VISCOUS FLOW COMPUTATIONS USING STRUCTURED AND UNSTRUCTURED GRIDS, ON THE INTEL-PARAGON

D. KOUBOGIANNIS, K.C. GIANNAKOGLOU, K.D. PAPAILIOU
NATIONAL TECHNICAL UNIVERSITY OF ATHENS, ATHENS, GREECE

Abstract

A computational technique for the numerical solution of the Navier-Stokes equations is ported on the distributed memory *Intel-Paragon* computing system. A key element in the present work is that the solution method possesses the capability of using either structured or unstructured grids, through a common finite-volume discretization technique and an explicit time integration scheme. The parallelization of the method is based on the multi-domain concept, where each subdomain is assigned to a different processor. Different discretization algorithms for the control volumes along or close to the interfaces and consequently different communication techniques are employed, depending on the type of the grid, in order to minimize the inter-processor communication cost. The laminar flow around an isolated NACA0012 profile, at zero incidence, infinite Mach number equal to 0.5 and a Reynolds number equal to 5,000 is analyzed, by making use of up to 40 compute nodes.

I Method Formulation

In a Cartesian coordinate system, the mass, momentum and energy conservation equations are written in a vector form, as follows ([Hirsch90])

$$\frac{\partial \overrightarrow{W}}{\partial t} + \frac{\partial \overrightarrow{F^{inv}}}{\partial x} + \frac{\partial \overrightarrow{G^{inv}}}{\partial y} - \frac{\partial \overrightarrow{F^{vis}}}{\partial x} - \frac{\partial \overrightarrow{G^{vis}}}{\partial y} = \overrightarrow{0} \tag{1}$$

by separately introducing inviscid (superscripted by *inv*) and viscous (superscripted by *vis*) terms. The solution variable array is denoted by $\overrightarrow{W}$, while $\overrightarrow{F}$ and $\overrightarrow{G}$ stand for the flux vectors in the x and y directions, respectively.

Equation 1 is integrated over control volumes which are formed around any grid node. For both structured and unstructured grids, the finite volumes are defined in a similar way, by successively connecting the midpoints of the segments incident upon the node at hand, with the barycenters of the (triangular or quadrangular) grid elements surrounding this node. In figure 1, the finite volumes for the structured and the unstructured meshes are denoted by the dashed areas.

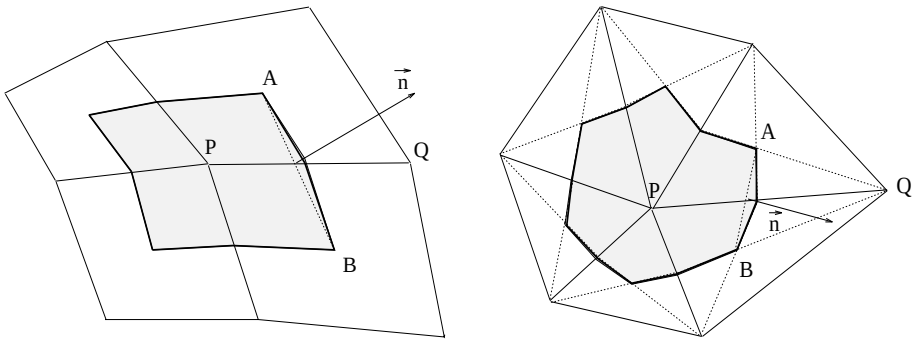

Figure 1 Grid nodes and grid cells, as well as finite volumes for structured and unstructured meshes

For the integration of equation 1 over each control volume C_P the Green's theorem is used, so

$$\iint_{C_P} \frac{\partial \overrightarrow{W}}{\partial t} dxdy + \sum_{Q \in K(P)} \int_{\partial C_{PQ}} (n_x \overrightarrow{F^{inv}} + n_y \overrightarrow{G^{inv}} - n_x \overrightarrow{F^{vis}} - n_y \overrightarrow{G^{vis}}) d\gamma = \overrightarrow{0} \quad (2)$$

where K(P) stands for the set of grid nodes which are directly linked to node P. The summation $\sum$ applies over the segments AB (figure 1) which correspond to the ∂C_{PQ} part of the control volume boundary ∂C_P. Associated with each part ∂C_{PQ}, which constitutes the interface between cells C_P and C_Q, is the normal outward vector $\overrightarrow{n} = (n_x, n_y)$, its length being equal to the length of the straight segment AB.

For the steady flows considered herein, the discretization of the above equation leads to

$$(\frac{\partial \overrightarrow{W}}{\partial t})_P \iint_{C_P} dxdy + \sum_{Q \in K(P)} (\overrightarrow{\Phi_{PQ}{}^{inv}} - \overrightarrow{\Phi_{PQ}{}^{vis}}) = \overrightarrow{0} \quad (3)$$

where the surface integral is equal to the area of the volume C_p, while $\overrightarrow{\Phi_{PQ}}$ is the numerical approximation of the flux directed from P to Q. The approximated flux vectors, either inviscid or viscous, are written in a compact form as

$$\overrightarrow{\Phi_{PQ}} = \frac{1}{2}(\overrightarrow{\Phi_P} + \overrightarrow{\Phi_Q}) + \overrightarrow{T_{PQ}} \quad (4)$$

The term $\overrightarrow{T_{PQ}}$ is zero for the viscous fluxes (this corresponds to the standard central differencing scheme, for the structured grids). For the inviscid fluxes, the Roe flux difference splitting scheme is used ([Roe81]), according to which the extra term $\overrightarrow{T_{PQ}}$ is calculated as follows

$$\overrightarrow{T_{PQ}} = \frac{1}{2} |A^*| (\overrightarrow{W_P} - \overrightarrow{W_Q}) \quad (5)$$

In equation 5, A^* denotes the flux jacobian which is built using the Roe-averaged quantities and is extended to second order accuracy through the standard MUSCL extrapolation technique ([vanLeer81]) in structured grids or an appropriate implementation of the latter for unstructured grids ([Fezoui89]).

In the structured grids, all nodes are swept and inviscid and viscous fluxes are computed along the right and the upper edges of the corresponding control cells. In the unstructured grids, the inviscid fluxes are computed via a loop over segments, during which contributions at their two nodal edges are collected. For the same grids, viscous fluxes are calculated by sweeping over triangular elements and collecting contributions to their three nodes. All quantities are stored at grid nodes.

The time derivative term in equation 3 is discretized by using the forward Euler scheme. Consequently, the solution is updated at each node in an explicit manner. Local time-stepping is used. Boundary conditions are employed through the adherence condition along the solid walls, while the fluxes leaving or entering the domain along the farfield boundaries are computed through a proper implementation of the Steger-Warming scheme ([Steger81],[Fezoui89]).

II Parallelization Aspects

1 The *Intel-Paragon* Computing System

The *Intel-Paragon* is a scalable distributed-memory multi-processing system. Its network topology is planar and supports Multiple Instruction/ Multiple Data (MIMD) applications. It is based on *Intel*'s i860XP/S RISC processors with a high-speed inter-connection network. In the present application, the *Intel-Paragon* machine is used in a Single Program/Multiple Data (SPMD) mode.

This computer has been installed in the Supercomputing Center of the National Technical University of Athens (NTUA) and is equipped with 51 nodes having a theoretical peak performance of 48x75 Mflops, in double precision computations. In the current configuration there are 48 compute (lying on a two-dimensional 4x12 mesh backplane) and 3 service nodes. Each node consists of two i860XP/S RISC processors, the first of them being the application processor while the other stands for the message processor. Both processors share the same memory, which is equal to 32 Mb per node with a cycle time of 50 MHz. The message processor is responsible for the message-passing operations, its role being to send to and receive messages from the other nodes via the network interface. The application processor is carrying out the primary work. Two RAID disks of 4.8 Gb each are connected to the two I/O nodes with SCSI interfaces.

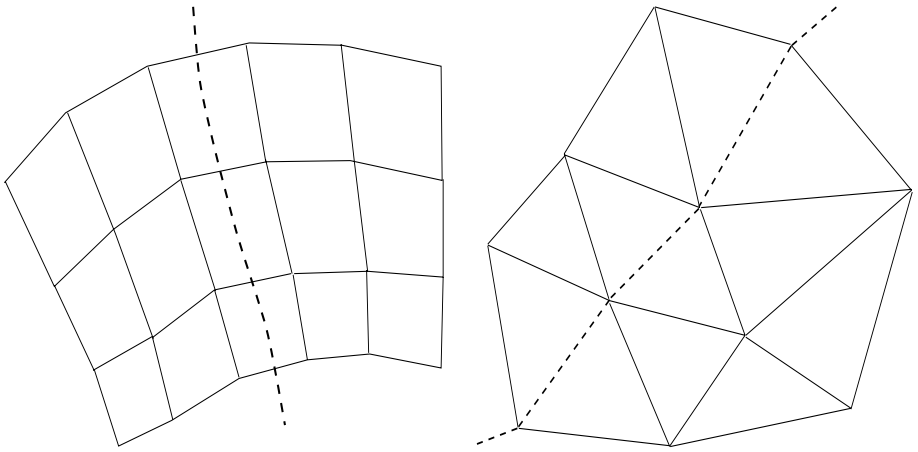

Figure 2 Grid partitioning for structured and unstructured meshes

In the configuration used in the present study, the 100 x 100 Linpack benchmark, executed on 36 processors, resulted to the following figures: $R_{max} = 1.558$ $Gflops$, $N_{max} = 10500$, $N_{1/2} = 2600$ and for the 1000 x 1000 Linpack: $R_{1000} = 0.237 Gflops$.

The interprocessor communication is carried out through a row-column routing. Since the disk I/O takes place via the same communication network ,it burdens the communication cost. Nevertheless, the communication time is only slightly affected by the number of nodes interfering between two communicating compute nodes. The communication cost is estimated as the cost of sending and receiving information between compute nodes. For the exchange of a double-precision variable message, this communication cost consists of a latency (being equal to about 65 μsec, regardless of the message length) in addition to a linear part which is proportional to the size of the message (the proportionality factor being equal to about 0.12 μsec per double-precision variable) ([Arbenz94]). The type of communication to be used is defined by the user. In the present work the sunchronous communication mode has been used by exploiting the existing NX library.

2 Grid Partitioning and Parallel Implementation

The approach adopted herein for the parallelization of the solution algorithm is based on the domain-decomposition concept. The global grid is first partitioned into a number of subdomains each of which is assigned to a different processor.

For the structured grids, the global mesh is partitioned by defining 'equidistant' (measured in terms of the number of grid lines in between them) grid lines, as interfacing boundaries. The subdomains are defined as in figure 2, where the dashed line plays the role of the interfacing boundary between the two adjacent

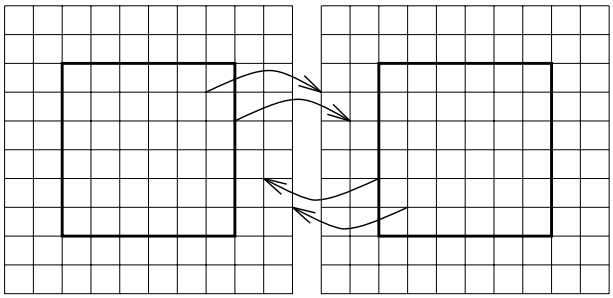

Figure 3 Communication patterns for two adjacent subdomains (and consequently processors). Structured mesh, second order accuracy

subdomains. As shown in figure 2, adjacent subdomains do not share common nodes, while each control volume strictly belongs to a single subdomain. The grid of each subdomain is perimetrically extended using two rows of fake nodes; the nodes along the first row coincide with the boundary nodes of the adjacent domains. The second row allows the application of second order accurate schemes. The communication task is associated with these two rows of nodes in order to properly assign them the dependent variables calculated at the boundary and near-boundary grid lines of the adjacent subdomains. Thus, at the end of each iteration, the dependent variables at the fake nodes of each subdomain are updated by receiving information from the adjacent domains. In this way, the numerical fluxes for all the nodes within each subdomain are computed using values already loaded to the corresponding processor. By retaining a well ordered communication schedule, any internal subdomain exchanges information only with four adjacent subdomains (east, west, north and south), while communication with four additional subdomains (in the cross directions, line north-west etc.) is avoided. Figure 3 shows the information exchange between two adjacent subdomains, in an illustrative manner.

For the purposes of the present study, the unstructured grid was generated by subdividing each quadrangular element of the structured grid into two triangles. The partitioning of the unstructured grid was carried out on the basis of the structured grid. Consequently, all internal subdomains communicate with four adjacent ones (by means of a common set of edges) and with four other subdomains (by means of a single common node). The control volumes corresponding to the interface nodes split among two or more adjacent subdomains (figure 2).

For the unstructured grids, the communication takes place in two phases. In the first phase, the gradients of the four primitive variables (to be used within the second order accurate scheme) and as well as time-steps are exchanged over each interface node. At the end of the first phase, the fluxes computation is taking place. However, the so calculated fluxes over the interface nodes which belong to more than one subdomains, are incomplete. For this reason, in the second phase, the four fluxes per node are exchanged. By adding these fluxes,

interface nodes are separately updated, as members of the subdomains where they belong to.

Due to the way the subdomains have been defined, the connectivity of either structured or unstructured subdomains retains the exact topology of the interprocessor communication network. Since C-type grids are in use, the aforementioned rule is perturbed only along the split line springing from the trailing edge, but this is of minor importance.

The above partitioning methods fulfill the standard requirements of equiloading the processors while minimizing the interprocessor communication.

III Assesment of the Parallel Method

The case used to assess the numerical tool ported on the *Intel-Paragon* is concerned with the two-dimensional flow around a NACA0012 profile. The flow is considered to be laminar, with a free stream Mach number of M=0.5 and zero incidence. The Reynolds number based on the free stream conditions and the airfoil chord, is 5000. The flow is subsonic and it is characterized by a tiny separation bubble near the trailing edge. The Reynolds number for this case approaches the upper limit for steady laminar flows prior to the onset of turbulence. Zero heat flux is prescribed along the airfoil surface. Numerical results for this case are available in [Mavriplis89].

As discussed in a previous section, the starting point for all grids used in the calculations is the same structured grid. This is a C-type mesh, generated using a standard hyperbolic grid generation method ([Rizzi81]). The grid dimension is 283x50 and the leading edge is located at the 44th node. The distance of the first grid node off the wall is equal to 0.2 percent of the chord, while the outer boundary is placed at a distance of 75 chords from the airfoil.

Some numerical aspects will be elaborated first. The solution obtained on the basis of the unstructured grid and a second order accurate scheme has been proved to be stable. On the contrary, the second order calculation on the structured grid appeared to be unstable and the instability occured after some thousands of iterations. The resulting unsteady flow, was attributed to the very high Reynolds number of the case examined. From a numerical point of view this is a consequence of the insufficient amount of artificial dissipation added by the upwind scheme itself, in this case. As expected, the first order solution scheme was always stable. So, in order to overcome the instability problems, a hybrid scheme was finally implemented. Thus, for the computation of the inviscid fluxes over the midnodes, using the equation

$$\overrightarrow{\Phi_{PQ}}^{\overrightarrow{inv}} = \overrightarrow{\Phi^{inv}}(\overrightarrow{W_P}, \overrightarrow{W_Q}) \tag{6}$$

a blend of first and second order accurate schemes was applied. This was realized, in practice, by expressing the solution variables array at the nodal points as

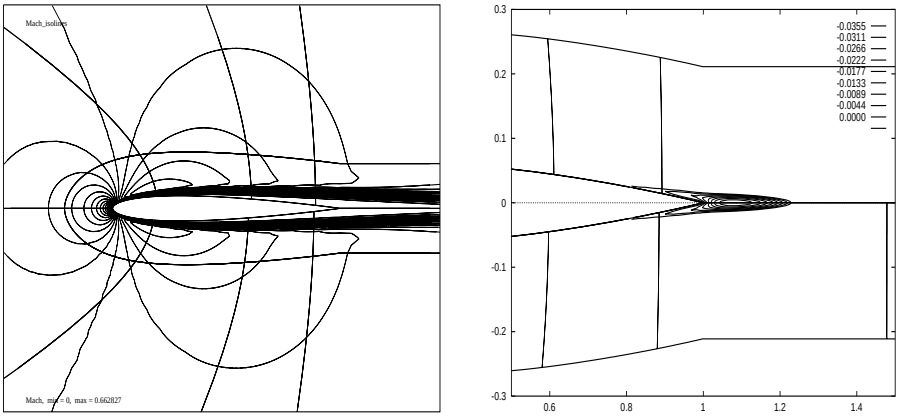

Figure 4 Mach number isolines in the vicinity of the airfoil and non-dimensional u-velocity contours close to the trailing-edge

$$\overrightarrow{W_P} = (1-\omega)\overrightarrow{W_P^{(1)}} + \omega\overrightarrow{W_P^{(2)}} \tag{7}$$

where $\overrightarrow{W_P^{(1)}}$ and $\overrightarrow{W_P^{(2)}}$ stand for the first and second order accurate approximations of the corresponding variable vectors. A similar expression is used for $\overrightarrow{W_Q}$ as well. The value $\omega = 0.8$ was found to be the maximum one which allowed a stable solution, while providing a very satisfactory comparison with some reference results.

Results to be used for the validation of the approach adopted herein are then presented. These results have been obtained by running on 30 processors for both the structured and the unstructured grids.

Figure 4 illustrates the iso-Mach contours in the vicinity of the airfoil and the negative non-dimensional u-velocity contours, in order to make clear the separation region that appears close to the trailing edge of the airfoil. Its origin is located at the 80 percent of the airfoil, which is in accordance with the reference calculations [Mavriplis89].

In figure 5, the predicted pressure C_p and friction C_f coefficient distributions along the airfoil wall are shown. For the C_p distribution, results obtained through the structured and the unstructured grids are illustrated. For the sake of convenience, the C_f distribution was also obtained using only the structured grid. The predicted distributions are compared with the reference results [Mavriplis89].

Figure 6 shows the convergence history of the structured grid algorithm for the flow case under consideration, running on 30 processors. Since the method is explicit, its convergence history is identical to that of the sequential algorithm running on a single processor.

Finally, the speedup S_p and the efficiency E versus the number of processors N_p are tabulated in Table 1 and plotted in figure 7, for both structured

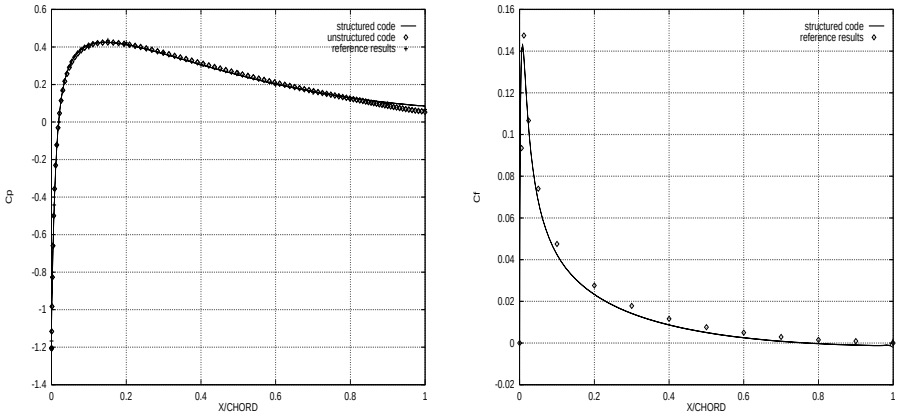

Figure 5 Pressure and friction coefficient distributions along the airfoil wall

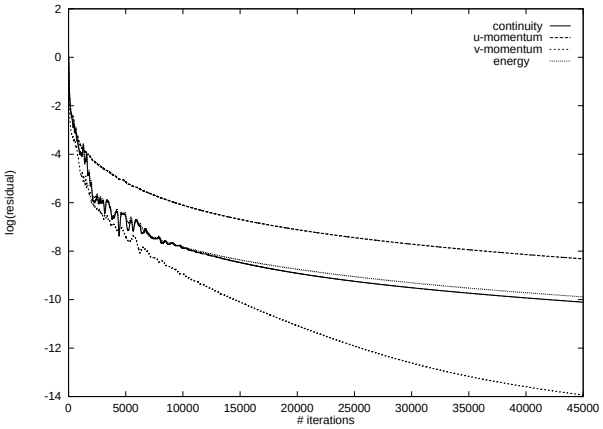

Figure 6 Convergence history for the structured grid on 30 processors

and unstructured grid computations. Table 1 was filled in after performing 5000 iterations, which is a typical number of iterations required for a convergence of about five orders of magnitude. The structured grid computation presents a slightly better speed-up. This can be attributed to (a) the fact that the code for structured grids is 25 % slower than the unstructured one when the same number of nodal points are used (this is due to the different ways used for the calculation of the viscous terms) and (b) the increased communication required by the unstructured grid concerning neighbouring subdomains in the cross direction. In this table, T_{ca} is the CPU time required for the calculation, T_{el} is the elapsed time and T_{co} is the communication time spent for the communication task. All these times have been measured in seconds.

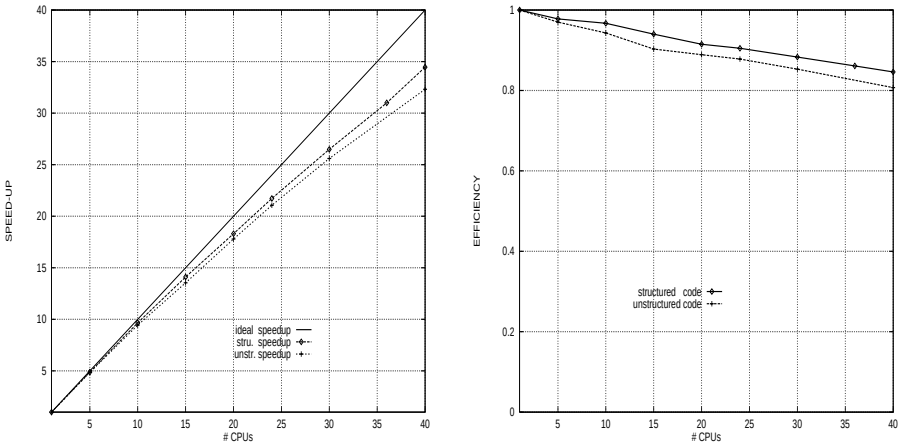

Figure 7 Speed-up and the efficiency curves for both structured and unstructured grid computations

	Structured grid					*Unstructured grid*				
N_p	E	S_p	T_{ca}	T_{el}	T_{co}	E	S_p	T_{ca}	T_{el}	T_{co}
1	1	1	40302	40302	0	1	1	30313	30313	0
5	0.98	4.89	41208	9000	822	0.97	4.85	31244	6680	940
10	0.97	9.67	41680	4800	1246	0.94	9.43	32128	3480	1819
15	0.94	14.10	42870	3550	2572	0.90	13.55	33569	2410	3243
20	0.91	18.30	44040	3000	3986	0.89	17.79	34084	1860	3789
24	0.90	21.72	44520	2650	4478	0.88	21.06	34536	1620	4213
30	0.88	26.48	45660	2400	5496	0.85	25.60	35520	1330	5214
40	0.85	34.46	46782	2233	7112	0.81	32.32	37519	1160	7245

Table 1 Calculation and communication time (sec) for 5000 iterations

IV Conclusions

An explicit, finite-volume solution algorithm for the solution of the compressible Navier-Stokes equations has been migrated to a distributed memory *Intel-Paragon* parallel computer. Aiming at a better understanding and assessment of the parallelizing techniques used, the laminar flow around an isolated profile was examined using either structured or unstructured grids. The unstructured grid calculations were performed by generating a pseudo-unstructured grid, resulting from the splitting of the C-type structured grid elements into triangles. Depending on the type of the grid, the solution and the communication algorithms are different, aiming at an optimum efficiency per case. For the structured grids, each domain is perimetrically extended using two rows of fake nodes, which

resulted to increased computational loading per subdomain and minimum communication. The upwind scheme used in conjuction with a structured grid was proved to be insufficient for a stable solution in the examined high Reynolds number case and a hybrid scheme, formulated by merging first and second order accurate schemes, was used instead. In the case of unstructured grids with second order accurate schemes, better efficiency was obtained through a two-phase communication scheme.

Acknowledgment

This work constitutes the contribution of the Laboratory of Thermal Turbomachines of NTUA in the Workshop entitled "MPP for Navier-Stokes Flows" held during the last phase of the ECARP Project (Concerted Action 4.2 "Cost Effective Solutions of the Compressible Navier-Stokes Equations on Massively Parallel Computers").

Bibliography

[Hirsch90] *Hirsch, C., (1990) :* , Numerical Computation of Internal and External Flows, John Willey & Sons

[Roe81] *Roe, P., (1981) :* , Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes Journal of Comput. Physics **43** , 357-371

[vanLeer81] *van Leer, B., (1972) :* , Towards the Ultimate Conservative Difference Scheme I : The Quest of Monotonicity Lecture Notes in Physics, **18** , 163

[Fezoui89] *Fezoui, L., et al., (1989) :* , Resolution Numerique des Equations de Navier-Stokes pour un Fluide Compressible en Maillage Triangulaire INRIA Report, No 1033, Programme 7

[Steger81] *Steger, P., Warming, R., F., (1981) :* , Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to the Finite-Difference Methods Journal of Comput. Physics **40** , 263-293

[Arbenz94] *Arbenz, P., (1994) :* , First experiences with the Intel Paragon Speed-up Journal **8** , No 2 , 53-58

[Mavriplis89] *Mavriplis, P., Jameson, A. and Martinelli, L., (1989) :* , Multigrid Solution of the Navier-Stokes Equations on Triangular Meshes ICASE Rep **89-11**

[Rizzi81] *Rizzi, A., (1981) :* , Computational Mesh for Transonic Airfoils, GAMM Workshop, Numerical Methods for the Computation of Inviscid Transonic Flows with Shock Waves, Notes on Numerical Fluid Dynamics **3** , 222-263