

# Higher-order Stabilized Finite Elements in an Industrial Navier-Stokes Code

Frédéric Chalot and Pierre-Elie Normand

**Abstract** This chapter covers Dassault Aviation's contribution to Workpackage 3 of the ADIGMA Project, which focuses on the extension of its stabilized finite element industrial Navier-Stokes code to higher-order elements. Mesh generation aspects are treated and especially the issue of highly-stretched curved elements close to the wall boundary of Navier-Stokes meshes. The high-order approach is carefully assessed using inviscid subsonic and transonic, laminar, and high Reynolds number turbulent flows.

## 1 Introduction

Over the past two decades, modern CFD has gone from producing pretty pictures to actually producing numbers which are crucial when improving the aerodynamic design of aircraft. Over this period of time, models have improved going from inviscid Euler calculations to laminar and then turbulent Navier-Stokes. Turbulent models have evolved from purely algebraic models to RANS models, to unsteady models like LES and DES which are slowly making their way into the industrial world [10]. The growth of computer power has also tremendously helped that change: the power of the vector supercomputers of the 80's is now available on laptop PC's, whereas the Top500 parallel computers are flirting with a few sustained Petaflops. Most industrial CFD codes and commercial packages have made the transition from early developments in finite differences to finite volumes, and are rapidly moving

---

Frédéric Chalot  
Dassault Aviation, 78 quai Marcel Dassault, CEDEX 300, 92252 Saint-Cloud CEDEX, France  
e-mail: frederic.chalot@dassault-aviation.com

Pierre-Elie Normand  
Université de Bordeaux, 341 cours de la Libération, 33405 Talence CEDEX, France  
e-mail: pierre-elie.normand@external.dassault-aviation.com

away from the apparent simplicity of structured and block-structured meshes to the flexibility of unstructured meshes, often at the price of a lesser efficiency.

A large amount of work was also performed to develop stable and accurate spatial numerical schemes for compressible flow calculations. Overly diffusive first-order schemes were rapidly abandoned for second-order accurate schemes with monotonic shock-capturing capabilities. Such a level of accuracy proved sufficient for most industrial codes, even for applications such as LES and DES which were at first developed with higher-order schemes [7]. Higher-order schemes seem reserved for specific fields (DNS, aeroacoustics) where the enhanced accuracy is mandatory.

Today's complex applications require an ever increasing number of grid points for which mesh convergence can seemingly never be attained.

The European project **ADIGMA** proposed a unique framework to address many of the accuracy and cost issues of current industrial CFD codes. The different partners have put together innovative higher-order methods which will constitute key ingredients for the next generation of industrial flow solvers. The participation of Dassault Aviation was twofold: higher-order stabilized finite elements described in the following sections, and adjoint-based adaptive mesh refinement detailed in Chapter ??.

## 2 Higher-order stabilized finite element schemes for the RANS equations

Although Dassault Aviation started from the beginning with unstructured meshes and a Navier-Stokes code based on a finite element formulation, the claim that finite elements can fairly effortlessly and in a straightforward manner go high in order was never fully exploited. We currently still use for all Navier-Stokes calculations linear elements which yield second-order accuracy [5, 9, 10]. A single but successful attempt was made to compute the flow past a supersonic ramp [2] using quadratic elements.

Higher-order (3<sup>rd</sup> and 4<sup>th</sup> order) finite elements in the SUPG/Galerkin-least squares framework will be revisited. We will present our numerical method in the following sections and highlight the adjustments required by higher-order elements.

### 2.1 *General description of our flow solver*

Dassault Aviation's Navier-Stokes code, called **AETHER**, uses a finite element approach, based on a symmetric form of the equations written in terms of entropy variables. The advantages of this change of variables are numerous: in addition to the strong mathematical and numerical coherence they provide (dimensionally correct dot product, symmetric operators with positivity properties, efficient precondi-

tioning), entropy variables yield further improvements over the usual conservation variables, in particular in the context of chemically reacting flows (see [4, 5]).

The code can handle the unstructured mixture of numerous types of elements (triangles and quadrilaterals in 2-D; tetrahedra, bricks, and prisms in 3-D). In practice mostly linear triangular and tetrahedral meshes are used.

Different one- and two-equation Reynolds-averaged turbulence models are available: Spalart-Allmaras,  $K-\varepsilon$ ,  $K-\omega$ ,  $K-l$ ,  $K-KL$ . . . These models are either integrated down to the wall, use a two-layer approach with a low-Reynolds modelization of the near wall region, or adopt a wall function treatment of the boundary layer. More advanced RANS models, such as EARSM and RSM, and extensions to LES and DES are also available (see [7], [8], and [10]).

Convergence to steady state of the compressible Navier Stokes equations is achieved through a fully-implicit iterative time-marching procedure based on the GMRES algorithm with nodal block-diagonal or incomplete  $LDU$  preconditioning (see [12]).

The code has been successfully ported on many computer architectures. It is fully vectorized and parallelized for shared or distributed memory machines using the MPI message passing library (IBM SP2 Series, IBM BlueGene, Itanium II- and Xeon-based Bull NovaScale) or native parallelization directives (NEC SX-4) (see [6]).

## 2.2 The symmetric Navier-Stokes equations

As a starting point, we consider the compressible Navier-stokes equations written in conservative form:

$$\mathbf{U}_{,i} + \mathbf{F}_{i,i}^{\text{adv}} = \mathbf{F}_{i,i}^{\text{diff}} \quad (1)$$

where  $\mathbf{U}$  is the vector of conservative variables;  $\mathbf{F}_i^{\text{adv}}$  and  $\mathbf{F}_i^{\text{diff}}$  are, respectively, the advective and the diffusive fluxes in the  $i^{\text{th}}$ -direction. Inferior commas denote partial differentiation and repeated indices indicate summation.

Equation (1) can be rewritten in quasi-linear form:

$$\mathbf{U}_{,i} + \mathbf{A}_i \mathbf{U}_{,i} = (\mathbf{K}_{ij} \mathbf{U}_{,j})_{,i} \quad (2)$$

where  $\mathbf{A}_i = \mathbf{F}_{i,i}^{\text{adv}}$  is the  $i^{\text{th}}$  advective Jacobian matrix, and  $\mathbf{K} = [\mathbf{K}_{ij}]$  is the diffusivity matrix, defined by  $\mathbf{F}_i^{\text{diff}} = \mathbf{K}_{ij} \mathbf{U}_{,j}$ . The  $\mathbf{A}_i$ 's and  $\mathbf{K}$  do not possess any particular property of symmetry or positiveness.

We now introduce a new set of variables,

$$\mathbf{V}^T = \frac{\partial \mathcal{H}}{\partial \mathbf{U}}$$

where  $\mathcal{H}$  is the generalized entropy function given by

$$\mathcal{H} = \mathcal{H}(\mathbf{U}) = -\rho s$$

and  $s$  is the thermodynamic entropy per unit mass. Under the change of variables  $\mathbf{U} \mapsto \mathbf{V}$ , (2) becomes:

$$\widetilde{\mathbf{A}}_0 \mathbf{V}_{,t} + \widetilde{\mathbf{A}}_i \mathbf{V}_{,i} = (\widetilde{\mathbf{K}}_{ij} \mathbf{V}_{,j})_{,i} \quad (3)$$

where

$$\begin{aligned} \widetilde{\mathbf{A}}_0 &= \mathbf{U}_{,\mathbf{V}} \\ \widetilde{\mathbf{A}}_i &= \mathbf{A}_i \widetilde{\mathbf{A}}_0 \\ \widetilde{\mathbf{K}}_{ij} &= \mathbf{K}_{ij} \widetilde{\mathbf{A}}_0. \end{aligned}$$

The Riemannian metric tensor  $\widetilde{\mathbf{A}}_0$  is symmetric positive-definite; the  $\widetilde{\mathbf{A}}_i$ 's are symmetric; and  $\widetilde{\mathbf{K}} = [\widetilde{\mathbf{K}}_{ij}]$  is symmetric positive-semidefinite. In view of these properties, (3) is referred to as a symmetric advective-diffusive system.

For a general divariant gas, the vector of so-called (physical) entropy variables,  $\mathbf{V}$ , reads

$$\mathbf{V} = \frac{1}{T} \begin{Bmatrix} \mu - |\mathbf{u}|^2/2 \\ \mathbf{u} \\ -1 \end{Bmatrix}$$

where  $\mu = e + pv - Ts$  is the chemical potential per unit mass;  $v = 1/\rho$  is the specific volume. More complex equations of state are treated in [3]. We would like to stress the formal similarity between the conservation variables  $\mathbf{U}$  and the entropy variables  $\mathbf{V}$ , which can be made more apparent if we write the conservation variables in the following form:

$$\mathbf{U} = \frac{1}{v} \begin{Bmatrix} 1 \\ \mathbf{u} \\ e + |\mathbf{u}|^2/2 \end{Bmatrix}$$

where  $v = 1/\rho$  is the specific volume.

Taking the dot product of (3) with the vector  $\mathbf{V}$  yields the Clausius-Duhem inequality, which constitutes the basic nonlinear stability condition for the solutions of (3). This fundamental property is inherited by appropriately defined finite element methods, such as the one described in the next section.

### 2.3 The Galerkin/least-squares formulation

Originally introduced by Hughes and Johnson, the Galerkin/least-squares (GLS) formulation is a full space-time finite element technique employing the discontinuous Galerkin method in time (see [1, 13]). The least-squares operator ensures good stability characteristics while retaining a high level of accuracy. The local control

of the solution in the vicinity of sharp gradients is further enhanced by the use of a nonlinear discontinuity-capturing operator.

Let  $\Omega$  be the spatial domain of interest and  $\Gamma$  its boundary. The semi-discrete Galerkin/least-squares variational problem can be stated as:

Find  $\mathbf{V}^h \in \mathcal{S}^h$  (trial function space), such that for all  $\mathbf{W}^h \in \mathcal{V}^h$  (weighting function space), the following equation holds:

$$\begin{aligned} & \int_{\Omega} \left( \mathbf{W}^h \cdot \mathbf{U}_t(\mathbf{V}^h) - \mathbf{W}_i^h \cdot \mathbf{F}_i^{\text{adv}}(\mathbf{V}^h) + \mathbf{W}_i^h \cdot \widetilde{\mathbf{K}}_{ij} \mathbf{V}_{,j}^h \right) d\Omega \\ & + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \left( \mathcal{L} \mathbf{W}^h \right) \cdot \boldsymbol{\tau} \left( \mathcal{L} \mathbf{V}^h \right) d\Omega \\ & + \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} v^h g^{ij} \mathbf{W}_i^h \cdot \widetilde{\mathbf{A}}_0 \mathbf{V}_{,j}^h d\Omega \\ & = \int_{\Gamma} \mathbf{W}^h \cdot \left( -\mathbf{F}_i^{\text{adv}}(\mathbf{V}^h) + \mathbf{F}_i^{\text{diff}}(\mathbf{V}^h) \right) n_i d\Gamma. \end{aligned} \quad (4)$$

The first and last integrals of (4) represent the Galerkin formulation written in integrated-by-parts form to ensure conservation under reduced quadrature integration.

The second integral constitutes the least-squares operator where  $\mathcal{L}$  is defined as

$$\mathcal{L} = \widetilde{\mathbf{A}}_0 \frac{\partial}{\partial t} + \widetilde{\mathbf{A}}_i \frac{\partial}{\partial x_i} - \frac{\partial}{\partial x_i} \left( \widetilde{\mathbf{K}}_{ij} \frac{\partial}{\partial x_j} \right). \quad (5)$$

$\boldsymbol{\tau}$  is a symmetric time-scale matrix for which definitions can be found in [13].

The third integral is the nonlinear discontinuity-capturing operator, which is designed to control oscillations about discontinuities, without upsetting higher-order accuracy in smooth regions.  $g^{ij}$  is the contravariant metric tensor defined by

$$[g^{ij}] = [\boldsymbol{\xi}_{,i} \cdot \boldsymbol{\xi}_{,j}]^{-1}$$

where  $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x})$  is the inverse isoparametric element mapping and  $v^h$  is a scalar-valued homogeneous function of the residual  $\mathcal{L} \mathbf{V}^h$ . The discontinuity capturing factor  $v^h$  used for linear elements is an extension of that introduced by Hughes, Mallet, and Shakib (see [11, 13]).

A key ingredient to the formulation is its consistency: the exact solution of (1) satisfies the variational formulation (4). This constitutes an essential property in order to attain higher-order spatial convergence.

## 2.4 Extension to higher-order elements

In principle everything is contained in the weighted residual given by Eq. (4). There is no new term to code, no interpolation technique specific to higher order to derive: everything is already there. We just have to compute the integrals of (4), taking into account the new higher-order shape functions.

The volume and surface integrals are numerically evaluated with quadrature rules. All is needed is the values of the shape functions (and their gradients) at the integration points. Higher-order functions only require more precise integration rules. In general, we use 3-, 6-, and 12-point rules, respectively for linear, quadratic, and cubic triangles. They have orders of accuracy which integrate exactly polynomials of degrees 2, 4, and 6 respectively.

For a given number of degrees of freedom, higher-order meshes contain much fewer elements than P1 meshes. The ratio is 1/4th for quadratic elements, and 1/9th for cubic. Although more integration points are required, the higher-order computation of (4) is actually cheaper. The extra cost comes from the implicit linear system which possesses a much larger bandwidth. For a regular 2-D mesh with six triangles connected to a given node, each line of the implicit matrix contains 7, 19, and 37 non-zero blocks, respectively for P1, P2, and P3 elements.

Preliminary quadratic and cubic element results obtained with the original stabilization and discontinuity capturing term used for linear elements, appeared too diffusive especially for MTC 2. This is an indication that the intrinsic time scale matrix  $\tau$  must be reduced for higher-order elements. Theoretical study of the 1-D scalar advection diffusion equation showed that the optimal  $\tau$  must indeed be reduced in the advective limit for any higher-order element. The shock capturing operator must also be tuned in a similar fashion.

In fact one term in the weighted residual must be specially treated in the context of higher-order elements. The last term in (5) vanishes to zero for linear elements. It appears in the second integral of (4). This term must be computed with higher-degree shape and test functions in order to preserve consistency. In practice, it is evaluated using an  $L_2$ -projection.

One-dimensional studies showed that there was no significant differences between SUPG and Galerkin/least-squares. We have chosen to concentrate solely on SUPG which is easier to implement.

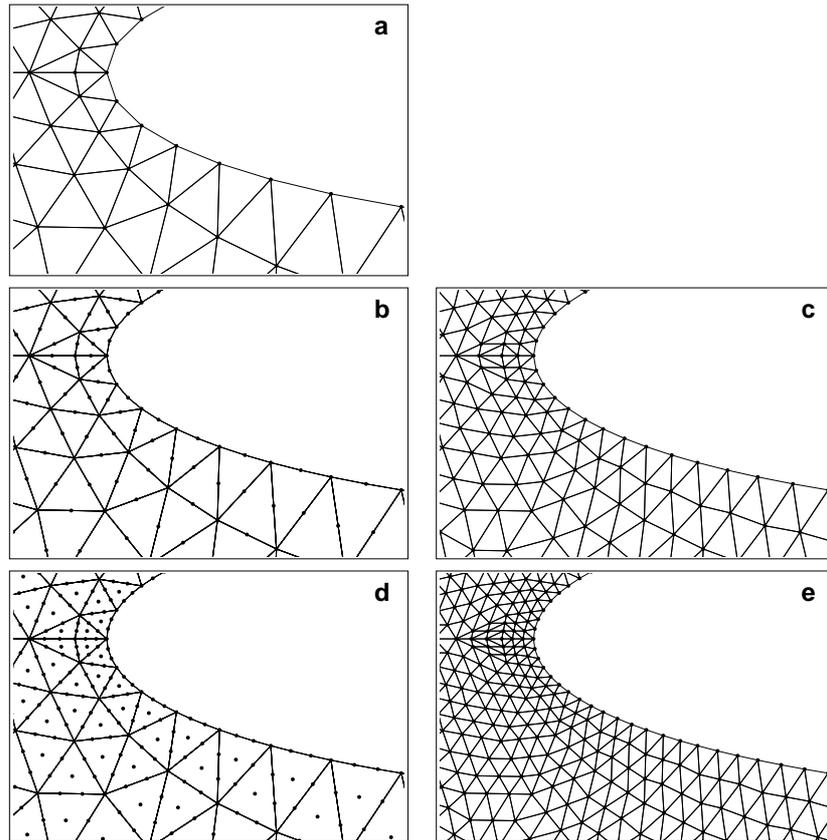
As a final remark, we want to stress the fact that whatever the order of the elements, all operations remain local (viz. at the element level). Consequently higher-order elements engender no implicitation nor parallelization issue (see [6]).

## 3 Isoparametric meshes with curved boundaries

We have made the seemingly obvious choice of higher-order **isoparametric** elements. One of the advantages of these elements, besides the higher-order shape functions, is the use of higher-order polynomials to represent curved boundaries.

They only ensure  $C^0$  continuity across elements, but locate all the nodes on the actual surface.

We had thought at first that the slope discontinuity across element boundaries could be minimized by adjusting the location of the extra nodes along the sides and the faces of elements beyond P1. In practice it is very easy to generate negative elements with “shamrock”-like edges if one tries to play with node location along edges to optimize curvature. Consequently we stuck in this study to elements with equally distributed nodes along the edges and faces.



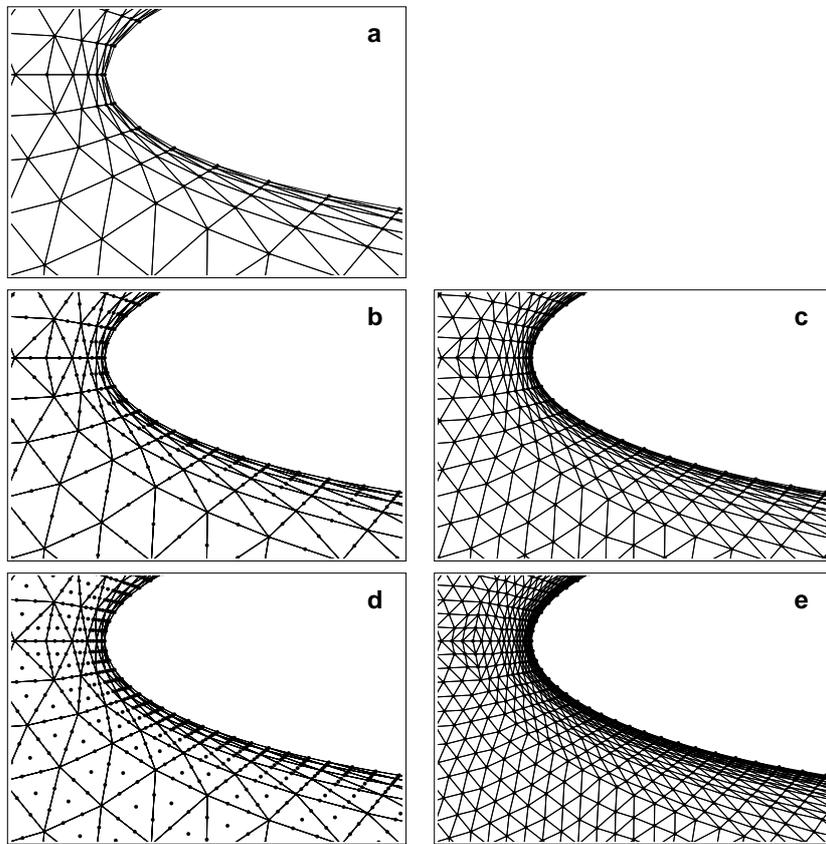
**Fig. 1** Higher-order mesh generation for inviscid test cases: original P1 mesh (a); P2 mesh and corresponding P1 mesh (b and c); P3 mesh and corresponding P1 mesh (d and e).

All higher-order meshes were obtained by adding nodes to a coarse initial P1 mesh. We had previously checked that quadratic and cubic triangular meshes would fit DASSAV data structure. Nested two-dimensional P1, P2, and P3 grids could be generated with equally distributed boundary nodes. Local node numbering was introduced into AETHER for cubic triangles, quadratic and cubic tetrahedra (linear

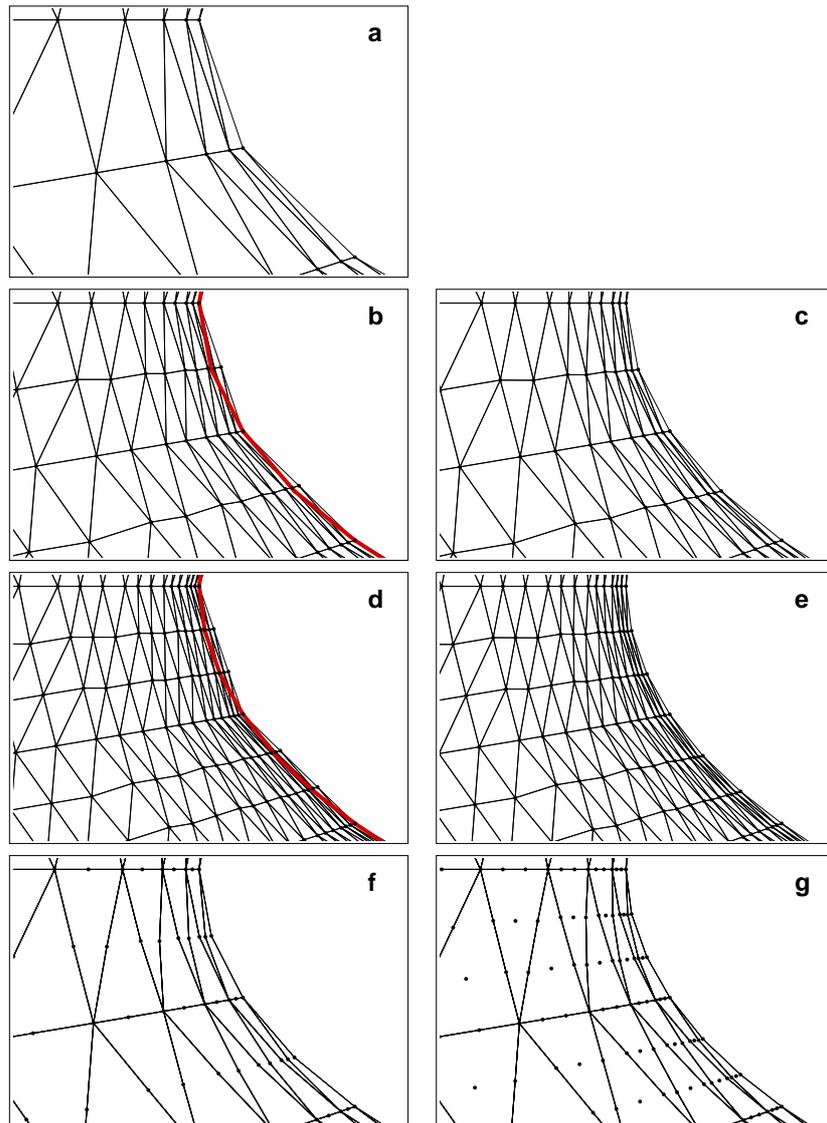
and quadratic triangles, and linear tetrahedra were preexisting). Corresponding face numbering for boundary elements was also introduced.

ARA provided sets of unstructured linear triangular grids for the Mandatory Test Cases. We had to generate new series of higher-order P1, P2, and P3 meshes. The first meshes used for the inviscid MTC's (1 and 2) are depicted in Figure 1.

All inviscid higher-order meshes were obtained by adding nodes to a coarse 1106-node P1 mesh (see Fig. 1 a). This yields a 4336-node P2 mesh and a 9690-node P3 mesh. The first P2 mesh is shown in Figure 1 b, whereas its P1 counterpart, which contains exactly the same number of grid points, is shown in Figure 1 c. Figures 1 d and e show the first cubic mesh and the matching linear grid. Four finer quadratic grids (up to 1,088,896 nodes) and two finer cubic grids (up to 775,386 nodes) were generated. All new nodes are added along the actual profile. This produces boundary elements with curved edges. Elements not connected to the boundary have straight edges.



**Fig. 2** Higher-order mesh generation for Navier-Stokes test cases: original P1 mesh (a); P2 mesh and corresponding P1 mesh (b and c); P3 mesh and corresponding P1 mesh (d and e).



**Fig. 3** Mesh deformation for isoparametric higher-order mesh generation for Navier-Stokes test cases: original P1 mesh (a); P1 iso-P2 mesh before (b) and after (c) deformation; P1 iso-P3 mesh before (d) and after (e) deformation; in (b) and (d) actual boundary represented with a bold red line; final P2 and P3 grids (f and g).

Navier-Stokes meshes with their stretched elements along the boundary bring a specific difficulty: extra nodes added along the boundary may produce negative elements. Figure 2 a presents the initial coarse 1533-node mesh which is the starting point of all grids generated for MTC 3. The corresponding 6034-node P2 and P1

iso-P2 grids, and 13,503-node P3 and P1 iso-P3 grids are shown respectively in Figures 2 b–e. Four finer quadratic grids (up to 1,521,184 nodes) and two finer cubic grids (up to 1,083,159 nodes) were generated.

Figure 3 depicts the mesh deformation technique used to generate stretched and curved higher-order elements close to the airfoil boundary for the Navier-Stokes cases. The initial grid is shown in Figure 3 a. P1 iso-P2 and P1 iso-P3 grids are constructed with straight edges (see Figs. 3 b and d). The bold red line represents the actual boundary. Figures 3 c and e presents the P1 meshes after deformation. The corresponding P2 and P3 grids are shown in Figures 3 f and g. Unlike the meshes built for the inviscid test cases, these meshes contains elements with curved faces in the volume away from the airfoil surface.

A series of P1, P2, and P3 meshes was also generated for MTC 5. The same mesh deformation technique used for MTC 3 grids was applied to obtain stretched and curved higher-order elements close to the airfoil boundary. Highly stretched elements are present close to the airfoil surface and in the wake with aspect ratios up to  $2 \times 10^6$ !

## 4 Numerical examples

Dassault Aviation computed four of the Mandatory Test Cases defined in Workpackage 2 of the ADIGMA Project. They cover a wide range of applications: from inviscid subsonic and transonic flows (MTC’s 1 and 2), to laminar Navier-Stokes (MTC 3), and finally a profile in transonic turbulent conditions (MTC 5). All four test cases were run with the baseline second-order version of Dassault Aviation’s industrial Navier-Stokes code AETHER and with the revisited or newly developed third and fourth order extensions.

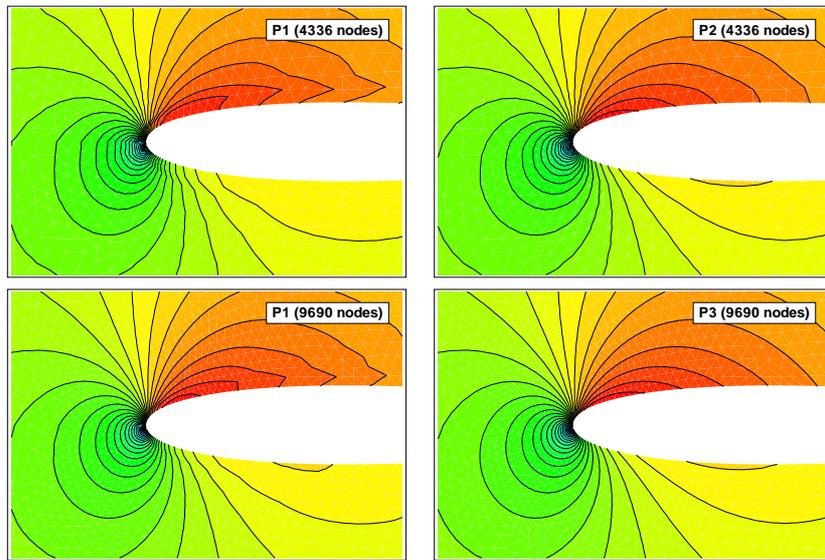
### 4.1 MTC 1: NACA0012, $M = 0.50$ , $\alpha = 2^\circ$ , *inviscid*

As an introductory comment, we should say that our code AETHER is really dedicated to Navier-Stokes applications. It can compute Euler flows but uses a strong slip boundary condition at the nodes with the true normals to the geometry. We impose a weak slip boundary condition at the trailing edge of airfoils and in regions where the definition of a single normal is tricky. A more natural way of imposing the inviscid slip condition in a finite element framework would be a weak condition through the boundary integral everywhere. Nevertheless inviscid test cases are valuable since they allow the assessment of the higher-order stabilization operator in the advection limit.

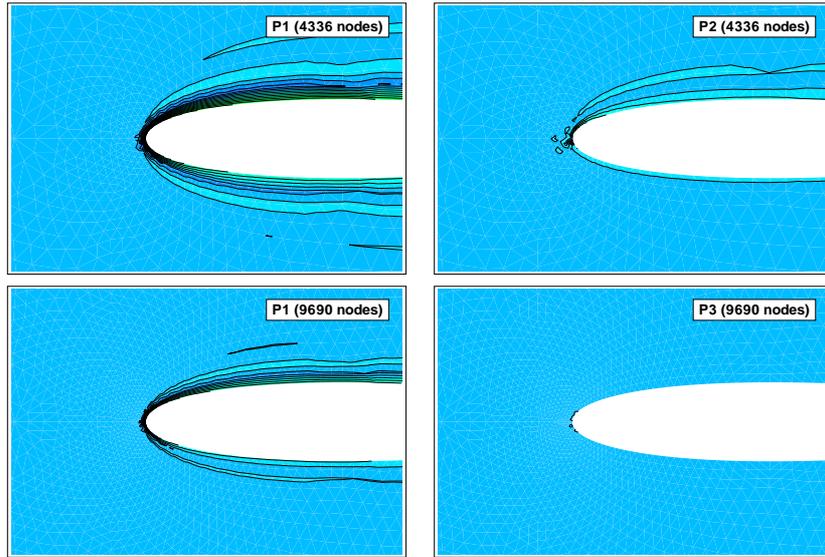
Higher-order MTC 1 results are compared with those obtained on the corresponding P1 mesh with the same number of nodes in Figures 4–5. They clearly show the advantage of the increased order of accuracy brought by quadratic and cubic ele-

ments. The entropy layer generated at the stagnation point is much reduced with quadratic elements and virtually disappears with cubic elements. This directly impacts the Mach number contours which traditionally present kinks near the wall on coarse P1 meshes. These kinks are removed from higher-order calculations, which also present much cleaner contours for the same number of degrees of freedom.

The kinks in Mach number contours observed in second-order solutions along the profile are not due to a lower degree of accuracy boundary condition or boundary integral computation as may have been suggested, but in fact to the level of spurious entropy generated at the leading edge. It is convected along the profile and affects the solution close to the airfoil. This fact will be confirmed in Chapter ??, where local mesh refinement in the sole leading edge region suppresses the spurious entropy production.



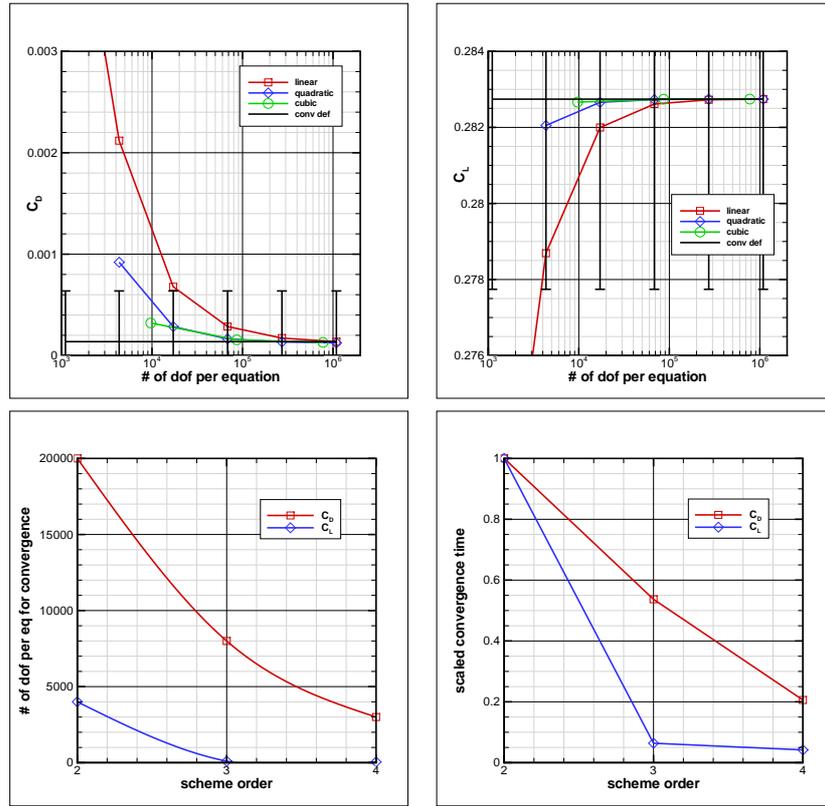
**Fig. 4** MTC 1: NACA0012,  $M = 0.50$ ,  $\alpha = 2^\circ$ , inviscid. Mach number contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.



**Fig. 5** MTC 1: NACA0012,  $M = 0.50$ ,  $\alpha = 2^\circ$ , inviscid. Entropy contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.

Figure 6 presents the convergence of the drag and lift coefficients with respect to the grid size given by its node number or “number of degrees of freedom per equation.” The error bars represent the convergence definitions provided for the test case: when a given coefficient reaches within the error bars, the solution is assumed converged for that particular coefficient.

We can notice a dramatic increase in convergence rate with the order of the scheme. Lift is converged for every tested higher-order mesh; drag requires more effort, and may still gain from an increase in scheme order beyond 4 as shown in the last plots of Figure 6. Even CPU time shows a gain with scheme order (note that a few higher-order values in these plots have been extrapolated). The times for convergence are scaled by the corresponding time for linear elements.



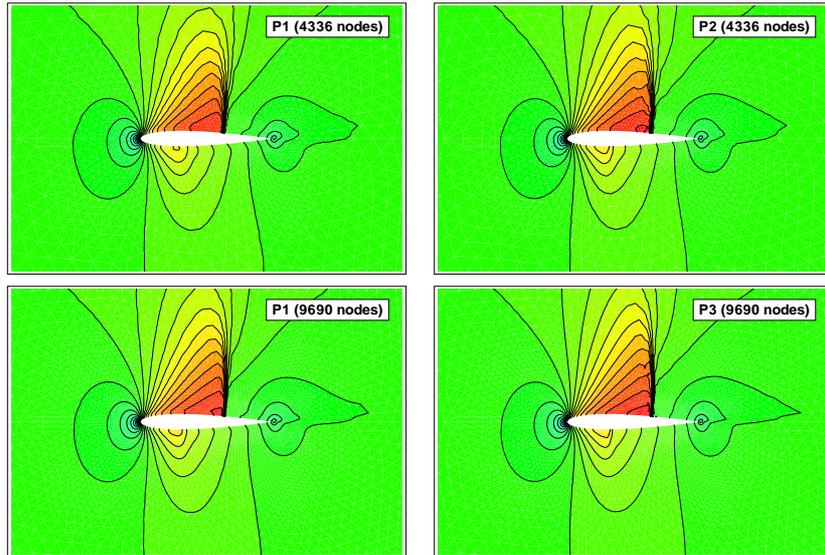
**Fig. 6** MTC 1: NACA0012,  $M = 0.50$ ,  $\alpha = 2^\circ$ , inviscid. Convergence of force coefficients for P1, P2, and P3 elements; estimated numbers of degrees of freedom and times for convergence.

#### 4.2 MTC 2: NACA0012, $M = 0.80$ , $\alpha = 1.25^\circ$ , inviscid

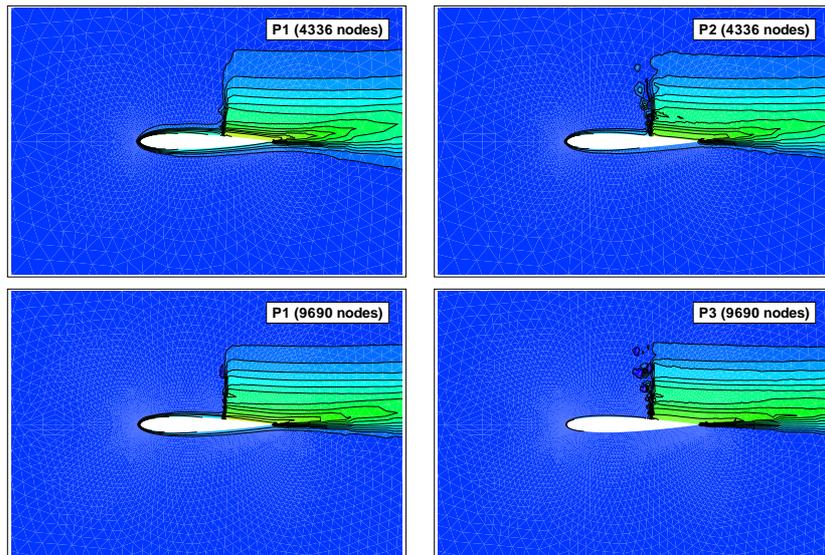
MTC 2 is a transonic inviscid test case. It is interesting in its own respect, since it can challenge the ability of higher-order elements to treat shocks with the help of the discontinuity capturing operator.

Figure 7 shows Mach number contours on the same set of meshes used for MTC 1. In spite of the presence of the shock wave, no obvious degradation in the solution quality can be observed. P3 elements even produce the best result with a well resolved slip line and a captured windward-side weak shock.

Entropy contours displayed in Figure 8 show a reduction in the production of spurious leading-edge entropy similar to MTC 1. However the entropy rise through the normal shock does not look as controlled with higher-order elements. The perturba-

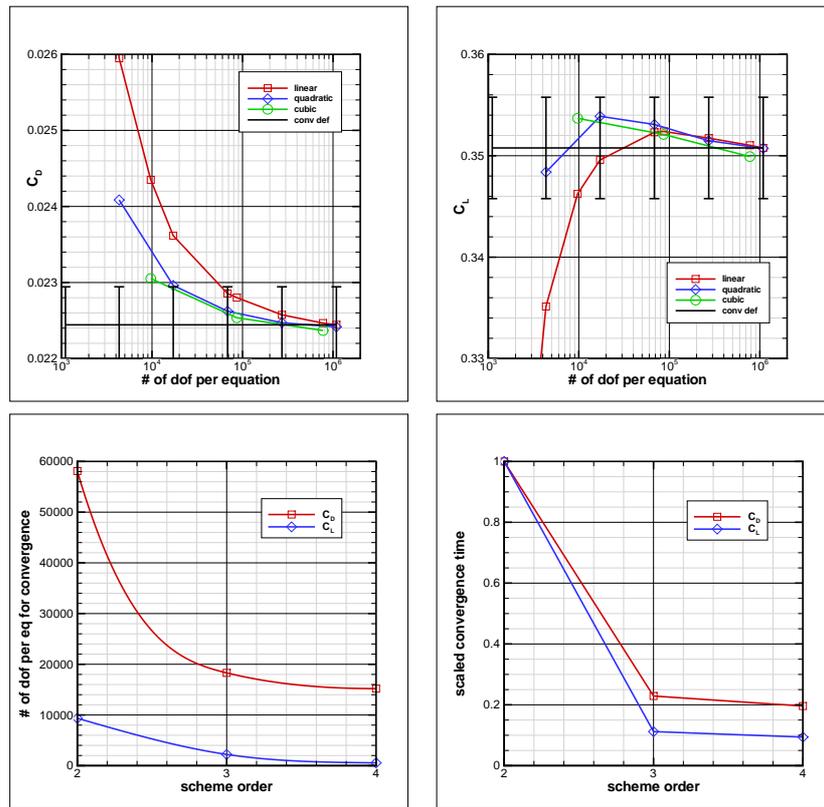


**Fig. 7** MTC 2: NACA0012,  $M = 0.80$ ,  $\alpha = 1.25^\circ$ , inviscid. Mach number contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.



**Fig. 8** MTC 2: NACA0012,  $M = 0.80$ ,  $\alpha = 1.25^\circ$ , inviscid. Entropy contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.

tions remain local though, thanks to the SUPG operator. Note that all these contours are plotted on P1 meshes. Actual higher-order contours might be smoother.



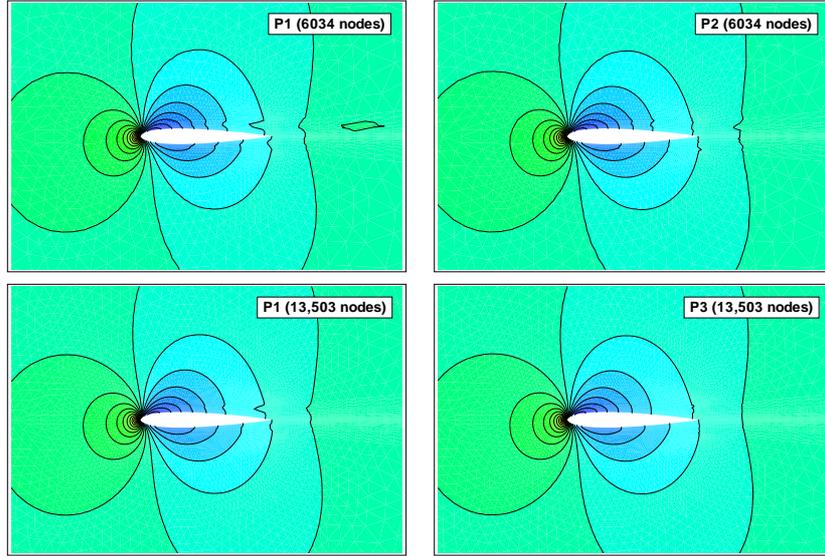
**Fig. 9** MTC 2: NACA0012,  $M = 0.80$ ,  $\alpha = 1.25^\circ$ , inviscid. Convergence of force coefficients for P1, P2, and P3 elements; estimated numbers of degrees of freedom and times for convergence.

Figure 9 presents the convergence of the drag and lift coefficients. As with MTC 1, all higher-order meshes display a converged lift coefficient, whereas drag requires more mesh points. The last two plots in Fig. 9 indicate that most of the gain is obtained with third order elements. On the average, CPU time to convergence is reduced by 80%.

### 4.3 MTC 3: NACA0012, $M = 0.50$ , $\alpha = 2^\circ$ , $Re = 5,000$

We now come to MTC 3, one of the most interesting test cases in the selection. It concerns the laminar computation of an airfoil. Although a Navier-Stokes test case, it is still far from concrete industrial applications. We will see however that it exemplifies the difficulty of getting converged Navier-Stokes solutions. One can anticipate an even greater challenge with complex 3-D RANS computations. . .

Figure 10 presents pressure contours obtained on the coarsest quadratic and cubic meshes. They are compared with results computed on corresponding linear meshes containing the very same numbers of grid points. P1 results show the difficulty of preserving a constant pressure through an underresolved boundary layer and highly stretched elements. This difficulty is alleviated with the increasing order of the elements.



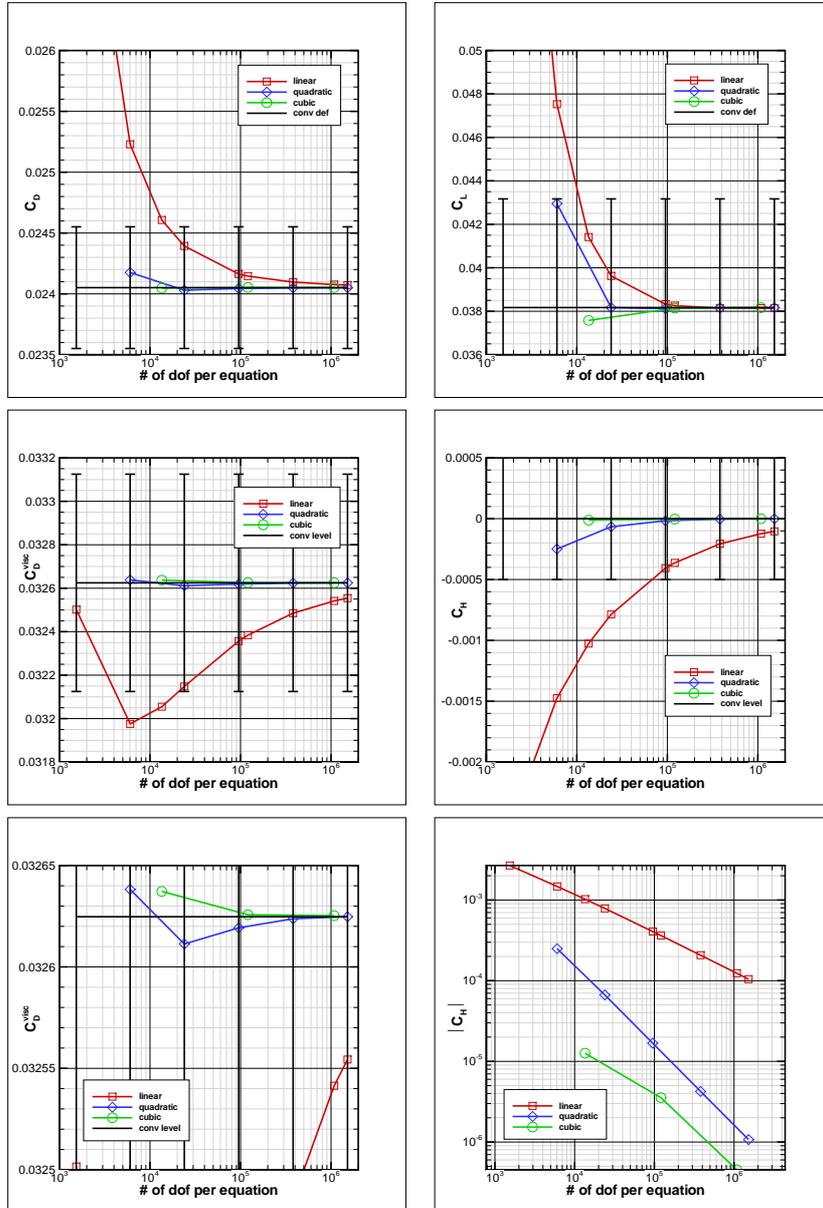
**Fig. 10** MTC 3: NACA0012,  $M = 0.50$ ,  $\alpha = 2^\circ$ ,  $Re = 5,000$ . Pressure contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.

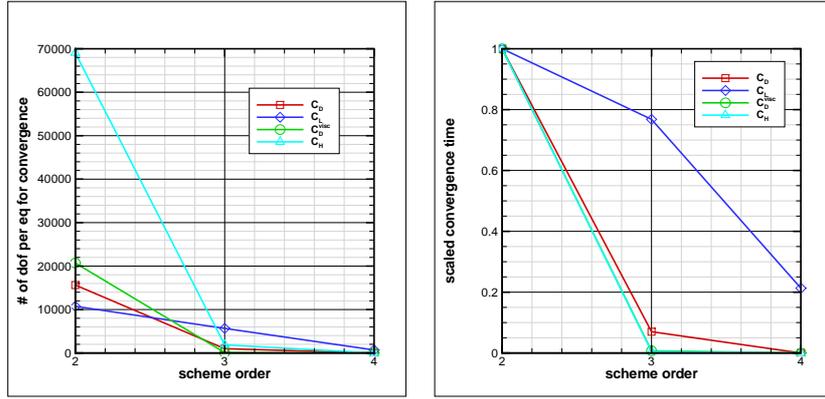
Figure 11 presents the convergence of force coefficients: pressure drag and lift, friction drag, and heat flux. The advantage of higher-order elements is even more blatant than for the inviscid test cases described previously. Pressure drag and lift converge faster with quadratic elements; cubic elements yield values close to the asymptotic limit for every computed grid, even the coarser ones.

Unexpectedly viscous fluxes appear as a real challenge for this laminar test case. Second order viscous drag is still not converged for the finest mesh which contains over 1.5 million nodes: the asymptotic value is provided by the quadratic results. The magnified plot is even more striking: linear elements have a hard time getting within one drag count of the asymptotic value of the friction drag, whereas as all higher-order results are within half of the same margin. Heat flux convergence plotted in log scale shows the substantial advantage of higher-order elements. The error in heat flux (which should be zero for an adiabatic wall condition) can be reduced by several orders of magnitude.

The number of nodes and the CPU time for convergence are again reduced with the order of the scheme used. Quadratic elements bring most of the reduction, except

for lift which seems to converge at a slower rate and may benefit from an element order beyond 3.





**Fig. 11** MTC 3: NACA0012,  $M = 0.50$ ,  $\alpha = 2^\circ$ ,  $Re = 5,000$ . Convergence of force and heat flux coefficients for P1, P2, and P3 elements; estimated numbers of degrees of freedom and times for convergence.

Regarding CPU cost and memory requirements, we can be more specific for this particular test case. For the same number of degrees of freedom, the extra cost of P2 elements over P1 is only 30%; P3 elements are 2 to 2.5 times as expensive as P1 elements. The overhead due to the  $L_2$  projection can be reduced. The CPU cost increase is overtaken by the drastic reduction in the number of nodes required for convergence. Consequently the CPU time for convergence decreases with the degree of the scheme. Memory requirements are mostly due to the implicit Jacobian matrix. They respectively gain 30% and 70% for quadratic and cubic elements.

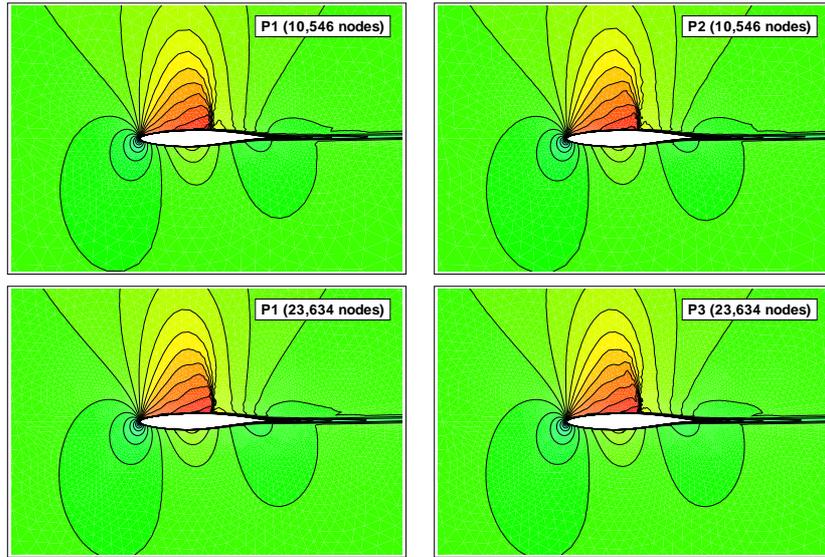
#### 4.4 MTC 5: RAE2822, $M = 0.734$ , $\alpha = 2.79^\circ$ , $Re = 6,500,000$

The final test case deals with a transonic high Reynolds number RANS problem.

In the numerical method described in Section 2.1, the turbulence equations are solved in a staggered manner, with a second-order residual distribution scheme, and are weakly coupled to the Navier-Stokes field through the turbulent viscosity  $\mu_t$ .

As a first step, for higher-order calculations, RANS turbulent equations are solved on an underlying P1 mesh, and thus remain second order accurate. These first results show the robustness of the SUPG finite element method. As for the more elementary MTC's (1, 2, and 3), the convergence of quadratic and cubic elements is similar to that obtained for linear elements with the same CFL settings. High aspect ratios (up to  $2 \times 10^6$  in the considered set of meshes) do not seem to be an issue.

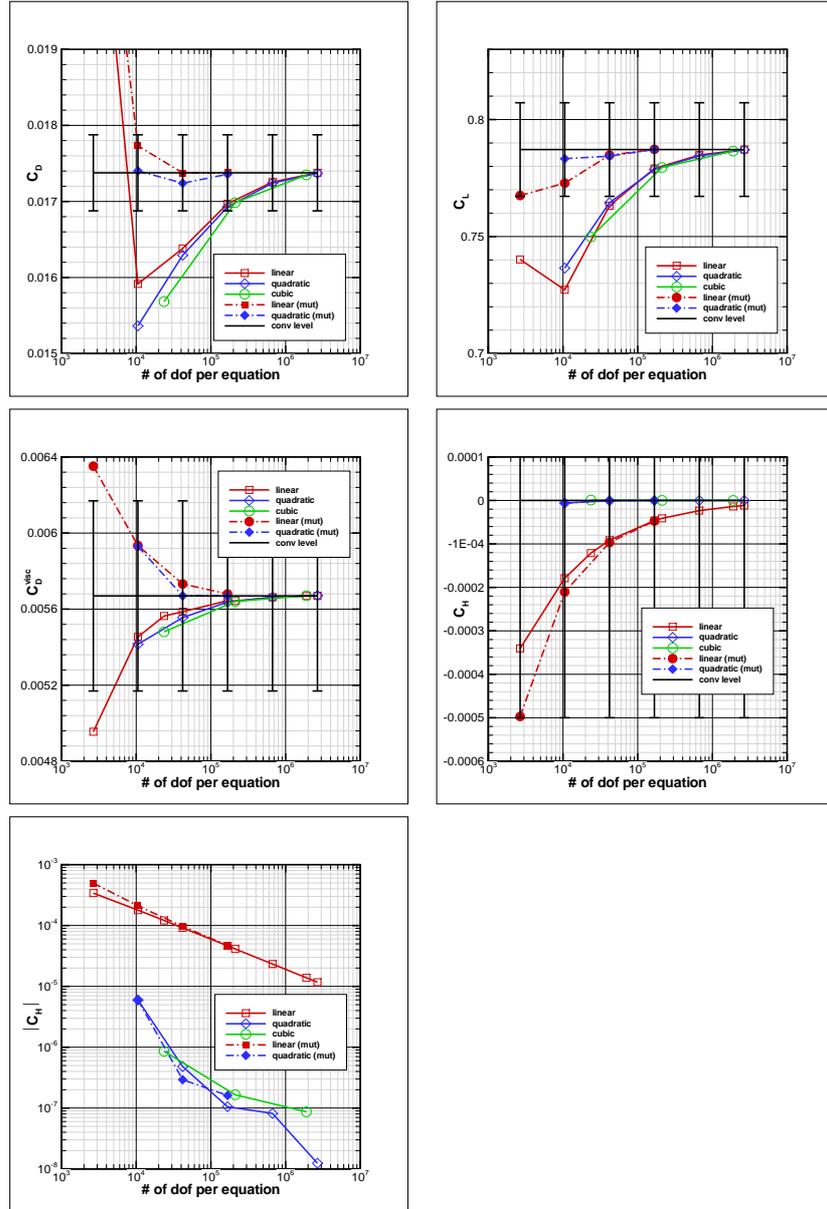
Figure 12 presents Mach number contours obtained with P1, P2, and P3 elements on matching grids. On these fairly coarse meshes, it's hard to see any difference between the solutions.



**Fig. 12** MTC 5: RAE2822,  $M = 0.734$ ,  $\alpha = 2.79^\circ$ ,  $Re = 6,500,000$ . Mach number contours on matching P1 iso-P2 and P2 grids, and P1 iso-P3 and P3 grids.

The force coefficient convergence plots are gathered in Figure 13. The open symbol curves represent the second-, third-, and fourth-order methods described above (with a second order turbulence model). There is no real distinction between the three schemes. They converge at the same rate toward the same asymptotic values. Nevertheless heat flux shows once more an indisputable advantage of higher-order elements over linear ones. The error is smaller by as much as three orders of magnitude. There is no additional benefit brought by cubic elements though.

In an attempt to simulate a “higher-order” turbulence model, we used the interpolation of the  $\mu_t$  field computed on the finest P1 mesh (2,669,536 nodes). The outcome of this test is indicated in the different convergence plots of Fig. 13 with filled symbols. We have only tested linear and quadratic elements. Results show that the turbulence model has a huge impact on the convergence of force coefficients. Quadratic elements have a slight edge over linear elements, especially for the coarsest meshes. Heat flux convergence is unaffected. This demonstrates the need for a higher-order turbulence model to fully exploit in RANS computations the benefit of higher-order elements observed in inviscid and laminar test cases.



**Fig. 13** MTC 5: RAE2822,  $M = 0.734$ ,  $\alpha = 2.79^\circ$ ,  $Re = 6,500,000$ . Convergence of force and heat flux coefficients for P1, P2, and P3 elements.

#### ***4.5 Concluding remarks on numerical test cases***

In this study, many firsts have been accomplished:

- the implementation of higher-order (quadratic and cubic) stabilized finite elements for compressible flows in an industrial code;
- the systematic convergence study of increasingly difficult test cases: inviscid, transonic, laminar, and turbulent flows;
- the proof that higher-order convergence can be achieved at a reasonable cost;
- the demonstration that higher-order elements are robust: same CFL rules were applied in our simulations with convergences similar to linear elements and sometimes significantly better;
- the verification that higher-order elements bring no particular complications in terms of implicitation nor parallel efficiency.

Difficulties were encountered with the RANS test cases. We believe they can be palliated with a stronger higher-order coupling between the Navier-Stokes solver and the turbulence model, or the use of a genuine higher-order scheme for solving the turbulence equations.

### **5 Towards industrial applications**

As a conclusion we'll comment on the transition towards industrial applications. The extension to 3-D is readily available. To make it industrially viable, one needs a dedicated way to generate higher-order meshes. Enriching P1 meshes yields way too fine higher-order mesh sets in 2-D. This is even more true in 3-D.

The cost of higher-order elements is reasonable (at most a factor of 2 for P3 with the same number of dof's), and it can be worked upon.

Higher-order elements can handle high aspect ratios and same CFL's as the standard second-order scheme with convergences often better than with linear elements. They engender no implicit, nor parallel issue, which is mandatory for industrial applications.

The coupling with RANS turbulence model must be improved. In the mean time, higher-order elements might show a unique potential for Large Eddy Simulations.

## References

1. A.N. Brooks and T.T.R. Hughes. Streamline Upwind Petrov Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, Vol. 32, pp 199–259, 1982.
2. F. Chalot, T.J.R. Hughes, Z. Johan, and F. Shakib, Application of the Galerkin/least-squares formulation to the analysis of hypersonic flows. I: Flow over a two-dimensional ramp. *Hypersonic Flows for Reentry Problems. Volume II: Test Cases – Experiments and Computations*. Proceedings of a Workshop held in Antibes, France, 22–25 January 1990, Springer Verlag, pp 181–200, 1991.
3. F. Chalot, T.J.R. Hughes, and F. Shakib. Symmetrization of conservation laws with entropy for high-temperature hypersonic computations. *Computing Systems in Engineering*. vol. 1, pp. 465–521, 1990.
4. F. Chalot and T.J.R. Hughes. A consistent equilibrium chemistry algorithm for hypersonic flows. *Computer Methods in Applied Mechanics and Engineering*. vol. 112, pp. 25–40, 1994.
5. F. Chalot, M. Mallet, and M. Ravachol. A comprehensive finite element Navier-Stokes solver for low- and high-speed aircraft design. Paper #94-0814. *AIAA 32nd Aerospace Sciences Meeting*. Reno, NV, January 10–13, 1994.
6. F. Chalot, Q.V. Dinh, M. Mallet, A. Naïm, and M. Ravachol. A multi-platform shared- or distributed-memory Navier-Stokes code. *Parallel CFD '97*. Manchester, UK, May 19–21, 1997.
7. F. Chalot, B. Marquez, M. Ravachol, F. Ducros, F. Nicoud, and T. Poinsot. A consistent Finite Element approach to Large Eddy Simulation. Paper #98-2652. *AIAA 29th Fluid Dynamics Conference*. Albuquerque, NM, June 15-18, 1998.
8. F. Chalot, B. Marquez, M. Ravachol, F. Ducros, and T. Poinsot. Large Eddy Simulation of a compressible mixing layer: study of the mixing enhancement. Paper #99-3358. *AIAA 14th Computational Fluid Dynamics Conference*. Norfolk, VA, June 28-July 1, 1999.
9. F. Chalot, Industrial aerodynamics. *Encyclopedia of Computational Mechanics*, Vol. 3, Computational Fluid Dynamics, chapter 12, E. Stein, R. de Borst, and T.J.R. Hughes editors, Wiley, 2004.
10. F. Chalot, V. Levasseur, M. Mallet, G. Petit, and N. Réau. LES and DES simulations for aircraft design. Paper #2007-0723. *45th AIAA Aerospace Sciences Meeting and Exhibit*. Reno, NV, January 8–11, 2007.
11. T.J.R. Hughes and M. Mallet. A new finite element formulation for computational fluid dynamics: IV A discontinuity-capturing operator for multidimensional advective - diffusive systems. *Comp. Meth. in Applied Mech. and Eng.* Vol. 58, pp. 329–336, 1986.
12. F. Shakib, T.J.R. Hughes, and Z. Johan. A multi-element group preconditioned GMRES algorithm for nonsymmetric systems arising in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*. Vol. 75, pp 415-456, 1989.
13. F. Shakib, T.J.R. Hughes, and Z. Johan. A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*. Vol. 89, pp 141–219, 1991.