

Adaptive Method Using Controlled Grid Deformation

Florin FRUNZULICA*¹

*Corresponding author

*“POLITEHNICA” University of Bucharest, Faculty of Aerospace Engineering,
Gh. Polizu Street 1-5, Bucharest, Sect.1, Romania

¹Institute of Mathematical Statistics and Applied Mathematics
13 September Street, Sect. 5, Bucharest, Romania
ffrunzi@yahoo.com

DOI: 10.13111/2066-8201.2011.3.3.6

Abstract. *The paper presents an adaptive method using the controlled grid deformation over an elastic, isotropic and continuous domain. The adaptive process is controlled with the principal strains and principal strain directions and uses the finite elements method. Numerical results are presented for several test cases.*

Key Words: *grid adaptation, principal strains, finite elements method.*

1. INTRODUCTION

In the last decade, remarkable progresses have been made for developing new methods to obtain adaptive mesh. The latter issued from the necessity to increase the spatial accuracy required to solve problems of physics, described by differential equations.

The purpose of an adaptive mesh is to provide a local increase of the number of nodes where the error distribution has large values and to thin the mesh where the error is minimal. Generally speaking, the error parameters are built on the basis of the gradient of a previous considered variable s , which supposes that one knows the values of variable on the initial mesh. The main ways to follow in order to obtain adaptive meshes are: 1) mesh regenerating, 2) moving the inner nodes, 3) adding or extracting a number of nodes, 4) a composed method and taking into account the previous 3 methods already presented. The author suggest the reader to consider references [1], [2] for the case 1), [5], [8], [9] for the case 2), [3], [4], [7] for the case 3), [6] for the case 4).

The present method belongs to the 2) category, because the nodes are moved towards the areas with large values of gradient of variable s . The deformation of the mesh is performed over an elastic, isotropic and continuous domain. The adaptive process is controlled with the principal strains and principal strain directions of the mesh elements.

2. DEFORMATION FUNCTION OF THE ELEMENT

The adaptive process consists in a controlled deformation of the mesh, which creates either increasing or decreasing the density of the nodes, as a function to a parameter which indicates the local error θ . For this indicator we propose the next formula:

$$\theta = \frac{|\nabla s| - |s|_{\min}}{|\nabla s|_{\max} - |s|_{\min}} \quad \nabla \quad (1)$$

where the physical variable s is chosen according to the type of the problem to be studied.

As one can notice, the values of θ range between 0 and 1.

Increasing or decreasing the density of the nodes supposes to modify the dimension of the element; by dimension of the element we mean the length in one-dimension, the area in two dimensions, and the volume in three dimensions.

Let us consider ϕ the function that characterises the dimensional modifications of the element: $V_N = \phi \cdot V_O$. The values of the function are: $\phi > 1$ when $\theta \rightarrow 0$, which means the growth of the element's "dimension", thus the mesh becoming locally less dense, and $\phi < 1$ when $\theta \rightarrow 1$, which means the decay of the element's "dimension", thus the mesh becoming locally more dense.

There are 2 forms proposed for the function ϕ :

$$\phi = \phi_0 (1 - \theta) + \phi_1 \theta \tag{2.a}$$

$$\phi = \phi_0^{1-\theta^2} \phi_1^{\theta^2} \tag{2.b}$$

The ratio $\phi_1/\phi_0 = \rho$ represents the ratio between the reduced size and the enlarged size of the adaptive mesh ($0 < \rho < 1$). When $\rho \rightarrow 0$, one will obtain a highly dense mesh in the area with large gradients. Being given the controlling parameter ρ , then the parameter ϕ_0 may be determined from the condition to preserve the dimensions: the length in 1D, the area in 2D, and the volume in 3D:

$$\int_V dV = \int_V \phi dV \tag{3}$$

If we suppose that the principal strains are estimated in the centre of each element (this is a simplifying hypothesis), then the function ϕ can be expressed by the means of the principal strains, as:

$$\phi = (1 + \varepsilon_1)(1 + \varepsilon_2)(1 + \varepsilon_3) = f_1 \cdot f_2 \cdot f_3 \tag{4}$$

Now we'll face the problem of choosing the functions f_1, f_2, f_3 which must verify the equation (4). For an element shaped as a parallelepiped it is convenient to consider the isotropic distribution of the function ϕ over the directions of the principal strains:

$$f_1 = f_2 = f_3 = \phi^{1/3} \tag{5}$$

If there is a preferred direction of the element's strain, then it leads to selection of the functions f_1, f_2, f_3 as:

$$f_1 = \phi^{k_1}, \quad f_2 = \phi^{k_2}, \quad f_3 = \phi^{k_3} \quad \text{where } k_1 + k_2 + k_3 = 1 \tag{6}$$

The direction given by the gradient of s , i.e. $\vec{g} = \nabla s / |\nabla s|$, may be considered as a preferred direction of the element's strain. We have to take into account that during the adaptive process, the direction \vec{g} given by the gradient is different from the maximum strain direction; therefore, we must alter the controlling function of the strains (2), such that:

$$f_1 = \phi_0^{k_1(1-\theta^2 n_1^2)} \phi_1^{k_1 \theta^2 n_1^2}, \quad f_2 = \phi_0^{k_2(1-\theta^2 n_2^2)} \phi_1^{k_2 \theta^2 n_2^2}, \quad f_3 = \phi_0^{k_3(1-\theta^2 n_3^2)} \phi_1^{k_3 \theta^2 n_3^2} \tag{7}$$

where (n_1, n_2, n_3) are the components of the direction \vec{g} , referred to a coordinate system attached to the principal strain directions.

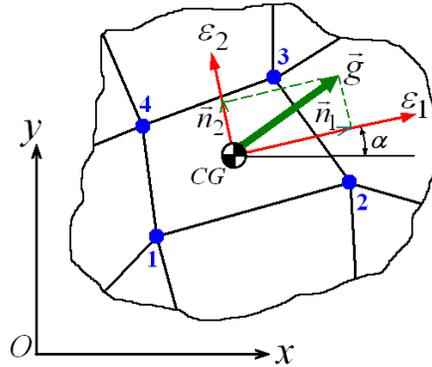


Fig. 1. The principal strains of the element

Relations (7) are used to compute the principal strains $\varepsilon_1, \varepsilon_2, \varepsilon_3$ and allow afterwards to obtain the linear strains $(\varepsilon_x, \varepsilon_y, \varepsilon_z)$ and the angular strains $(\gamma_{xy}, \gamma_{yz}, \gamma_{zx})$, expressed in the global reference system of the mesh (figure 1).

3. DETERMINING THE NODAL DISPLACEMENTS OF THE MESH

In order to determine the nodal displacements (u, v, w) of the mesh, being known the strains on the elements, one may suppose that the computational domain is an isotropic, homogeneous and elastic material. The mesh models are: 1D- a set of beams, submitted to extension/compression stress, or springs; 2D- a flat plate, submitted to flat strains and flat stress; 3D- a deformable solid.

The idea is to determine the nodal forces which induce the supposed set of strains. When computing the displacements with the aid of the finite element method, for the 2D and 3D model, the notion of “a concentrated force acting in the nodes of the elements” is some how exaggerated, in order to allow the obtaining of a simple model for an adaptive mesh.

We'll suppose that both the coordinates (x,y,z) and the displacement function (u,v,w) on each element, may be represented by a set of interpolating function (of Pascal type):

$$\begin{aligned}
 x &= \sum_{i=1}^p N_i(\xi, \eta, \zeta) x_i, \quad y = \sum_{i=1}^p N_i(\xi, \eta, \zeta) y_i, \quad z = \sum_{i=1}^p N_i(\xi, \eta, \zeta) z_i \\
 u &= \sum_{i=1}^p N_i(\xi, \eta, \zeta) u_i, \quad v = \sum_{i=1}^p N_i(\xi, \eta, \zeta) v_i, \quad w = \sum_{i=1}^p N_i(\xi, \eta, \zeta) w_i
 \end{aligned}
 \tag{8}$$

where (ξ, η, ζ) represents the natural coordinates attached to the element.

Between the strains and the vector of the nodal displacements on each element, stands the relation:

$$\{\varepsilon\} = [B] \{d\}
 \tag{9}$$

where $\{d\}$ is the vector of the nodal displacements on each element and $[B]$ is the matrix of the derivatives of the interpolating functions N_i .

One may prove that for an element, the stiffness matrix can be expressed as:

$$[K_e] = \int_{V_e} [B]^T [E] [B] dV
 \tag{10}$$

based on the principle of the stationary value of the potential energy.

Matrix $[E]$ is the elasticity matrix. Being known the field of the strains, for an element one may write the following relation:

$$[K_e]\{d_e\} = \int_{V_e} [B]^T [E] [B] dV \{d_e\} = \int_{V_e} [B]^T [E] [B] \{d_e\} dV = \int_{V_e} [B]^T [E] \{\varepsilon\} dV \quad (11)$$

The concentrated forces applied in the node of order “ i ”, which induce the local deformation on the mesh, may be estimated after the addition of the contributions of all the elements adjacent to node “ i ”:

$$\{F_i\} = \sum_{e=1}^{e_i} \int_{V_e} [B]^T [E] \{\varepsilon\} dV \quad (12)$$

Next step will be to establish the nodal displacements of the mesh, with the aid of the finite element method (FEM). Therefore, for each element, one may write:

$$[k_e] \{d_e\} = \{f_e\} \quad (13)$$

where $[k_e]$ is the stiffness matrix of the considered element and $\{f_e\}$ is the vector of the nodal forces. After performing the assembling process, one obtains the global stiffness matrix $[K]$ and $\{F\}$ vector of forces, represented just by the nodal concentrated forces (fig. 2).

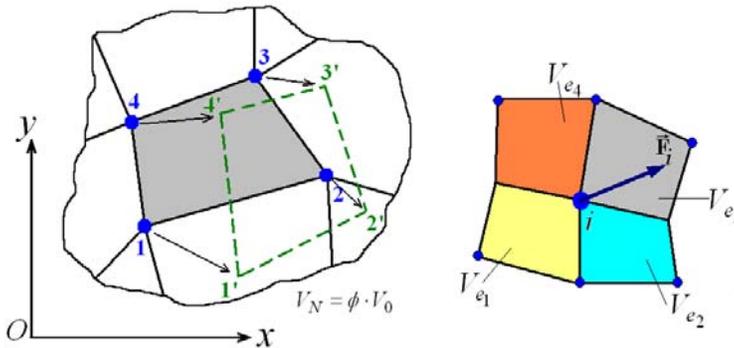


Fig. 2. Node displacements under force \vec{F}_i

There are necessary boundary conditions. Thus, the nodes on the boundary can either “frozen”, or moved according to an algebraic law or are allowed to a “slip” along the boundary S_w (“... $i-1, i, i+1$...” nodes – fig. 3).

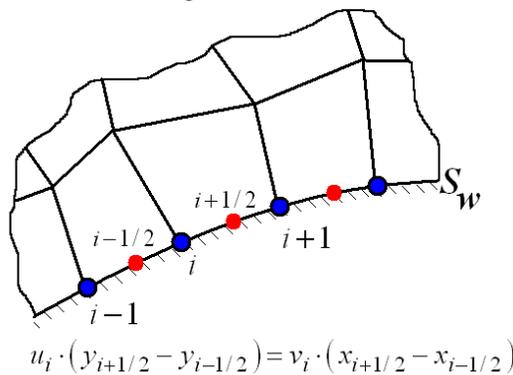


Fig. 3. Boundary condition

The linear system:

$$[K] \{d\} = \{F\} \quad (14)$$

has a band-matrix $[K]$, and it can be solved with the aid of either direct eliminating Gauss method or by an iterative process (Gauss-Seidel or conjugated gradient).

The solution of the system (14) offers the values for the nodal displacements; next, the position of the nodes in the mesh is updated:

$$x_i^N \leftarrow x_i^O + u_i, \quad y_i^N \leftarrow y_i^O + v_i \quad (15)$$

4. ADAPTIVE ALGORITHMS

Input data: original mesh, the values of s estimated in nodes, the characteristics of the elastic medium (E and ν). Steps (fig.4): 1) estimate the value of ∇s and select the principal strain directions; 2) consider the values of the principal strains given by the functions (7) according to the values of the error parameter θ ; 3) calculate the linear and angular strains; 4) calculate the nodal forces $\{F\}$, by means of (12); 5) determine the nodal displacements by using FEM solver; 6) establish the position of the mass-centres for the elements of the new mesh; 7) update the values of s in nodes and ∇s .

The iterative process continues with step 1) until the new position of the nodes will be obtained with the imposed precision.

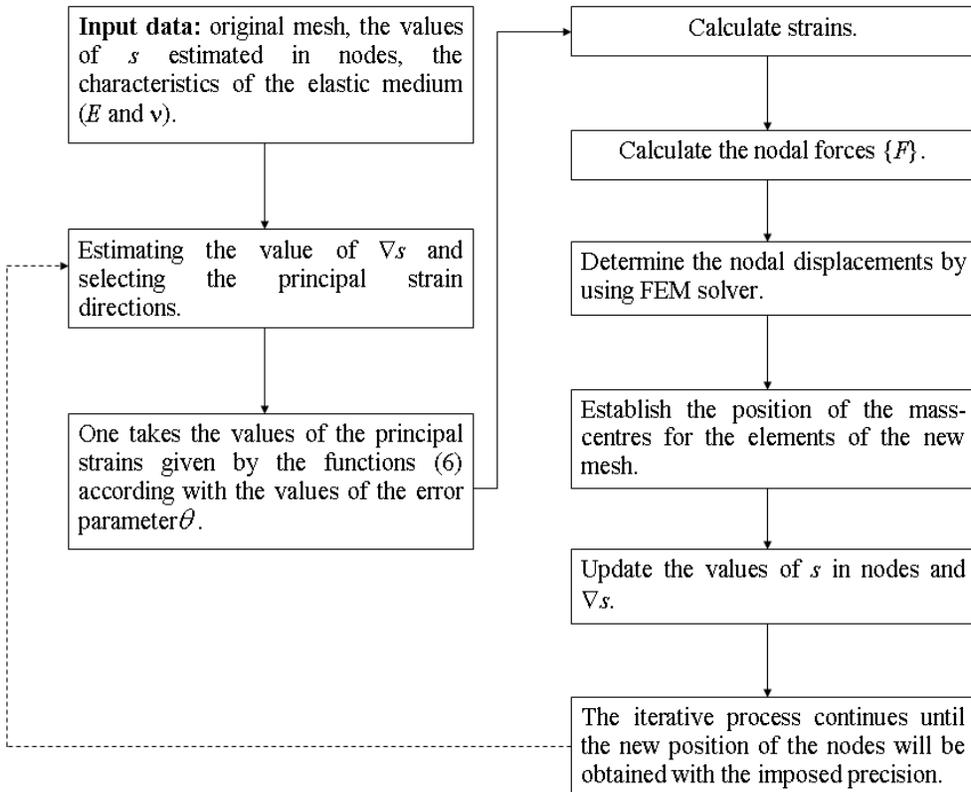


Fig. 4. Adaptive algorithm

5. RESULTS

The purpose of the first case study is to verify the form of the two functions proposed (2.a and 2.b). The variation of s is described by the function:

$$f(x) = \begin{cases} 1 & \text{for } x < -\varepsilon \\ -\sin\left(x \cdot \pi / (2 \cdot \varepsilon)\right) & \text{for } -\varepsilon \leq x \leq \varepsilon \\ -1 & \text{for } x > \varepsilon \end{cases} \quad (16)$$

where $\varepsilon = 0.4$.

Figure 5 shows the variations of the function $f(x)$ and the parameter θ ; also, the adaptive mesh is presented for different values of the parameter ρ . The mesh has 41 equally spaced nodes and the adaptive process is performed in 5 iterations. The variation of the 2 functions looks alike, as regards the increase of the mesh density when $\theta = 1$.

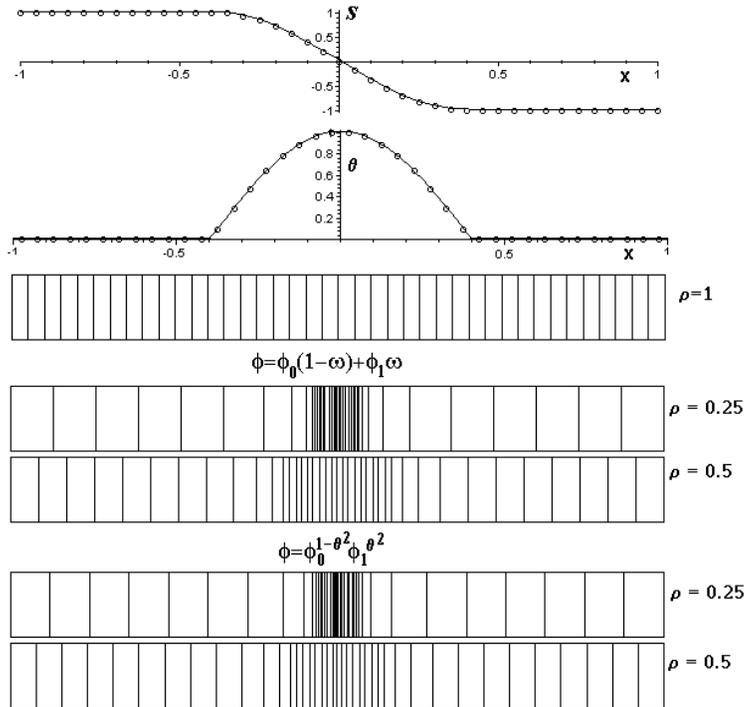


Fig. 5. Adaptive mesh in 1D

The second case study deals with an adaptive mesh built for a square, having 35 x 35 equally spaced nodes. The variation of s is described by:

$$f(r) = \begin{cases} 0 & \text{for } |r| < \varepsilon \\ C e^{-\frac{\varepsilon^2}{\varepsilon^2 - r^2}} & \text{for } |\varepsilon| \leq r \end{cases} \quad (17)$$

where $\varepsilon = 0.85$, $C \int_{-\varepsilon}^{\varepsilon} e^{-\frac{\varepsilon^2}{\varepsilon^2 - \eta^2}} d\eta = 1$ and $r = \sqrt{x^2 + y^2}$.

Boundary conditions: “freezing” the 4 corners; considering the slip condition over all the edges.

The adaptive mesh obtained in 3 iterations is shown in Figure 6. One may notice an increase of the nodes number in the areas where ∇s has large values.

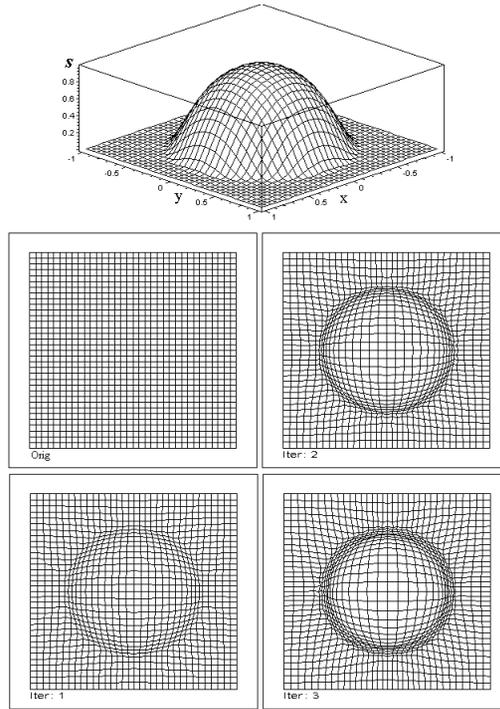


Fig. 6. Adaptive mesh built for a square ($\rho=0.4$)

Steady supersonic flow in 2D channel

The third test was conducted on a 2D channel (fig. 7), in which inviscid flow is supersonic (Mach = 1.6). The two-dimensional flow is described using the unsteady Euler equations:

$$\frac{\partial U}{\partial t} + \nabla \vec{F} = 0 \quad (18)$$

where

$$\begin{aligned} U &= \{\rho \quad \rho u \quad \rho v \quad \rho E\}^T \\ \vec{F} &= f \cdot \vec{i} + g \cdot \vec{j} \\ f &= \{\rho u \quad \rho u^2 + p \quad \rho uv \quad (\rho E + p)u\}^T \\ g &= \{\rho v \quad \rho uv \quad \rho v^2 + p \quad (\rho E + p)v\}^T \end{aligned} \quad (19)$$

and

$$E = \frac{1}{2}(u^2 + v^2) + \frac{1}{\gamma - 1} \frac{p}{\rho} \quad (20)$$

for a perfect gas.

For the conservative variables (U), the integral formulation of Eq.18 is given by

$$\frac{\partial}{\partial t} \int_{\sigma_e} U d\sigma + \int_{\partial\sigma_e} (f(U)n_x + g(U)n_y) ds = 0 \quad (21)$$

where $\vec{n} = n_x \vec{i} + n_y \vec{j}$ is the outward normal of the element boundary $\partial\sigma_e$.

In this form, is attractive to use a Finite Volume Method (FVM), especially for the case of the convective dominant flows.

The main feature of the FVM is the integration of the flux on the surface of the control volume σ_e .

For the convective terms, the problem becomes 1-D in the direction of the normal to the surface. The one-dimensional Euler system may be easily diagonalized and then the surface flux can be evaluated in an upwind purpose.

At the wall, for inviscid flows, the boundary condition is:

$$\vec{V} \cdot \vec{n} = 0 \tag{22}$$

and as a consequence the convective flux vanishes.

This condition is implemented using a fictive cell on the solid frontier where the velocity has the opposite sign with respect to the velocity in the neighbouring cell, while pressure, density and energy remain constant. A numerical correction for the interior component of velocity is required:

$$\vec{V} = \vec{V}_c - (\vec{V}_c \vec{n}) \vec{n} \tag{23}$$

where \vec{V}_c is the calculate velocity.

To obtain the second order spatial accuracy, we have adopted a MUSCL formulation (Monotone Upstream-centred Schemes for Conservation Laws) for nonuniform structured grid. The limiter used in MUSCL reconstruction (at the interface level) was proposed by van Albada [14], because it has simple mathematical form and smooth properties.

Choosing $r=0.3$ and maximum allowed reduction order 0.35 between two consecutive iterations, the adapted mesh looks similar to the one in figure 7 (a-b). There is also a comparison between the isobar curves ($p / (\rho_\infty V_\infty^2)$) in the two situations (fig. 7 c-d,e-f).

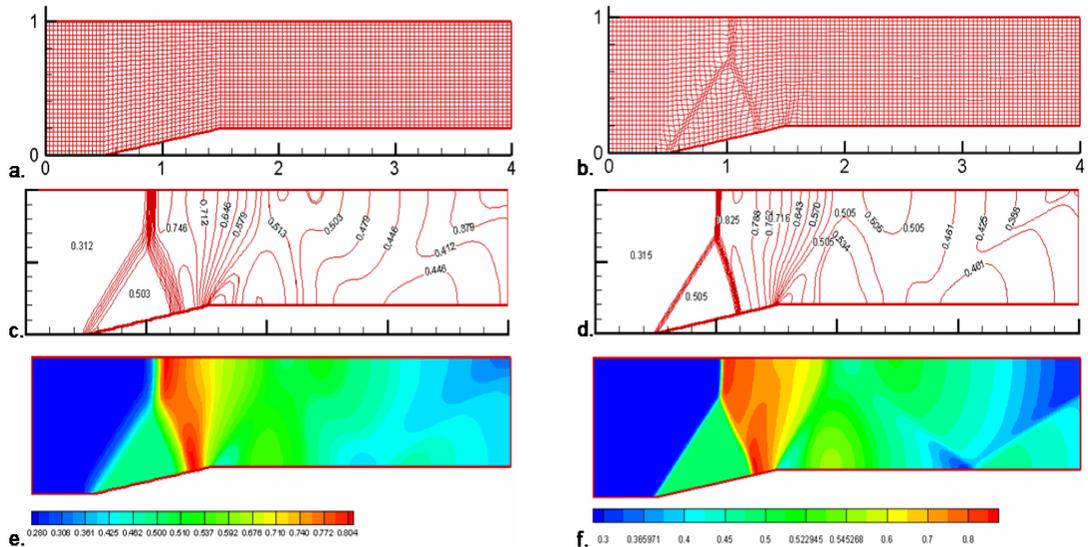


Fig. 7: Mesh before (a) and after (b) the adaptive process; isobar contours - (c-e) original mesh; (d-f) adapted mesh

6. CONCLUSIONS

The tests, which have been done over 1D and 2D meshes, show the capability of the algorithm to increase the number of nodes whenever high values of ∇s are obtained. Also, the proposed method shows a good accuracy for the iterative process and being fast convergent has convenient computing time.

The adaptive algorithm is robust and relatively quick. Basically, for a 2D mesh, the algorithm converges after a few iterations. Keeping the number of nodes and mesh connectivity, the necessary hardware resources for the adaptive process are relatively small. The Euler solver converges fairly quickly on the adapted mesh, having as initial data the solution obtained on the original mesh, and then extrapolated on the adapted mesh. The adaptive procedure is also suitable for multi-zoned meshes, the adaptive process being done in this case on sub-domains.

REFERENCES

- [1] D. A. Anderson, J. Steinbrenner, Generating Adaptive Grids with a Conventional Grid Scheme, *AIAA Paper 86-0427*, New York 1986.
- [2] D. A. Anderson, Equidistribution schemes, Poisson generations and adaptive grids, *Appl. Math. Comput.* **24**, pp. 211-277, 1987.
- [3] M. Berger, J. Olinger, Adaptive mesh refinement for hyperbolic partial differential equations, *J Comput Phys* **53**, pp. 484-512, 1984.
- [4] R. Biswas, R. C. Strawn, Tetrahedral and hexahedral mesh adaptation for CFD problems. *Appl. Num. Math.* **26**, no. 1, pp. 135-151, 1998.
- [5] R. Carcaillet, Optimization of three-dimensional computational grids and generation of flow adaptive computational grids, *AIAA Paper 86-0156*, Reno USA, 1986.
- [6] Wu SJ Hwang, Global and Local Remeshing Algorithms for Compressible Flows, *Journal of Computational Physics* **102**, pp. 98-113, 1992.
- [7] R. J. LeVeque, Wave propagation algorithmus for multi-dimensional hyperbolic systems, *J Comput Phys* **131**, pp. 327-353, 1997.
- [8] G. Liao, D. Anderson, A new approach to grid generation, *Appl Anal* **44**, pp. 285-298, 1992.
- [9] G. Liao, T-W. Pan, J. Su, A numerical grid generator based on Moser's deformation method, *Numer. Methods Partial Differential Equations* **10**, pp 21-31, 1994.