Implementation of an adjoint-based error estimation and grid adaptation module in the DLR TAU code

Jorge Ponsin
Instituto Nacional de Técnica Aeroespacial
13 PUBLICATIONS **64** CITATIONS
SEE PROFILE

Esther Andres
Ingeniería de Sistemas para la Defensa de España
49 PUBLICATIONS **142** CITATIONS
SEE PROFILE

Carlos Lozano
Instituto Nacional de Técnica Aeroespacial
53 PUBLICATIONS **438** CITATIONS
SEE PROFILE

IMPLEMENTATION OF AN ADJOINT-BASED ERROR ESTIMATION AND GRID ADAPTATION MODULE IN THE DLR TAU CODE

J. Ponsin*, A. Caloto*, E. Andrés**, P. Bitrián***, C. Lozano***
***National Institute of Aerospace Technology (INTA), Spain – Fluid Dynamics Branch,**
****Ingenieria y Servicios Aeroespaciales S.A. (INSA),**
***** Technical consultant**

Keywords: *Adjoint, grid adaptation, error estimation*

Abstract

An adjoint-based error prediction and grid adaptation methodology has been implemented in a CFD-software (DLR-TAU). The potential and drawbacks of this method are examined with an eye to its industrial application. Two strategies for an adjoint-based adaptive process are compared in terms of the computational efficiency. The influence of the surface mesh reconstruction method, which is a relevant issue for industrial applications, is also examined. The method is applied to several 2D and 3D configurations, including a more realistic complex 3D industrial geometry. The effectiveness of the method is demonstrated for inviscid flows.

1 Introduction

The necessity to accurately predict the global aerodynamic coefficients using CFD is a crucial question for the aeronautical industry nowadays. The increasing use of CFD in the aeronautical industry allows to reduce the design cycle and thus the time to market, to optimize and improve the quality of the product in terms of energy efficiency and contamination reduction and finally, to save a lot of money. Absolute drag coefficient prediction within ten drag counts ($\pm 5.10^{-4}$) was defined as representative of the desired accuracy level by the EU aerospace industry within the EU project ADIGMA [1, 17] for some simple geometry test cases. Such drag accuracy may translate to variations of ±1.5% in the total predicted drag for a typical long range aircraft configuration. It has been shown [2] by using the simple Breguet-range equation, that a 1% under-prediction of the drag produces a 1% shortfall in range which may be recovered, assuming the same initial fuel weight by an 8% of payload weight reduction. This translates, for such scenario, to a reduction of over 30 passengers for a large transport aircraft. These figures denote the importance of the drag prediction accuracy, especially when using CFD tools for design and optimization.

In the current aeronautical industrial environment, the standard CFD solvers are based on second order accurate finite-volume structured or unstructured grid schemes. For such solvers, excluding physical modeling error sources, the mesh resolution remains the key aspect to obtain accurate and reliable predictions of engineering outputs such as drag or lift. The popular CFD practitioner credo “grids are everything” still remains valid nowadays. Generating a suitable (appropriate) grid for the accurate computation of a given aircraft is not a trivial task, even for a simple wing-body configuration [3]. The grid generation activity demands a lot of user expertise. Thus, the development of methods to automatically generate optimal grids for the accurate estimation of any engineering output is still a necessity and a challenge nowadays.

The adjoint error estimation and mesh adaptation methodology [4] has gained increasing popularity in recent years as a promising post-processing tool to improve the accuracy of CFD predictions. While this methodology has been successfully implemented and applied to test cases involving simple geometries by several groups [5, 6, 7, 8], few results have been shown for complex test cases of industrial interest [9, 10]. There are still several critical drawbacks, which are inherent in the methodology itself, that make the method difficult to apply at production level within industrial environments. Most adjoint adaptation approaches [5-10] require the construction of a uniformly refined embedded grid. This requirement produces appreciable memory overheads for large initial grids. Moreover, the construction of such embedded meshes is challenging for the viscous areas in Navier-Stokes meshes. In addition to this, robustness and computational efficiency of the adjoint solver is still an issue to be addressed for the systematic application of the adjoint technology in the industry environment.

The present paper describes the implementation of an adjoint-based adaptation method into a CFD code which is being used on a daily basis in the European industry. Some issues regarding the adaptation strategy, surface reconstruction, mesh quality, etc., are examined with the aim of highlighting the potentials and drawbacks of the method with an eye to its application in an industrial environment.

This paper is organized as follows: Section 2 describes briefly the adjoint-based adaptation methodology implemented into DLR-TAU. Section 3 is focused on implementation issues. Section 4 discusses several applications of the method, as well as various challenging issues such as adaptation strategies, surface reconstruction, mesh quality, etc. which stand out as the main drawbacks for industrial application.

2 Adjoint-based adaptation method

An adjoint-based error correction method due to Dwight [1] is already available within the DLR TAU software suite. In this method, the error indicator is derived in a somewhat heuristic way using the fact that the dissipation term introduced explicitly for stabilizing the solution process is the major source of error in the output functional prediction. Therefore, the proposed error indicator only accounts for errors originated from dissipation terms (e.g. JST-central dissipation scheme). On the positive side, the method does not rely on an auxiliary refined grid (embedded grid), which saves memory for large initial mesh cases and avoids the burden of generating uniformly refined meshes for the anisotropic hybrid meshes typically used in Navier-Stokes calculations. Despite of this attractive advantage, Dwight's method is not general in the sense that the discretization errors may be originated from sources other than the artificial dissipation. Therefore, in the current work we have implemented into DLR-TAU an alternative adjoint-based method which follows the original formulation proposed by Giles [13] and adapted by Venditti and Darmofal [5,11] to the finite volume method framework. In the next section we briefly review the mathematical formulation of the method.

2.1 Adjoint-based error correction formulation

The implemented adjoint-based error estimation follows the work of Venditti [11]. Let's consider an output functional $J(U)$ of physical interest derived from the solution U of the system of partial differential equations (PDE's) $R(U)$ under consideration. In the present case, the functional output is usually an integral quantity (e.g. lift, drag, etc.), U is the flow solution and the PDE's are the flow equations (e.g. Navier-Stokes). Due to the complexity of the flow equations, the solution is obtained by numerical methods. In the present case, an unstructured finite volume method [12] is used to solve the flow on a mesh Ω_H of characteristic size H. Let's denote the functional output of the numerical solution U_H on this mesh by $J_H(U_H)$. Next, pick a fine mesh Ω_h obtained e.g. by uniform h-refinement of the coarse grid Ω_H. The error correction method seeks to approximate the output functional on the mesh Ω_h, i.e. to obtain $J_h(U_h)$, without solving the flow equations on that mesh, $R_h(U_h)=0$. By combining Taylor expansions of the flow equations and the output functional around an approximated flow solution $\tilde{U}_h$ defined on mesh Ω_h it is possible to obtain a (first order) approximation to the error in the output functional

$$J_h(U_h) \approx J(\tilde{U}_h) - (\psi_h)^T R(\tilde{U}_h) \qquad (1)$$

where ψ_h is the solution of the discrete adjoint solution system given by

$$\left[\frac{\partial R_h}{\partial \tilde{U}_h}\right]^T \psi_h = \left[\frac{\partial J_h}{\partial \tilde{U}_h}\right]^T \qquad (2)$$

Obviously, obtaining an approximation of the functional by using expression (1) is still too expensive computationally because one has to solve the discrete adjoint system (2) on the fine embedded mesh. Instead, suppose that we have an approximation $\tilde{\psi}_h$ to the adjoint solution on the fine mesh. The expression (1) may be then re-written as

$$J_h(U_h) \approx J_h(\tilde{U}_h) - (\tilde{\psi}_h)^T R_h(\tilde{U}_h) - (\delta\psi_h)^T R_h(\tilde{U}_h) \, (3)$$

where $\delta\psi_h = \psi_h - \tilde{\psi}_h$. The first term on the right hand of Eq. (3) can be computed because $\tilde{\psi}_h$ and $\tilde{U}_h$ are assumed to be known. This term is known in the literature, e.g. [1], as the "Computable Correction" (CC). The second term represents the error introduced in the correction when an approximation of ψ_h is used to compute the functional output correction. This term is known as the "Remaining Error" (RE). Note that this error component is not known and should be approximated, usually, by using higher order interpolation approximations to ψ_h.

2.2 Reconstruction of primal and dual solutions

In the previous section an expression for the functional error correction has been derived. This expression depends on approximations of the primal and dual solutions on the fine grid Ω_h. These approximations are obtained by projection of the coarse grid solutions onto the fine grid as follows. First, consider that the embedded fine grid Ω_h is generated by uniform h-adaptation of the coarse one, i.e. by inserting a new node at the midpoint of each existing edge of the grid Ω_H. The next step is to find approximations to $\tilde{U}_h$ and $\tilde{\psi}_h$ on that embedded mesh. Linear and higher order interpolant operators, L_h^H and Q_h^H, respectively, are used for projecting the coarse primal and adjoint solutions, U_H and ψ_H, respectively, onto the embedded fine grid. In the present implementation, the linear operator is a simple linear interpolation at the edge midpoint using the two nodal edge values. For the higher order operator, a cubic Hermitian polynomial is used along the edge to approximate the value of the midpoint from the edge nodal values [6]. This interpolation is fourth order accurate provided that the derivates are exact at the nodes. The estimation of the gradients of the variables at the nodes of the grid Ω_H is performed using a weighted least square method which is second order accurate and exact for linear functions. Therefore, the higher order interpolation reduces to third order accuracy overall. This compact edge higher order interpolation has been chosen because it is quite robust, it does not produce computational overheads and it is especially well suited for using together with an edge-based structure in an unstructured solver. Substituting the reconstructed solution into (3) gives the following expression for the estimated functional on the embedded mesh

$$J_h(U_h) = J_h(L_h^H U_H) - (L_h^H \psi_H)^T R_h(L_h^H U_H) \; (4)$$

Note that the linear operator has been finally chosen for both, the approximation of the primal and the adjoint solutions. The main reason for this choice is to maintain robustness, especially in regions where the flow and adjoint solutions may exhibit steep gradients, discontinuities and oscillations.

The grid adaptation strategies implemented in the new TAU module will be described in the next section. One of the strategies is to define a local adaptation parameter based on the local remaining error which is given by

$$RE = (\delta\psi_h)^T R(\tilde{U}_h) \qquad (5)$$

In the present implementation the remaining error has been approximated using the higher order projection operator Q_h^H described above yielding the following expression for the estimated remaining error

$$RE = ((Q_h^H \psi_H)^T - (L_h^H \psi_H)^T) R(L_h^H U_H) \qquad (6)$$

2.3 Grid adaptation strategies

The adjoint-based error analysis can also serve to drive an adaptive refinement process, in which the grids are especially tuned to compute the chosen functional. Two main adjoint-based adaptation strategies may be devised. The first one uses the computable correction (functional output correction) as a driver in the adaptive refinement [7,13]. The goal is to reduce the magnitude of the error correction term in each adaptation cycle. We call this strategy the *CC-strategy*. The second strategy, called *RE-strategy*, is based on improving the accuracy of the corrected output in (3) by reducing the remaining error after correction [5-10]. Both strategies have been implemented in the adjoint-based correction and adaptation module. A comparison between these two strategies will be presented in section 4.

Let us consider a local adaptation parameter ε_i for a node *i* of the grid Ω_h, which is strategy dependent and is defined for the *CC-strategy* as

$$\varepsilon_{h,i} = \left| \tilde{\psi}_{h,i}^T R(\tilde{U}_{h,i}) \right| \qquad (7)$$

and for the *RE-strategy* by

$$\varepsilon_{h,i} = \left| \delta\tilde{\psi}_{h,i}^T R(\tilde{U}_{h,i}) \right| \qquad (8)$$

The corresponding projection operators are applied to obtain the approximated values of $\tilde{\psi}_h$ and $\tilde{U}_h$. The adaptation parameter defined by (8) is a simpler version of the estimate suggested by Venditti et al. [2], who include an additional term based on the residual of the adjoint equation. The sum of the adaptation parameters over the nodes of the grid gives a conservative estimation of the computable correction if definition (7) is used, or an upper bound for the remaining error after correction for (8). The used adaptation strategy aims at reducing and equi-distributing the value of the defined adaptation parameter throughout the computational domain.

Assume that the local error in the fine grid is transferred back to the coarse grid with a certain prolongation operator (to be described in the next section)

$$\varepsilon_{H,i} = \tilde{P}_H^h \varepsilon_{h,i} \qquad (9)$$

With the aid of this operator, the adaptation parameter, which mimics the local error, or the incertitude of the error, respectively, has been returned to the original mesh. We define the total error as

$$\varepsilon = \sum_{i=1}^{N_p} \varepsilon_i \qquad (10)$$

where N_P is the number of nodes of the coarse grid. Given a user specified global tolerance *tol* for the functional of interest, we specify a target adaptation level for each node as

$$t = \frac{tol}{N_P} \qquad (11)$$

A node of the original coarse mesh is flagged for refinement if

$$\varepsilon_{H,i} > \lambda t \qquad (12)$$

where $\lambda \geq 1$ is an optional global threshold factor used to scale the refinement parameter. This parameter can be useful to reduce the number of flagged nodes in the initial cycles of the adaptation process [10], especially for the CC- strategy adaptation.

A cell or edge of the coarse mesh Ω_H is selected for refinement if it has at least one node flagged for refinement. Once the mesh is refined, the solution is computed on the new adapted grid and a new adaptation cycle begins.

Thus, an adaptation cycle consists of the computation of the error correction and adaptation parameters on the embedded mesh, the transfer of this adaptation parameter back to the starting coarse mesh and the mesh refinement itself using the adaptation parameter. No termination criterion other than a user-prescribed number of cycles has been implemented in the adaptive process in the present version. Three input parameters, which must be prescribed by the user, control the adjoint-based error correction and adaptation process: number of adaptation cycles, desired functional output tolerance, *tol*, and relaxing threshold, λ.

3 Implementation issues

The adjoint-based error correction and adaptive refinement process has been implemented in a new TAU module called *Adj_error_adapt*. The complete process involves the interaction of several TAU modules: Primal and adjoint solvers [12,14], as well as the adaptation [12] module. The adaptation process has been integrated using a python script shown in Fig. 1.

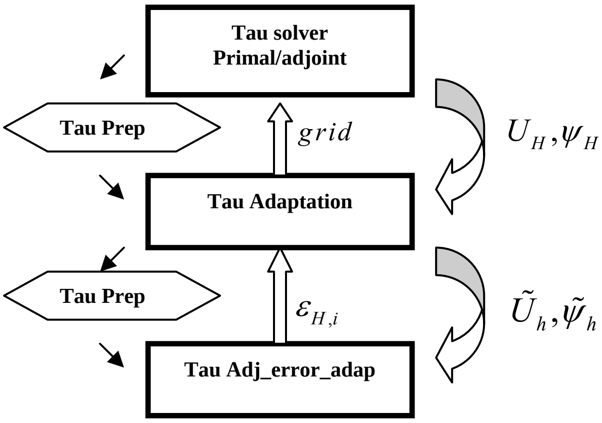

Fig. 1. Adjoint-based adaptation process.

In this way, the process is fully automatic and easy to use since the only three required inputs are the number of adaptation cycles, the prescribed tolerance for the chosen output functional and the relaxing threshold. The automation and ease of use of a CFD process are two of the chief requirements for a process to be successfully used in industrial environments.

The inputs for the *Adj_error_adapt* module are the embedded dual grid file and the interpolated primal and adjoint solution files. A key point of the process is the transfer of the adaptation parameter back to the original coarse mesh, i.e., Eq. (9). To accomplish this operation it is necessary to keep track of the information of the original coarse mesh in the embedded one. This task is carried out by introducing a new *inverse* permutation index which maps the embedded grid onto the original coarse one. In this way, an edge of the embedded grid "knows" whether its endpoints are new points or "old" coarse grid points (with the corresponding old connectivity). Following a common practice employed in transferring residuals within multigrid methods, the projection operator, $\tilde{P}_H^h$, is constructed in such a way that the total error transferred to the coarse mesh is equal to the total error computed in the fine grid, i.e.

$$\sum_{i}^{N_{P,H}} \varepsilon_{H,i} = \sum_{i}^{N_{P,h}} \varepsilon_{h,i} \tag{13}$$

The new adjoint error correction module has been implemented to work in either serial or parallel modes alike. In the current implementation, the data exchange among the TAU modules is through file I/O rather than memory. However, the python script may be extended in a straightforward manner to perform the exchanges in memory using the TAU-python API.

4 Discussion of results

Adaptation results are shown for three cases: 2D and 3D Euler flows over an airfoil and wing, respectively, and an inviscid 3D flow over a wing-body-pylon geometry. The first two cases can be found in many studies reported in the specialized literature, and they are normally used for validation. The third test case was selected for the demonstration of the adjoint-based error estimation and adaptation (AEA) method on a more representative (from the industrial viewpoint) configuration. Two functional outputs have been investigated in these studies, the total drag and lift.

4.1 Grid adaptation strategies

As explained above, two adjoint-based adaptation strategies are available (CC and RE), depending on the adaptation sensor being used, which is the correction term in (3) for the CC-strategy, and the remaining error term (5) for the RE-strategy. The question naturally arises as to which of them is the most appropriate strategy to be used in adjoint-based adaptation methods.

In the present work, we have tried to find an answer to the former question from a practical point of view (keeping always in mind the industrial application). A comparative study between the two adaptation strategies (CC and RE) has been performed for 2D and 3D inviscid transonic flows. For the 2D case, a NACA 0012 airfoil at Mach number $M_\infty = 0.8$ and $\alpha = 1.25^o$ incidence was adapted using the AEA method. The initial grid had 5990 nodes. The target functional was the total drag coefficient. The user prescribed tolerance was set to 0.001 (10 drag counts), with a threshold $\lambda = 1$ which was held constant throughout the adaptation cycles. For the 3D case, an inviscid transonic flow over an ONERA M6 at angle of attack of $\alpha = 3.06^o$ and Mach number M_∞ = 0.84 was tested. The adaptation parameters are the same as the 2D case and the initial grid has over 50000 nodes.

Both the CC and RE-strategies were compared for the two test cases. The adapted drag values are displayed in Fig 2. This figure also shows the drag values obtained with the default TAU adaptation method, which is a feature-based adaptation based on the reduction of the local differences of some selected flow variables [15]. The uniform h-adaptation is also displayed in order to show the grid convergence properties of this case for the selected numerical scheme, as well as the drag value for asymptotic convergence. The results show that the AEA method can reach the final drag level with two orders of magnitude less points compared with uniform refinement and one compared to the feature-based adaptation. The RE-strategy is clearly superior to the CC-strategy, particulary in the 2D case, where the asymptotic value is reached with ten times less points; as for the 3D case, the differences are not so large, but the RE-strategy is still more competitive.

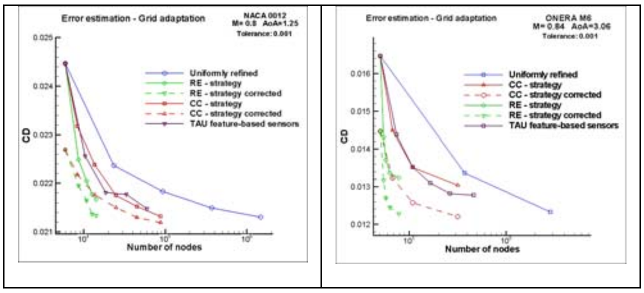

Fig. 2. Drag convergence for the 2D case (left) and the 3D case (right) using two different adaptation strategies.

For all the cases tested in the present study the CC-strategy always gives rise to adapted grids with more nodes than the RE-strategy. From an industrial point of view, the incremental percentage of points in each adaptation can be critical if the initial mesh is large (which is usually the case for complex test cases). Therefore, the RE-strategy is, from this viewpoint, the most desirable.

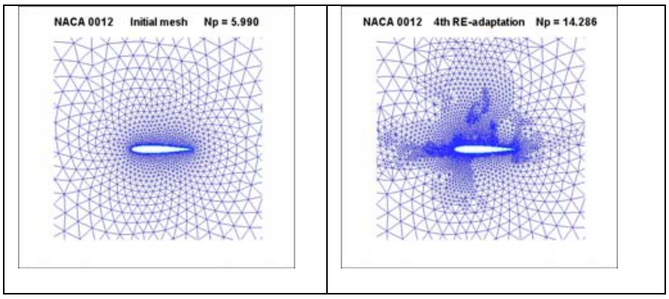
NACA 0012 Initial mesh Np = 5.990
NACA 0012 4th RE-adaptation Np = 14.286

Fig. 3. Initial (left) and final (right) grids for the NACA0012 with adjoint-based adaptation using the RE-strategy.

At any rate, both strategies clearly outperform the feature-based TAU adaptation strategy. In transonic cases, for example, such strategy would refine the shock wave indefinitely owing to the unbounded derivates across the discontinuity. Hence, the feature-based strategy, after several cycles, over-refines the shock areas but not other areas which can have a greater impact on the final drag value such as the leading and trailing edge regions of the airfoil. This should be compared with the adapted grids produced by the adjoint-based approach. The final grids after four adaptation cycles (using the RE-strategy) are shown in Fig. 3. and 4. for the NACA0012 airfoil and ONERA M6 wing respectively. Fig.4. shows that, unlike the feature-based scheme, the AEA method avoids the over-refinement of the shock area and concentrates instead on regions where the discretization errors may have a greater influence on the computation of the selected functional output (e.g. the region upstream of the airfoil or the area neat the trailing edge).

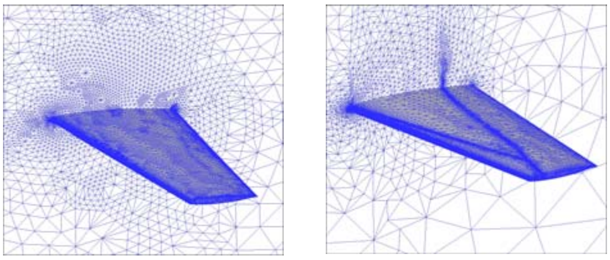

Fig. 4. Final grids for the ONERA M6 wing with adjoint-based adaptation using the RE-strategy (left) and feature-based adaptation (right).

4.2 Influence of surface reconstruction

An important issue of the adaptation process is in general the surface reconstruction strategy. This question is particularly tricky in the case of adjoint-based adaptation approaches, which admits initial grids significantly coarser than those used in feature-based approaches. The TAU adaptation module employs cubic surface reconstruction (via edge-based cubic spline interpolation using smoothed normal surface vectors at the edge nodes), and the question arises as to whether this reconstruction strategy is accurate enough for the adjoint-based approach. The ONERA M6 wing test case has been tested with adjoint adaptation for the drag functional using the original reconstruction module. The examination of the resulting chord-wise Cp distributions at different span sections for the final adapted grid (starting from a relative coarse grid with 50000 nodes) reveals pressure oscillations in the mid and aft location of the sections (see Fig. 5.).

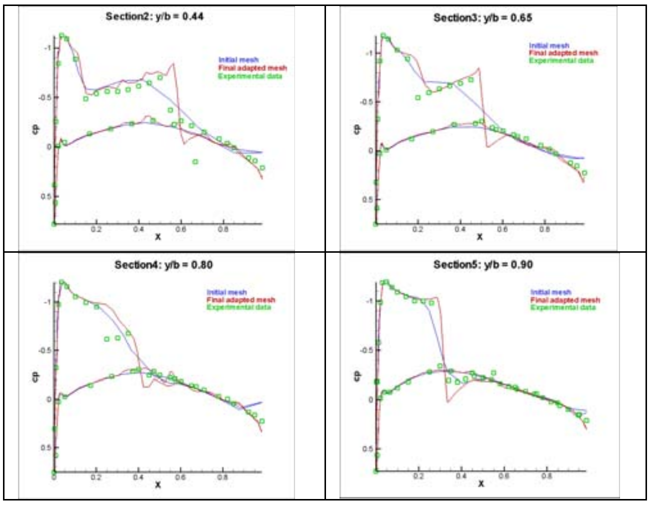

Fig. 5.Pressure distributions of adapted grid using local surface reconstruction for the ONERA M6 wing.

These oscillations stem from the type of surface grid reconstruction used by the TAU adaptation module. This type of reconstruction has a minor effect on the computed pressures when the surface grid resolution is sufficient to produce small errors in the interpolated points (see the outermost section, $y/b = 0.9$, in Fig. 5). However, when the resolution is too low to support the interpolation, the reconstructed surface has curvature oscillations which are reflected in the computed pressure distributions.

In order to assess this effect, the adaptation process was repeated using CAD-based reconstruction of the surface grid. A set of modules developed by INTA [16] were introduced in the adaptation chain to tackle the surface reconstruction. A CAD-based description of the wing, based on surface Non-Uniform Rational B-Splines (NURBS), is used to reconstruct the surface grid by performing a projection of the new nodes (created by refinement) onto the NURBS representation. The volume mesh must also be deformed in order to obtain good quality cells near the surface. This process proves to be reasonably robust for small surface deformations in the case of Euler meshes.

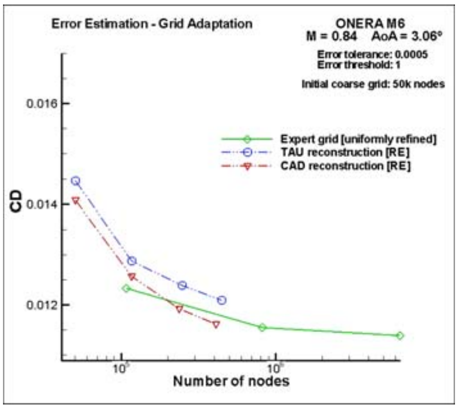

Fig. 6. Drag convergence for the ONERA M6 wing with two types of surface reconstruction.

The results obtained for the ONERA M6 wing with the CAD-based surface reconstruction technique are shown in Fig. 6, which also depicts the values obtained with the cubic-spline surface reconstruction of TAU, both starting from a mesh with 50000 nodes. The adjoint-based adapted and corrected values are improved significantly with the CAD-improved technique. The asymptotic drag level obtained using CAD-based surface reconstruction is nearly the same as that obtained by uniform h-adaptation from an initial mesh with 10^5 nodes. Fig.7. shows the chord-wise wing surface pressure distributions for the adjoint-based adapted grid with CAD-based surface reconstruction. The oscillations are no longer present in the pressure distributions.

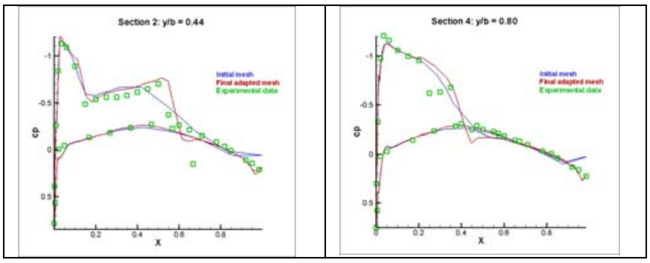

Fig. 7. Pressure distributions for section of an ONERA M6 wing with CAD surface reconstruction.

4.3 Non-expert initial grids

One of the most attractive features of the AEA method is the capability to generate optimized grids for the chosen functional starting from relatively coarse grids. This is related to an old dream of the CFD engineers, i.e. to remove the dependence on user expertise in the grid generation phase. The adjoint-based method provides an approach that localizes mesh refinement to only those flow features that actually influence the accuracy of the chosen functional, and is largely independent of the initial grid. This section describes the results obtained for the adaptation process of an ONERA M6 wing starting from three different, extremely coarse grids generated with the prescription of having the minimum resolution necessary to solve the geometric details (we call those grids non-expert grids). The drag convergence results are displayed in Fig. 8.

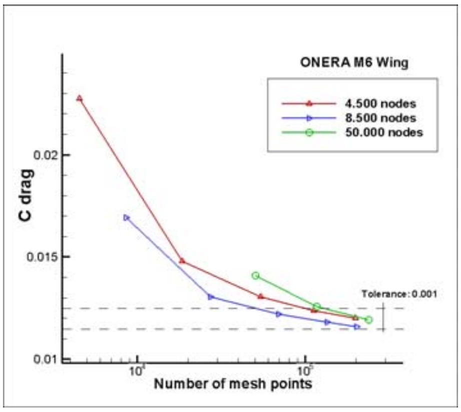

Fig. 8. Drag convergence for the ONERA M6 wing with three initial non-expert grids.

The RE-strategy with a prescribed tolerance of 0.001 was used in all cases. The corrected drag values converge to the same final asymptotic value within the user prescribed tolerance. Thus, the adjoint-based adaptation is able to adapt very poorly resolved grids to obtain accurate estimations of the selected functional output.

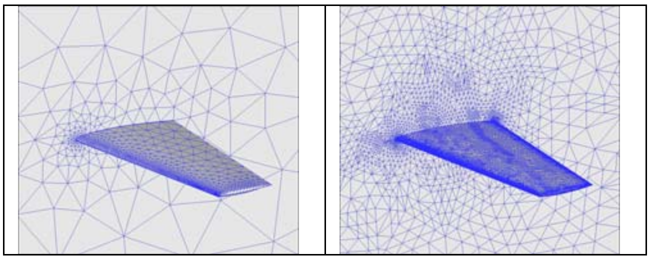

Fig. 9. Initial and final grids (after 4 adjoint adaptation cycles) for the ONERA M6 wing. The initial grid has only 8500 nodes.

Fig. 9. shows the initial and final grids for the ONERA M6 wing starting from a coarse non-expert grid with only 8500 nodes. The mesh was generated with CentaurSoft with the only requirement of having a minimum resolution at the leading edge. In the final grid, the leading and trailing edge were refined as well as the shock wave areas.

4.4 Complex geometry test case: Lift adaptation of the DLR-F6 WBNP

The adjoint-based adaptation procedure is applied to a complex aircraft configuration in order to explore the potential application of this method to more realistic aircraft geometries. The chosen configuration is the DLR-F6 wing-body-nacelle-pylon (WBNP) configuration. This configuration was introduced in the AIAA Drag Prediction Workshop II to investigate the drag increment due to engine installation. Transonic flow conditions of M_∞ = 0.8 and 0 degrees incidence were selected to test the adaptation process. Such off-design conditions were selected with the sole purpose of verifying the adaption process and not to validate with experimental results. In the present work, results are shown only for inviscid flows. Such a restriction does not prevent difficulties from appearing, as we will see momentarily. An initial grid containing about 275000 nodes was generated by INTA using Centaur Soft. Although this grid is relatively coarse it has the necessary resolution to resolve the main geometrical features of the configuration (see Fig. 10).

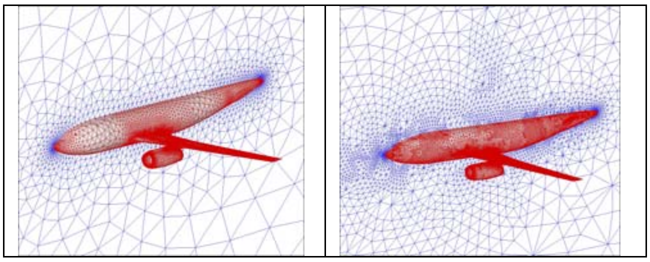

Fig. 10. DLR-F6 WBNP configuration. Left: initial grid (270 knodes). Right: final adapted grid after three adjoint adaptations.

The lift error prediction and grid adaptation procedure with user specified error tolerance of 0.03 (and constant threshold of 1) is plotted as a function of mesh size in Fig. 11. The uniform h-adaptation and TAU's feature-based adaptation are also displayed in the same figure. The correct lift coefficient is obtained on the adjoint-based adapted mesh with less than one-tenth of the size of the uniformly refined grid (7.5×10^5 vs. 10.4×10^6 nodes, respectively). The feature-based method is also able to adapt the mesh to yield reasonable results but the final mesh contains over three and a half more nodes than the adjoint-based one.

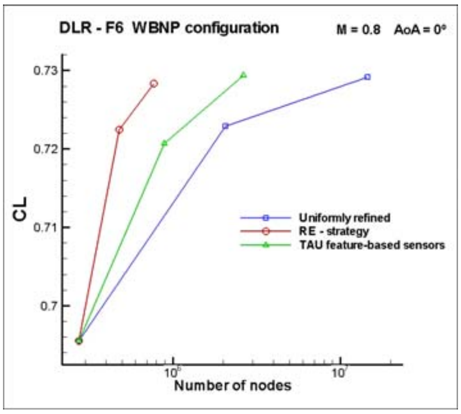

Fig. 11. Lift convergence for the DLR-F6 WBNP configuration.

Only two adjoint adaptation cycles were performed. The quality of the adapted mesh deteriorated after each adaptation cycle, which eventually prevented the convergence of the adjoint solution and consequently any further adjoint analysis. The improvement of the grid quality after each adaptation step (through grid smoothing techniques) and the improvement of the robustness of the discrete adjoint solver are two issues which have to be addressed in order to make the adaptation procedure robust enough to be used in industrial environments.

The initial and adjoint-based adapted surface meshes are shown in Fig. 10, where the mesh on the symmetry plane also is also shown. Observe that the areas located on the fore and aft fuselage have been refined. A detail of the wing surface refinement is shown in Fig. 12. The adjoint-based method refines all relevant areas which may affect the lift coefficient (wing leading and trailing edge, shock location). Note that the wing was refined especially on the leading edge at the inboard wing area. The pylon and the nacelle were also refined. The Mach contour map is also depicted in Fig. 12.

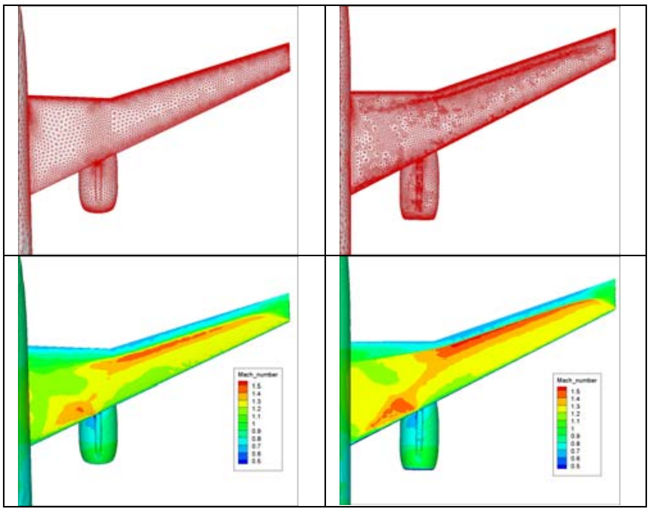

Fig. 12. Detail of initial and final adapted grids for the DLR-WBNP (above), and Mach number contours (below) on the suction side of the wing.

4.5 Drag adaptation of the DLR-F6 WBNP

Finally, the adjoint-based methodology is tested in an adaptation process with total drag as the target output function. The same off-design flow conditions as the in previous section were used for this case. A non-constant threshold scaling strategy was applied, in order to prevent a rapid growth of the number of nodes of the adapted grid during the first adaptation steps. The user specified error tolerance was prescribed to 10 drag counts (0.001). The initial threshold value was set to 30 and reduced successively to 10 and 5. The last adaptation was also obtained with a threshold value of 5, thus the true error tolerance is equivalent to 50 drag counts. Fig. 13 shows the results for adjoint-based, feature-based and uniform adaptation. The drag estimation is improved but the convergence failure of the discrete adjoint solver in the last adaptation cycle prevented the obtention of an asymptotic value within the tolerance range. Nevertheless, the final corrected value for the drag is nearly the same as the one obtained by uniform h-adaptation.

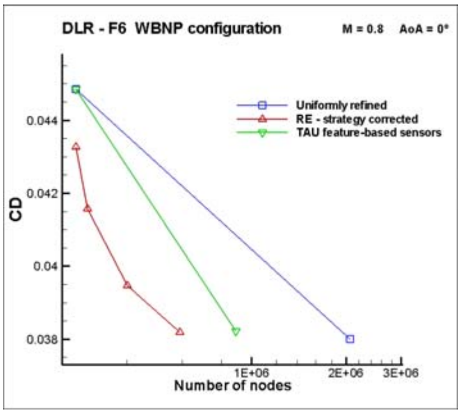

Fig. 13. Drag convergence for the DLR-F6 WBNP configuration.

4.6 Robustness and computational efficiency

Two relevant aspects (qualities) of every CFD technique are the robustness and the computational efficiency. The more robust and efficient a CFD technique is, the more likely it is that it will make it into industrial production environments. According to [17] the robustness of the (adaptation) process may be defined as its likelihood to produce the result (adapted mesh). On the other hand, the computational efficiency may be linked, for this process, to the additional CPU-time and memory overhead involved in the adaptation process.

The experience gained through the application of the present adjoint adaptation method to several test cases showed us that the robustness of such method is mainly linked to the robustness of the direct and, especially, the adjoint solvers. For the present case, the robustness (or lack thereof) may be associated to the quality of the grids in each adaptation cycle. Table 1 shows the quality of the adapted meshes (by means of two different quality parameters, the mean ratio and the minimal dihedral angle [15]) at each adaptation cycle for the DLR-F6 WBNP case. What can be deduced from Table 1 is that the quality of the mesh is degrading through the adaptation process, especially when local surface reconstruction is used. The quality of each mesh can be improved if an additional module for mesh smoothing is introduced in the adaptation chain –see Fig.1.

	DLR - F6 configuration. Initial mesh: 270K points (Lift adaptation)				
	MESH	N(q<0,2)	N(alpha<5º)	Nº points	Size (MB)
UNIFORM ADAPTATION	Initial mesh	0	0	275561	32,13
	1st Adaptation	0	0	2061293	384,64
	2nd Adaptation	454	24	4547849	892,55
REMAINING ADAPTATION	Initial mesh	0	0	275561	32,13
	1st Adaptation	0	0	480446	85,53
	2nd Adaptation	29	0	772098	145,64
TAU feature-based adaptation	Initial mesh	0	0	275561	32,13
	1st Adaptation	145	0	892380	168,08
	2nd Adaptation	280	15	2647671	531,73

Table 1. Bad elements (in red) generated in the adaptation process for the DLR-F6 WBNP configuration.

As for the computational efficiency, table 2 shows a CPU-time profiling (in % of the total process time) of the adaptation process for the DLR-F6 WBPN case. It can be seen that most of the CPU-time overhead is due to the Adjoint solver (which is of the order of the direct solver). This situation is even worse for Navier-Stokes solutions if no especial convergence techniques are employed for obtaining the adjoint solutions. Therefore, this kind of adaptation seems unsuitable when the initial meshes are too large (typical sizes in industry) because they are very time consuming. The present adaptation technique seems to be more appropriate for coarse initial meshes.

F6 WBNP configuration (772 Knodes)	
2nd RE - adaptation of a complete aircraft	
STEP	% TOTAL TIME
Flow Solver	45,8
Adjoint Solver	45,9
Uniformly Adaptation	3,4
Error - estimation	1,5
Adjoint Adaptation	1,3
Smoothing	2,2

Table 2. CPU-time profiling (percentages) for the DLR-F6 WBNP configuration adaptation process.

Likewise, even though the construction of the embedded uniform mesh does not seem to be very time consuming, memory requirements involved in this step may suppose a bottle neck given the sizes of the meshes typically used in real applications. This can be illustrated with the following simple example: Consider a typical coarse grid used in industry for a RANS computation of a wing-body configuration. Such grid may have about 3.8×10^6 nodes and 10.5×10^6 cells (4.6×10^6 tetrahedra and 5.8×10^6 prisms for a mesh with 25 viscous layers). The corresponding (uniformly refined) embedded grid would have about 60×10^6 elements and 18.5×10^6 nodes (assuming that in the embedded grid the number of viscous layers is held constant). The preprocessor, adaptation and solver modules have to manage meshes with about 20×10^6 points already at the first adaptation cycle!

5 Conclusions

We have implemented and tested an adjoint-based error prediction and grid adaptation methodology in a widely used CFD-software (DLR-TAU).

The method outperforms conventional feature-based adaptation in that comparable accuracy is obtained with far smaller meshes. Moreover, the method can be applied to extremely coarse initial grids, provided that integrity of the geometry is preserved with a suitable surface reconstruction technique. The adjoint adaptation technique turns out to be a powerful method for obtaining grids especially tuned for accurate functional output calculations but at the same time it is a computationally expensive technique.

The approach has a few potential drawbacks, which are inherent in the methodology itself, that could make the method difficult to apply at production level within industrial environments. On the one hand, the dependency on an auxiliary, uniformly refined mesh which has to be generated at each adaptation step can render the method unaffordable for high Re viscous flows around complex configurations, not only for the size of the involved auxiliary meshes but for the difficulty inherent in the refinement of highly anisotropic hybrid meshes. On the other hand, robustness and computational efficiency of the adjoint solver is still an issue to be addressed for the systematic application of the adjoint technology in the industry environment.

6 Contact Author Email Address

ponsinj@inta.es

Acknowledgemets

The research described in this paper has been supported by the Ministry of Defence-INTA under the activity Termofluidodinámica (IGB99001), by the Ministry of Education and Science under projects DOMINO (CIT-370200-2007-22) and by Airbus-Spain under project DOVRES/FUSIM-E. The TAU Code is property of the Deutsches Zentrum für Luft- und Raumfahrt (DLR), developed at the Institute of Aerodynamics and Flow Technology at Göttingen and Braunschweig, and has been licensed to INTA through a research and development cooperation agreement.

Copyright Statement

References

[1] Dwight R. Heuristic a posteriori estimation of error due to dissipation in finite volume schemes and application to mesh adaptation. *Journal of Computational Physics,* 227, pp 2845-2863, 2008.

[2] Vassberg J, DeHaan M, Sclafani T. Grid Generation Requirements for Accurate Drag Predictions Based on OVERFLOW Calculations. *AIAA Computational Fluid Dynamics Conference,* Orlando, Florida, AIAA 2003-4124, 2003.

[3] Mavriplis D, Vassberg J, Tinoco E, Mani M, Brodersen O, Eisfield B, Wahls R, Morrison J, Zickuhr T, Levy D, Murayama M. Grid Quality and Resolution Issues from the Drag Prediction Workshop Series. *Journal of Aircraft,* Vol. 46, No. 3, pp 935-950, 2009.

[4] Fidowski K, Damorfal D. Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics: Overview and Recent results. *AIAA Aerospace Science Meeting,* Orlando, Florida, AIAA 2009-1303, 2009.

[5] Venditti D, Damorfal D. Anisotropic Adaptation for Functional Outputs of Viscous Flow Simulations. *Computational Fluid Dynamics Conference*, Orlando, Florida, AIAA 2003-3845, 2003.

[6] Park M. Adjoint-based, Three-Dimensional Error Prediction and Grid Adaptation. *Fluid Dynamics Conference*, Florida, AIAA 2002-3286, 2002.

[7] Kim H-J, Nakahashi K. Output-based Error Estimation and Adaptive Mesh Refinement Using Viscous Adjoint Method. *44th AIAA Aerospace Science Meeting,* Orlando, Florida, AIAA 2005-1395, 2005.

[8] R Balasubramanian, Newman J. Comparison of Adjoint-based and Feature-based Grid Adaptation for Functional Outputs. *Applied Aerodynamics Conference*, San Francisco, California, AIAA 2006-3314, 2006.

[9] Lee-Rausch E-M, Park M, Jones W, Hammond D, Nielsen E. Application of Parallel Adjoint-based Error Estimation and Anisotropic Grid Adaptation for Three-Dimensional Aerospace Configurations. *Applied Aerodynamics Conference*, Toronto, Ontario, AIAA 2005-4842, 2005.

[10] Nemec M, Aftosmis M. Adjoint-based Adaptive Mesh Refinement for Complex Geometries. *46th AIAA Aerospace Science Meeting,* Reno, Nevada, AIAA 2008-725, 2008.

[11] Venditti D. Grid Adaptation for Functional Outputs of Compressible Flow Simulations. *Ph.D. thesis,* Massachusetts Institute of Technology, Cambridge, MA, 2002.

[12] Gerhold T, Galle M, Friedrich O, Evans J. Calculation of Complex 3D Configurations Employing the DLR TAU-code. AIAA 97-0167, 1997.

[13] Muller J, Giles M. Solution Adaptive Mesh Refinement Using Adjoint Error Analysis. *AIAA Computational Fluid Dynamics Conference,* Anaheim, California, AIAA 2001-2550, 2001.

[14] Dwight R, Brezillon J. Effect of Approximations of Discrete Adjoint on Gradient-Based Optimization. *AIAA Journal,* Vol. 44, No. 12, pp 3022-3031, 2006

[15] Technical Documentation of the DLR TAU-code. Release 2010.1.0. March 2010.

[16] Bitrian P. Guía del Usuario de la suite DOMINO. INTA Technical Report no. AT/TNO/4410/009/INTA/09. 2009.

[17] Kroll N. ADIGMA-A European Project on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications. *AIAA Aerospace Science Meeting,* Orlando, Florida, AIAA 2009-176, 2009.