Proceedings of
ASME TURBOEXPO 2000
May 8-11, 2000, Munich Germany

2000-GT-636

Three-Dimensional Investigations for Axial Turbines by an Implicit Unstructured Multi-block Flow Solver

P. Adami and F. Martelli

Energetic Dept. "S. Stecco"- University of Florence – Italy
Fax. +39 055 4796342 – e-mail: adami@ing.unifi.it

V. Michelassi
Department of Mechanical and Industrial Engineering
University of Roma 3

ABSTRACT

An unstructured 3D implicit approach (HybFlow) is briefly described for gas turbine applications. The numerical approach is based in an upwind finite volume scheme with an implicit time marching algorithm. The linear solver is GMRES with right preconditioning obtained by the ILU(0) incomplete factorization. A two *k-w* turbulence model is considered for viscous flows. The new multiblock extension of the code is here considered for improving the computational efficiency of the basic procedure in view of the application to complex 3D geometries. Particular concern has been placed in the new approach for memory saving while keeping at the same time a great geometrical flexibility and grid transparency. The strategy developed is therefore described considering the whole process and all phases required in the solution of internal flows.

Both memory and CPU time costs are addressed to verify the performances of the multiblock scheme as also computational accuracy for 2D tests. The flow conditions investigated range from the low speed regime up to the transonic one to prove the accuracy of the approach using hybrid unstructured configurations. Finally the application to the transonic cooled annular IGV blade row is considered to assess the multiblock strategy for a 3D case.

NOMENCLATURE

E	Total internal energy
GMRES	Generalised Minimal Residual Method
ILU	Incomplete LU factorisation
$\vec{n}$	Face normal direction
k	Turbulent kinetic energy
p	Static pressure
U	Conserved variables vector
R	Residual vector
u_i	Cartesians velocity components
Greek symbols	
ρ	Density
σ	Schmidt number
τ	Viscous stresses
ω	Turbulent specific dissipation rate

INTRODUCTION

Computational fluid dynamics has become an effective tool to analyse complex flows and ease the design of more efficient machinery components. The advantage of the numerical approach lays in the chance offered for investigating many different working conditions from which global or detailed information about the flow can be extracted. In fact particular geometrical situations (inner cooling ducts, tip clearance passages…) and some flow quantities can result costly or difficult to be studied experimentally. This is also the case of transonic turbine cascades which represent an important and demanding application for a CFD numerical procedure. In fact, complex patterns can develop with shocks systems interacting strongly with the flow especially when boundary-layer transition or even separation is observed. Therefore, both efficiency and accuracy are needed in order to simulating such kind of realistic flows. To this aim, great importance has to be paid to the turbulence modelling adopted in the computation but also to the numerical scheme. In this regard many applications of structured codes for 3D turbine investigations have been reported. Some examples are represented by Michelassi and Belardini, 1999 using implicit ADI factorisation, Arnone, 1993, Ho, 1996 and Koiro, 1998 using explicit multigrid techniques. A common drawback of all the structured approaches is the crude way required to improve accuracy in confined region of rapid gradient which also imply a costly grid refinement involving a large amount of the flow domain. A more sensible approach would instead refine the mesh locally only in region where a sharp variation of the solution is effectively expected. The use of unstructured mesh for flow computations originates from the finite element area and has been focused for a number of years to incompressible viscous flows. The first examples of compressible simulations come from the external aerodynamics (Jameson And Baker, 1987, Frink, 1991, Parthasarathy and Kallinderis, 1994,) mainly for inviscid flows. More recently viscous extensions where reported by Kwon and Hah, 1995 and Mavriplis, 1995. Probably the first application of a 3D unstructured solution adaptive methodology in the turbomachinery area is represented by the work of Dawes, 1992 and increasing effort has been addressed since then to the use of similar approaches for internal flows. The main reason for this interest lays not only in the more rational mesh refinement allowed but also in the higher geometrical flexibility allowing complex configurations to be represented and easily handled by the solution algorithm. In this regard most of known works actually do not fully take advantage of the great flexibility offered by the unstructured meshing. Only recently the concept of grid transparency and hybrid mesh generation is being considered for practical computations (Connel and Braaten, 1995, Delanaye and Essers, 1997, and Haselbacher *et al.* 1999). It is in fact easily understood (Habashi *et al.* 1997) that the best advantage for the mesh discretization capabilities can be achieved not only through a local refinement but also with a solution adapted choice and orientation of the elements.

The present work considers the application of an implicit unstructured hybrid solver HybFlow (Adami *et al*. 1998) for internal

viscous flows. The turbulence model here used is based on the two equation *k-w* proposal of Wilcox, 1993. The extension for transition by Wilcox, 1994 is also considered along with the improvements suggested by Kato and Launder, 1993 and Durbin, 1996. As commonly observed for unstructured approaches, despite allowing a more efficient and flexible disposition of the elements, they still require a considerable computational costs. This drawback has been observed also for the code HybFlow posing serious limitations especially for large 3D viscous computations. Therefore, a multiblock approach has been considered with the main aim to improve the numerical efficiency of the code. The new features of the multiblock solver are here described and verified. The performances improvements are demonstrated computing a 2D IGV turbine blade SNECMA ESCA CD01 (Bario and Beral, 1998), while accuracy for the application to transonic conditions is addressed through the LS89 blade (Arts *et al.*, 1990). Finally the application and assessment of the numerical procedure for 3D computations is considered for a transonic IGV 3D blade (Kapteijn *et al.1996*, Dunker, 1995).

THE GOVERNING EQUATIONS AND THE IMPLICIT SOLVER

The flow solver HybFlow is based on a numerical scheme developed for the compressible Navier-Stokes equations cast in strong conservative form (Adami, 1998). The turbulent version of the solver refers to a classical eddy-viscosity approach using the two-equation model $k-\omega$ suggested by Wilcox, 1994. Adopting the perfect gas model and a Favre mass averaging the whole set of governing equations characterises the flow field through the solution vector U for $\rho, \rho u_i, \rho E$ and the turbulent quantities $\rho k - \rho\omega$. All the governing equations within the flow field satisfy the same conservation laws obtained accounting for the accumulation, convection, diffusion, and production phenomena. The variant proposed by Wilcox, 1994 has been also implemented into the code for computing turbulent transition. The production term correction suggested by Kato-Launder, 1993 and the physical realizability constraint of Durbin, 1996 are considered for the turbulence production terms.

The solver has been developed and validated in the frame of internal compressible/incompressible flow regimes. In the present work a brief description of the basic numerical scheme will be provided while a more detailed discussion can be found in the suggested references (Adami *et al.* 1998, Adami 1999). The solver performs a numerical discretization of the spatial gradients for both convective and diffusive terms of the balance equations using a cell centred finite volume scheme. The spatial discretization has been specifically designed to be grid transparent in order to achieve a spatial discretization, which is independent on the cell type and therefore suitable for unstructured hybrid grids. To this aim the computational domain is subdivided into an arbitrary set of 3D elements which can be selected among hexahedra, prisms, pyramids and tetrahedra. The whole of all elements covers without overlapping the computational domain. Consequently it is also possible to use unstructured and structured parts in the same grid although no special treatment is considered in the solver for these last and therefore no advantage is achieved from the local ordered disposition of the elements. The governing equations are integrated over each element allowing the spatial derivatives to be expressed as flux functions through its boundary faces.

The numerical fluxes are computed by a midpoint quadrature formula. Concerning the inviscid fluxes a reconstruction evolution approach is used to define the solution on every face midpoint. With the reconstruction phase the solution stored on the cell centres is interpolated onto faces mid-point considering a linear law. During the following phase, the reconstructed values interact assuming an analogy with the evolution of a 1D Riemann problem having the same initial states.

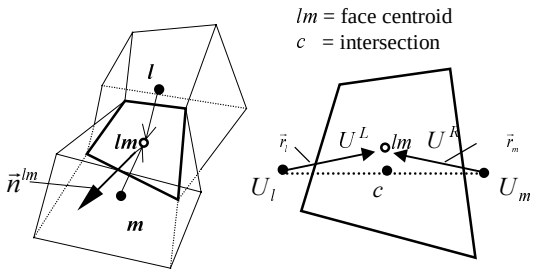

Figure 1: finite volume method

Then the midpoint fluxes are computed from the solution of the Riemann problem using the Roe approximated method (Roe, 1986 and Barth, 1991). The monotonicity is enforced thanks to a slope limiting of the gradients used during the reconstruction phase according the scheme derived from the idea of Barth, 1991. The viscous transport terms of the governing equations are added integrating by the same numerical formula the diffusive fluxes over the faces of the control volumes. The gradients of solution needed to define the viscous stresses are approximated directly onto the faces midpoint using finite differences. In this case, the solution derivatives on the face midpoint are given by $\left.\frac{\partial U}{\partial n}\right|_{lm} = \frac{U_l - U_m}{|\vec{r}_l| + |\vec{r}_m|}$.

The steady state flow field is obtained solving the non-linear time dependent system of differential equations resulting from the spatial discretization. If U is the solution vector for all elements and R the residual coming from the spatial discretization then the systems can be expressed as:

$$\frac{\partial U}{\partial t} + R(U) = 0$$

The implicit iterative Newton method is considered for computing the steady solution. The following linear system is therefore solved at each time step:

$$\left[\frac{I}{\Delta t} + \frac{\partial R(U)}{\partial U}\right]^n \Delta U^n = -R(U^n)$$

Stability of the numerical algorithm is provided by the time-marching relaxation term appearing in the implicit operator and resulting from the unsteady approximation in time of the governing equations. The matrix of the implicit method is computed numerically approximating the derivatives of the vector R with respect to the solution U. In this regard the solution vector is locally perturbed by a small quantity and the residual recomputed. The Jacobian derivatives are then obtained from a finite difference approaximation. Since with this approach only the face fluxes are needed, then the implicit matrix and the residual vector can be assembled with a single loop over all faces. This approach allows an efficient and grid transparent scheme that do not need any local information about the element type.

The linear system of the method is solved by the iterative method GMRES (Saad, 1994). To obtain an efficient convergence of the linear solution a right preconditioning is coupled with the iterative method. The preconditioning matrix is computed performing the incomplete ILU(0) factorisation of the implicit matrix (Saad, 1994). It is worthwhile to remember that for all the matrices involved the whole procedure GMRES-ILU(0) makes use of a condensed storage format considering only non-zero elements. As far as spatial discretization, no differences arise between the flow equations and these of the turbulence model. Conversely concerning the iterative

time-marching approach the two transport equations of the turbulence model are solved in a decoupled fashion. Therefore three separated and consecutively iteration steps are performed to march in time both the Navier-Stokes equations and the turbulent ones. In the first passage the 5x5 coupled system for mass, momentum and energy is considered and following two decoupled implicit iterations are performed respectively for the turbulent kinetic energy and its specific dissipation rate. In these last cases, the implicit iteration matrix is build up for each equation using the residual derivatives with respect to the current solution variables while the other components of the solution are kept frozen.

THE MULTI-BLOCK EXTENSION

The basic solver described in the previous paragraph has been used for computing internal and external flows in both compressible and incompressible regimes. In this last case stability and accuracy of the computation was achieved using a low mach preconditioning scheme (Adami, 1999). A common feature shared by the code in all the applications was a high computational cost in terms of both CPU time and memory requirements. This aspect represents the main drawback of the code and can be considered a direct consequence of the implicit-unstructured architecture. In fact, the most demanding phase of the solution algorithm lays in the implicit matrix storage and inversion required at every iteration step of the time marching procedure. Despite the attention paid to the use of efficient numerical strategies such as the Krylov sub-spaces (GMRES iterative solver) the matrix preconditioning (ILU(0)) and the compact storage, it was concluded by Adami *et al.*, 1998 that around 75÷80% of CPU time per iteration is required for the linear inversion while about 88% of the code memory demand is needed for the implicit arrays storage. These costs rapidly increase beyond modern computer capacities when complex three-dimensional viscous flows are considered. Therefore, to improve the computational efficiency a multiblock procedure has been implemented into HybFlow with the primary objective of reducing the memory required by the solver. The numerical efficiency improvement is pursued by this approach limiting the maximum dimension of the implicit matrices to the dimensions of the biggest block obtained from the mesh partitioning. The multiblock strategy here developed considers three distinct phases indicated as pre-processing, solution and post-processing. In this contest the pre-processing phase includes the grid meshing, the domain decomposition, the metrics computation and the mesh optimisation. With this pre-processing routines the solver retain instead a fixed architecture which is sequentially applied to march in time the flow in every block defined within the whole original mesh. The post-processing phase is needed to join again the solution in all the blocks and to extract the computational result for visualisation and analysis.

The multiblock implemented in the code has a vertical mode structure and a complete iteration step of the time marching scheme is obtained processing sequentially every single block of the grid. The main advantage of this approach is that each block can be computed independently from the other regardless its position and connections inside the grid. The memory allocation strategy of the multiblock scheme considers two sets of variables having a global and local extension over the computational domain. Global variables are univocally accessible from any point in the algorithm and refer to information related to the whole grid and set of blocks. Examples of global variables are the solution vector and the grid metrics. Conversely, the arrays referring to the implicit time marching scheme (GMRES, ILU) are local variables allocating memory space during the processing of each single block. These local variables are not needed from outside the solution phase of a block and then the local memory storage is continuously allocated and freed moving from block to block. The memory cost of local arrays is bounded by the dimensions of the biggest block in the mesh. Since the local variable allocation strategy involves directly the most storage demanding matrices of the implicit-unstructured solver, then the memory costs involved in the multiblock version are expected to be drastically reduced increasing the number of sub-domains. The best efficiency is clearly obtained providing a well balanced decomposition where all blocks have roughly the same number of nodes.

To obtain a stable time marching scheme it is worthwhile defining an auxiliary global array to store the solution on the elements interfacing neighbouring blocks. During the sequential processing, this auxiliary vector defines the solution to be considered for every given block as domain boundary condition. The auxiliary vector represents actually the only link of the solution among the blocks and must be updated at the end of every cycle when all sub-domains have been processed. The matching performed through the auxiliary vector is therefore of a typical explicit character. Consequently, the multiblock code is expected to reduce progressively the robustness of the scheme with increasing number of inner boundaries and in the end with increasing number of blocks. Nevertheless considering the total CPU time costs for convergence the new procedure should also allow cases in which better performances can be obtained. This expectation is based on the evidence that the CPU cost of the code per iteration increases almost quadratically with the number of elements processed implicitly at the same time. Theoretically, for a given number of time step cycles and if the local domain dimension are kept frozen, then the multiblock solver should have a linear CPU time increase with the total number of elements involved in the computation.

THE UNSTRUCTURED GRID MANAGMENT

As described the analysis of a given flow field problem by the new multiblock procedure includes three main phases. The architecture of the approach follows the scheme reported in Fig. (2) where the main routines involved in the analysis are also indicated.

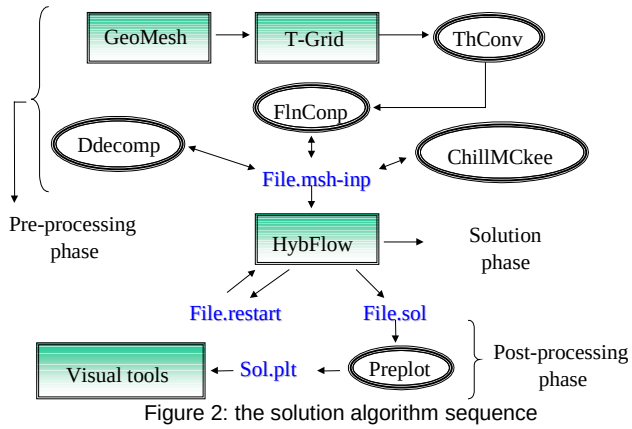

Figure 2: the solution algorithm sequence

The first pre-processing phase is the more relevant feature of the multiblock procedure developed and is entirely devoted to the grid management needed before the solution algorithm. The grid generation is performed through the commercial software T-Grid with a single unstructured mesh covering the flow domain. The required information from the T-Grid output file are extracted and converted by the routine ThConv. The following algorithm FlnComp contains the subroutines originally implemented inside the code HybFlow for processing the mesh and computing all the metric information needed by the implicit-FVM solution algorithm.

The output file is then fed into the domain decomposition program (Ddecomp) which generates finally the input file containing all required data as the number of blocks, the connections, the

metrics and boundary conditions. The Ddecomp routine is based on an empirical strategy derived from the scheme suggested by Farhat, 1993.

Several searching laws have been considered in the decomposition approach with the aim to obtain a multiblock grid which minimises the extension of internal interfaces and provides at the same time a well balanced distribution of elements among all subdomains created. Optionally the routine ChillMCkee can be applied to rearrange the element numbering in the blocks in order to reduce the bandwidth of the matrices assembled by the implicit scheme. This last routine closely follows the automatic reordering scheme suggested by CutHill and McKee, 1969.

APPLICATION OF THE MULTI-BLOCK SOLVER

Three application of the multiblock code will be here considered. A performance analysis will be addressed to verify the improvements allowed by the procedure developed in terms of memory and CPU time costs. The comparison with experimental measurements is also reported considering the velocity profiles in the low speed regime. Wall pressure distributions are compared against experiments in the subsonic and transonic conditions. The third application investigates in more details the grid generation and management problems arising for a 3D transonic blade. For the same blade, more details will be also provided concerning the flow field computation performed with trailing edge coolant injection.

ANALYSIS OF PERFORMANCES

The SNECMA ESCA blade is a an hydraulically smooth blade having a stagger angle of 41.5 deg, an outlet angle of 63.5 deg and a pith to chord ratio of about 0.625. Details for this large scale test configuration (the blade chord is 1.4 m) as also the experimental results can be found in the work of Bario and Beral, 1998. Concerning the computation, a structured O type mesh has been placed around the blade to provide an accurate and efficient discretization of the boundary layer. Outside this first structured region, an unstructured mesh composed by triangles covers the remaining part of the computational field. As far as the mesh decomposition, the blocks were generated expanding the boundaries with a searching criterion that gradually included the elements having a smaller distance from the initial starting location of each block. The decomposition outcome using this approach is shown in Fig. 3 for a 14 blocks grid.

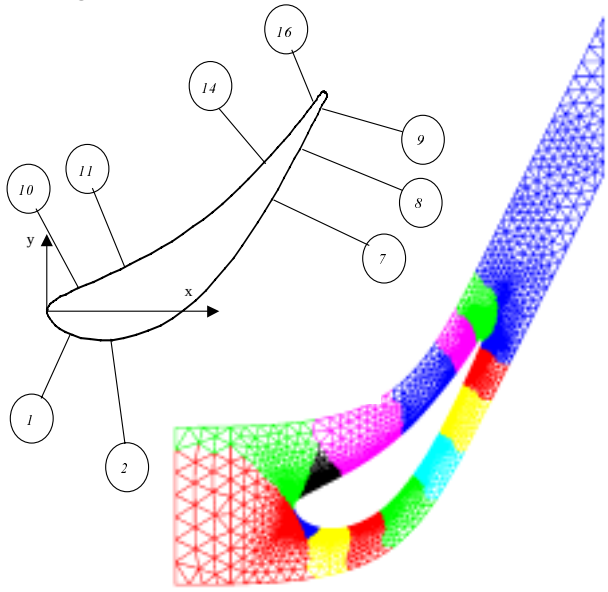

Figure 3: grid decomposition and blade measurement stations

The computation has been compared with the set of measurements carried out for an inlet turbulence level of 5%. The turbulence length scale is also known to be 65 mm while the outlet Reynolds number is 1.31×10^6. The outlet Mach number is about 0.04 and the low speed preconditioning technique implemented in the solver was used to allow a stable and accurate computation (Adami, 1999). A grid made of 15600 mixed elements has been considered for investigating boundary layer velocity profiles against measurements. The first computed grid node is placed in the range of $y^+=5\div10$ along the blade profile. In the first two stations along the suction side a laminar boundary layer develops. Despite the application of the Kato-Launder correction or the Durbin's physical realizability constraint, the two-equation model shows an exceeding turbulent production close to the stagnation point.

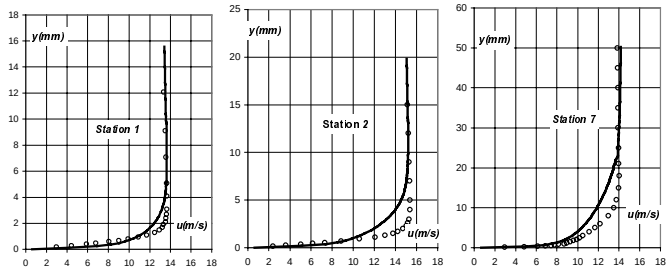

Figure 4: suction side velocity profiles

This over-prediction produces higher values of turbulent kinetic energy around the blade leading edge, which result in the wrong indication of boundary layer thickness observed in Fig. 4. This effect was confirmed by a full laminar computation that conversely produced in the same stations a result in close agreement with the experiments. This effect is felt with decreasing importance also in the downstream sections of the suction side until turbulent transition occurs close to station 7.

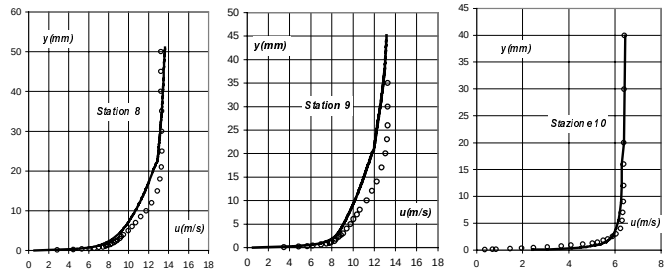

Figure 5: TE and LE velocity profiles

From this location (Fig. 5) both the computation and the experiments show the same velocity profiles although the computed boundary layer is still slightly higher than expected. Conversely on the suction side (Fig. 5-6) all sections shows a close agreement with the experimental outcome.

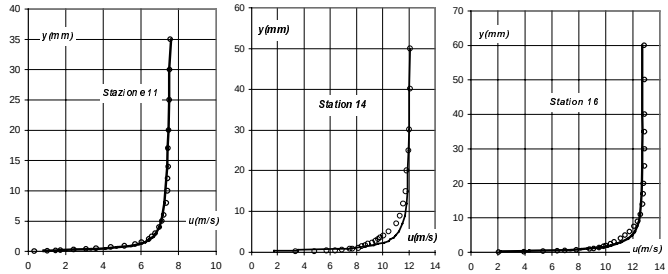

Figure 6: pressure side velocity profiles

To analyse the code performance five block decompositions of the grid are here considered as shown in Tab (1). For a total number of 9700 mixed elements and a fixed CFL of 50, the single block computation needs quite high memory storage. Conversely, starting from the five blocks decomposition this requirement drastically reduces to one third of the original value. With more domains involved in the decomposition clearly the storage demand is further lowered although the gain becomes progressively less relevant as shown passing from 14 to 20 blocks. The CPU time cost per iteration is also reported in the same table. According with the memory demand also the CPU cost for the same CFL shows a steep decrease from the single to the five block configuration, but now the improvements prove to reach a limit threshold already for the ten blocks case.

Blocks	CPU/iter (secs)	Mem (Mbyte)	CPU(secs)/(iter×node)	Conv. Iter.
1	31.4	43.5	0.0031	80
5	9.2	12.6	0.00093	126
10	8.4	8.6	0.000845	150
14	8.7	7.4	0.00088	162
20	8.9	6.9	0.00089	188

Table 1: multiblock performances

This difference in CPU time costs per iteration is easily explained considering that increasing the domain splitting progressively the gain obtained for the implicit matrix inversion is reduced. At the same time, the cost overhead introduced by the internal boundaries becomes increasingly important until finally it overcomes the former effect. Similar conclusion can be drawn considering the CPU/(iterxnode) parameter of Tab. 1. As expected, increasing the blocks number deteriorates the convergence performance and more iterations are needed to reach a stated residual level. Despite this effect is quite remarkable especially passing from the one to the twenty blocks configuration, all multiblock cases still shows a total CPU time requirement lower than the single domain computation. These results are also shown in Fig. 7 and 8 where the convergence histories are reported in terms of iteration number and CPU secs. Globally the memory improvements range from 71 to 84 % while the CPU reduction is in the range of about 50% (5 and 10 blocks) to 34% in the worst case (20 blocks). It is worthwile mentioning that the multiblock and single block computation at convergence gave the same solution

A comparison for variable CFL number was also performed. In this case, the 14 blocks computation performed the best with a time reduction against the constant CFL case of about 25%.

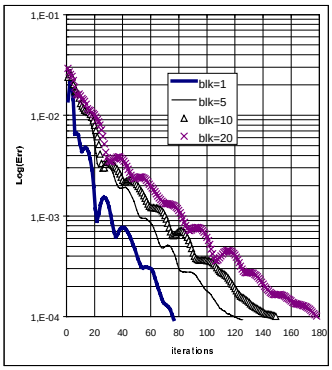

Figure 7: conv. vs iterations

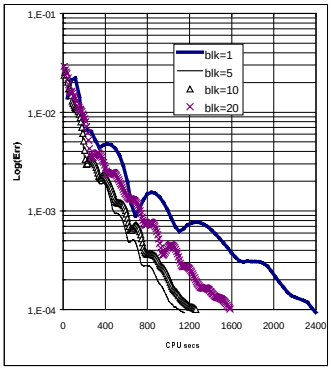

Figure 8: conv. vs CPU time

THE TRANSONIC BLADE LS89

The following application considers the 2D blade LS89 investigated by Arts *et al.*, 1990. The grid features are shown in the following Fig. 9 referring to the 7000 elements coarse grid. The same hybrid approach has been considered with the structured O type region around the blade wall and the unstructured part covering the remaining flow domain.

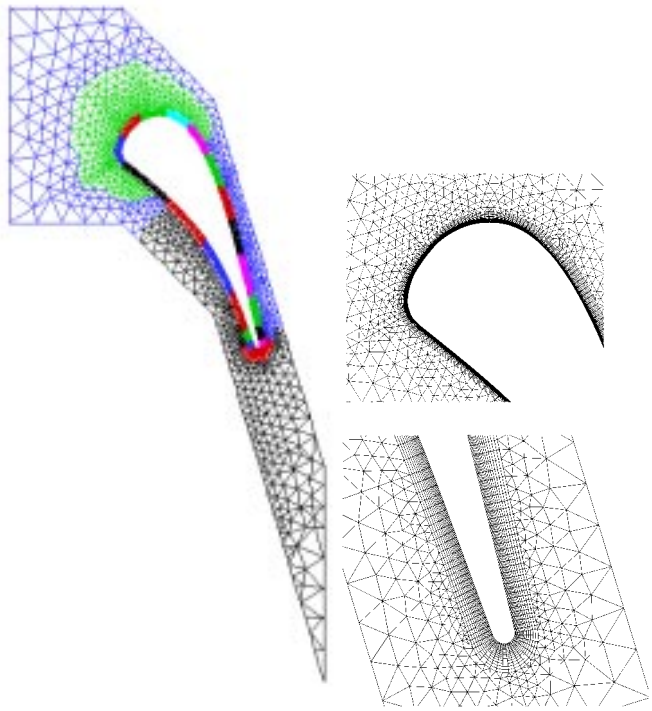

Figure 9: LS89 grid features

A refined grid of 11000 elements has also been obtained from the coarse one increasing the elements in the O type region along with the normal wall direction. The decomposition approach still considers an expanding criterion based on minimum elements distances. Nevertheless in this case the outer inviscid field and the viscous substrate where considered separately as shown in Fig. 9 in order to preserve as much as possible the integrity of the grid in the inviscid flow field. The main characteristics of both the blade and the test conditions are here reported while for a more detailed description the work of Arts *et al.* 1990 should be considered. The blade chord is 67.647 mm with a pitch/chord ratio of 0.85; the inlet total temperature is 420K for a turbulence level of 1% and a length scale of 1% the pitch. The Reynolds number is 1.0E6 and two outlet pressure conditions have been here investigated. The first condition termed MUR43 corresponds to an outlet isentropic Mach of 0.84 while the second, MUR47, has a higher outlet value of 1.02. For the MUR43 test, the flow field developed is subsonic and a stable convergence for the laminar and turbulent computations is usually achieved within 150 global iterations. According with the conclusion drawn in the previous test a constant value of 40 has been used for the CFL number through all the computation for the Navier-Stokes equations. A lower constant value of CFL$_t$ =5 has been applied for the turbulent *k-w* update to ensure stability. In order to maintain a globally balanced time-marching the two turbulent equations have been repetitively processed with more sub-iterations for each main iteration of the Navier-Stokes relations and the number of sub-iterations has been set as follows: $n_{turb} = \text{int}(CFL/CFLt)$. The total CPU time required on a DEC ALPHA 500/500 for a steady solution on the refined grid is 30 minutes. The computed pressure profile is compared against experiments in Fig. 10 for the coarse grid. A satisfactory agreement is clearly obtained over all the blade profile although the pressure recovery on the last part of suction side is

slightly displaced ahead of the experimental measures. This effect is probably linked with the noticeable flow expansion computed close the blade trailing edge on the pressure side. In fact, the wake shown in Fig. 11 behind the blade is probably experiencing an excessive flow underturning which also explains the small discrepancy on the suction side pressure profile.

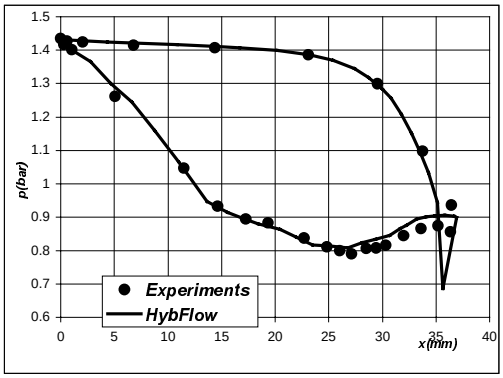

Figure 10: wall pressure profile for MUR43

Since no separation of the boundary layer is experienced by the flow in this test condition then the computation is expected to show no appreciable influence on pressure profiles by the turbulent field characteristics. In fact, despite the turbulence solution actually behaves differently considering the standard *k-w* model, the variant with transition, the Kato-Launder production correction or the Durbin realizability, the wall pressure has not shown differences also for the refined grid computation.

The outlet pressure condition in the test MUR47 correspond to a transonic flow developing in the blade passage with a normal shock compression over the suction side. In this case, the turbulent computation differs considerably from the laminar one owing to the shock-boundary layer interaction. The laminar flow experiences a boundary-layer separation across the first shock with a following acceleration and a second shock compression. This last shock before the trailing edge is needed by the flow on the suction side to match downstream the flow coming from the pressure side. Clearly, a sharp normal shock is placed at the same blade position observed in the experiments. From the mach isolines it is argued that this shock compression causes a sudden boundary layer thickening which, differently from the laminar computation, does not detach appreciably from the solid surface. The shock presence introduces a remarkable effect also in the turbulent flow field. Considering in fact the turbulent kinetic energy from Fig. 12 it is evident that a strong production is starting at the shock position and is propagating downstream until the blade trailing edge.

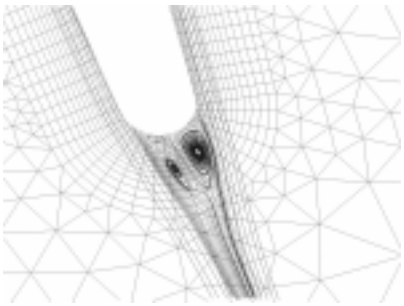

Figure 11: trailing edge grid and streamlines

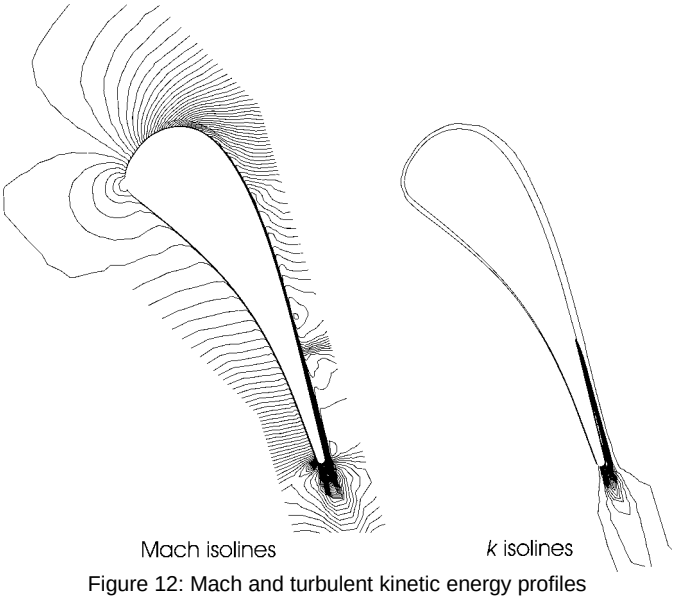

Figure 12: Mach and turbulent kinetic energy profiles

The *k*-contours here reported refer to a computation using the transitional model of Wilcox. The Kato-Launder correction and Durbin's physical realizability constraint were also considered but no appreciable difference has been observed for the wall pressure profiles. These agree well with experiments on the pressure side and suction side until the shock. This last is placed in the same exact position as experiments, but, as for the MUR43 test, the pressure recovery is slightly shifted upstream the experimental prevision. This behaviour proved to be scarcely affected by any modification in the turbulence model parameters or changing from the coarse to the refined grid computation.

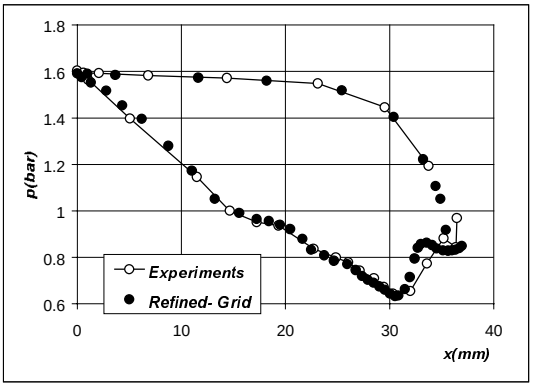

Figure 13: turbulent wall pressure profile (transonic case MUR47)

COMPUTATION OF THE 3D IGV

The 3D IGV blade here considered has investigated within the frame of a BRITE EURAM project (Dunker, 1995, Kapteijn *et al*., 1996) The main features and test conditions of the row are the following: inlet total pressure of 1.62 bar, inlet total temperature of 440 K, outlet midspan isentropic Mach number of 1.05, axial inlet, Re=10E6 and 1% inlet turbulence level. The blade is cylindrical with a midspan pitch of 54.0425 mm. A cut for coolant ejection spans all the blade height at the trailing edge on the pressure side.

The computational grid follows the basic generation and management strategy applied for the 2D tests. In this case, there are three types of solid boundaries: the blade surface, the hub and shroud end walls. An accurate prediction of 3D effects developing through the row implies a reasonable resolution of all boundary layers. To this aim, beside the structured O type grid surrounding the blade, a grid clustering has been considered also on both the hub and shroud

end walls surfaces. The hybrid grid configuration used is represented in Fig. 14. Three different zones can be here defined: the first one is the structured mesh around the blade composed mainly of hexaedra. This ordered region is then matched with the core flow of the passage that is made up of unstructured tetrahedral elements. The elements interfacing these two regions are pyramids having a quadrilateral base and triangular sidewalls. Finally, the third regions types are the prismatic unstructured strips placed on the hub and shroud end walls.

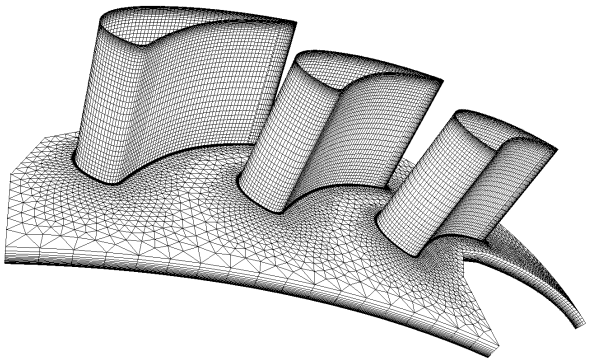

Figure 14: 3D unstructured mesh of VKI annular row

A special grid treatment is used in the cut area of the blade as shown in the detail of Fig. 15. The resulting grid, made of 340000 elements (about 1 million of faces), has been divided with the Ddecomp routine following the approach of Fig. 3.

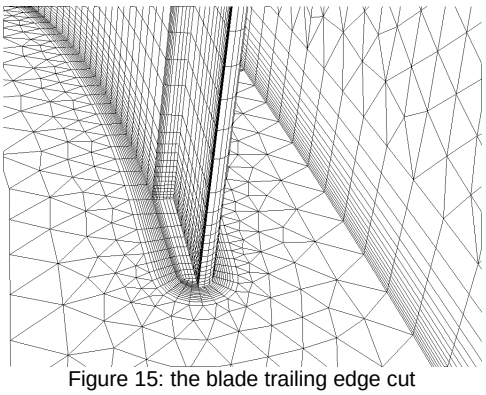

Figure 15: the blade trailing edge cut

A quite high number of 100 blocks has been used in the decomposition for the grid with a memory cost of 184 MB. To this aim the decomposition routine must be as much as possible automatic and reliable since small possibilities are left to the program user for an easy verification of the grid quality resulting from the decomposition in 3D geometries.

Differently from the 2D problems, some inaccuracies in the mesh generation procedure have been here observed. As expected the T-Grid generation software does not ensure quadrilateral faces to be planar. To verify the influence of this problem a check of the grid has been implemented in the conversion algorithm integrating over the whole field a constant solution. From this test, it has been found that in several prismatic and pyramidal elements the warping of the faces causes a considerable systematic numerical error of 10E-3. Therefore, to improve the grid quality, the warping of quadrilateral faces is checked during the computation of the grid metrics within the pre-processing algorithms. In case of excessive warping, the quadrilateral face is actually divided in the two triangles. Differently form the original approach with this modification two elements can arbitrarily share one or two faces requiring only minor modification to the solver architecture.

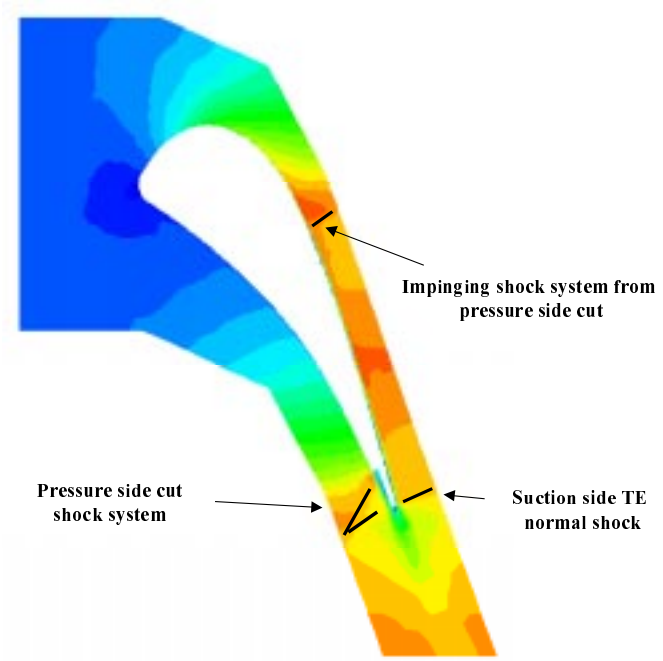

Figure 16: mid-span Mach and pressure profiles

Solution here described refers to the condition of no coolant injection and the isolines for the Mach number are given in Fig. 16 at midspan. The wall pressure computed at the same section is also compared in Fig. 17 with experiments using the isentropic Mach number. As shown the computation agrees well at the blade leading edge and on the pressure side. Conversely, on the suction side the Mis peaks are not exactly reproduced. In fact, the fist one is slightly hinted while the second one close the blade trailing edge is clearly underpredicted. The pressure profile on the first part of the suction side, also owing to the supersonic regime, is strongly dependent on the blade cut geometry and on the flow recirculation here developed. From the Mach field of Fig. 16, it is seen that the flow accelerates as a consequence of the expansion caused by the pressure side edge of the cut. This expansion ends up with a shock system that leads to an oblique shock propagating from the pressure side towards the suction side. The impinging of this shock onto the suction side is felt in the Mis profiles of Fig. 17 for x/c ≈ 0.4. The expansion at the cut it is clearly driven by the recirculation vortex pattern within the cut area. For this reason since from Fig. 17 the flow expansion at the pressure side of the trailing edge seems to be slightly underpredicted then some discrepancies with experiments arise also for the suction side pressure profile at x/c ≈ 0.4.

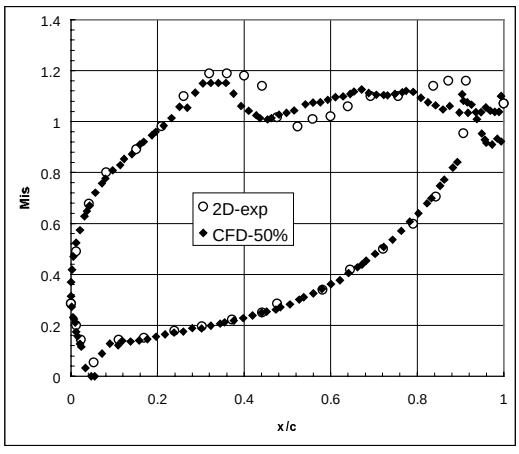

Figure 17: computed and experimental wall isentropic profiles

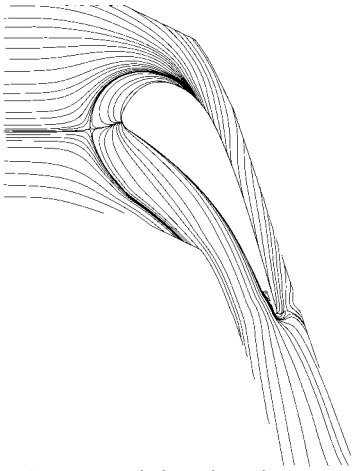

Figure 18: computed shroud section streamlines

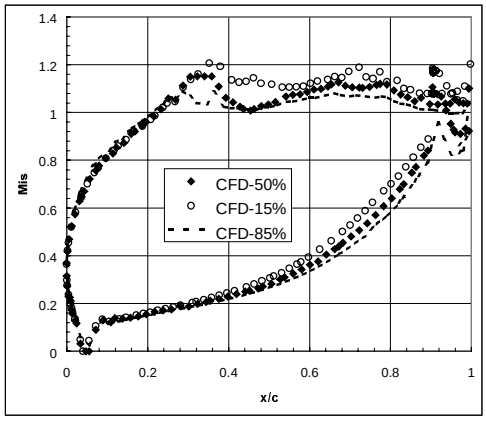

Figure 19: spanwise Mis profile

At hub and shroud the computed flow field evolves quite similarly and in both sections a strong horseshoe vortex develops ahead of the blade leading edge. The saddle point is placed upstream the blade as shown in the figures while, as expected, its two branches behave differently interacting downstream with the passage vortex. The pressure branch moves towards the opposite side of the vane and, after impingement onto the blade suction wall, it cancels the other branch of the horseshoe vortex. The same pattern is reported for the same IGV row by Sieverding *et al.* in Dunker, 1995. The angle at the blade leading formed by the stagnation streamline is of is about 10-11 degrees for both the hub and shroud sections. It has also to be pointed out that the prediction capability on distribution of Mis-isolines is well tested on such complex geometry, which involve separation at the blade pressure cut. The Mis profiles at three radial sections are finally reported in Fig. 19 for this case showing the same tendency also observed by Michelassi in Dunker, 1995. The isentropic Mach number peak of the suction side is larger in the tip than in the hub. After about 40% the axial distance on the pressure side, Mis values increase from hub to tip.

COMPUTATION WITH THE COOLANT EJECTION

For investigating the coolant jet ejection from the blade trailing edge, a first computation has been performed at midspan using a 2D mesh. The mesh topology follows the same ideas of the previous 2D tests and it is composed by 14000 elements. The geometrical detail at the injection point is shown in Fig. 20. The structured O type ring at the cut and at the trailing edge is interrupted and matched with the unstructured mesh. This solution has been applied considering that in these regions the flow is expected to be separated and fully turbulent. Further, the unstructured region allows a flexible mean for locally increasing the grid accuracy. It also couples the three different structured parts placed upstream in order to solve the coolant jet duct, the pressure and the suction side boundary layers.

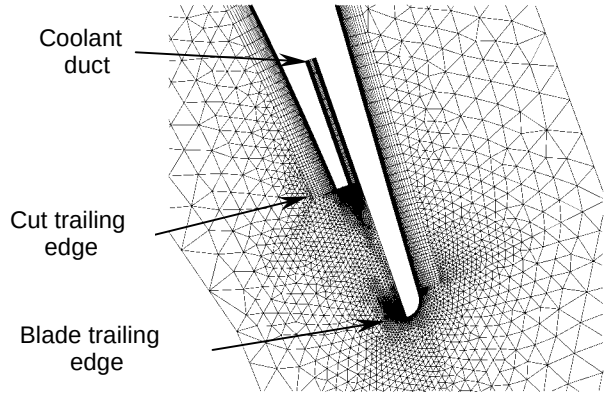

Figure 20: 2D grid at the leading edge

The steady flow pattern computed at the trailing edge is shown in the following figure for a 3% coolant flow rate of the total mass flow and an outlet Mis=1.05. The base pressure p_b resulting from the computation has been compared against the measurements carried out by Kapteijn *et al*. 1996. The measured value of the pressure coefficient $Cp = (p_b - p_{exit})/0.5 \cdot \rho_{exit} V_{exit}^2$ at the cut trailing edge is about 0.18 while the computation gives a slightly higher value of 0.2. At the blade trailing edge, the computation matches the same value of -0.06 as experiments.

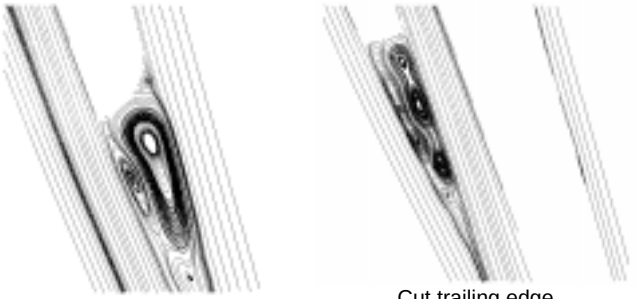

Figure 21: trailing edge streamlines

The isentropic Mach number of Fig. 22 shows the same tendency observed for the 3D computation without ejection.

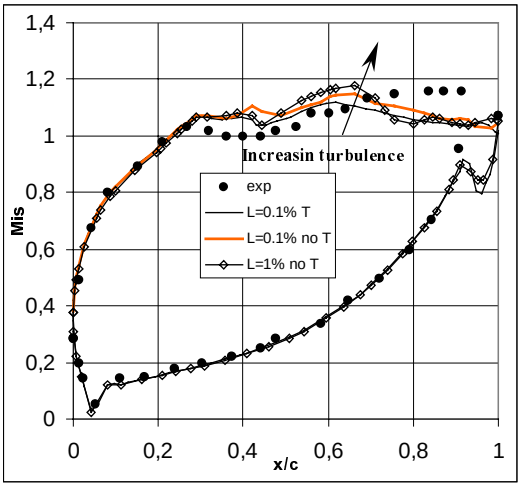

Figure 22: wall Mis profiles

According with the experiments the shock system at the blade cut is considerably reduced by the presence of the coolant jet as indicated by the Mis profile on the suction side for x/c=0.4. On the pressure side, the agreement is satisfactory and the higher base pressure obtained for the cut trailing edge is here confirmed. On the suction side for x/c=0.7 the flow decelerates while in the experiments the wall pressure level attains an almost constant value before the final shock. A similar effect was observed in Fig. 17 for the 3D computation without injection. The turbulence level proves to have some impact on the flow pattern in this region but despite different turbulent conditions were considered the flow shows almost the same tendency to anticipate the compression. All the modifications considered to the turbulence model (the transition model of Wilcox, T in the legend of Fig. 22, the Kato-Launder and the Durbin's correction) affect the solution on the suction side as far as they increase the turbulence eddy viscosity of the flow. In this regarda similar effect is obtained from the incoming turbulence length scale L (as percentage of the inlet pitch) used to define the inlet dissipation: $\omega \propto \sqrt{k}/L$. The turbulence intensity reduces the thickness of the boundary layer on the suction side allowing the flow to accelerate further after x/c=0.5 until a shock compression of increasing strength develops at x/c≈0.7. The flow pattern computed for this configuration is shown in Fig. 23 where the shocks developing in the vane are underlined. As also observed in the experiments, a first oblique shock starts from the blade trailing edge and moves towards the suction side for x/c=0.4. Then this shock is smeared and reflected further downstream against the blade wake. In the following sections, the flow accelerates until the shock placed close to x/c≈0.7. Finally, as also confirmed by the experiments, another shock detaches from the suction side trailing edge in order to match the same conditions of static pressure and velocity direction with the flow coming from the blade pressure side. On the suction side for x/c=0.3 the solution is controlled by the pressure side shock system originating from the trailing cut edge. In this regard, a preliminary unsteady computation has been carried out for the same test computing the trailing edge vortex shedding. From this computation, it has been observed that the wake structure modifies appreciably the shock system at the cut trailing edge. Owing to the vortex shedding interaction with the tail-fish shock, the unsteady pressure profiles on the suction side between x/c=0.3 and x/c=0.5 show a better agreement with experiments (Fig. 24).

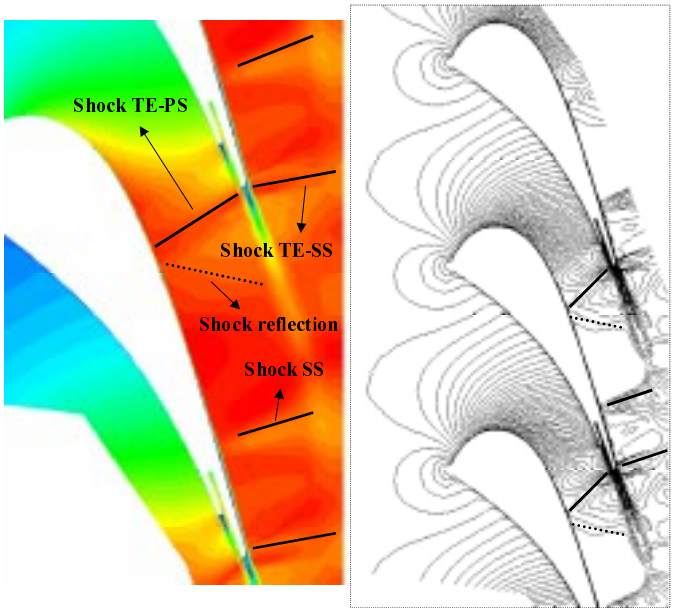

Figure 23: Mach number isolines

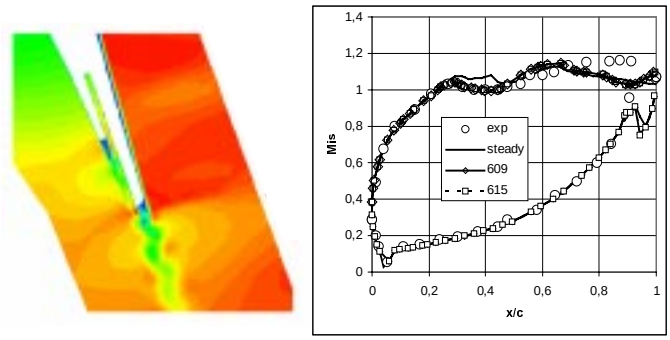

Figure 24: unsteady Mach number solution

The presence of the vortex shedding clearly impacts the pressure side branch of the blade TE shock improving the solution on the suction side for x/c=0.4 where this shock meets the wall of the neighbouring blade. The influence of this phenomenon appears to be limited to this effect while the pressure disagreement downstream x/c=0.7 remains almost unaffected by the flow unsteadiness.

CONCLUSIONS

The implicit unstructured solver HybFlow has been modified implementing a multiblock strategy. The approach developed is composed by three main phases needed to prepare the grid information, to compute the solution and to post-process the data. The vertical multiblock approach used with the solver HybFlow has been primarily designed to reduce the memory costs while keeping at the same time the high geometrical flexibility offered by the original solver.

In case of a well-defined block decomposition of the computational grid, the tests performed have demonstrated a quite remarkable improvement of both memory and CPU performances from the original solver. In fact a reduction of more than 70% has been achieved for the memory requirements while the CPU time saving has been kept in the range of about 50%. The unstructured hybrid features of the scheme have been considerably stressed in the present work and all the tests performed took advantage as much as possible from an adapted disposition of the discretization elements.

Accuracy of the numerical scheme has been addressed using 2D cascades with subsonic and transonic regimes. A 3D application of the procedure has been also considered. A well-adapted grid has been prepared for this testcase in order to investigate accurately the unstructured advantages offered by the numerical approach. The pressure profiles at midspan compared accurately with experiments while some inaccuracies observed have been attributed to the discretization of the flow within the trailing edge blade cut. The secondary flows structure has been represented at hub and shroud sections meeting the expected streamlines pattern. In this regard the hybrid unstructured meshing provides local adapted elements which are clearly suitable to capture the main flow aspects. Therefore, the hybrid approach has to be considered not only useful for memory reduction but also effective for an accurate characterisation of 3D flows. The coolant ejection has been also computed allowing a detailed simulation of the trailing edge area of the blade. In this case unsteady vortex shedding from the blade trailing edge proved to have an important role on the solution pressure profile of the suction side.

In conclusion, consistently with the turbulence model, the multiblock approach has proved to be robust, efficient and accurate in several flow configurations with relevant viscous effects.

AKNOWLEDGMENTS

This work was sponsored by EEC in the frame of the BRITE-EURAM project IMT Area 3, TATEF BRPR-CT97-0519. The

authors would like to thank Dr. P. Cardamone for his competent support. Mr. F. Rubechini is also acknowledged for the help in the 3D VKI mesh generation.

REFERENCES

Adami P., Michelassi, V., Martelli, F., 1998 “Performances of a Newton-Krylov scheme against implicit and multi-grid solvers for inviscid flows” AIAA paper 98-2429.

Adami, P., 1999 “Computations for internal flows with a low-mach preconditioned Newton-Krylov scheme”, Third IMECHE Conference, London, March, 1999.

Arnone, A. Liou, M.S. and Povinelli, L.A., 1993, “Multigrid Calculation of a Trhee-Dimensional Viscous Cascade” Journal of prop. And Power, Vol.9, No.4.

Arts, T, Lambert de Rouvroit M. and Rutherford, A.W., 1990, “Aerothermal Investigation of a Highly Loaded Transonic Linear Turbine Guide Vane Cascade” VKI tech. Not. 174.

Bario, F., Beral, C., 1998, “Boundary layer measurements on the pressure and suction sides of a turbine inlet guide vane”, Exp. Thermal and Fluid Sci.,17, 1-9.

Barth, T.J.,1991 "Numerical Aspects of Computing Viscous High-Reynolds Number Flows on Unstructured Meshes", AIAA Paper 91-0721, Jan.

Connel, S.D. and Braaten, M.E., 1995, “Semistructured Mesh Generation for Three-Dimensional Navier-Stokes Calculations” AIAA J., Vol. 33, No. 6, June.

Cuthill, E. and McKee, J., 1969, “Reducing the Band Width of sparse Symmetric Algorithms”, *Porc. ACM Nat. Conference*.

Dawes, W.N., 1992, “The Simulation of Three-Dimensional Viscous Flow in Turbomachinery Geometries Using a Solution-Adaptive Unstructured Mesh Methodology” Transaction of ASME Vol. 114, July.

Delanaye, M., Essers, J.A., 1997, “Finite Volume Scheme with Quadratic Reconstruction on Unstructured Adaptive Meshes Applied to Turbomachinery Flows”, J. of Turbomachinery, Vol. 119.

Durbin, P.A., 1996, ”On the *k*-ε Stagnation point Anomaly”, Int. J. Heat and Fluid Flow, 17,89-90.

Farhat, C. and Lesoinne, M., 1993, “Automatic Partitioning of Unstructured Meshes for the Parellel Solution of Problems in Computational Mechanics”, Int. J. for Num. Methd. in Eng., 36.

Frink,N.,T.,1991"Upwind Scheme for Solving the Euler Equations on Unstructured Tetrahedral Meshes", AIAA J.Vol.30,1.

Habashi, W.G., Fortin, M., Dompierre, J., 1997, “Anisotropic Mesh Optimization for Structured and Unstructured Meshes” VKI LS 1997-02 Von Karman Institute.

Haselbacher, A., McGuirk, J.J., Page, G.J., 1999, “Finite Volume Discretization Aspects for Viscous Flows on Mixed Unstructured Grids” AIAA J., Vol. 37, No. 2.

Jameson, A. and Baker, T.J., 1987, “Improvements to the Aircraft Euler Method”, AIAA paper No. 87-0452.

Kapteijn, C., Amecke, J., Michelassi, V., 1996 “Aerodynamic Performanche of a Transonic Turbine Guide Vane with Trailing Edge Coolant Ejection: Experimental Approach” ASME J. Of Turbomachinery, Vol. 118

Kato, M. and Launder, B.E., 1993, “The modelling of turbulent flow around stationary and vibrating square cylinders” 9th Symposium on *Turbulent Shear Flows*, Kyoto, Japan August.

Koiro, M. and Lakshminaranayana, B., 1998, “Simulation and Validation of Mach Number Effects on Secondary Flow in a Transonic Turbine Cascade Using a Multigrid, *k*-ε solver” Journal of Turbomachinery, April, Vol. 120.

Kwon, O.J. and Hah, C., 1995, “Simulation of Three-Dimensional Turbulent Flows on Unstructured Meshes” AIAA J. Vol.33, No. 6, June.

Mavriplis D.J. 1995, “Three-Dimensional Multigrid Reynolds Averaged Navier Stokes Solver for Unstructured Meshes” AIAA J., Vol.33, No. 3.

Michelassi, V. and Belardini, E., 1999, “Numerical Simulation of 3D inlet Guide Vanes”, Third IMECHE Conference, London, March.

Parthasarathy, V. and Kallinderis, Y., 1994, “New Multigrid Approach for Three-Dimensional Unstructured, Adaptive Grids” AIAA J.,Vol. 32, No. 5, May.

Pulliam T.,H., Rogers, S., Barth, T., 1996, "Practical Aspects of Krylov Subspace Iterative Methods in CFD", *Prog. and Challenges in CFD Methods and Algorithms*, Seville, AGARD CP-578.

R. Dunker Ed., 1995, “*Advances in Engine Technology*”, John Wiley and Sons.

Roe P.L., 1986 “Characteristic based scheme for Euler equations” , Ann. Rev. Fluid Mech., 18.

Saad, Y., 1994 "Krylov Subspace Thechniques, Coniugate Gradients, Preconditioning and Sparse Matrix Solvers", CFD VKI LS 1994-05 VonKarman Institute for Fluid Dynamics.

Saad, Y., 1994, "Krylov Subspace Techniques, Coniugate Gradients, Preconditioning and Sparse Matrix Solvers", CFD VKI LS 1994-05 VonKarman Institute for Fluid Dynamics.

Sieverding, C.H., 1994,”Recent Progress in the Understanding of Basic Aspects of Secondary Flows in Turbine Blade Passages”, ASME Paper 84-GT-78.

Slack, D.,C., Whitaker, D.,L., Walters, R.,W., 1994, "Time Integration Algorithms for the Two-Dimensional Euler Equations on Unstructured Meshes", AIAA Journal Vol. 32, No. 6, June.

Venkatakrishnan, V., 1995, "A Perspective on Unstructured Grid flow Solvers", Technical report AIAA Paper 95-0667, AIAA 33rd Aerospace Sciences Conference, Reno, NV.

Wilcox, 1993, D.C., “*Turbulence modelling for CFD*” DWC Industries, Inc., La Canada.

Wilcox, D.C., 1994, “Simulation of transition with a two-equation turbulence model” AIAA J., Vol. 32, No. 2, Feb.

