terms $\mathbf{V}$, $\mathbf{H}$ and $\mathbf{f}$ are integral expressions that can be computed by assembling the contributions of all the elements. Then the sensitivities of each term can also be computed by assembly of all the elemental sensitivities. For instance, the matrix $\mathbf{V}$ can be computed as

$$\mathbf{V} = \sum_e \mathbf{V}_e = \sum_e \int_{\Omega_e} \mathbf{N}^{\mathrm{T}}\mathbf{A}_i\frac{\partial \mathbf{N}}{\partial x_i}\,\mathrm{d}\Omega = \sum_e \int_{\Omega_\xi}\left(\mathbf{N}^{\mathrm{T}}\mathbf{A}_i\frac{\partial \mathbf{N}}{\partial x_i}\right)|\mathbf{J}|\,\mathrm{d}\xi_1\,\mathrm{d}\xi_2, \tag{31}$$

where the last integral has been transformed into the isoparametric domain as in expression (17) corresponding to the potential flow equations. Then the sensitivities of (31) can be obtained by

$$\frac{\partial \mathbf{V}}{\partial \mathbf{s}} = \sum_e \frac{\partial \mathbf{V}_e}{\partial \mathbf{s}} = \sum_e \int_{\Omega_\xi}\left(\mathbf{N}^{\mathrm{T}}\mathbf{A}_i\frac{\partial^2 \mathbf{N}}{\partial x_i \partial \mathbf{s}}\right)|\mathbf{J}|\,\mathrm{d}\xi_1\,\mathrm{d}\xi_2 + \sum_e \int_{\Omega_\xi}\left(\mathbf{N}^{\mathrm{T}}\mathbf{A}_i\frac{\partial \mathbf{N}}{\partial x_i}\right)\frac{\partial |\mathbf{J}|}{\partial \mathbf{s}}\,\mathrm{d}\xi_1\,\mathrm{d}\xi_2, \tag{32}$$

where the sensitivities of $|\mathbf{J}|$ can be obtained using (18).

The second derivatives of $\mathbf{N}$ with respect to x_i and $\mathbf{s}$ can also be computed in terms of the nodal coordinate sensitivities (see Reference 3 for more details).

Error estimation and adaptivity

For one-dimensional problems the local error e in the solution can be defined as the difference between the exact solution w (where w denotes any of the field variables) and the approximated solution obtained after the finite element discretization, $\hat{w}$:

$$e = w - \hat{w}. \tag{33}$$

If linear elements are used, as is normally the case in compressible flow problems, and assuming that the solution is exact at the nodes, the error can be expressed within each element as

$$e = Ch_e^2 \frac{\partial^2 \hat{w}}{\partial x^2}, \tag{34}$$

where h_e is the size of the element and C is a constant.[7] The remeshing strategy aims to obtain a new mesh where this error is equally bounded everywhere. The most popular strategy for the definition of the new mesh is to distribute this error uniformly over the domain. In this case the requirement is that the total error should be equally distributed between all the elements, i.e.

$$h^2 \frac{\partial^2 \hat{w}}{\partial x^2} = \text{constant}. \tag{35}$$

Equation (35) suggests that the new mesh should be generated with a local spacing δ determined from the condition

$$\delta^2 \frac{\partial^2 \hat{w}}{\partial x^2} = e_p, \tag{36}$$

where e_p is normally chosen in order to get 'reasonable' meshes avoiding too small elements. Typically $\delta_{\min}^2(\partial^2 \hat{w}/\partial x^2)_{\max}$, where $\delta_{\min}$ is the minimum element size chosen and $(\partial^2 \hat{w}/\partial x^2)_{\max}$ is the maximum value of the second derivative in the finite element mesh.

For two-dimensional problems the second derivatives of a typical variable of the problem, U, can be computed at each node as

$$D_{jk} = \frac{\partial^2 \hat{U}}{\partial x_j \partial x_k}. \tag{37}$$

The local principal directions $\mathbf{l}_1$ and $\mathbf{l}_2$ of the Hessian matrix $\mathbf{D}$ can be determined and the corresponding quantities can be obtained

$$\lambda_1 = \frac{\partial^2 \hat{U}}{\partial l_1^2}, \qquad \lambda_2 = \frac{\partial^2 \hat{U}}{\partial l_2^2}, \qquad |\lambda_1| > |\lambda_2|. \tag{38}$$

The one-dimensional process outlined above can be applied along each of the principal directions, leading to the requirement that

$$\delta_1^2 |\lambda_1| = \delta_2^2 |\lambda_2| = \delta_{\min}^2 |\lambda|_{\max}, \tag{39}$$

where δ_1 and δ_2 denote the new spacings in the directions $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$. These spacings computed at each nodal point define the characteristics of the new mesh.

The control variable $\hat{U}$ can be chosen to be one of the nodal variables of the problem, e.g. the density, or a combination of any of them, e.g. the Mach number or the modulus of the velocity.

The computation of the second derivatives of $\hat{U}$ needs some explanation. For linear elements these second derivatives are zero within the elements and take an infinite value at their boundaries. One

solution to this problem is to define a second-derivative field interpolated with the same shape functions as in the discretization of the flow problem, i.e.

$$\frac{\partial^2 \hat{U}}{\partial x_j \partial x_k} \approx \mathbf{N}\left(\frac{\partial^2 \bar{U}}{\partial x_j \partial x_k}\right). \tag{40}$$

This new second-derivative field is adjusted to the original one by a least squares method as

$$\int_\Omega \mathbf{N}^{\mathrm{T}}\left(\mathbf{N}\frac{\partial^2 \bar{U}}{\partial x_j \partial x_k} - \frac{\partial^2 \hat{U}}{\partial x_j \partial x_k}\right)\mathrm{d}\Omega = 0. \tag{41}$$

After integrating by parts, (40) gives

$$\frac{\partial^2 \bar{U}}{\partial x_j \partial x_k} = \mathbf{M}^{-1}\left(\int_\Omega \mathbf{N}^{\mathrm{T}} \frac{\partial^2 \hat{U}}{\partial x_j \partial x_k}\right) = -\mathbf{M}^{-1}\left(\int_\Omega \frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial x_j}\frac{\partial \mathbf{N}}{\partial x_k}\,\mathrm{d}\Omega\right)\hat{U}, \tag{42}$$

where $\mathbf{M}$ is the mass matrix defined in (26).

The computation of the sensitivities of the second-derivative field in terms of the sensitivities of the nodal variables can be obtained following the same rules as in the sensitivity analysis of the flow variables.

The process outlined above provides the standard remeshing strategy or the analysis of compressible fluid flow problems. We will show next how this strategy can be introduced into an optimum design process to define the characteristics of the meshes for successive designs.

DESIGN ENHANCEMENT AND DEFINITION OF THE NEW MESH

The objective function sensitivities are used to get improved values of the design parameters by means of a standard minimization method.[3] Depending on the optimization algorithm, it may be necessary to use second- order sensitivities. The design variables corresponding to the improved design will usually be found as

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \theta \mathbf{s}^k, \tag{43}$$

where θ is an advance parameter. The direction of change, $\mathbf{s}^k$, can be obtained using a BFGS quasi-Newton method or a GMRES method which only requires first derivatives of the objective function. The value of θ can be obtained by a line search procedure. One possibility is to use a second-order sensitivity analysis in the direction $\mathbf{s}^k$. The objective function can then be approximated along this direction using a second-order Taylor expansion whose minimization provides the value of θ. Details of this algorithm can be found in Reference 3.

Once the new design has been defined, all the relevant variables for the adaptive remeshing strategy can be projected from the old design to the new one using the sensitivity analysis. Second-order sensitivities can be used if required. The co-ordinates of the projected mesh are obtained using

$$(x, y)^{k+1} \approx (x, y)^k + \theta\left(\frac{\partial x}{\partial \mathbf{s}}, \frac{\partial y}{\partial \mathbf{s}}\right) + \frac{1}{2}\theta^2\left(\frac{\partial^2 x}{\partial \mathbf{s}^2}, \frac{\partial^2 y}{\partial \mathbf{s}^2}\right). \tag{44}$$

For the case of incompressible potential equations the projections of the error estimator and the 'energy' can be obtained from

$$\|e\|^{2^{k+1}} = \|e\|^{2^k} + \theta \frac{\partial \|e\|^2}{\partial \mathbf{s}} + \frac{1}{2}\theta^2 \frac{\partial^2 \|e\|^2}{\partial \mathbf{s}^2}, \tag{45}$$

$$\|U\|^{2^{k+1}} = \|U\|^{2^k} + \theta \frac{\partial \|U\|^2}{\partial \mathbf{s}} + \frac{1}{2}\theta^2 \frac{\partial^2 \|U\|^2}{\partial \mathbf{s}^2}. \tag{46}$$

For the case of the Euler equations any of the nodal variables can be projected using

$$\left(\frac{\partial^2 \bar{U}_i}{\partial x_j \partial x_k}\right)^{k+1} = \left(\frac{\partial^2 \bar{U}_i}{\partial x_j \partial x_k}\right)^{k} + \theta \frac{\partial}{\partial \mathbf{s}}\left(\frac{\partial^2 \bar{U}_i}{\partial x_j \partial x_k}\right) + \frac{1}{2}\theta^2 \frac{\partial^2}{\partial \mathbf{s}^2}\left(\frac{\partial^2 \bar{U}_i}{\partial x_j \partial x_k}\right). \tag{47}$$

These projections provide a good approximation of each of the above values for the next design configuration prior to any new computations. In fact, the projected values provide the necessary information to perform a remeshing over the next design *even before any new computation is attempted*. In that sense we have changed an error estimator computed *a posteriori* into an *a priori* error estimator for the definition of a new mesh.

This projection is of paramount importance since it allows us to control the quality of the mesh for each new design. Only a single mesh is generated in each new design analysis step. Thus the extra computation cost involved in the control of the mesh quality is very cheap.

The projected values are used to create the background mesh information needed to generate the mesh corresponding to the new design geometry. This operation closes the iterative process which will lead to the optimum design geometry after convergence.

APPLICATION EXAMPLES

In order to show the good behaviour of the presented methodology applied to the resolution of aerodynamic shape optimization problems, the results of two reconstruction cases are presented. The flow results have been obtained using the potential flow model.

Korn aerofoil

The example chosen is the recovery of a Korn aerofoil at an angle of attack of 0° as proposed in Reference 3. The target pressure coefficient C_p^{target} has been obtained by a direct computation of the Korn aerofoil with a finite element code including adaptive remeshing with a maximum global error of 0.1%. The initial design corresponds to an NACA 64A410 profile.

The inverse problem has been solved using a minimization approach. The cost functional to be minimized has been defined as

$$f = \int_0^2 \left[C_p(x) - C_p^{\text{target}}\right]^2 \mathrm{d}s. \tag{48}$$

This integral is extended around the profile and the integration variable is the arc s.

The geometry of each design has been defined using 25 design variables. These variables are the y-co-ordinates of 25 points distributed around the profile which are used to interpolate a B-spline. Figure 6 shows the initial shape and the finite element mesh used for the initial design. The 25 points used to define the shape of each design are all the nodes lying on the profile in Figure 6, with the exception of the trailing edge which is fixed.

In this example particular emphasis has been put on

(1) The accuracy of the numerical solution
(2) The efficiency of the process.

Figure 7 shows the final mesh and shape obtained for this problem using a global error limited to 0.1% of the total potential norm.

Figure 6 shows a superposition of the C_p distributions corresponding to the target (Korn) aerofoil, the initial (NACA 64A410) aerofoil and the final design obtained with a 0.1% error tolerance. As can be observed, the agreement between the target and the final design is almost perfect.

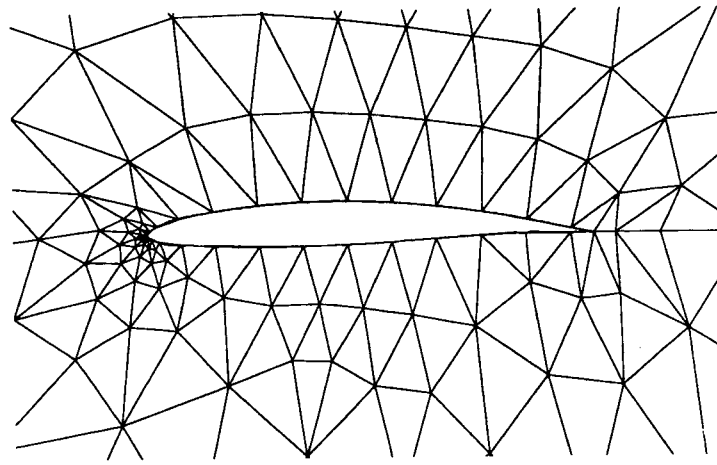

Figure 6. Initial shape, initial mesh and definition of design variables for the Korn aerofoil problem.

Figure 9 shows the convergence of the normalized cost functional using different global error norm values and P2 triangular finite elements. It can be observed that it is necessary to substantially decrease the global error norm to produce a significant improvement in the final value of the cost functional (48).

Figure 10 shows the computing time necessary to perform 100 iterations of the BFGS algorithm for different values of the global error required. These computations have been performed on a Silicon Graphics Indigo R4000 workstation. It is seen that an increment in the precision of the numerical solution involves a big increment in the computing time. On the other hand, it is possible to obtain a quite good solution in a short time (1 h) allowing for a bigger global error. Once a solution with a given global error norm is obtained, it is also possible to restart the process from this solution using a smaller global error.

Figure 11 shows the number of finite elements used for each iteration and for different global errors. The relationship between the global error required and the increase in the number of elements for each mesh can be observed. This explains the big increase in the computing cost for high precisions. In any case the percentage of error is perfectly controlled and always below the imposed limitation.

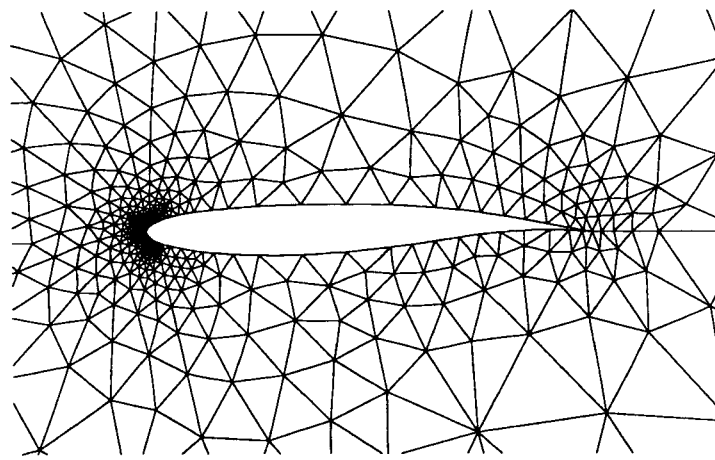

Figure 7. Final shape and mesh for a global error of 0.1% for the Korn aerofoil problem.

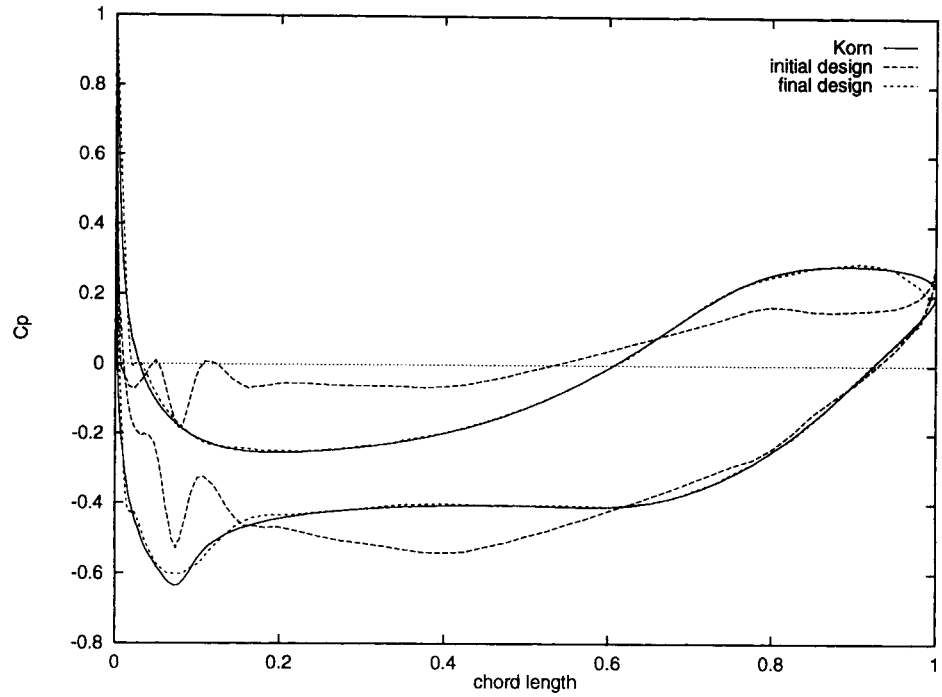

Figure 8. Superposition of the C_p distributions corresponding to the Korn aerofoil, the initial design and the final design.

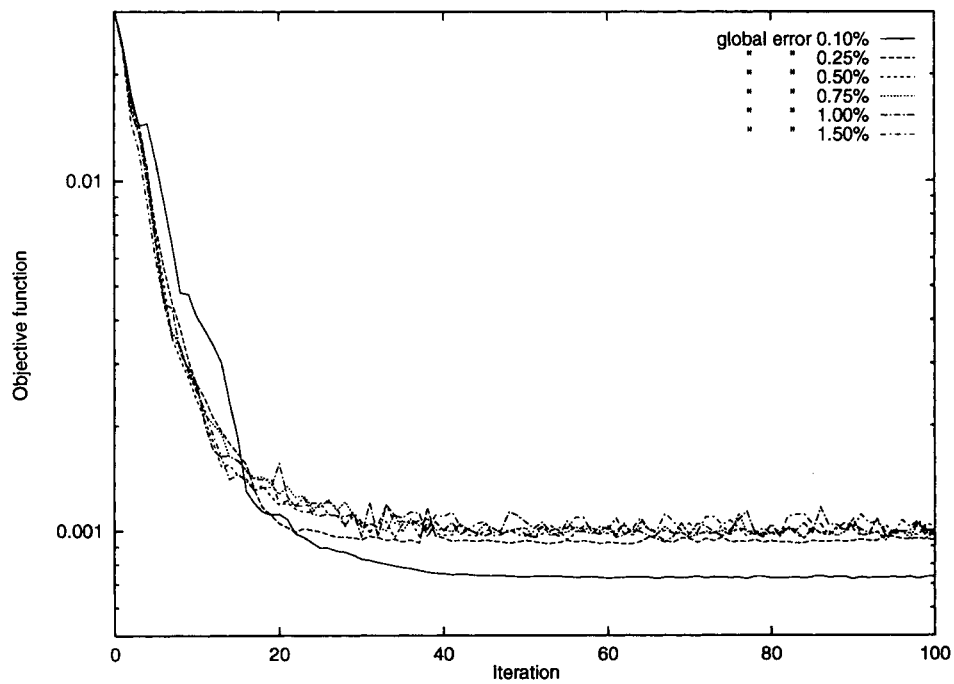

Figure 9. Evolution of the cost functional for the Korn aerofoil problem.

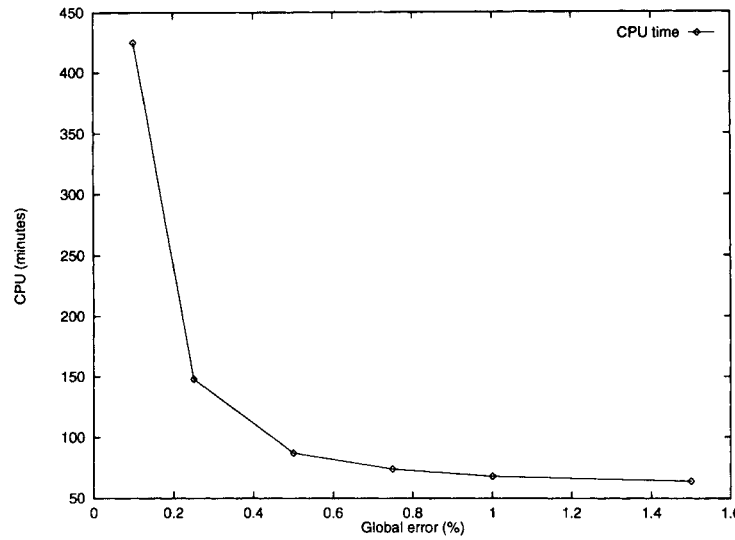

Figure 10. CPU time for different percentages of global error.

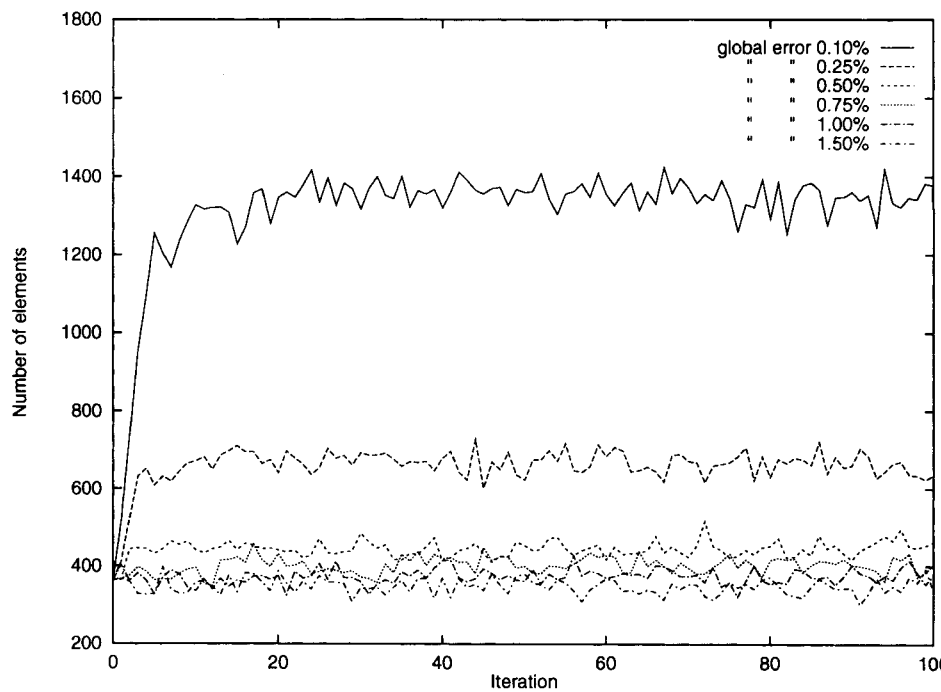

Figure 11. Number of elements for each mesh.

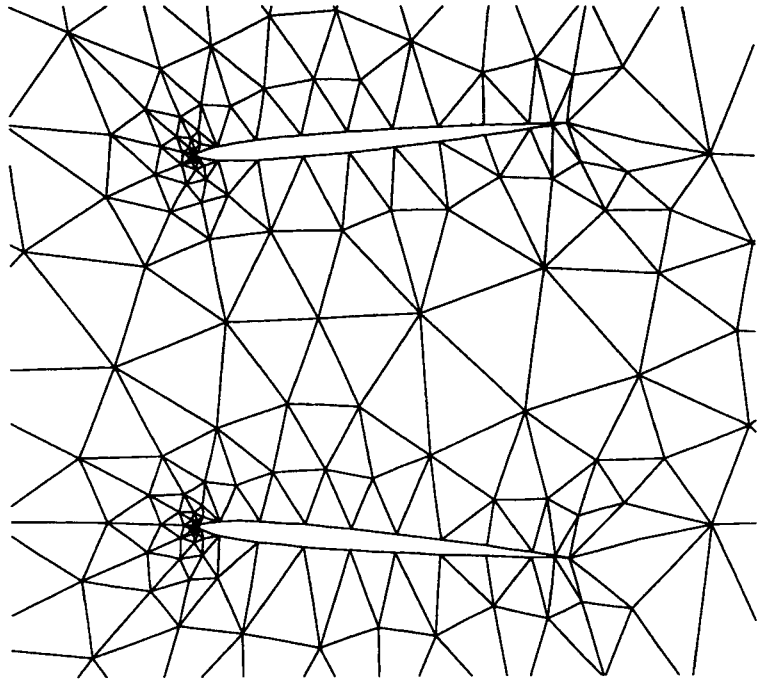

Figure 12. Initial shape and mesh for the two-element problem.

Two-element problem

This case consists of recovering two NACA 0012 profiles at an angle of attack of 0° positioned as shown in Figure 13. The starting design is formed by the two profiles shown in Figure 12. Each profile has been obtained from an NACA 0012 by reducing its thickness to one-half and rotating it 5° around its leading edge.

The target pressure coefficient C_p^{target} has been obtained by a direct computation of the target design as in the previous example.

The inverse problem has been solved using a minimization approach. The cost functional to be minimized has been defined as in equation (48), extending the integral to the boundaries of both aerofoils.

The geometry of each aerofoil has been defined using 18 design variables. These variables are the y-co-ordinates of 18 points distributed around each profile which are used to interpolate a B-spline. Figure 12 shows the initial shape and the finite element mesh used for the initial design. The 18 points used to define the shape of each profile are all the nodes lying on them in Figure 12, with the exception of the trailing and leading edges. For each profile its rotation around its leading edge has been defined as an additional design variable. This means that two angles have been defined as design variables. The total number of design variables is thus 38. The maximum global error during the minimization process has been limited to 0.2% of the total potential norm.

The iterative process has been considered as converged after 100 iterations. The final shape and mesh can be observed in Figure 13. The whole problem has taken around 100 h of CPU time on a Convex C-3480 computer using a single processor. It is important to note that the code has not been adapted to take advantage of the vectorial capabilities of the computer, which could substantially reduce the computational cost.

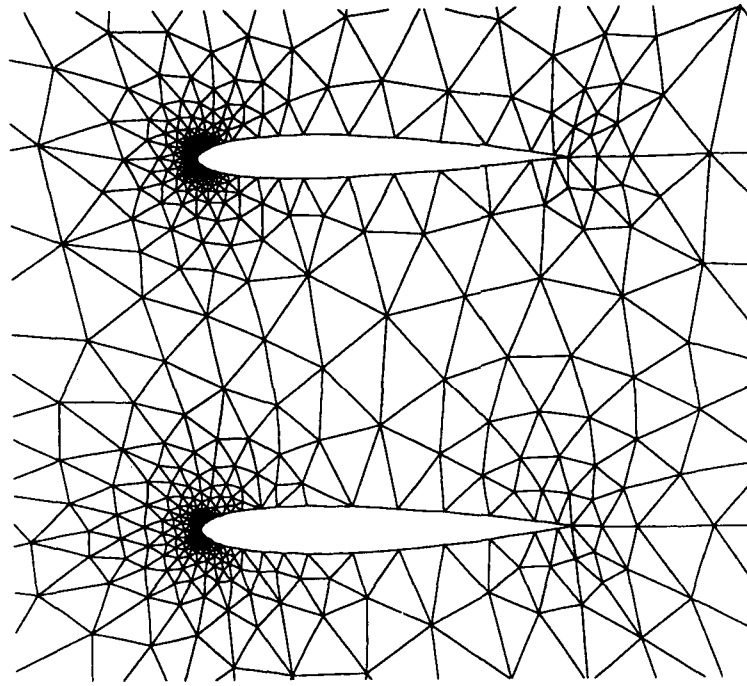

Figure 13. Final shape and mesh for the two-element problem.

The evolution of the normalized cost functional can be seen in Figure 14. The meshes used for the computations have around 2700 nodes and 1300 quadratic elements and the percentage of error is always below the imposed limitation of 0.2%. Figures 15 and 16 show superpositions of the C_p distributions for the target profile and the initial and final designs for each aerofoil.

Figure 14 shows good convergence of the minimization process. The cost functional has been diminished by more than three orders of magnitude in 100 iterations.

The comparisons between the target and computed C_p distributions shown in Figures 15 and 16 are quite good, especially taking into account that the C_p distribution corresponding to the initial design is very far away from the final one.

CONCLUSIONS

A new methodology for the resolution of aerodynamic shape optimization and inverse problems has been developed and assessed. This methodology is able to optimize the design and the analysis mesh in a joint manner in order to produce a final design computed with a proper mesh.

Good quality results are obtained using a single mesh for each design without any remeshing. This considerably reduces the additional cost of the mesh control.

The presented methodology has provided excellent results for all the cases analysed, leading to an accurate final solution with a good final mesh.

The use of a single and different 'optimal' meshes for each optimum design step seems to be especially interesting for application of this methodology to more realistic flow models where the control of the mesh quality is crucial. This could be particularly attractive in the presence of shocks or separation points. Further assessment of this methodology in the context of both Euler and Navier–Stokes flows will be reported in the near future.

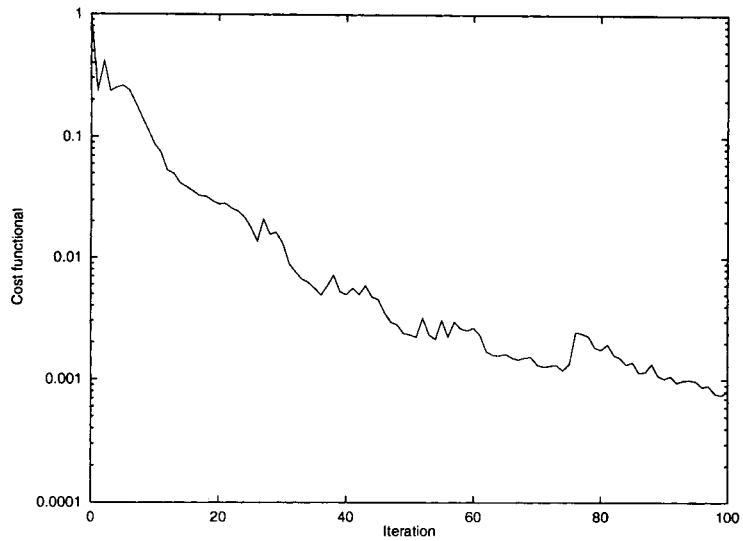

Figure 14. Evolution of the normalized cost functional for the two-element problem.

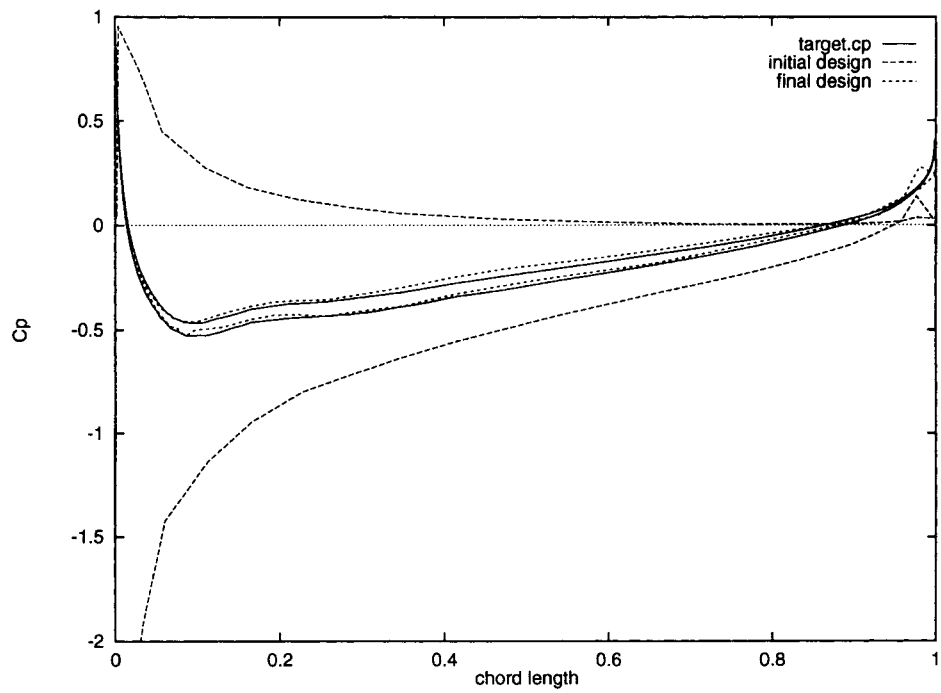

Figure 15. Superposition of the C_p distributions corresponding to the upper element of the target profile, the initial design and the final design.

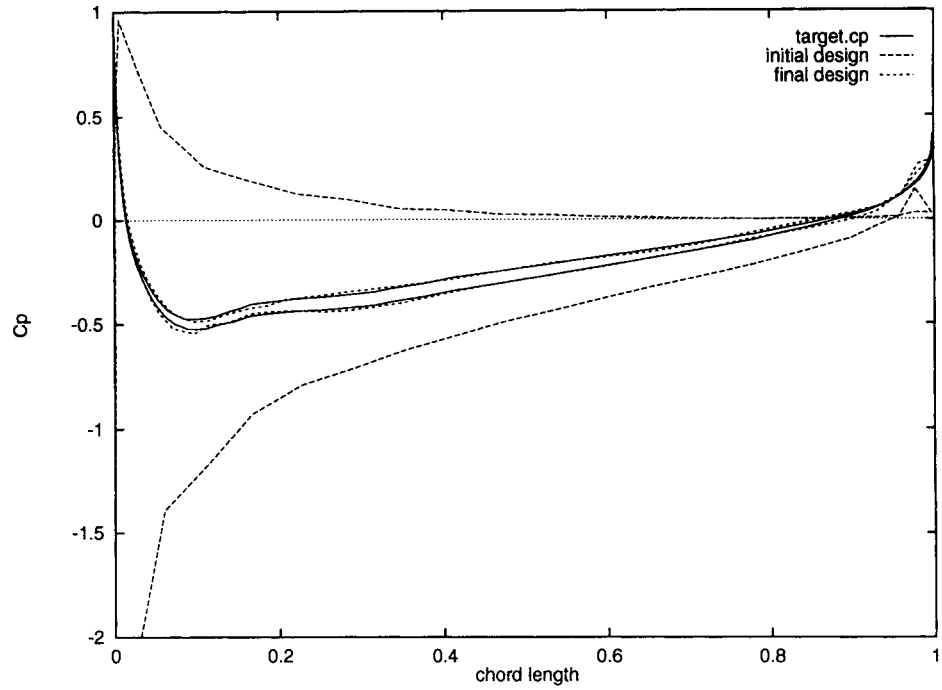

Figure 16. Superposition of the C_p distributions corresponding to the lower element of the target profile, the initial design and the final design.

ACKNOWLEDGEMENTS

This work was partially funded by the EUROPT I and ECARP projects in the Brite–Euram Aeronautics programme of the EEC.

REFERENCES

1. F. Beux, A. Dervieux, M. P. Leclercq and B. Stoufflet. 'Techniques de contrôle optimal pour l'optimisation de forma en aérodynamique avec calcul exact du gradient', *Revue Scientifique et Technique de la Défense*, 1993.
2. E. Hinton and J. Sienz. *Adaptive finite element Analysis and Shape Optimization*, Saxe-Coburn, Edinburgh, 1995.
3. G. Bugeda and D. Joannas. 'Aerodynamic shape optimization using automatic adaptive remeshing', in J. Periaux *et al.* (eds), *Notes on Numerical Fluid Mechanics, Proc. Optimum Design Final BRITE/EURAM Workshop*, Vieweg, Braunschweig, 1993.
4. I. D. Faux and M. J. Pratt. *Computational Geometry for Design and Manufacture*, Ellis Horwood, Chichester, 1987.
5. J. Peraire, M. Vahdati, K. Morgan and O. C. Zienkiewicz. 'Adaptive remeshing for compressible flow computations', *J. Comput. Phys.*, **72**, 449–466 (1988).
6. O. Pironneau. *Méthodes des Eléments Finis pour les Fluides*, Masson, Paris, 1988.
7. O. C. Zienkiewics and R. L. Taylor. *The Finite Element Method*, Vol. 2, 4th edn, McGraw-Hill, New York, 1991.
8. G. Bugeda and E. Oñate. 'New adaptive techniques for structural problems' *Proc. First Eur. Conf. on Numerical Methods in Engineering*, Brussels, September 1992.
9. G. Bugeda and E. Oñate. 'Adaptive mesh refinement techniques for aerodynamic problems, in H. Alder, J. C. Heinrich, S. Lavanchy, E. Oñate and B. Suárex (eds), *Numerical Methods in Engineering and Applied Sciences*, CIMNE, Barcelona, 1992.