HIGH VELOCITY FLOW SIMULATIONS FOR THE DESIGN OF AIRCRAFT AND SPACE VEHICLES: FINITE ELEMENT METHODS AND APPLICATIONS.

F.Chalot, Q.V.Dinh, M.P.Leclercq, M.Mallet, B.Mantel,

J.Periaux, P.Perrier, M.Ravachol, P.Rostand, B.Stoufflet

Dassault Aviation, 78 Quai M. Dassault - 92214 Saint-Cloud (FRANCE)

Abstract

In this paper a review of the CFD methodologies and numerical tools developped in the Aerodynamic Department of Dassault Aviation for the design of aircraft and space vehicles is presented.

This paper covers the description of three solvers: full potential, Euler and Navier Stokes possibly with chemistry or turbulence coupling. The accuracy of these pieces of software relies on unstructured finite element meshes and their efficiency on implicit time integration.

The complexity of the shapes and also of the flow structures requires adapted unstructured meshes to capture the main features of the solution using strategies based on physical or mathematical criteria.

A multizone approach is used in an industrial environment to compute at reasonable cost the detail of the flow field around an aircraft or a space vehicle. Inviscid computations with boundary layer corrections are performed on global meshes surrounding the whole aircraft or space plane to predict aerodynamic coefficients while estimation of heat fluxes is achieved via the solution of the Navier Stokes equations on local meshes.

The role of databases such as the European Hypersonic Data Base is described as a systematic tool for analysis and comparisons, which contributes significantly to the code validation process and the evaluation of uncertainty margins in Aerospace Industry.

Three dimensional computations of flows around complete civil and military aircraft such as the Falcon Jet and Mirage 2000, and also space vehicles such as the Hermes Space Plane are presented to illustrate the possibilities of the above methodologies.

Significant improvements of these finite element solvers are expected on massively parallel architectures using novel ingredients based on domain decomposition or fictitious domain techniques. Recent numerical experiments on store separation for the simulation of 3-D unsteady potential flows around rigid moving bodies are also discussed.

1. Introduction

The prediction of compressible flows over complex geometries necessitates mastering efficient computational codes that incorporate various physical models (e.g., chemically reacting flows, turbulent flows, etc.) and various related advanced numerical techniques.

At the end of the 70' the aerospace industry was experimenting successfully the first 3-D potential transonic solvers for the design of aircraft vehicles ([1]) and in the early 80' a considerable amount of theoritical work was conducted to solve the Euler ([2]) and Navier-Stokes ([3]) equations. A decade after, these theoretical studies provided efficient Euler and Navier stokes solvers for transonic, supersonic and also hypersonic flow simulations with unstructured finite element meshes. In this context and in close cooperation with INRIA ([4]) and Stanford University ([5]) Dassault Aviation developed techniques relying on unstructured meshes and using an implicit upwind formulation combined with adaptive mesh refinements. One important aspect in the organization of this type of calculation is the coordination of global and local simulations. Local simulations can account for rather complex and costly models (Navier Stokes with turbulence or chemistry models) while global simulations are limited to the simpler inviscid models (potential or Euler with chemistry models).

We present in this paper the multi-zone and multi-code strategies used on a routine basis to compute at reasonable cost all the needed details of the flow field (pressures forces and heat transfer) around aircrafts and space vehicles in an industrial environment. Methodologies including mathemat-

ical modeling and numerical methods used in the solvers are described in Section 2. Mesh adaption procedures based on the gradients of physical quantities or on a posteriori errors estimates are presented in Section 3. The role of database as an interactive validation tool is explained in Section 4. and computational results on complete aircrafts and space vehicles or local configurations are presented in Section 5.

2. Methodologies: mathematical modelling and finite element solutions

2.1-Potential flow

2.1.1-The flow solver

Let ω be a bounded two-dimensional domain and $\rho, \vec{u}, \phi$ the density, velocity and velocity potential ($\vec{u} = \vec{\nabla}\phi$). For irrotational flows, mass conservation reads :

$$\rho_t + \vec{\nabla} \cdot \left(\rho \vec{\nabla}\phi\right) = 0 \quad \text{in} \quad \omega \qquad (1)$$

where $(.)_t$ denotes time derivative. Bernoulli's law for unsteady potential flows gives the density ρ as a function of the potential ϕ :

$$\rho = \left[1 - \frac{\gamma - 1}{2} M_\infty^2 (2\phi_t + |\vec{\nabla}\phi|^2 - 1)\right]^{\frac{1}{\gamma - 1}} \qquad (2)$$

In (2), γ is the ratio of specific heats (1.4 for air) and M_∞ designates the freestream Mach number.

Boundary conditions, usually mass fluxes, are prescribed at the boundary $\partial\omega$ of the flow domain ω. Typically, $\partial\omega$ includes the outer limit (γ_o), solid bodies surfaces (γ_b) and wake cuts for lifting airfoils (γ_c). For flows with moving bodies, ω must be time-dependent.

2.1.2-A finite element discretization

Let $V_h(\omega)$ be the standard finite element space of piecewise linear functions on a triangulation $T_h(\omega)$ of ω. We use first-order time discretisation with time step $\Delta t > 0$ and derive a Galerkin *weak* formulation of (1).

Boundary conditions are handled using approximate Riemann solvers restricted to potential flows.

We use an extension an Osher-type mass flux biaising procedure in a finite element framework to capture shocks and supersonic pockets. In the presence of moving bodies, deformations of ω should be taken into account : this is usually done through a *Lagrangian* step where deformations of the boundary $\partial\omega$ yield remeshing and subsequent projection of the solution field onto the new mesh. Another point is that the boundary flux function should depend on the solid bodies'velocities.

2.1.3-A cartesian grid finite element discretisation

We would like now to propose a way to circumvent the remeshing task. The idea is to have a fixed mesh by taking into account the motion of γ_b and γ_c through a projection operator. It goes back to the notion of *fictitious-domain* methods ([6], [7]) which has seen lately some interesting developments ([8]).

We define a bounded domain Ω which forms an embedding of the flow domain ω. A regular cartesian grid is set up in Ω with its companion triangulation $T_h(\Omega)$.

Simple linear extensions are used for $\tilde{r}$ except for wake cuts where nodes are artificially doubled so that we can still use approximate Riemann solvers to compute boundary fluxes.

Newton's method can still be applied : numerically, the linearized operator is better conditioned due to an overall better element aspect ratio.

More interestingly, the motion of solid bodies can be taken into account in the computation of the characteristic function χ_ω and the boundary flux function: no remeshing is needed !

2.1.4-Mesh management

In these *fictitious-domain* methods, we have a trade-off (as in most Computational Fluid Dynamics techniques !) between solution accuracy and grid flexibility. In the framework of unstructured meshes, it seems natural for us to use adaptive mesh refinement to regain some accuracy : in fact, we have replaced the notion of *body-fitted* meshes by the notion of *body-adapted* meshes.

Starting with our regular cartesian grid, we use hierarchical data data structures developed for unstructured meshes in the PLTMG finite element package ([9]).

At each time iteration, to compute the characteristic function χ_ω, intersections between the bodies'shapes, defined by Ferguson splines, and the mesh should be done very carefully. This mesh is usually a result of different refinement and derefinement steps during the motion of bodies. To avoid highly skewed elements, smooth variations are insured between the different mesh levels.

Mesh refinement/unrefinement and smooth mesh variations are activated in the PLTMG package via the definition of an *error* criteria. In our present work, this criteria is purely geometrical.

2.2-Euler with chemistry

2.2.1-Lagrange-Galerkin formulation

The following model convective/diffusive equation (or system of equations as Navier-Stokes model) is considered:

$$R(U) = U_t + \nabla \cdot F(U) - \nabla \cdot (K \nabla U) = 0 \quad (3)$$

Let T_h be a triangulation of the computational domain $\omega \subset I\!R^2$ with boundary $\partial\omega$ of unit normal $\vec{\nu}_{\partial\omega}$. We denote by T a current element, in case of a triangle by $\vec{n}_i^T$ the inward integrated normal opposite to node N_i , $K(i)$ the set of neiboughring nodes of node N_i and $supp(i)$ the support of the basis function ϕ_i associated to node N_i. Let V_h^d be a set of piecewise polynomial functions from $I\!R^2$ with values in $I\!R^d$ that are continuous. Further, the basis of V_h^d is the set of functions ϕ_j satisfying the Lagrange interpolation conditions. Let us consider the following abstract family of schemes for the spatial approximation of the hyperbolic system:

$$\int_\omega U_t \phi \, dv + \int_\omega \phi \nabla \cdot F(U) dv + \int_\omega K \nabla U \nabla \phi dv = 0 \quad (4)$$

where $F(U)$ is considered as an element of V_h^d (group representation) and ϕ is a test function.

2.2.2-Finite Volume Galerkin

Lagrange-Galerkin methods may be interpreted as finite-volume schemes in some extended sense; indeed the divergence operator can be written as (boundary terms excluded):

$$\int_\omega \phi_i \nabla \cdot F(U) = \sum_{j \in K(i)} \Phi^{centered}(U_i, U_j, \vec{\eta}_{ij}) \quad (5)$$

where
$\vec{\eta}_{ij} = \int_{supp(i) \bigcap supp(j)} (\phi_i \vec{\nabla} \phi_j - \phi_j \vec{\nabla} \phi_i) dv.$

Upwinding can then be introduced by replacing the centered flux in considering a Riemann problem with U_i and U_j as left and right states and η_{ij} defining the interface normal. Extensions of previous schemes to second-order accuracy are performed through MUSCL Finite Element interpolations. An improved formulation with a modified definition of control cells has been proposed in [10].

2.2.3-Generalized Flux Vector Splitting

We have shown that a Lagrange-Galerkin approximation of a general hyperbolic system can be interpreted consistently in other formulations which lead to different integration schemes. Combination of above formulations applied to an adequate splitting of the Euler equations of gas dynamics are investigated.

In that direction, an attractive approach is to investigate schemes based on a separation of convective and pressure fluxes as mentionned in the introduction. The convective part can be discretized by either distributive or SUPG schemes which have truely multidimensional property whereas the pressure part can be treated by Finite Volume Galerkin techniques. An early study proves to be promising.

2.3-Navier-Stokes with chemistry turbulence model

2.3.1-Governing equations

Let ρ, $\vec{u}$, and E be respectively the density, the velocity, and the total energy per unit mass of fluid. The Navier-Stokes equations read:

- conservation of mass

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0, \quad (6)$$

- Newton's second law

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u}) = \nabla \cdot \sigma, \quad (7)$$

- conservation of energy

$$\frac{\partial \rho E}{\partial t} + \vec{\nabla} \cdot (\rho E \vec{u}) = \nabla \cdot (\sigma \vec{u}) - \vec{\nabla} \cdot \vec{q}, \quad (8)$$

where σ is the Cauchy stress tensor and $\vec{q}$ is the heat-flux vector. This set of partial differential equations is subsidized with appropriate constitutive relations and state equations which we will describe in the following sections. We can notice that up to this point no particular alteration to the usual Navier-Stokes equations is made to accommodate for turbulence and/or high-temperature effects. Thus the strategy developed for the solution of laminar flows of a perfect gas can be formally applied. The numerical method used is discussed in the next section.

2.3.2-The Galerkin/least-squares formulation

This formulation has developed into a general approach for a wide class of problems. The basic idea can be understood by considering the steady scalar advection-diffusion model problem:

$$\mathcal{L}u = \vec{a} \cdot \vec{\nabla} u - \vec{\nabla} \cdot K \vec{\nabla} u = o.$$

where $\vec{a}$ and K are constant parameters. For simplicity we assume that u vanishes on the boundary. The Galerkin method is defined as:

Find $u^h \in V^h$ *such that for all* $w^h \in V^h$, $B(w^h, u^h) = 0$ *where*

$$B(w^h, u^h) = \int_\omega (w^h \vec{a} \cdot \vec{\nabla} u^h + \vec{\nabla} w^h \cdot K \vec{\nabla} u^h) d\omega$$

The Galerkin/least-squares method can be defined by the following variational equation:

$$B(w^h, u^h) + \sum_e \int_{\omega^e} \mathcal{L} w^h \tau \mathcal{L} u^h d\omega^e = 0$$

The additional term is the sum of integrals over element interior (a finite element discretization of the domain is assumed). It adds stability to the Galerkin formulation without upsetting the consistency of the method.

For the multidimensional case, the numerical diffusion is characterized by the diffusivity matrix $K^{\text{num}} = \vec{a}\ \tau\ \vec{a}^T$ where $\tau = \frac{h}{2} \frac{f(Pe)}{|\vec{a}|}$ and $f(Pe) = \coth(Pe) - 1/Pe$ is a doubly asymptotic function of the element Peclet number ($Pe = |\vec{a}|h/2|\vec{K}|$) going to zero when diffusion dominates and to one when advection dominates.

The doubly asymptotic behavior is present in each mode in the numerical diffusion. This ingredient of the method is critical in establishing the convergence results presented in Hughes, Franca and Mallet ([11]) for linear systems of advection-diffusion equations. The formulation can be applied to the compressible Navier-Stokes equations which can be written in the form of a symmetric advective-diffusive system in terms of entropy variables, as we will see shortly.

2.3.3-Entropy variables

We define the generalized entropy function $\mathcal{H}$ by $\mathcal{H} = \mathcal{H}(\vec{U}) = -\rho s$, where s is the physical entropy per unit mass. $\mathcal{H}$ is a strictly convex function of the vector of conservative variables,

$$\vec{U}^t = \frac{1}{v}(1,\ \vec{u},\ e + |\vec{u}|^2/2)$$

where $v = 1/\rho$ is the specific volume. Consequently, the relation $\vec{V}^T = \partial\mathcal{H}/\partial\vec{U}$ constitutes a legitimate change of variables. $\vec{V}$ is referred to as the vector of *(physical) entropy variables.* They were originally derived with the perfect gas case in mind, but were recently extended to take chemistry and high-temperature effects into account ([12]).

2.3.4- Discontinuity capturing operator

Although the Galerkin/least-squares method is a stable method, oscillations may occur in the vicinity of strong gradients. A nonlinear discontinuity-capturing operator based on the gradient in the element local coordinate system is added to the formulation

2.3.5- Implicit iterative time-marching algorithm

Convergence to steady-state of the compressible Navier-Stokes equations is achieved through an implicit iterative time-marching algorithm. At each discrete time t_n, the finite element discretization leads to a system of nonlinear equations. This system is solved by performing a linearization through a truncated Taylor series expansion. $\vec{R}$ is the residual of the nonlinear problem and $\vec{J}$ is the consistent Jacobian associated with $\vec{R}$. The consistent Jacobian is often replaced by a Jacobian-like matrix $\mathcal{J}$ leading to a more stable time-marching algorithm. The system of equations $\mathcal{J}\vec{p} = -\vec{R}$ is preconditioned by a nodal block-diagonal preconditioner and solved using the Generalized Minimal RESidual (GMRES) algorithm

2.3.6 Application to turbulent flows

The partial differential equations used to describe the mean flow field are the mass-averaged Navier-Stokes equations of a compressible fluid. The closure of the Reynolds stress tensor and heat flux is obtained using a classical Boussinesq hypothesis and the concept of eddy viscosity. The eddy viscosity is computed through a two-equation turbulence model, thus the Navier-Stokes equations are augmented by two additional partial differential equations for the turbulence quantities.

The turbulence models used belong to the $k-\epsilon$ family. The extra equations needed are convection-diffusion equations mainly coupled through their source terms. The turbulence quantities are positive, the enforcement of this constraint can be achieved through the use of a monotone discrete advective operator and a special time discretization for the source term ([13]).

The discretized mean flow equations and the turbulence equations are integrated using a splitting method . At a current time step, we solve the Navier-Stokes equations using turbulence data evaluated at the previous time while the turbulence equations are solved using the flow variables computed at the previous time.

In the near wall regions, the effects due to the molecular viscosity have to be accounted for. We use a two-layer model to do so. This model has been introduced by Chen and Patel and modifies the k equation in the near wall region while the ϵ equation is replaced by an algebraic definition of

the dissipation ϵ, away from the wall the standard $k-\epsilon$ equations are conserved. Since the Navier-Stokes equations are integrated down to the wall, separation can be accurately computed.

3. Mesh adaptation with physical and mathematical criteria

3.1- Anisotropic mesh adaptation

The mesh adaptation process discussed below is guided by a control space. This contains information about the size and the shape of the expected elements depending on their location.

3.1.2-General background

Let Ω be the bounded domain of $\mathbb{R}^2$ or $\mathbb{R}^3$ we want to triangulate. According to P.L. George and F.Hecht [INRIA Rocq.], we define a control space as follows:

Definition: $(\mathcal{O},\ H)$ is a control space for the domain Ω under consideration if:

- Ω is an open subset of $\mathbb{R}^2$ or $\mathbb{R}^3$ containing $\mathcal{O}$,
- H is a real valued function defined at each point P of $\mathcal{O}$ and in each direction d, $H(P,d)$ represents the desirable step size of the mesh at P in d direction and must always be positive.

If $H(P,d)$ is independent of d, the control is said to be *isotropic*. In this case, a good triangulation of Ω will be constituted of nearly equilateral triangles of a given size. On the contrary if $H(P,d)$ is d-dependent, the control is *anisotropic* and stretched elements are desirable.

3.1.3-A particular class of control spaces

Here, a particular class of control spaces is considered, including some anisotropic properties. We assume that $\mathcal{O}$ is partitioned by a mesh whose vertices are $(S_i,\ i=1,..,N)$. A n x n-matrix M_i symmetric and positive definite is associated with each vertex S_i (n = 2 or 3 is the dimension of the space). A new metric is defined by the following norm

$$\| X \|_{S_i} = \sqrt{X^t M_i X} \qquad (9)$$

Then, for each direction d, the function H is given at point S_i by

$$H(S_i,d) = \frac{\| X_d \|}{\| X_d \|_{S_i}} \qquad (10)$$

where X_d is a vector in the d direction. The control space $(\mathcal{O},\ H)$ is now fully defined by interpolating the matrices on each element of the partition. Only continuous interpolations are considered to avoid jumps of the control function over $\mathcal{O}$. The simplest one is linear on each element.

3.1.4-Riemannian metric

The open set $\mathcal{O}$ can be considered as a C^∞-manifold and matrices $M(P)$, $P \in \mathcal{O}$ as bilinear forms. For a control space of the considered class, these forms define a *Riemannian metric* on $\mathcal{O}$. The length of a parametric curve $\gamma(\tau)$, $\tau \in [0,1]$ in such a metric is the value of

$$\int_0^1 \sqrt{\gamma(\tau)^t\ M(\gamma(\tau))\ \gamma(\tau)\ d\tau)}$$

Consequently, the change from Euclidean metric to Riemannian metric reduces mesh control to an isotropic and constant step size control. From the point of view of this metric, a mesh satisfying the function H all over the domain $\Omega \subset \mathcal{O}$ is constituted of equilateral elements of diameter equal to 1.

3.1.5- Adaptation

The mesh generator needs a fully defined control space. For this, a function H can be given over $\mathcal{O}$ from a priori knowledge of the physical solution. With a view to generating adaptive meshes, H is determined from a computed solution obtained on a previous (coarse) mesh T_0. This mesh becomes a natural partition of Ω to support the definition of the control function H. Following J. Peraire et al. [?], we base the mesh adaption on an error estimation for a selected key variable σ. More precisely H is determined in order to equi-distribute the interpolation error. The linear interpolation error depends on second derivatives of σ. This leads to computing the desirable directional step size H at each vertex S_i according to

$$H(S_i,d)^2 \mid \frac{\partial^2 \sigma}{\partial d^2}(S_i) \mid = c_0 \qquad (11)$$

where c_0 is the acceptable error chosen by the user.

The previous algorithm can be applied with one of these metrics instead of the Euclidian one. The obtained mesh is adapted to variable σ in the sens that the interpolation error becomes equally distributed all over the domain and in each direction.

3.2- A posteriori error estimate for mesh adaptation

For the Navier-Stokes equations, we calculate the a posteriori error estimate based on the solution of a local generalized Stokes problem. The

degrees of freedom for the error estimate of both velocity and pressure are chosen at the midpoint of each side of triangle with the quadratic functions. Particularly, the whole frame of the right hand side in the generalized Stokes problem, which is used as one step in θ-scheme, an iterative algorithm for Navier-Stokes equations, has to be placed as one part of the named right hand side of the equations about estimation so that the nonlinear convection term is considered. With the solutions of the error estimates at the three midpoints, we can evaluate an integrated energy norm as the value of error estimate for the element.

The threshold is defined in the sense of geometric average over the error estimate of all the elements in the triangulation on every level. Without any artificial intervention, we can refine all the triangles with the error estimate greater than the threshold, and the whole progressive refining process goes on automatically. But sometimes, especially at the beginning af our adaptation procedure, there are fewer elements exceeding the calculated threshold, and we might "push" it towards the minimum value of error estimate distribution so as to speed up the refinement.

As for the refinement schemes, the main way is to divide an element into four similar ones by pairwise connecting the midpoints of the three edges and these regular products are to be kept. In order to form an admissible mesh, we are obliged to divide some elements into two by bisecting a single edge in the middle of which a new node has been generated in the previous regular refinement. This irregular scheme is to be replaced with being refined regularly if necessary.

In the course of refinement, two guidelines should be respected as the complement of the defined threshold. One is called "2-neighbors rule". As the name suggests, it means a regular refinement for the existing element with two neighbors that have been regularly refined. If there is a triangle which has not been regularly refined but already one of whose sides has contained more then one irregular vertex, the other guideline, "1-irregular rule" makes this kind of triangle refined regularly.

4.- Scientific code validation with data bases

A European Hypersonic Data Base (EHDB) installed at INRIA Sophia Antipolis and containing critical problems originating from a series of workshops on hypersonic flows for reentry problems operates since early 1993 ([14]). The available data offer an attractive tool to the scientific and Industrial communities involved in Aerospace activities to evaluate accuracy and efficiency of hypersonic codes.

The EHDB contains general information and information related to the Hypersonic workshop test cases.The general information provides specifications of test cases, output formats, practical definition of physical models for computations (e.g. chemically reacting flows), grids made available to contributors ,analysis software and a chart defining the access and usage rules of the base.This information is accessible by network via a mailserver, or anonymous ftp, or wais ([15]):

to: hdb-quest@sophia.inria.fr
subject: guide.ps
request: hdb
topic:guide.ps
request: end.

Additionally the EHDB contains computerized data associated with experimental or computational solutions to the workshop test cases.The main important test cases used for Euler and Navier-Stokes code validation include the flow over a ramp (Problem 3), the flow over a double ellipse/Ellipsoid (Problem 6) and the flow over a delta wing (Problem 7).

The ramp flow (Re= 148000/m; β = 15°; Mach= 10) was proposed to study the shock/Boundary layer interaction in a separated flow and has been tested by Delery, ONERA, France. The solution computed with the above Navier Stokes solver compares quite well with available experiments; however from the collection of results compared with experiments, this test case revealed rather controversial in the separated region with respect to Stanton number predictions.

Test case 6.3 (Mach=25; α = 30°; Z=75 km) was employed to test different possible models for the diffusion terms applicable in the case of the mixture of reacting gases. A certain scatter in the stagnation-point heating has been observed due to the different models of diffusion terms.At present the EHDB contains 3-D viscous equilibrium flows solutions including the on computed with the solver described in section 11.3.1. but only 3-D non equilibrium boundary layer calculations.

The laminar vortex flow around a Delta Wing (test case 7.4: Mach= 8.7; α = 30°; Re= 2.25*10**6/m) which has been tested experimentally at the DLR Köln Wind Tunnel H2K by Henckels has shown computed Stanton numbers noticeably higher than the experimental value.

The EHDB will now be enriched with a new set

of more complicated test casesin both supersonic and hypersonic regimesfor which experimental data are or will be available. The precise description of related test cases will appear in the announcement of a forthcoming workshop ([16]).

Such workshop and database activity plays a major role in the evaluation of uncertainty margins and is essential to improve confidence in reliable thermodynamics data in flight conditions.

5.- Applications: Flowfields of interest

The following examples of 3-D flow computations around complete aircrafts and space vehicles show the possibilities of the codes with the above methodologies:

5.1- Euler calculations

This code can be used iteratively in order to design and optimize air intakes or/and after bodies shapes. Figure 1 shows the surface mesh on MIRAGE 2000 and Figure 2 the surface pressure distribution from an Euler computation at $M_\infty = .9$, $\alpha = 2°$.

The computation shows in Figures 3 and 4 was performed on a Falcon 900 spatial mesh composed of 43,000 nodes and 258,000 elements. The objective of this calculationwas two-fold: first, to evaluate the pressure distribution over the whole aircraft; then to investigate with great care the transonic interaction phenomenon between the wing trailing edge and the propulsion ensemble (side nacelle and its fuselage linking pylon).

The design of a space plane like Hermes requires the prediction of the aerodynamics forces in flight conditions, to be accessible in the form of a data base covering all possible Mach numbers, angles of attack, yaw angle, altitudes and control surface deflections. Figure shows the pressure distribution at high Magh number and high altitude. To generate this data base, the Euler equations, with chemical effects when needed, have to be solved a large number of times; consequently the unit cost of the simulation has to be low.

Figure 8 and 10 present a 2-D horizontal cut through a scramjet air intake. The diffusion in the inlet is insured by 6 degree angle side walls. Fuel injection struts with vertex 6 degree half angle also contribute to the fluid compression. Extensive mesh adaptation was used in order to capture the complex shock wave interaction.

5.2- Navier-Stokes calculations with chemistry

It is also necessary to obtain estimations of heat fluxes in regions where boundary layer treatment is not possible. In such regions the full Navier-Stokes equations have to be solved. A typical example is the flow over the canopy of Hermès, which contains complicated three dimensional separation and reattachments, causing local peaks on the skin temperature.

In order to obtain such Navier Stokes solutions at a reasonnable cost, we use "local meshes", imbedded in the global ones, which will represent the full details of the space plane in a few critical regions.

Because of the complexity of the shapes involved, we use unstructured meshes, made either of tetrahedras, prisms or hexahedras. First a mesh of the skin of the aircraft is generated, then the three dimensional volume is filled by a front marching technique ([17]).

Euler calculations are usually initiated by a uniform flow, taken at the free stream conditions which also define the boundary conditions. Implicit solutions are obtained in less than 200 time steps, and CPU consumption ranges from a few minutes to a few hours, for the largest meshes (200 000 nodes), on an IBM ES 9000/820.

Navier-Stokes calculations are performed on very dense meshes, which represent only a small part of the aircraft, and are embedded in a coarser grid covering all the domain of dependency of the local calculation. They are initiated by a flow field obtained by merging the Euler solution and the boundary layer solution; this initial flow also defines the boundary conditions for the Navier-Stokes simulation.

Figure 6 shows the flow over the canopy: we present here the friction lines. This calculation shows the window temperature to be over the design limit for the particular shape calculated. We can see that the flow is separated, with a very complex reattachment in the upper window region. The mesh includes 180 000 nodes. Radiative boundary conditions were used (including non convex reradiation).

5.3- Navier-Stokes calculations with turbulence model

The case presented is a 3D computation on a generic cylindrical air intake at $Re_{/\mathrm{m}} = 1,100,000.$ and $Mach = 0.25$ at 15° of angle of attack. The mesh has more then 60,000. nodes. In Figure 7 we present the streamwise velocity with a visualization of the separated region in the air intake. One can see the complete 3D structure of the separation.

5.4- Unsteady potential calculations

with fictitious domain method

The 3-D mesh around the wing and the store has 48,619 nodes and 275,041 tetrahedra. For the potential flow computation the parameters were $M_\infty = .82$, $\alpha = 0°$. Figure 9 shows the pressure distribution on the initial configuration for body-fitted mesh and fictitious domain (below).

6. Conclusion

We have shown how several robust and efficient flow solvers which have been developped and validated in an industrial environment are used for the design of aircrafts and hypersonic vehicles. An important feature of these solvers including real gas effects and turbulence models is their ability to be interfaced with each other using a multi-zone/multicode strategy.

From recent calculations we strongly believe that in the case of transonic and supersonic flows cost effective solution can be obtained through multi level preconditioning on structured meshes simultaneously associated to unstructured meshes for accuracy. An important part of the efficiency can be achieved via domain decomposition methods combined with fictitious domain techniques on MIMD parallel architectures.

Future applications in Aerospace Engineering will deal with the computation of active flow control using smart structures via intelligent algorithms.

Acknowledgements

The results and methods described in this lecture have been obtained in cooperation with many colleagues at Dassault Aviation and abroad through fruitful discussions. We would like to mention most particularly R. Abgrall, R. Bank, A. Dervieux, A. Désidéri, R. Glowinski, T.J.R. Hugues, O. Pironneau and M.G. Vallet and emphasize on the important role played by D. Alleau, A. Naïm, J.M. Hasholder, G. Heckman and E. Teupootahiti who have integrated in an industrial environment most of the methods described in this paper.

The partial support of DRET under several contracts and ESA/CNES under R&D Hermes grants is also acknowledged.

References

[1] M.O. Bristeau, R. Glowinski, J. Périaux, P. Perrier, O. Pironneau and G. Poirier. On the numerical solution of non linear problems in fluid dynamics by least squares and finite element methods (II). Applications to transonic flow simulations. Computer Methods in Applied Mechanics and Engineering (51), 1985.

[2] GAMM Workshop on numerical methods for compressible inviscid fluids. INRIA Rocquencourt (France), December 1983.

[3] GAMM Workshop on the numerical Simulation of compressible Navier-Stokes Flows, M.O. Bristeau, R. Glowinski, J. Périaux, H. Viviand, Eds, Vol.18, Notes on Numerical Fluid Mechanics, Vieweg, 1985.

[4] B. Stoufflet, J. Périaux, L. Fézoui and A. Dervieux. Numerical simulation of 3-D hypersonic Euler flows around space vehicles using adapted finite elements, AIAA paper 87-0560, Reno (Nevada), 1987.

[5] T.J.R. Hughes et al. A new finite element formulation for Computational Fluid Dynamics: I. symmetric Forms of the Compressible Euler and Navier Stokes equations and the Second Law of Thermodynamics. Computer Methods in Applied Mechanics and Eng., 54, 1986.

[6] Q. V. Dinh, J. He. A cartesian grid finite element method for potential flows. 2nd symposium On High Performance Computing, October 1991, Montpellier, France, Durand and El Dabaghi Eds, North Holland.

[7] R. Glowinski, T.W. Pan, J. Périaux. A one shot domain decomposition /fictitious domain method for the solution of elliptic equations. Proceedings of the Parallel CFD '93 Conference. Paris , May 10-12 1993, Elsevier, to appear.

[8] Q.V. Dinh, B. Mantel and Jiwen He. A cartesian grid finite element method for aerodynamics of moving rigid bodies. Computational Fluid Dynamics, Proceedings of the First European CFI Conference, Volume 2, September 7-11 1992, Brussels, C. Hirsch,J. Périaux and W. Kordulla Eds, Elsevier.

[9] R.E. Bank, PLTMG: a software package for solving elliptic partial differential equations, User's guide 6.0, SIAM, 1990.

[10] Kasbarian, C.,Leclercq, M.P., Ravachol, M. and Stoufflet, B.- Improvements of Upwind Formulations on Unstructured Meshes, in 4th International Conference on Hyperbolic Problems, Taormina(Italy), 3-8 April 1992.

[11] T.J.R Hughes, L.P. Franca and M. Mallet: A New Finite Element Formulation For Computational Fluid Dynamics: I Symmetric Forms of the Compressible Euler and Navier Stokes Equations and the Second Law of Thermodynamics. North-Holland, Computer Methods in Applied Mechanics and Engineering, 54, 1986.

[12] 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

[13] F. Chalot, M. Mallet, and M. Ravachol, "A comprehensive finite element Navier-Stokes solver for low and high speed aircraft design," *32nd Aerospace Sciences Meeting and Exhibit*, Paper AIAA 94-0814, Reno, NV, Jan. 1994.

[14] R. Abgrall, J.A. Désidéri, R. Glowinski, M. Mallet, J. Périaux, Eds. Hypersonic Flows for Reentry Problems, Vol. III, Springer, 1992.

[15] R. Abgrall, J.A. Désidéri, M. Mallet, J. Périaux, B. Stoufflet. The European Hypersonic dataBase: an efficient Code Validation Tool for High Velocity Flow Analysis, ISTS 94-d-48v, Yokohama, Japan, May 1994.

[16] Joint US/Europe Conference Workshop on the High Speed Flowfields Data Base, University of Houston, November 1995.

[17] D. Alleau, J.P.Figeac, B. Mantel and J. Périaux. Unstructured mesh generation and adaption techniques for 3-D compressible flows. Computational methods in Applied Sciences, First European CFD Conference, 1992, Brussels, Ch. Hirsch,J. Périaux and E. Onate Editors, Elsevier.

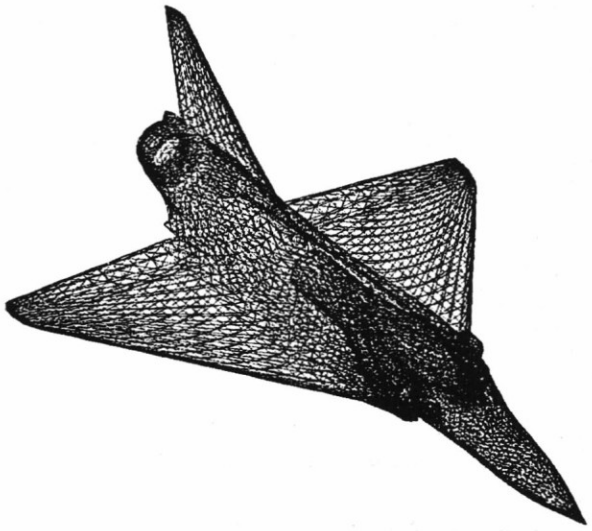

Figure 1: Surface mesh of the Mirage 2000.

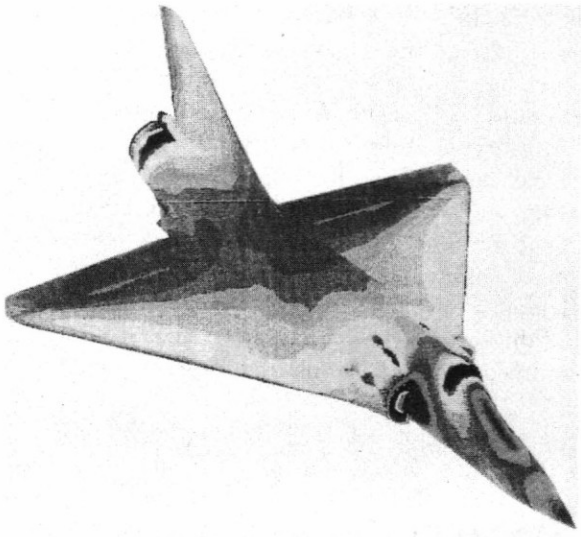

Figure 2: Mach number distribution. M=.7, $\alpha = 0°$

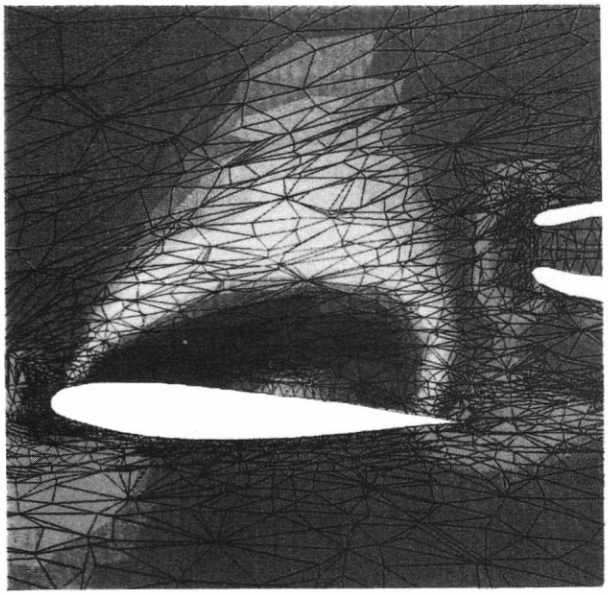

Figure 3: Enlargement of the Mach number distribution in vertical cross section. M=.85, $\alpha = 3°$

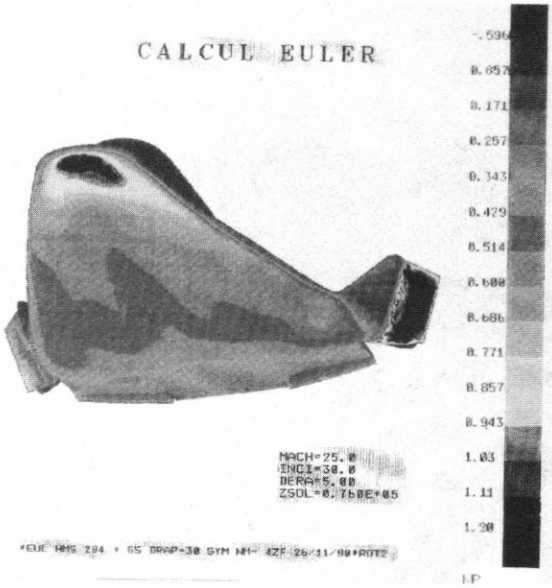

Figure 5: Pressure distribution on Hermes.

Figure 4: Mach number distribution on the Falcon 900. M=.85, $\alpha = 3°$

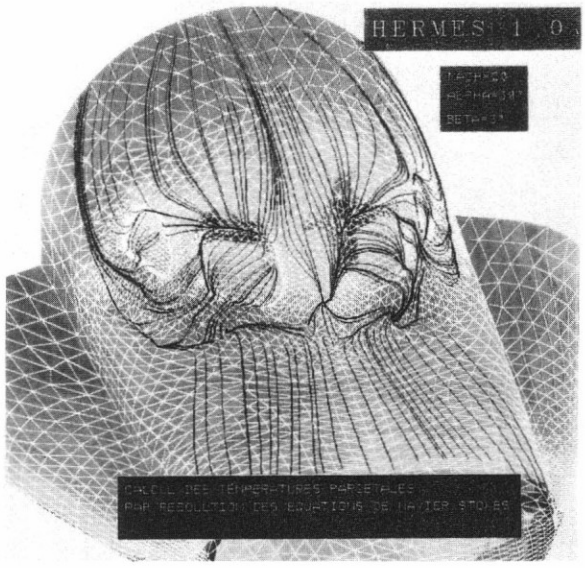

Figure 6: Temperature distribution on the canopy of Hermes. M=20., $\alpha = 0°$, $\beta = 3°$

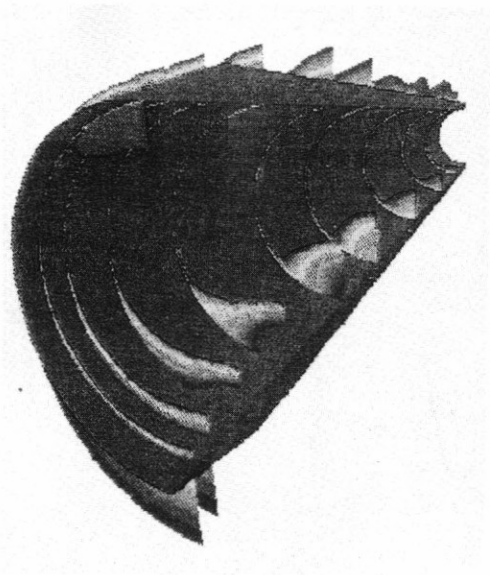

Figure 7: 3D Cylindrical air intake. M=0.25, $\alpha = 20°$, Re=1.110^6

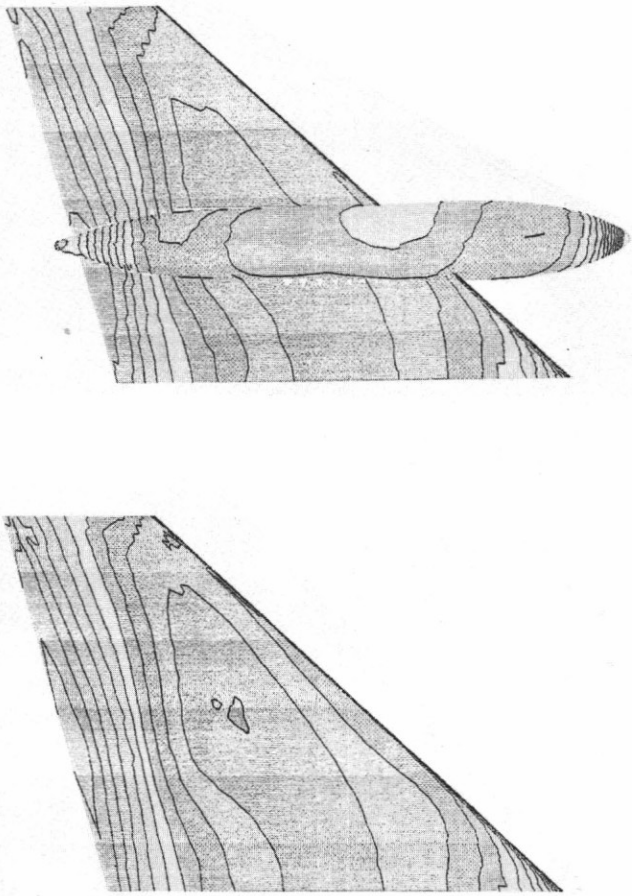

Figure 9: Comparison of flow solution on conformal mesh and fictitious domain

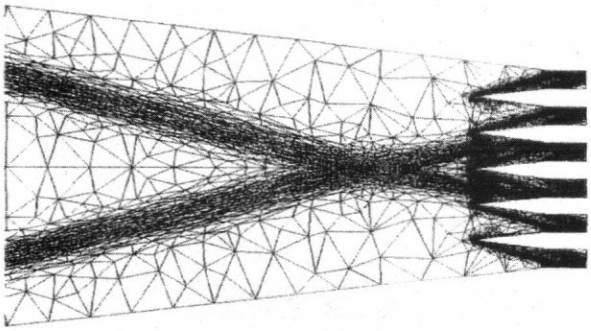

Figure 8: Mesh adaptation of shock-shock interaction.

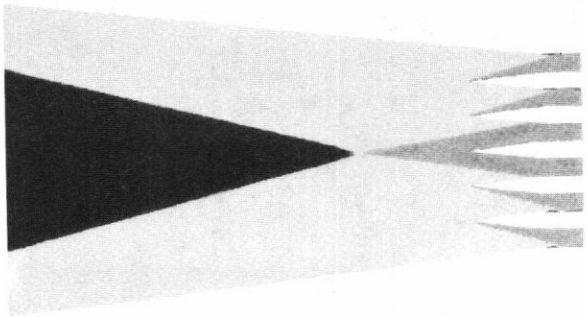

Figure 10: 2D scramjet inlet, Mach distribution. M=6., $\alpha = 0°$