

VON KARMAN INSTITUTE FOR FLUID DYNAMICS

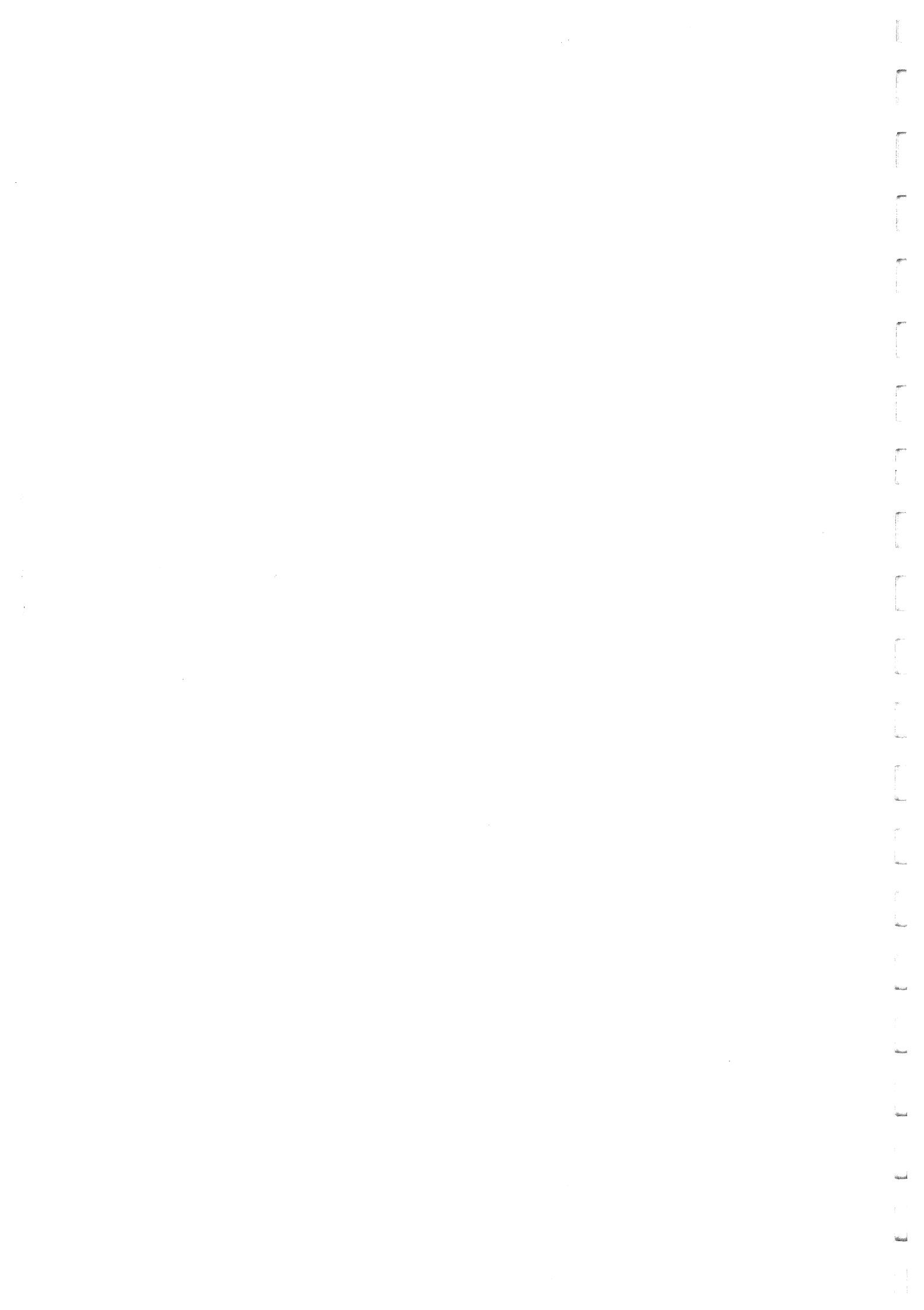
LECTURE SERIES 1988-05

COMPUTATIONAL FLUID DYNAMICS

MARCH 7 - 11, 1988

AN IMPLICIT APPROXIMATE FACTORIZATION FINITE VOLUME  
TECHNIQUE WITH IMPROVED ACCURACY FOR  
THE COMPRESSIBLE NAVIER-STOKES EQUATIONS

Mart Borsboom  
VKI, Belgium



## ABSTRACT

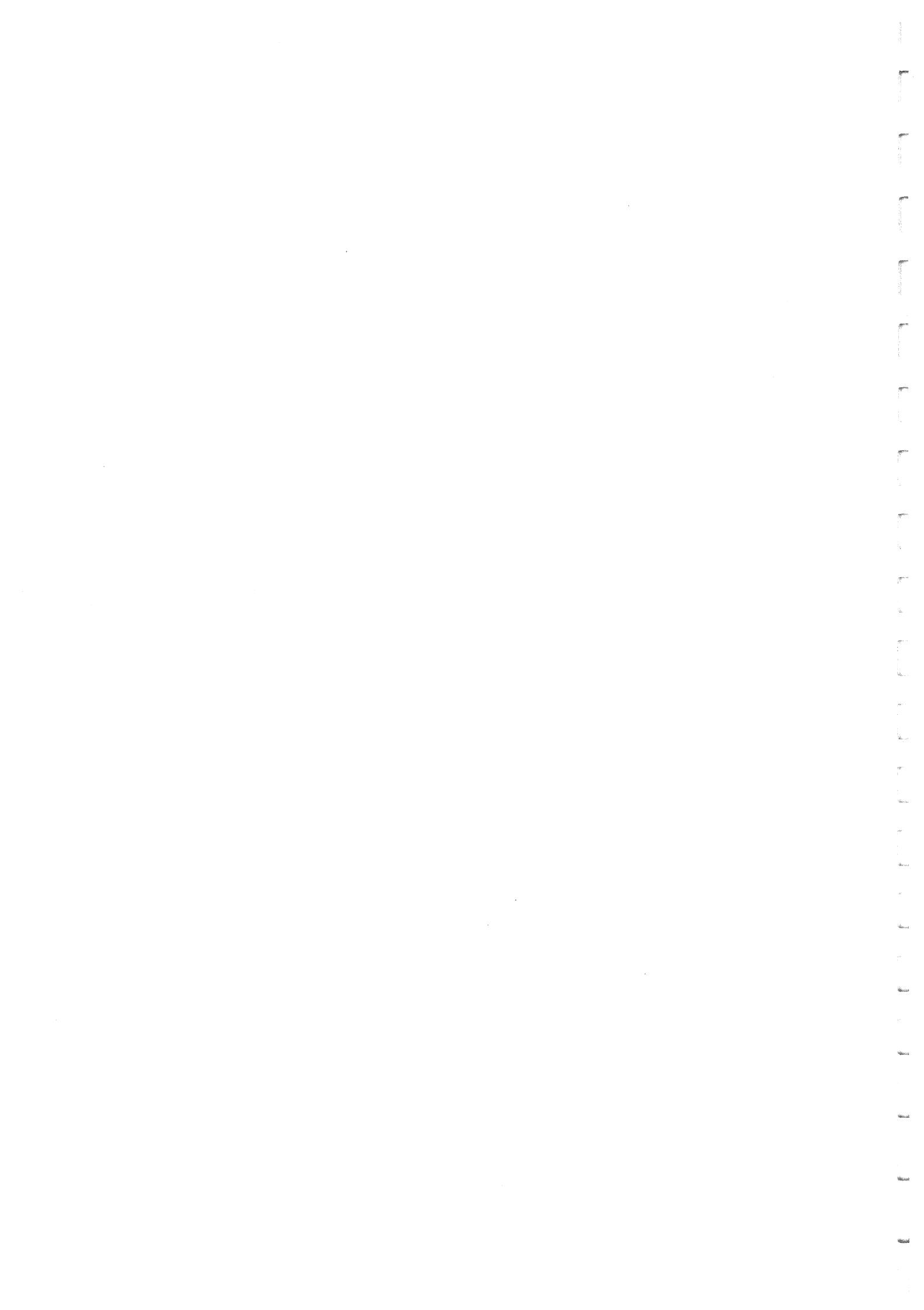
Any discretization method may be considered as a technique to transform continuous space operators into a finite set of algebraic equations that (locally) prescribe a relation between a discrete number of unknowns evaluated at nodal points or in volumes. The errors in this type of numerical approximation lead to truncation error terms in the equivalent equation, and it is argued that the magnitude of the truncation error, rather than the order of accuracy of the discretization method, is significant for the determination of the accuracy of the method.

In the case of a conservative discretization scheme, these truncation error terms can be determined by means of Taylor series expansions in the Gauss' quadrature point(s), because the integral form of the differential problem may be approximated by a discretization, as well as by an appropriate Gauss' integration rule. Hence a conservative discretization scheme is equivalent to a (sum of) finite difference discretization(s), and this has been verified both analytically and experimentally, for low-order as well as for higher-order methods.

This concept has been used to develop an optimal low-order conservative scheme on a three-times-three point computational molecule, which is almost free of numerical diffusion, even on highly distorted grids, while still relatively easy to program. The scheme is therefore considered to be a useful compromise for Navier-Stokes calculations. This has been confirmed by the results that have been obtained so far.

An implicit time-marching technique has been used as an iterative solver for the discretized system of equations. It can easily be shown that the backward Euler scheme has the best convergence properties. For two-dimensional applications, however, this scheme has to be linearized and factorized, in order to reduce the resulting nonlinear system of equations to a series of linear block-tridiagonal systems, for which efficient direct solvers exist. An analytical expression for the optimal time step was derived and applied everywhere on the domain (local time stepping), leading to optimal convergence properties of the scheme. Excellent results were obtained with the calculation at moderately high Reynolds number of a steady subsonic viscous flow over a backward facing step and through a cascade of NACA0012 profiles.

The method has been extended for the efficient numerical simulation of unsteady flow by Markus Kloker. A new approach has been tested, in which the numerics has been uncoupled from the physics. In this way, the numerical stability limit on the physical time step could be removed, leading to very efficient unsteady calculations. This has been demonstrated by the calculation of the self-induced vortex shedding behind a cascade of circular cylinders.



## CHAPTER I

### ***INTRODUCTION***

When somebody would like to develop a numerical simulation of some physical phenomenon, he has to be aware of the fact that his simulation will always be an approximation of what is happening in reality. First of all he has to find himself a sufficiently accurate mathematical formulation of the phenomenon, which automatically brings about the necessity of a compromise to be found between the complexity and completeness of the formulation on one hand and the number of simplifications to be made on the other hand. Depending on the type of phenomenon our fellow wants to study, one or more simplifying assumptions may probably be introduced without deteriorating too much the accuracy of the mathematical model, but others will at least strongly limit the usefulness of the chosen formulation. As for applications in the area of fluid dynamics we may think of simplifications like ignoring the variations in the third dimension, the compressibility effects and the viscous forces, and assumptions like constant total enthalpy and stationary flow, with or without using the Reynolds-averaged Navier-Stokes equations with a suitable turbulence model.

Once the mathematical model has been established, the model equations have to be transformed by some means into something that may be considered as their numerical counterpart. In order to obtain this, the equations have to be discretized, since a computer can handle only a finite number of variables and unknowns.

Here, we are faced with a second problem, since a general and accurate numerical representation of the model equations, and a simple and

straightforward discretization method, are two things that do not match very well. The purpose of the CHAPTERS II and III is to treat this problem to some extend for the case of the steady-state Navier-Stokes equations.

It is a well-known fact that we do not know very much about the exact solution of the Navier-Stokes equations. Exact solutions are known only for some particular cases, all laminar, such as certain types of boundary-layer flow and duct flow [1, 2]. Asymptotic techniques have been used with some success for the development of approximative and local solutions [3, 4, 5]. Some theory has been developed concerning the existence and uniqueness of solutions of the incompressible Navier-Stokes equations [6], but very little can be said analytically about the solution of the Navier-Stokes equations in general [1].

Yet, the solution of the Navier-Stokes equations is considered to be very important, because they form the mathematical model that describes in a very general way the behavior of a compressible, viscous flow in any domain, as long as the fluid may be considered as a continuous medium. As such they contain valuable information concerning all kinds of convective, viscous and nonlinear effects that occur in a flow. However, we would like to have this information in a more explicit form, in order to be able to predict in a quantitative way things like transport phenomena, losses, body forces, shocks, dispersion, etc. It is generally believed and accepted that the solution of the Navier-Stokes equations exists and is well determined under a wide variety of conditions, since viscous, compressible flows exist and can be found everywhere around us. Nature is solving the equations every day, and we see it happening around us, without being able to fully understand what is going on.

Given the reality that we cannot solve the Navier-Stokes equations analytically, we may at least try to solve them in a rough manner, by means of numerical approximation techniques. A very long list of references could be given here, but we will content ourselves by refering the reader to some of the more recent papers [7-34], and to the excellent, but not very recent review of Peyret and Viviand [35]. However, some of the basic requirements for an accurate numerical simulation may already be formulated.

First of all, the flow domain in its discretized form should be able to reveal all the flow details we are interested in, and which have been retained in the model equations. But also the set of differential equa-

tions, once discretized, should be such that the most essential properties of the original equations are conserved.

It will be clear that this is not always possible. For example, for the description of all the small-scale, unsteady structures in a turbulent flow field, down to the smallest eddies, we would need such an enormous amount of grid points and time levels, that no existing computer would be able to handle all that data. Also very steep gradients like vortex sheets, shocks, boundary layers and shear layers can only be represented on very fine grids, which in practice is (still) impossible. It is common practice to smooth out such details, by means of some added artificial and/or turbulence viscosity, and we may consider this as part of the mathematical modeling process of the physics. Such techniques are very useful, and permit an acceptable representation on a relatively coarse grid in the sense that some flow details that are supposed to be of less or only local importance are sacrificed in order to drastically reduce the computational effort.

Summarizing, we may state that a computational method should be such that it is capable of generating a good approximation to the exact solution of the differential problem at hand (provided that such a solution exists), at a minimal cost. In the case of the Navier-Stokes equations, this means that an easy-to-program, low-order discretization scheme, that is generally applicable and rather insensitive to the grid structure, is to be preferred. Especially this last point is of crucial importance [36]. The number of unknowns is proportional to the number of grid cells, and should be kept as small as possible in order to minimize the computer time needed to solve the discretized system of equations. On the other hand, the grid should also provide a sufficiently high resolution [37]. If we consider for a moment the complicated structures that may appear in compressible, viscous flows, we get a good appreciation of what that signifies for the grid design.

In many fields of application, small regions of steep gradients may be present in the flow domain, even when the variables have been Reynolds-averaged. To capture such details numerically, local concentrations of grid points are necessary, while at the same time the total number of points should be minimized. This can only be achieved by using highly distorted grids, but then a low-order scheme may not be sufficiently accurate anymore [36, 38-40]. This makes the generation of grids that are suited to numerical viscous flow simulation a very difficult

task, and in practice one always tries to keep the grid as smooth as possible [7-34, 36-39]. It will be clear that the optimal way to handle this problem will be some sort of compromise, and we cite Thompson, who says that "at the same time that effort is made to generate better grids, a similar effort should be made to develop hosted algorithms that are more tolerant of the grids" [36].

In these notes we propose to discuss to some extend the numerical modeling of the steady-state Navier-Stokes equations on an arbitrary grid, and we will present a technique that was found to be very useful in determining the "optimal" compromise. The equations may be in their original or in a more simplified form, the latter being obtained by including in the equations models for steep-gradient phenomena such as turbulence, boundary layers and shocks.

Once the numerical model has been established, obtained upon applying on a suitable grid the derived low-order discretization method to the physical differential problem, the task remains of solving the resulting system of nonlinear algebraic equations. The development of an efficient iterative method that can handle this problem will be the subject of the CHAPTERS IV and V. The method is essentially a so-called *time-marching* scheme, which are nowadays very popular as iterative solution techniques for the discretized steady-state Euler and Navier-Stokes equations.

Time-marching schemes consist of a discretization of the *unsteady* flow equations, and obtain the solution of the stationary equations by numerically simulating the physical decay of the perturbations introduced in the system by the initial conditions. They make use of the fact that the unsteady flow equations are hyperbolic/parabolic in time [35], so the initial-value problem is well posed. The discretization of the time derivative is in general rather inaccurate and not always consistent with the physics, but this is irrelevant for steady-state calculations. Normally, the time derivative should vanish once a steady state is reached, so the convergence of the scheme may be improved by a suitable *artificial* discretization of this physical term.

The method that will be proposed here is an implicit time-marching scheme, (approximately) factorized in order to reduce the amount of work per iteration step considerably, while still maintaining a reasonable convergence speed. Implicit approximate factorization schemes have been

used already before for Navier-Stokes calculations by Beam and Warming, McDonald et al., Pulliam, Dawes, and many others [12-20, 41-44], but in all these cases simple finite differences or finite volumes were used for the discretization of the space derivatives. The implicit approximate factorization concept had to be generalized, in order to combine this iterative technique with the spatial discretization methods that were developed in the first part of this work.

A scheme with good convergence properties was obtained by optimizing the size of the time step analytically. The validity of such an analysis is unfortunately rather limited due to the severe simplifications that have to be made, since analysis tools for the convergence properties of general nonlinear (and even linear) algebraic systems are quasi nonexistent. Yet the optimal time step turned out to be predicted quite well by this analysis, notwithstanding its limitations. It is still necessary to gain more insight in this matter by means of heuristic reasoning and extensive testing of the new scheme.

As for the numerical solution method for the *unsteady* Navier-Stokes equations, we have used a second-order accurate, implicit discretization in time (trapezoidal rule), which leads again, once the space operators have been discretized also, to a system of nonlinear algebraic equations, to be solved this time at every physical time level. To this end, the same implicit approximate factorization scheme has been used as for the steady-state flow simulation, by adding to the equations a second *unphysical* time derivative.

In comparison to the classical approach of using a time-marching scheme in a time-accurate way, this technique has the advantage that the discretization in the *physical* time depends only on the physical problem at hand. The restrictions to the time step of a classical time-marching scheme are in general much more severe, because of stability conditions or factorization errors. In that case, the time step is not limited anymore by the *physics*, but only by the *numerics*. This is especially true when highly stretched grids in space will be used.

It will be shown that due to these numerical restrictions, the classical numerical simulation of unsteady, viscous flows by means of a time-marching method should be considered as *inefficient*. The details of this study, as well as the newly developed, time-accurate scheme, have been reported by Kloker [45], which reference has been included in these

lecture notes. Notice that this new technique of unsteady numerical flow simulation has the additional advantage of allowing a full optimization of the discretization of both the physical and the numerical time derivative, both in space and in time, since both effects have been completely uncoupled.

## CHAPTER II

### ***LOW-ORDER CONSERVATIVE SCHEMES ON A NON-UNIFORM GRID***

It is a straightforward matter nowadays, to design discretization methods for use on a uniform grid, that are both accurate and simple. In practice, however, the usefulness of uniform grids is rather limited, especially for Navier-Stokes calculations. We will therefore analyse in this chapter the use of some one-dimensional schemes on a non-uniform grid, and extend the analysis to discretization methods for two-dimensional differential equations (e.g. the Navier-Stokes equations) on non-uniform grids in the next chapter.

Some loss in accuracy may be expected, and our objective is to limit this loss as much as possible, while still keeping the discretization scheme relatively simple. For this reason, only low-order discretization schemes will be investigated, that discretize the equations on a computational molecule of (three times) three grid points. For a one-dimensional non-uniform grid, such a molecule is given in FIGURE 1, together with the notations that will be employed. A *low-order scheme* will be defined as a discretization scheme that is obtained using an at most piecewise linear representation of the unknowns, whereby the order of the equations is normally reduced by writing the equations in an integral form and applying Gauss' theorem.

We will restrict our investigations to centered discretization schemes. Non-centered schemes always introduce some form of artificial dissipation in the numerical model of a convection-diffusion equation, that may obscure the physical dissipation effects completely [46, 47]. However, it seems that at present it is still not possible to construct

and to use computational grids that are fine enough to avoid the necessity of the introduction of any artificial dissipation, when high Reynolds number flows are to be simulated numerically. In other words, "a suitable finite difference scheme for a high Reynolds number flow will be one that maintains good accuracy in the convection-dominated region; in the regions where the convection terms and the viscous terms in the original differential equation should balance each other, which cannot be done exactly numerically, the truncation error should enhance the weight of the viscous terms." (Shyy [48]).

For a better understanding of how well a convection-diffusion equation has been modeled numerically, it is essential to know the magnitude of the artificial dissipation relative to the physical one. We believe therefore that for the numerical simulation of the Navier-Stokes equations *only* centered discretization schemes should be used, on a non-staggered grid, since *only* these schemes can be completely free of any artificial dissipation effect. Artificial dissipation may be added locally in an explicit way, and if this is done in the form of second derivative terms, it can directly be checked afterward how well the viscous phenomena were modeled by the numerics, by comparing the artificial and physical diffusion terms. In addition, only central discretization techniques lead, for a given computational molecule, to the most accurate schemes (compact schemes).

Artificially added terms have also the advantage that they give us more freedom in designing the optimally dissipative scheme. For example, although other quantities are numerically discontinuous over a shock or a boundary layer, the mass flow is not, so it does not seem to be appropriate to add artificial dissipation terms to a mass conservation equation, also because this is highly unphysical. Concerning the other flow equations, we think that a good way of implementing artificial dissipation may be to artificially increase the physical stresses in the direction of a numerical discontinuity, in order to obtain the necessary smoothing. In this way, the artificially added dissipation can be compared directly with the physical one. This topic will be discussed further during the presentation (see also [49]).

Only conservative schemes will be considered, that discretize the set of equations written in conservative form. This restriction is not only to ensure a correct shock-capturing, but also to reduce the

computational effort. The set of equations will be written in integral form rather than in differential form, and flux terms rather than derivatives will be discretized. Depending on the number of space dimensions, the computational domain will be divided into a finite number of intervals, surfaces, or volumes, that are adjacent to each other, but may be overlapping. This has the advantage that every evaluation of a numerical approximation of a flux term can be used twice, because it will be both ingoing flux and outgoing flux for two adjacent subdomains. We will refer to the small subdomains, that normally are of the size of a grid cell, as *finite volumes*, independently of the number of space dimensions.

We will see in the next chapter that the optimal discretization scheme we will finally arrive at, will not be strictly conservative. It seems that this is the price to pay if we try to combine accuracy with a low-order discretization technique. However, this does not necessarily have to pose any problem, since a non-conservative scheme is not necessarily incapable of capturing shocks correctly. See [45] for a discussion about this matter.

We will consider the linear differential equation  $d\Phi/dx = f$ , but written in the integral form:

$$\int_V d\Phi = \int_V f dx , \quad (2.1)$$

which equation should hold for any finite volume (that is here an interval)  $V$  belonging to the domain of definition of  $\Phi$  and  $f$ .

This is a so-called weak formulation of the differential problem [51], since it accepts also discontinuous functions  $\Phi$  as solution, in contrast with the differential form of the equation. This is of importance for the Euler equations, but not for the compressible Navier-Stokes equations, as its solution will be sufficiently differentiable anyway [46, 47]. In that case there is no difference between a weak solution and a genuine solution.

Discretized over the finite volumes  $[x_{j-1}, x_{j+1}]$ , equation (2.1) yields:

$$\Phi_{j+1}^h - \Phi_{j-1}^h = \int_{x_{j-1}}^{x_{j+1}} f dx . \quad (2.2)$$

Clearly, the discretization is exact if the integral of  $f$  will be evaluated exactly, and any order of accuracy can be obtained with this

scheme, dependent only on the order of accuracy with which the solution  $\Phi^h$  is prolonged in between the exact nodal point values  $\Phi_j^h$ . However, for numerical purposes, also the right-hand side of (2.2) should be discretized, and the simplest way to do this is by approximating  $f$  over the whole interval  $[x_{j-1}, x_{j+1}]$  by its value at the point  $x_j$ . This yields the discretization:

$$\Phi_{j+1}^h - \Phi_{j-1}^h = (h_j + h_{j+1}) f_j , \quad (2.3)$$

which would also have been obtained by transforming the physical coordinate  $x$  to a computational coordinate  $\xi$ , such that  $x_j = x(\xi=j)$ , and discretizing the differential equation  $d\Phi/dx = f$ , transformed to the computational  $\xi$ -domain, by means of finite differences, whereby the transformation coefficient  $dx/d\xi$  is approximated by the centered difference  $(x_{j+1} - x_{j-1})/2$ . The accuracy of discretization (2.3) for the transformed equation is  $O(\Delta^2 \xi)$ , since centered discretizations have been used on a grid that is uniform in  $\xi$ , where  $\Delta \xi$  denotes the constant length of each grid cell.

The order of accuracy of (2.3) on the original domain with variable grid spacing in  $x$  is not so obvious. The technique of expanding  $\Phi(x)$ ,  $\Phi^h(x)$  and  $f(x)$  in Fourier series cannot be used anymore on a non-uniform grid. However, there is still the possibility of determining the order of magnitude of the truncation error of the discretization, which corresponds to the order of accuracy of the discretization scheme itself.

For this purpose Taylor series expansions could be used, and in order to find out around which point(s), it suffices to realize that the integral in (2.2) can be approximated equally well by means of a Gauss' integration formula. For example, if a piecewise constant or a piecewise linear approximation of the integrand is used in order to arrive at a discretized form of (2.2) that will be at most third-order accurate, the integral may as well be approximated by means of a one-point quadrature rule, that is also third-order accurate. This yields:

$$\int_{x_{j-1}}^{x_{j+1}} f dx = 2\bar{h}f(\bar{x}) + O(\bar{h}^3) , \quad (2.4)$$

and we conclude that the finite volume discretization (2.3) is equivalent to a finite difference discretization times  $2\bar{h}$  of  $d\Phi/dx = f$  at the point  $x = \bar{x} = (x_{j+1} + x_{j-1})/2$ , that is of the same order of accuracy.

We see directly around which point the algebraic equation (2.3) should be expanded in a Taylor series, in order to determine up to  $O(\bar{h}^2)$  the equivalent differential equation that is satisfied by the numerical approximation  $\tilde{\Phi}^h$ . A Taylor series expansion around the point  $x = \bar{x}$  shows that (2.3) approximates the left-hand side of (2.1) up to at least  $O(\bar{h}^3)$ , while the approximation of the right-hand side yields an error term of  $O(\bar{h}h_j - \bar{h}h_{j+1})$ . Upon replacing the first derivative of  $\tilde{f}$  by the second derivative of  $\tilde{\Phi}^h$ , using the derivative of the equation itself, the equivalent equation of (2.3) is obtained:

$$\frac{d\tilde{\Phi}^h}{dx} = \tilde{f} + \frac{h_j - h_{j+1}}{2} \frac{d^2\tilde{\Phi}^h}{dx^2} + O(\bar{h}^2). \quad (2.5)$$

It follows that the truncation error of (2.3) is  $O(\Delta h, h^2)$ , where  $\Delta h$  denotes the difference in grid spacing over one computational molecule.

At first sight, the conclusion would be that discretization (2.3) is first-order accurate on a non-uniform grid, and second-order accurate on a uniform grid. However, the former conclusion is only valid if we suppose that  $\Delta h$  is proportional to the grid size  $h$ , for any value of  $h$ .

This may be true for some strange construction of the grid, but in practical applications the grid will never be like that, and we may even state that in practice any grid refinement will cause at the same time a grid uniformization. So it will be more realistic to say that  $\Delta h$  will be at least proportional to  $h^2$ , and that (2.3) will be a second-order accurate discretization of (2.1).

This corresponds also to the order of accuracy of (2.3) for the transformed problem on the  $\xi$ -domain. It can easily be verified that if the uniform grid in  $\xi$  is refined, the non-uniform grid in the  $x$ -domain is indeed uniformized!

Apparently, the order of accuracy of a numerical scheme is not well defined on a non-uniform grid (cf. [36]). We will therefore refer to the order of accuracy of a discretization scheme as to its asymptotic error behavior for a sufficiently smooth solution on a *uniform* grid, whose grid size  $h$  tends to zero. Since in practice any numerical grid will tend to a uniform grid when  $h \rightarrow 0$ , the addition of this condition seems to be appropriate.

Although scheme (2.3) will in practice almost always behave as a second-order accurate scheme, the fact remains that its truncation error is only first order in  $\Delta h$ . This implies that, because of dimensional reasons, the coefficient of the second-derivative error term in this truncation error will only be zero when discretization (2.3) is used on a uniform grid, but that it may be very large on a non-uniform grid, indicating the presence of *numerical diffusion*. Although this may have not too drastic consequences in case the equation  $d\Phi/dx = f$  is to be solved numerically, it may deteriorate completely the accuracy of the numerical approximation if a discretization of a second derivative is added to the scheme in order to model numerically some convection-diffusion phenomenon.

As mentioned in the introduction of this chapter, it may be necessary to add artificial dissipation terms (e.g. second-derivative error terms) to the numerical scheme for reasons of accuracy. However, it has been argued also that these terms should be kept as small as possible, if we want to study numerically equations containing second derivatives. Otherwise, the artificially added second derivative dissipation may be of unacceptably high influence on the final numerical result.

For the same reason, any source that contributes to this error in an uncontrollable way should be reduced to a level at which its influence will be negligible. Numerical diffusion is such a source, and should therefore be eliminated as much as possible, especially for Navier-Stokes calculations. Numerical diffusion may cause the over- and underestimation of the physical diffusion, which will make the numerical simulation very inaccurate. It is a quantity that in general is very difficult to control, because it depends on the structure of the grid and may change sign. It would therefore be best to use discretization schemes and grids that are introducing as less numerical diffusion as possible.

For scheme (2.3) this would imply that the use of an (almost) uniform grid is imperative. But this has the consequence that we lose all the freedom in grid design, which is definitely not desirable.

Alternatively, we may also try to develop a discretization scheme for (2.2), that does not introduce that much numerical diffusion, when applied on a non-uniform grid. This can be achieved by using a more accurate approximation than a piecewise constant interpolation for the integrand  $f$ , for example by using a piecewise linear interpolation, or even a quadratic interpolation. The resulting discretization scheme is:

$$\Phi_{j+1}^h - \Phi_{j-1}^h = h_j \frac{f_{j-1} + f_j}{2} + h_{j+1} \frac{f_j + f_{j+1}}{2}, \quad (2.6)$$

in case of a piecewise linear approximation of  $f$ , and:

$$\begin{aligned} \Phi_{j+1}^h - \Phi_{j-1}^h &= \\ &= (h_j + h_{j+1}) \left\{ \frac{2h_j - h_{j+1}}{h_j} \frac{f_{j-1}}{6} + \frac{(h_j + h_{j+1})^2}{h_j h_{j+1}} \frac{f_j}{6} + \frac{2h_{j+1} - h_j}{h_{j+1}} \frac{f_{j+1}}{6} \right\}, \end{aligned} \quad (2.7)$$

if a quadratic approximation is used. On a uniform grid, the discretizations (2.6) and (2.7) correspond respectively to a Box-scheme-type discretization, and to a discretization obtained by applying the piecewise linear Galerkin finite element method to the problem.

Since discretization (2.6) has been obtained using second-order accurate, linear interpolations, its truncation error can still be found by means of a Taylor series expansion around the single quadrature point  $x = \bar{x}$ . This shows that the scheme is  $O(h^2)$ , independently of the grid structure, and does not contain any numerical diffusion.

Another way of obtaining a second-order accurate conservative discretization scheme that is free of numerical diffusion is by evaluating the integrand of the right-hand side of (2.2) at the quadrature point  $\bar{x}$ . This yields, using a quadratic interpolation of  $f$  in order to determine its value at  $x = \bar{x}$ :

$$\begin{aligned} \Phi_{j+1}^h - \Phi_{j-1}^h &= \\ &= (h_j + h_{j+1}) \left\{ \frac{h_j - h_{j+1}}{h_j} \frac{f_{j-1}}{4} + \frac{(h_j + h_{j+1})^2}{h_j h_{j+1}} \frac{f_j}{4} + \frac{h_{j+1} - h_j}{h_{j+1}} \frac{f_{j+1}}{4} \right\}. \end{aligned} \quad (2.8)$$

Like (2.3), scheme (2.8) corresponds to the classical finite difference discretization of the problem, but only when the grid is uniform. It differs from the former in that it does not introduce any numerical diffusion.

Scheme (2.8) is very interesting from a computational point of view, since it establishes a kind of *optimal mass lumping*. It maximizes the ratio between the weight of the value of  $f$  taken at the "central" grid point  $x_j$ , and the weights at the two other points, while still being free of numerical diffusion.

With regard to scheme (2.7), it may be assumed that this discretization of (2.2) will be the most accurate one possible on a three-point computational molecule. Since it is based on a third-order accurate interpolation of  $f$ , and fourth-order accurate on a uniform grid, we expect its truncation error to be  $O(h^2\Delta h, h^4)$ .

Because of the high-order interpolation of  $f$ , a higher-order quadrature rule is to be used in order to find the relation between this global finite volume discretization and pointwise finite difference discretizations. The two-point quadrature rule, applied to (2.2), yields:

$$\int_{x_{j-1}}^{x_{j+1}} f \, dx = \bar{h}f(\bar{x}-\bar{h}/\sqrt{3}) + \bar{h}f(\bar{x}+\bar{h}/\sqrt{3}) + O(\bar{h}^5). \quad (2.9)$$

This shows that it should in principle be possible to split (2.7) in two finite difference discretizations at the two quadrature points  $x = \bar{x} \pm \bar{h}/\sqrt{3}$ , such that a Taylor series expansion of each of those will yield the expected truncation error. This appeared indeed to be possible, which proves that in the truncation error of (2.7) neither second-derivative error terms, nor third-derivative error terms appear, while the fourth-derivative error term vanishes on a uniform grid.

A rather remarkable discretization scheme is obtained by adding one third of scheme (2.3) to two thirds of scheme (2.6). It can easily be verified that this scheme is fourth-order accurate on a uniform grid, but nevertheless *not* free of numerical diffusion when used on a non-uniform grid, because of scheme (2.3).

The discretization we talk about is a *finite element* scheme, that is obtained by means of a Galerkin finite element discretization of  $d\phi/dx = f$ , using piecewise linear basis functions. This proves that, although the finite element technique yields very accurate discretization schemes on *uniform* grids, the technique may be completely unsuited for use with non-uniform grids.

The same conclusion applies also to the finite difference method, albeit to a lesser extend. For this reason such techniques should preferably be used on smoothly varying grids [36, 38-40, 52]. However, in practice they will still behave as a second-order accurate scheme (or even better) on any grid, because a grid refinement means in general automatically a grid uniformization [53-55].

## II.1 CONCLUDING REMARKS

We believe that it is mainly to avoid the introduction of an excessive amount of (negative!) numerical diffusion that one always tries to keep computational grids as smooth as possible. Because of the dimension of the coefficient of the second-derivative truncation error term (proportional to the grid spacing), all second- or higher-order accurate discretization schemes of  $d\Phi/dx = f$  must be (almost) free of numerical diffusion on an (almost) uniform grid, and this conclusion can easily be extended to the numerical simulation of the Euler and Navier-Stokes equations.

However, although it is rather easy to design a discretization scheme for these equations that is free of numerical diffusion on a uniform grid, the results presented here show that it is not so evident to obtain this for a non-uniform grid. Especially the classical discretization techniques do not seem to be able to generate generally applicable discretization schemes of sufficient accuracy.

We think that this is essentially due to the assumptions that are *a priori* made by such methods. For example, the finite difference technique uses centered differences for a transformed set of differential equations on a uniform computational grid, ignoring the effect that the transformation may have on the accuracy (*not* the order of accuracy) of the numerically obtained solution. The drawback of the finite element method is that in general everywhere on the grid the same type of basis functions and weight functions are used, independent of the shape of the computational molecule. Although it is legitimate to do so on a uniform grid, it seems questionable whether the same procedure should be followed when the molecules are not identical anymore, as is the case on a non-uniform grid, but also at boundaries. The fact that the Galerkin finite element scheme was found to be fourth-order accurate on a uniform grid, while introducing spurious second derivative terms in the numerical solution procedure whenever a non-uniform grid is used, supports this idea quite well.

For these reasons, we have followed an alternative approach for the development of the optimal low-order conservative scheme for the Navier-Stokes equations, by trying to develop a discretization scheme without fixing any *a priori* assumption, with the exception perhaps of the assumption that the scheme should be as free of numerical diffusion as

possible. The technique, as well as the resulting discretization scheme, will be presented in the next chapter.

It should be mentioned that the analytical results of this chapter have all been verified by means of a series of numerical tests, discretizing and solving approximately the linear equation (2.1) on a non-uniform grid with a  $\Delta h$  of order  $h$ . It was found that the theory predicted the observed error behavior very well, *including* the errors due to the discretization of the numerical boundary condition. The results of these tests will be presented during the lectures.

## CHAPTER III

### **THE OPTIMAL LOW-ORDER CONSERVATIVE DISCRETIZATION SCHEME FOR THE STEADY-STATE COMPRESSIBLE NAVIER-STOKES EQUATIONS**

In this chapter discretization methods in space for the steady-state compressible Navier-Stokes equations in two dimensions will be considered. The system of equations we will be dealing with may be written in the general conservative form:

$$\frac{\partial E}{\partial x} + \frac{\partial G}{\partial y} = \frac{\partial}{\partial x} \left[ V_1 \frac{\partial V_2}{\partial x} + V_3 \frac{\partial V_4}{\partial y} \right] + \frac{\partial}{\partial y} \left[ W_1 \frac{\partial W_2}{\partial x} + W_3 \frac{\partial W_4}{\partial y} \right] + S . \quad (3.1)$$

Here, the vectors  $E$ ,  $G$ ,  $V_i$ ,  $W_i$  and  $S$ , and the matrices  $V_i$  and  $W_i$  are in general functions of the vector of unknowns  $\underline{y}$  and the space and time variables  $x$ ,  $y$  and  $t$ . The precise form of all these vectors and matrices is irrelevant for the development of a discretization method, and is also rather difficult to give, as (3.1) not only comprises the continuity, momentum, and energy equation, but may also contain differential equations concerning the turbulence model, chemical reactions, two-phase flow, etcetera.

The integration of (3.1) over some finite volume  $V$  belonging to its domain of definition yields:

$$\oint_E dy - \oint_G dx = \oint \left[ V_1 \frac{\partial V_2}{\partial x} + V_3 \frac{\partial V_4}{\partial y} \right] dy - \oint \left[ W_1 \frac{\partial W_2}{\partial x} + W_3 \frac{\partial W_4}{\partial y} \right] dx + \iint_S S dxdy , \quad (3.2)$$

where the Gauss' theorem has been applied in order to reduce the order of the integrands, and where  $S$  is the contour of  $V$ .

The purpose of this chapter is to put equation (3.2) in a discretized form in such a way that an accurate numerical approximation of the solution of (3.1) can be obtained by solving the resulting system of algebraic equations, provided that this is possible. This is a rather difficult thing to ask, since the accuracy of a discretization is not only determined by the discretization method that has been employed, but depends also on the grid.

We have mentioned briefly in the previous chapter that the accuracy of a numerical approximation is limited if the grid does not yield enough resolution. We have also seen that the accuracy of a numerical scheme may depend strongly on the smoothness of a grid. Both problems should be solved, but here we will only treat the latter one, and try to develop a discretization scheme that is sufficiently accurate for Navier-Stokes calculations, independently of the grid structure.

This means in the first place that the amount of spurious diffusion introduced by the numerical scheme should be sufficiently small, especially in those regions of the flow where the viscous effects are quite strong. This is a rather severe demand if viscous flows at high Reynolds number are to be simulated numerically. On the other hand we would also like to use a simple low-order scheme, which may inevitably lead to the introduction of some numerical diffusion. The question is: does an acceptable compromise exist?

In general, a low-order discretization method may only be expected to be free of any truncation error for functions that are linear in the coordinates  $x$  and  $y$ , which indicates that the truncation error will normally contain error terms of second and higher derivatives [46, 47, 56]. So for a low-order discretization of (3.2), the equivalent equation is expected to be of the form:

$$\begin{aligned}
 \frac{\partial \tilde{F}^h}{\partial x} + \frac{\partial \tilde{G}^h}{\partial y} &= \frac{\partial}{\partial x} \left\{ (1+\tilde{a}_{01}) \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} + (1+\tilde{a}_{02}) \tilde{V}_3^h \frac{\partial \tilde{V}_4^h}{\partial y} \right\} + \\
 &+ \frac{\partial}{\partial y} \left\{ (1+\tilde{a}_{03}) \tilde{W}_1^h \frac{\partial \tilde{W}_2^h}{\partial x} + (1+\tilde{a}_{04}) \tilde{W}_3^h \frac{\partial \tilde{W}_4^h}{\partial y} \right\} + \\
 &+ \tilde{a}_{11} \frac{\partial^2 \tilde{F}^h}{\partial x^2} + \tilde{a}_{12} \frac{\partial^2 \tilde{F}^h}{\partial x \partial y} + \tilde{a}_{13} \frac{\partial^2 \tilde{F}^h}{\partial y^2} + \\
 &+ \tilde{a}_{14} \frac{\partial^2 \tilde{G}^h}{\partial x^2} + \tilde{a}_{15} \frac{\partial^2 \tilde{G}^h}{\partial x \partial y} + \tilde{a}_{16} \frac{\partial^2 \tilde{G}^h}{\partial y^2} + \tilde{S}^h + \text{HOT} .
 \end{aligned} \tag{3.3}$$

The numerical approximations of the functions in (3.1) are denoted by a tilde, and are such that their values at the grid points (denoted by  $\tilde{E}_j^h$  or  $E_j^h$ , etc.) satisfy the equations (3.1) and (3.2) in their discretized form. The second derivatives of  $\tilde{s}^h$  have not been included in (3.3), since higher derivatives of the numerical approximation of the source term can be replaced by higher derivatives of the other unknowns.

The  $\tilde{a}_{\ell m}$ -coefficients are only functions of the coordinates  $x$  and  $y$ , because the difference operators should be, like the differential operators, linear in the unknown functions. The order of magnitude of the  $\tilde{a}_{\ell m}$  in terms of the grid size can therefore be determined easily.

For the first derivative  $\partial F / \partial x$ , for example, any difference approximation must be of dimension  $\Delta F / \Delta x$ , and as a consequence  $\tilde{a}_{11}$ ,  $\tilde{a}_{12}$  and  $\tilde{a}_{13}$  are proportional to  $\Delta x$ ,  $\Delta y$  and  $\Delta^2 y / \Delta x$  respectively. Here,  $\Delta x$  and  $\Delta y$  indicate the size in  $x$ - and  $y$ -direction of the computational molecule used for the discretization of  $\partial F / \partial x$ . If we denote by  $h_j$  the diameter of the molecule, then each of the  $\tilde{a}_{1m}$  is  $O(h_j)$ , where the actual value depends on the structure of the molecule. Likewise we find that the  $\tilde{a}_{0m}$  are  $O(1)$ .

Notice that some of the  $\tilde{a}_{\ell m}$  may very well be equal to zero. It is well known that on a uniform grid any low-order discretization method for (3.1) of practical interest is at least second-order accurate. Such a method must therefore be free of any numerical diffusion, since the  $\tilde{a}_{\ell m}$  ( $\ell \leq 1$ ) are at most  $O(h)$ . See also the concluding remarks of the previous chapter.

In consequence, the  $\tilde{a}_{1m}$  are  $O(\Delta_{1m} h_j)$  rather than  $O(h_j)$  when a discretization method of second-order accuracy is employed, or else these coefficients would not vanish on a uniform grid. Here, the  $\Delta_{1m} h_j$  indicate in some sense the change in grid size per computational molecule. Their values, that depend on the type of scheme used, vanish on a sufficiently regular molecule (cf. CHAPTER II).

Similarly, we have that the  $\tilde{a}_{0m}$  will be  $O(\Delta_{0m} h_j / h_j)$  in that case. Remark that the coefficients  $\tilde{a}_{0m}$  and  $\tilde{a}_{1m}$  depend in general in different ways on the non-uniformity of the molecule, which has been denoted by the subscripts of the molecule-dependent variables  $\Delta_{\ell m} h_j$ .

### III.1 AN EXAMPLE OF WHAT CAN BE OBTAINED

As an example of what is possible, we will consider the low-order conservative discretization  $v_{\alpha\beta\gamma\delta}$  of the convective part of (3.1). The evaluation of the flux integrals at the left-hand side of (3.2) on the quadrilateral of FIGURE 2, using an isoparametric bilinear interpolation for  $E^h$  and  $\xi^h$ , yields:

$$\begin{aligned} \oint_{\alpha\beta\gamma\delta} E dy - \oint_{\alpha\beta\gamma\delta} \xi dx \approx v_{\alpha\beta\gamma\delta} &= \frac{E_\alpha^h - E_\gamma^h}{2}(y_\beta - y_\delta) + \frac{E_\beta^h - E_\delta^h}{2}(y_\gamma - y_\alpha) + \\ &+ \frac{\xi_\alpha^h - \xi_\gamma^h}{2}(x_\delta - x_\beta) + \frac{\xi_\beta^h - \xi_\delta^h}{2}(x_\alpha - x_\gamma). \end{aligned} \quad (3.4)$$

Expression (3.4) approximates the integrals up to  $O(VOL \cdot h^2)$ , where VOL denotes the surface (the finite volume) of the quadrilateral of FIGURE 2. An approximation of the same order of accuracy can be obtained by means of a one-point Gauss' integration rule, where the integrands are evaluated at the quadrature point  $((x_\alpha + x_\beta + x_\gamma + x_\delta)/4, (y_\alpha + y_\beta + y_\gamma + y_\delta)/4)$ . Similarly to what has been observed in the previous chapter, we find here that the finite volume discretization  $v_{\alpha\beta\gamma\delta}$  is equivalent up to  $O(VOL \cdot h^2)$  to a finite difference discretization times VOL at the quadrature point, and that its truncation error can be determined by a Taylor series expansion around that point. As expected, it was found to be  $O(\Delta_{1m} h, h^2)$  in general, and free of any numerical diffusion if and only if:

$$x_\alpha + x_\gamma = x_\beta + x_\delta \quad \text{and} \quad y_\alpha + y_\gamma = y_\beta + y_\delta, \quad (3.5)$$

i.e. when the quadrilateral is a parallelogram.

To verify this experimentally, a rectangular domain was divided into  $n$  times  $n$  equal subrectangles ( $n$  varying from 1 to 100), and in each subrectangle the grid module of FIGURE 3 was placed. Only on such a repetitive grid it is possible to have a  $\Delta h$  of order  $h$ , even when  $h \rightarrow 0$ .

For consider for example a one-dimensional grid as in FIGURE 1, with grid spacings defined as  $h_j = A^{j-1}h$ ,  $j = 1, \dots, n$ , where  $A$  is some constant smaller than 1. On such a grid the difference  $\Delta h$  between the length of each two consecutive grid cells is  $O(h)$ . But the total length of the domain that is covered by the grid equals  $h(1-A^n)/(1-A)$  and is thus proportional to  $h$  when the number of grid points tends to infinity. It is therefore impossible to let the grid size  $h$  tend to zero. In

consequence some regularity must be introduced if we would like to study the asymptotic error behavior of a discretization scheme on a non-uniform grid. However, this regularity should *not* be such that  $\Delta h$  becomes  $O(h^2)$ ! Otherwise the numerical test will indicate a second- or higher-order accuracy [55, 57], which does *not* imply that the scheme is free of numerical diffusion! The test that is proposed here satisfies this criterion.

The equation that we have solved on the repetitive grid is  $\partial\Phi/\partial x + \partial\Phi/\partial y = 0$ . Two versions of  $V_{\alpha\beta\gamma\delta}$  were tested, namely  $V_{1208} = 0$  (see FIGURE 3), calculating the value of the unknown function  $\Phi^h$  at point 0 from its values at the points 1, 2 and 8, and  $V_{2468} = 0$ , where the value of  $\Phi^h$  at point 6 was determined in function of its values at the points 2, 4 and 8.

Both tests started with the exact solution  $\Phi = ((y-x+2)/3)^3$  at lower and left boundary, to find  $\Phi - \Phi^h$  at the upper-right corner. The results of FIGURE 4 show the asymptotic first-order behavior of  $V_{2468}$ , and a clear  $O(h^2)$  for  $V_{1208}$ , which is in perfect agreement with the theory.

In many computational fluid dynamics applications, grid cells form in general approximately parallelograms, although the grid as a whole can be highly stretched and bent [7-34, 37]. This makes that the conservative discretization of (3.4) is much more useful in its  $V_{1208}$ -configuration.

However, the low-order Galerkin finite element technique (GFEM) for the convective part of (3.1), using piecewise bilinear shape and weight functions, would yield the discretization  $(V_{1208} + V_{2340} + V_{0456} + V_{8067} + V_{2468})/6$ , as can easily be verified, while the classical finite difference scheme for that part is equal to  $V_{2468}/VOL$ . This makes that for both methods the grid design is very critical for flow calculations at high Reynolds number, if we want to keep the numerical discretization of (3.1) reasonably free of spurious diffusion. Yet these discretizations generate only fourth-order and second-order error terms respectively when applied on a uniform grid, which coincides with what we found already in the previous chapter. The test proves that in general both methods yield only first-order accurate discretizations of first derivatives, which is in perfect agreement with the theory [36, 58p255].

### III.2 THE ACCURACY OF A LOW-ORDER CONSERVATIVE SCHEME FOR CONVECTIVE TERMS

In the foregoing we have shown analytically, and verified experimentally, that the concept of Taylor series expansions around quadrature points is indeed the correct way to determine the truncation error of a conservative discretization scheme, although other investigators seem to have a different opinion about that [40, 53]. We would therefore like to use this concept to determine in an analytical way a low-order conservative discretization scheme, that introduces no or only a minimal amount of numerical diffusion in the equivalent equation (3.3). Such an analysis is in fact only practical for low-order methods, where it suffices to expand the whole discretization formulae around one single quadrature point.

We will only consider unknowns (and functions of unknowns) defined at grid points. Low-order discretization schemes based on global values, such as the average of the unknowns and their variation per volume, are intrinsically less accurate, due to the fact that linear interpolations for the flux terms are much more difficult to realize [59].

Starting with the convective terms of (3.1), we will first write the discretization of the corresponding flux integrals of (3.2) in the general form:

$$\oint_S \underline{F} dy \approx \sum_i \alpha_i \tilde{\underline{F}}_i^h, \quad \oint_S \underline{G} dx \approx \sum_i \beta_i \tilde{\underline{G}}_i^h, \quad (3.6)$$

where the discretizations are based on the values  $\tilde{\underline{F}}_i^h$  ( $= \underline{F}_i^h$ ) and  $\tilde{\underline{G}}_i^h$  ( $= \underline{G}_i^h$ ) of the approximation functions  $\tilde{\underline{F}}^h$  and  $\tilde{\underline{G}}^h$  at a certain number of grid points  $(x_i, y_i)$ .

Since the differential operators  $\partial/\partial x$  and  $\partial/\partial y$  are linear, the  $\alpha_i$  and the  $\beta_i$  should all be functions of the coordinates  $x_i$  and  $y_i$  only. Because of the similarity between the two expressions in (3.6) we should have:

$$\alpha_i(x_j, y_k) = \beta_i(y_j, x_k), \quad \text{for all } i. \quad (3.7)$$

We will not fix any finite volume shape, and will define the point  $(x^*, y^*)$  to be equal to the quadrature point. A Taylor series expansion of (3.6) around  $(x^*, y^*)$  yields:

$$\begin{aligned}
& \oint_S \underline{F} dy - \oint_S \underline{G} dx \approx \sum_i \alpha_i \tilde{F}_i^h - \sum_i \beta_i \tilde{G}_i^h = \\
& = E^* \sum_i \alpha_i + F_x^* \sum_i (x_i - x^*) \alpha_i + F_y^* \sum_i (y_i - y^*) \alpha_i + \\
& - G^* \sum_i \beta_i - G_x^* \sum_i (x_i - x^*) \beta_i - G_y^* \sum_i (y_i - y^*) \beta_i + \\
& + E_{xx}^* \sum_i \frac{(x_i - x^*)^2}{2} \alpha_i + E_{xy}^* \sum_i (x_i - x^*)(y_i - y^*) \alpha_i + E_{yy}^* \sum_i \frac{(y_i - y^*)^2}{2} \alpha_i + \\
& - G_{xx}^* \sum_i \frac{(x_i - x^*)^2}{2} \beta_i - G_{xy}^* \sum_i (x_i - x^*)(y_i - y^*) \beta_i - G_{yy}^* \sum_i \frac{(y_i - y^*)^2}{2} \beta_i + \text{HOT}
\end{aligned} \tag{3.8}$$

Here, the superscript \* denotes values of  $\tilde{F}^h$ ,  $\tilde{G}^h$  and their derivatives taken at the point  $(x^*, y^*)$ , where the derivatives with respect to  $x$  and  $y$  are indicated by the subscripts.

For the discretization (3.6) to be a numerical-diffusion-free approximation of (3.1), it is necessary that the coefficients in (3.8) satisfy:

$$\sum_i \alpha_i = \sum_i \beta_i = 0 \quad ; \tag{3.9}$$

$$\sum_i (y_i - y^*) \alpha_i = \sum_i (x_i - x^*) \beta_i = 0 \quad , \tag{3.10}$$

$$\iint_V dx dy = \sum_i (x_i - x^*) \alpha_i = - \sum_i (y_i - y^*) \beta_i = \text{VOL} \quad , \tag{3.11}$$

$$\sum_i (y_i - y^*)^2 \alpha_i = \sum_i (x_i - x^*)^2 \beta_i = 0 \quad , \tag{3.12}$$

$$\sum_i (x_i - x^*)(y_i - y^*) \alpha_i = \sum_i (x_i - x^*)(y_i - y^*) \beta_i = 0 \quad , \text{ and} \tag{3.13}$$

$$\sum_i (x_i - x^*)^2 \alpha_i = \sum_i (y_i - y^*)^2 \beta_i = 0 \quad , \tag{3.14}$$

with VOL the exact value of the surface of the finite volume.

Some interesting conclusions may be drawn from these conditions. First of all, combining (3.9) and (3.10) yields:

$$\sum_i y_i \alpha_i = \sum_i x_i \beta_i = 0 \quad . \tag{3.15}$$

Hence the  $\alpha_i$  and the  $\beta_i$  can only be functions of the  $y_i$  and the  $x_i$  respectively, as this condition is supposed to hold for any set of points  $(x_i, y_i)$ . Notice, however, that these functions may very well depend on the geometry of the computational molecule, and therefore be functions of the coordinates  $x_i$  and  $y_i$  themselves.

Next, upon combining (3.9) and (3.11), we obtain:

$$\sum x_i \alpha_i = -\sum y_i \beta_i , \quad (3.16)$$

and it can easily be shown that, in view of the previous conclusion, this condition can only be satisfied in a general way if the  $\alpha_i$  and the  $\beta_i$  are linear functions of the  $y_i$  and the  $x_i$  respectively. Again, the functions themselves may vary with the geometry of the computational molecules. This result is rather obvious, since for the discretizations at the right-hand sides of (3.6) to be consistent with the integrals at the left-hand sides, the  $\alpha_i$  and the  $\beta_i$  should represent length scales in  $y$ - and in  $x$ -direction. As these length scales vanish for identical points  $(x_i, y_i)$ , in which case VOL becomes equal to zero, we have in addition that the  $\alpha_i$  and  $\beta_i$  should vanish for equal  $y_i$  and  $x_i$ . It follows that, also because of (3.7), the  $\alpha_i$  and  $\beta_i$  can be written as:

$$\alpha_i = \sum_j \theta_{ij} y_j , \quad \beta_i = \sum_j \theta_{ij} x_j , \quad \sum_j \theta_{ij} = 0 , \quad (3.17)$$

where the  $\theta_{ij}$  are constants to be determined.

Finally, upon combining (3.9), (3.10) and (3.12) we obtain:

$$\sum y_i^2 \alpha_i = \sum x_i^2 \beta_i = 0 . \quad (3.18)$$

From this it follows immediately that if we approximate the flux terms of every finite volume of the computational domain by the same discretized form (3.6) by keeping the  $\theta_{ij}$  in (3.17) constant over the whole domain (i.e. by using a low-order discretization technique based on piecewise linear interpolations), it will in general *not* be possible to have a diffusion-free approximation of  $E_x^*$  and  $G_y^*$  on every finite volume. The summation of products of both a quadratic and a linear function in the  $y_i$  and the  $x_i$  should vanish, which cannot be the case for arbitrary  $x_i$  and  $y_i$ .

This actually proves that on an arbitrary grid any low-order conservative discretization of first derivatives based on grid point values causes the presence of spurious diffusion terms in the equivalent differential equation. For such a discretization scheme, the coefficients  $\tilde{a}_{1m}$  in (3.3) are indeed in general not equal to zero, and may therefore pose a problem in any numerical solution method for the Navier-Stokes equations at high Reynolds number.

If we would put certain restrictions to the choice of the points  $(x_i, y_i)$ , both condition (3.15) and (3.18) may be fulfilled, and the discretization may then be completely free of any numerical diffusion. Such restrictions may however not be possible in practice, since the distribution of points in a computational grid also has to suit the physical problem to be modeled.

Alternatively, it would also be possible to make the  $\theta_{ij}$ -values variable, and to use a finite volume discretization technique where these coefficients are functions of the grid points  $(x_i, y_i)$ . For every computational molecule the  $\theta_{ij}$  should then be such, that the conditions (3.9) to (3.14) would all be satisfied. However, great care should be taken in order not to destroy the conservation property of the finite volume approach. The only way to realize this would be by using a higher-order conservative discretization scheme, but this would make a computer code considerably more complex, and would demand storage space and a calculation procedure for all the  $\theta_{ij}$ .

For these reasons the possibility of molecule-dependent  $\theta_{ij}$  to obtain diffusion-free conservative discretizations for convective terms on any two-dimensional grid has not been considered any further. Instead, we have limited our investigations to the development of an optimal low-order conservative scheme on a structured grid, and looked for the most accurate  $E_x$ - and  $E_y$ -discretization possible with fixed  $\theta_{ij}$ , on a computational molecule of three times three points.

The latter restriction is in order to keep not only the numerical scheme, but also its implementation (relatively) simple, which is especially of importance if in the future the extension to three-dimensional problems will be made. Larger molecules are possible of course, but then one may as well design a higher-order discretization procedure, in order to obtain a scheme that is completely free of numerical diffusion, independently of the grid structure.

### III.3 DERIVATION OF OLOCS - THE OPTIMAL LOW-ORDER CONSERVATIVE DISCRETIZATION SCHEME

For the derivation of the Optimal Low-Order Conservative Scheme, consider a three-times-three-point computational molecule on a structured grid, with the central grid point numbered 0, and the outer points numbered 1 to 8, starting the numbering at one of the four corner points and from there on moving in counterclockwise direction (FIGURE 5). Many different shapes of finite volumes are possible on such a configuration, as is shown in the figure, but they all have in common that for the approximation of the flux through one part of the finite volume boundary, unknowns and coordinates from at most six grid points may be used.

For a conservative discretization, any flux approximation through a finite volume boundary in the domain should be used twice, namely once in each of the two finite volumes that have this boundary in common. So for the approximation only those points should be used that are shared by the two computational molecules to which the finite volumes belong. Hence a maximum of six points per flux approximation, depending on the overlapping of the molecules.

On a three-times-three-point molecule, four of such six-point groups may be discerned: the group of points 1, 2, 0, 6, 7 and 8, the points 3, 4, 0, 8, 1 and 2, the points 5, 6, 0, 2, 3 and 4, and the group formed by the points 7, 8, 0, 4, 5 and 6. The total flux traversing the boundaries of any finite volume defined on the molecule can be split into four parts, where every part is discretized using one of those six-point groups. So instead of the general form (3.6) we can now write:

$$\begin{aligned}
 \oint_S F dy &= \int_{S_W} F dy + \int_{S_S} F dy + \int_{S_E} F dy + \int_{S_N} F dy \approx \\
 &\approx \alpha_{1W} \tilde{F}_1^h + \alpha_{2W} \tilde{E}_2^h + \alpha_{3W} \tilde{F}_0^h + \alpha_{4W} \tilde{E}_6^h + \alpha_{5W} \tilde{F}_7^h + \alpha_{6W} \tilde{E}_8^h + \\
 &+ \alpha_{1S} \tilde{F}_3^h + \alpha_{2S} \tilde{E}_4^h + \alpha_{3S} \tilde{F}_0^h + \alpha_{4S} \tilde{E}_8^h + \alpha_{5S} \tilde{F}_1^h + \alpha_{6S} \tilde{E}_2^h + \quad (3.19) \\
 &+ \alpha_{1E} \tilde{F}_5^h + \alpha_{2E} \tilde{E}_6^h + \alpha_{3E} \tilde{F}_0^h + \alpha_{4E} \tilde{E}_2^h + \alpha_{5E} \tilde{F}_3^h + \alpha_{6E} \tilde{E}_4^h + \\
 &+ \alpha_{1N} \tilde{F}_7^h + \alpha_{2N} \tilde{E}_8^h + \alpha_{3N} \tilde{F}_0^h + \alpha_{4N} \tilde{E}_4^h + \alpha_{5N} \tilde{F}_5^h + \alpha_{6N} \tilde{E}_6^h ,
 \end{aligned}$$

where, according to (3.17), the  $\alpha_i$  are linear functions of the  $y_i$ , with:

$$\alpha_{iW} = \alpha_{iW}(y_1, y_2, y_0, y_6, y_7, y_8) ,$$

$$\alpha_{iS} = \alpha_{iS}(y_3, y_4, y_0, y_8, y_1, y_2) ,$$

$$\alpha_{iE} = \alpha_{iE}(y_5, y_6, y_0, y_2, y_3, y_4) , \text{ and}$$

$$\alpha_{iN} = \alpha_{iN}(y_7, y_8, y_0, y_4, y_5, y_6) , \quad i = 1, \dots, 6 .$$

It is not necessary to consider also the discretization of  $\int_S E dy$ , because of (3.7).

In principle different functions for the  $\alpha_i$  could be used on different molecules in the domain, but we excluded this possibility, for reasons that were discussed in the previous section. Because only centered schemes are to be considered, each flux part will be discretized in the same way, which means that in (3.19) all the functions  $\alpha_{iW}$ ,  $\alpha_{iS}$ ,  $\alpha_{iE}$  and  $\alpha_{iN}$  will be taken equal to the function  $\tilde{\alpha}_i$ .

Each flux part at the left-hand side of (3.19) can be discretized in two different ways, since they are also to be approximated for the discretizations on each of the four adjacent molecules. For the first flux part we have (cf. (3.19)):

$$\int_{S_W} E dy \approx \tilde{\alpha}_1 \tilde{E}_1^h + \tilde{\alpha}_2 \tilde{E}_2^h + \tilde{\alpha}_3 \tilde{E}_0^h + \tilde{\alpha}_4 \tilde{E}_6^h + \tilde{\alpha}_5 \tilde{E}_7^h + \tilde{\alpha}_6 \tilde{E}_8^h , \quad (3.21)$$

$$\text{with: } \tilde{\alpha}_i = \tilde{\alpha}_i(y_1, y_2, y_0, y_6, y_7, y_8) , \quad i = 1, \dots, 6 ,$$

while the same flux term appears also in the flux integral of one of the adjacent finite volumes, where this term will be approximated by:

$$\int_{-S_W} E dy \approx \tilde{\alpha}_1 \tilde{E}_6^h + \tilde{\alpha}_2 \tilde{E}_7^h + \tilde{\alpha}_3 \tilde{E}_8^h + \tilde{\alpha}_4 \tilde{E}_1^h + \tilde{\alpha}_5 \tilde{E}_2^h + \tilde{\alpha}_6 \tilde{E}_0^h , \quad (3.22)$$

$$\text{with: } \tilde{\alpha}_i = \tilde{\alpha}_i(y_6, y_7, y_8, y_1, y_2, y_0) , \quad i = 1, \dots, 6 .$$

For the finite volume discretization to be conservative, both fluxes should be equal, and this can only be when:

$$\tilde{\alpha}_1(y_1, y_2, y_0, y_6, y_7, y_8) = -\tilde{\alpha}_4(y_6, y_7, y_8, y_1, y_2, y_0) ,$$

$$\tilde{\alpha}_2(y_1, y_2, y_0, y_6, y_7, y_8) = -\tilde{\alpha}_5(y_6, y_7, y_8, y_1, y_2, y_0) , \text{ and} \quad (3.23)$$

$$\tilde{\alpha}_3(y_1, y_2, y_0, y_6, y_7, y_8) = -\tilde{\alpha}_6(y_6, y_7, y_8, y_1, y_2, y_0) .$$

A similar analysis of the other three fluxes leads to the same result.

We are left now with only three different  $\tilde{\alpha}_i$ , each of them being a linear function of six different  $y$ -coordinate values, but because of the restriction in (3.17) only 15 degrees of freedom remain in the approximation (3.19) of the flux:

$$\begin{aligned}\tilde{\alpha}_i(y_1, y_2, y_3, y_4, y_5, y_6) &= \\ &= \theta_{i1}y_1 + \theta_{i2}y_2 + \theta_{i3}y_3 + \theta_{i4}y_4 + \theta_{i5}y_5 + \theta_{i6}y_6,\end{aligned}\quad (3.24)$$

$$\text{where: } \sum_j \theta_{ij} = 0, \quad i = 1, 2, 3. \quad (3.25)$$

Some more restrictions to the choice of the constants  $\theta_{ij}$  may be found if we require that the scheme should be at least free of numerical convection, i.e. the  $\tilde{\alpha}_i$  should satisfy at least the conditions (3.9) and (3.10). Condition (3.11), the expression for the surface of the finite volume, will follow automatically, once we have determined the flux approximations.

Transformed in  $\theta_{ij}$ -relations, condition (3.9) reads:

$$\sum \theta_{i3} = \sum \theta_{i6} \quad \text{and} \quad \sum (\theta_{i1} + \theta_{i5}) = \sum (\theta_{i2} + \theta_{i4}), \quad (3.26)$$

while (3.10) yields the conditions:

$$\begin{aligned}\theta_{33} &= 0, \quad \theta_{11} = \theta_{22}, \quad \theta_{12} + \theta_{21} = \theta_{23} + \theta_{32}, \quad \theta_{15} = \theta_{24}, \\ \theta_{31} + \theta_{35} &= \theta_{26} - \theta_{13}, \quad \text{and} \quad \theta_{32} + \theta_{34} = \theta_{16} - \theta_{23}.\end{aligned}\quad (3.27)$$

Conditions (3.26) and (3.27) are all necessary for (3.9) and (3.10) to hold for any set of coordinate values. Together with (3.25), they form a system of nine linearly independent equations, leaving nine degrees of freedom to the choice of the  $\theta_{ij}$ , although only six linearly independent finite volume discretizations result from it:

$$\oint_S F dy \approx E_x^* \cdot VOL \approx \quad (3.28)$$

$$\begin{aligned}\text{Scheme I: } \approx \tilde{F}_1^h(y_2 - y_8) + \tilde{F}_2^h(y_3 - y_1) + \tilde{F}_3^h(y_4 - y_2) + \tilde{F}_4^h(y_5 - y_3) + \\ + \tilde{F}_5^h(y_6 - y_4) + \tilde{F}_6^h(y_7 - y_5) + \tilde{F}_7^h(y_8 - y_6) + \tilde{F}_8^h(y_1 - y_7) = \\ = 2(V_{1208} + V_{2340} + V_{0456} + V_{8067}),\end{aligned}$$

Scheme II:  $\approx (\tilde{E}_2^h - \tilde{E}_6^h)(y_4 - y_8) + (\tilde{E}_4^h - \tilde{E}_8^h)(y_6 - y_2) = 2V_{2468}$ ,

Scheme III:  $\approx (\tilde{E}_1^h - \tilde{E}_5^h)(y_3 - y_7) + (\tilde{E}_3^h - \tilde{E}_7^h)(y_5 - y_1) = 2V_{1357}$ ,

Scheme IV:  $\approx \tilde{E}_1^h(y_4 - y_6) + \tilde{E}_2^h(y_5 - y_7) + \tilde{E}_3^h(y_6 - y_8) + \tilde{E}_4^h(y_7 - y_1) +$   
 $+ \tilde{E}_5^h(y_8 - y_2) + \tilde{E}_6^h(y_1 - y_3) + \tilde{E}_7^h(y_2 - y_4) + \tilde{E}_8^h(y_3 - y_5)$ ,

Scheme V:  $\approx \tilde{E}_1^h(y_4 - y_2) + \tilde{E}_2^h(y_1 - y_7) + \tilde{E}_3^h(y_6 - y_4) + \tilde{E}_4^h(y_3 - y_1) +$   
 $+ \tilde{E}_5^h(y_8 - y_6) + \tilde{E}_6^h(y_5 - y_3) + \tilde{E}_7^h(y_2 - y_8) + \tilde{E}_8^h(y_7 - y_5)$ ,

Scheme VI:  $\approx \tilde{E}_1^h(y_0 - y_6) + \tilde{E}_2^h(y_5 - y_0) + \tilde{E}_3^h(y_0 - y_8) + \tilde{E}_4^h(y_7 - y_0) +$   
 $+ \tilde{E}_5^h(y_0 - y_2) + \tilde{E}_6^h(y_1 - y_0) + \tilde{E}_7^h(y_0 - y_4) + \tilde{E}_8^h(y_3 - y_0) +$   
 $+ \tilde{E}_0^h(y_2 + y_4 + y_6 + y_8 - y_1 - y_3 - y_5 - y_7)$ .

The surface discretizations, corresponding to each of these finite volume schemes, can simply be found by the substitution of  $x_i$  instead of  $\tilde{E}_i^h$  in their respective formulas.

Similar results would have been obtained if we had considered the discretization of  $\oint_S G dx$ , as the finite volume approximation of  $-G_x \cdot VOL$ . For the schemes arising from the analysis we have, as in the continuous case where  $\oint_S x dy = -\oint_S y dx = \iint_V dxdy = VOL$ :

$$\sum_i x_i \alpha_i(y_j) = -\sum_i y_i \alpha_i(x_j) = VOL. \quad (3.29)$$

Hence the same surface for both the  $E_x$ - and the similar  $G_y$ -discretization, and the condition (3.11) has automatically been satisfied.

It is evident that any linear combination of the above-mentioned six schemes will also be free of numerical convection. To give a few examples: one half times Scheme II is obtained if we take the finite volume indicated by the dotted lines in FIGURE 5, while the dashed and the dashed-dotted lines in the figure represent finite volumes that lead to 1/2 Scheme I and 1/2 Scheme III respectively. The combination of 2/3 times Scheme I minus 1/6 of Scheme III comes from the large solid-line finite volume, using an isoparametric biquadratic interpolation to

approximate the unknowns. Finally, in order to determine the fluxes through the boundaries of a finite volume, we may also use piecewise bilinear approximations over each grid cell, in order to define  $E^h$  (and all the other functions) everywhere on the domain. In the case of the small solid-line volume this gives a discretization composed of 1/16 Scheme I plus 1/8 Scheme II, and the composition of 1/6 times Scheme I plus II if the dotted-line volume is considered. This last composition arises also from the use of GFEM, as was already mentioned before.

It is remarkable that the shapes of the finite volumes associated with the Schemes IV, V and VI are not well defined. But this is not really necessary either. As long as every approximation of a flux term is used twice, once as an ingoing and once as an outgoing flux, a scheme is conservative, just like any other more conventional finite volume discretization technique.

All schemes listed in (3.28) create spurious numerical diffusion, and so does any linear combination, due to the restrictions that we have put on this group (constant  $\theta_{ij}$ -coefficients). Fortunately, for practical applications this does not have to be a serious drawback. The design of a computational grid will often be such that highly irregular and arbitrary-looking shapes will not be present, or this situation can at least always be avoided when a structured grid is used. We may assume that in that case grid lines will be in general (nearly) parallel, at least locally, or will deviate only slightly and incidentally from that "ideal" situation. So the grid molecule of FIGURE 5, with its parallel sides, may be considered as a typical example of a molecule that occurs in practical applications.

On a molecule with parallel sides (side 1-8 parallel to side 2-0 parallel to side 3-4, etcetera) only Scheme I and III do not introduce numerical diffusion. The other schemes do, as well as any linear combination of them. This can easily be verified by checking if the conditions (3.12), (3.13) and (3.14) can all be satisfied for some value of  $x^*$  and  $y^*$ , which would determine then at the same time the quadrature point  $(x^*,y^*)$ . In general this will not hold, and in that case the coordinates of the \*-point will be defined such that both (3.13) and (3.14) are equally well approximated.

Obviously, the most accurate low-order conservative discretization scheme on a three-times-three-point computational molecule is to be

sought in the group of linear combinations of the Schemes I and III. By means of an order-of-magnitude estimation of the errors, introduced in both schemes by means of a small off-parallel deformation of the molecule, it was found that the optimal combination is Scheme I -  $\frac{1}{4}$  Scheme III. This is not a very surprising result, since this combination, apart from a factor  $2/3$ , corresponds to the large solid-line finite volume of FIGURE 5, with the fluxes approximated by means of an isoparametric, biquadratic interpolation. We will refer to this optimal combination as OLOCS: the Optimal Low-Order Conservative Scheme.

An interesting feature of OLOCS is that it is also free of numerical diffusion on a computational molecule where the coordinates satisfy the relations:  $x_2 = (x_1 + x_3)$ ,  $y_2 = (y_1 + y_3)$ ,  $x_4 = (x_3 + x_5)$ , etc. (see FIGURE 6). On such a molecule, the interpolation in between the grid point values is actually third-order accurate, showing that the scheme becomes of higher-order. So in some sense OLOCS is the low-order limit of a higher-order scheme. This knowledge may be useful for grid-generation techniques, where in "difficult" regions molecules like in FIGURE 6 can be used in order to avoid excessive spurious diffusion.

The numerical diffusion coefficients of OLOCS and GFEM are compared with each other in the FIGURES 7 and 8, for the grid that we have used to solve the steady viscous flow over a backward facing step. We see that, apart from the step corner where both methods are of about equal (in)accuracy and where the grid could have been designed better, OLOCS is indeed much more accurate than the equivalent GFEM, based on the same computational molecule.

### III.4 AN ACCURATE CONSERVATIVE DISCRETIZATION SCHEME FOR SECOND-DERIVATIVE TERMS

Concerning a discretization method for the diffusion terms of (3.1), it is not possible to derive in the same rigorous way as for the convective terms an optimal scheme, due to the nonlinear character of the problem. Instead, we have followed a more heuristic approach, by trying to combine optimal accuracy with ease of programming.

The first problem to be solved is to find an accurate approximation of the integrands of the diffusion integrals in (3.2). It can easily be seen, that integrals of nonlinear functions are to be evaluated, if the

vectors  $\underline{v}_i$  and  $\underline{w}_i$ , and the matrices  $V_i$  and  $W_i$ , are approximated piecewise bilinearly. This would necessitate the use of Gauss' integration formulas, like in the finite element technique, but this would increase the computational work and the required memory storage space considerably.

By using the knowledge acquired in the previous sections, a much simpler discretization technique could be designed, that has a strong resemblance with the optimal scheme for the convective terms. Namely, at the quadrature points of all nine quadrilaterals on a three-times-three-point computational molecule that have (approximately) parallel sides, the integrands can be discretized accurately by means of the finite volume discretization (3.4), where the coordinates of each quadrature point are given by  $((x_\alpha + x_\beta + x_\gamma + x_\delta)/4, (y_\alpha + y_\beta + y_\gamma + y_\delta)/4)$  (cf. FIGURE 2). So the integrands can be approximated well at point A with coordinates  $((x_1 + x_2 + x_0 + x_8)/4, (y_1 + y_2 + y_0 + y_8)/4)$ , by using the values of the unknowns at the points 1, 2, 0 and 8; at point B with coordinates  $((x_1 + x_3 + x_4 + x_8)/4, (y_1 + y_3 + y_4 + y_8)/4)$ , by using the values of the unknowns at the points 1, 3, 4 and 8; etcetera (FIGURE 9).

Obviously, only two schemes are to be considered for the discretization of the diffusion integrals. They are the equivalent of Scheme III and Scheme I, and obtained by a piecewise linear interpolation of the discretizations of the integrands over the sides of the volumes indicated in FIGURE 9:

$$\int_{S_W} \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy \approx \quad (3.30)$$

$$\text{Scheme VII: } \approx \frac{y_A - y_G}{2} \left[ (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_A + (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_G \right],$$

$$\begin{aligned} \text{Scheme VIII: } &\approx \frac{y_A - y_H}{2} \left[ (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_A + (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_H \right] + \\ &+ \frac{y_H - y_G}{2} \left[ (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_H + (\tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x})_G \right], \end{aligned}$$

where the integral is a part of the contour integral:

$$\begin{aligned} \iint_V \frac{\partial \tilde{V}_1^h}{\partial x} \frac{\partial \tilde{V}_2^h}{\partial x} dx dy &= \oint_S \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy = \quad (3.31) \\ &= \int_{S_W} \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy + \int_{S_S} \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy + \int_{S_E} \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy + \int_{S_N} \tilde{V}_1^h \frac{\partial \tilde{V}_2^h}{\partial x} dy. \end{aligned}$$

The other integrals at the right-hand side of (3.31), as well as the other diffusion terms of (3.2), are approximated similarly.

The integrand in (3.30) is discretized like in (3.4), by means of:

$$(\tilde{v}_1^h \frac{\partial \tilde{v}_2^h}{\partial x})_A \approx \frac{\tilde{v}_{11}^h + \tilde{v}_{12}^h + \tilde{v}_{10}^h + \tilde{v}_{18}^h}{4} \frac{(\tilde{v}_{21}^h - \tilde{v}_{20}^h)(y_2 - y_8) + (\tilde{v}_{22}^h - \tilde{v}_{20}^h)(y_0 - y_1)}{(x_1 - x_0)(y_2 - y_8) + (x_2 - x_8)(y_0 - y_1)}, \quad (3.32)$$

where the second subscript indicates at which grid point the values of the unknowns are taken. Both discretization schemes are such, that no numerical diffusion is present in the truncation error when the grid cells have parallel sides. Otherwise, the schemes introduce a small inconsistency error of  $O(\Delta h/h)$  in the equivalent equation (3.3).

The optimal combination of the two schemes was found to be Scheme VII - 2/5 Scheme VIII; it introduced an at most 8 % excessive viscosity in our calculation of the viscous flow over a backward facing step (FIGURE 10). The surface of the finite volume of this scheme can be found by substituting  $x$  for  $\tilde{v}_1^h$  in (3.30), putting  $\partial \tilde{v}_2^h / \partial x$  equal to 1.

If the ratio between this surface and the surface of OLOCS would be constant, the scheme as a whole would be strictly conservative. This will approximately hold on the type of grids necessary for the developed methods to be sufficiently free of numerical diffusion.

The discretization of the integral over the source term  $\xi$  in (3.2) is rather simple, once we have determined the coordinates of the \*-point. Like in (2.8), we can interpolate in between the values of  $\xi^h$  at the nine grid points of the molecule, to determine as accurately as possible its value at the \*-point. This yields again an optimal mass lumping scheme, which may be of considerable importance for the discretization in space of a time derivative. Certainly it would be important if we want to do unsteady calculations, because of reasons of accuracy, but also if we would like to use a time-marching method as solution procedure to converge to the steady-state solution of (3.1) the optimal mass lumping technique may prove useful.

### III.5 CONCLUDING REMARKS

The above-mentioned accurate discretization in space of a physical time derivative has been implemented and tested already in the framework of the present numerical Navier-Stokes solver by Kloker (see his contribution to these lecture notes). Good results were obtained, although at present we think that the integration over the whole finite volume of a higher-order interpolation of the  $s^h$  at the grid points, like in (2.7), may perform much better. It would make the discretization in space of a time derivative fourth-order accurate, which may reduce considerably the phase error of an unsteady numerical flow simulation, much like in unsteady finite element calculations the use of the consistent mass matrix is to be preferred over the use of the lumped mass matrix [60, 61]. As this topic is presently under study, we will probably be able to present some preliminary analysis results during the lectures.

Concerning the discretization in space of a *numerical* time derivative, it is common use in time-marching methods to assign the calculated corrections at each (non-physical) time step to the unknowns at the inner grid points of the molecules (point 0 in FIGURE 5). With respect to these points, the method may contain a large quantity of numerical diffusion, with the exception of very regular shapes.

Although this does not influence the steady state solution, where the (pseudo-)time derivative should normally vanish, it may have considerable consequences for the convergence speed of the solution procedure. Every time step, the solution corrections will be calculated by means of an unsteady discretization that introduces a molecule-dependent diffusion, which may be very large and negative! This may result in an iterative scheme that is very sensitive to the grid structure, with convergence properties that are far from the analytically predicted ones, due to the (local) ill-posedness of the unsteady equivalent equation. It may be very difficult to cope with this effect, because of its local grid-structure dependency.

Alternatively, we could also evaluate the numerical time step at the \*-points, not so much as to gain accuracy, but more as an attempt to stabilize and improve the performance of time-marching methods. It is expected that this procedure will make such iteration methods far less grid sensitive.

## CHAPTER IV

### **TOWARD A FIRST-ORDER IMPLICIT RELAXATION SCHEME FOR THE COMPRESSIBLE NAVIER-STOKES EQUATIONS**

A series of physical and numerical arguments leads to the conclusion that a solution method for the discretized steady-state compressible Navier-Stokes equations that combines robustness, general applicability and good convergence properties is to be sought in the class of first-order implicit relaxation schemes [47]. Explicit methods, on the contrary, must be expected to be intrinsically slow, due to their rather severe stability limits [47, 62].

The most convenient way of constructing such an implicit relaxation scheme is by means of a backward Euler discretization in time of the unsteady equations, as this scheme yields the highest convergence rates. For an infinitely large time step  $\Delta t$ , such a scheme reduces to a direct solver, that has of course an infinite speed of convergence. However, a direct solver is not feasible anymore for nonlinear and/or multi-dimensional problems.

In the first case one cannot really speak of a "direct" solver anymore, since the system of equations has to be linearized in order to be solvable. A Newton-Raphson iterative procedure could be used, which will converge quadratically to the solution  $\Phi^*$  of the system, if the initial solution  $\Phi^0$  is sufficiently close to  $\Phi^*$ . Else a finite  $\Delta t$  must be used, in order to relax the (too) strong steady-state perturbations, introduced in the system by  $\Phi^0$ . Moreover, due to the complexity of certain expressions (e.g. the models for the physical, turbulent, and artificial viscosity coefficients), some terms often can only be treated explicitly,

and hence would make the use of a true Newton-Raphson method impossible anyway.

In the second case, a direct solver is still possible when the multi-dimensional problem is linear, but will be very uneconomical, due to the sparseness of the matrix to be inverted [63,p167]. So the differential problem should be discretized implicitly in time, but such that the resulting system of equations can be solved very efficiently. On the other hand, the system should still approximate the backward Euler discretization rather closely. Both demands can be fulfilled by a special form of the implicit system of equations, based on a splitting of the difference operators in space. The system of equations can then be written as a couple of nested systems of equations that can all be solved very efficiently.

This technique is known as *implicit approximate factorization* [13, 42, 43]. At convergence, the correct steady-state equation will be solved, so factorization does not influence the steady-state solution of the problem. However, it creates a factorization error, and due to this error the iterative scheme does not reduce to a direct solver when  $\Delta t \rightarrow \infty$ , but on the contrary tends to a system with zero speed of transfer [64]. The choice of the relaxation parameter  $\Delta t$  becomes therefore very critical. The purpose of this chapter is to develop a theoretical basis for the estimation of the optimal time step for this implicit approximate factorization scheme (AF).

#### IV.1 OPTIMIZATION OF A FIRST-ORDER IMPLICIT RELAXATION SCHEME FOR A TWO-DIMENSIONAL CONVECTION-DIFFUSION EQUATION

The following linear model equation is considered:

$$\frac{\partial \Phi}{\partial t} + u \frac{\partial \Phi}{\partial x} + v \frac{\partial \Phi}{\partial y} = v_x \frac{\partial^2 \Phi}{\partial x^2} + v_y \frac{\partial^2 \Phi}{\partial y^2} + f, \quad (4.1)$$

$$\Phi = \Phi(x, y, t), \quad f = f(x, y).$$

The unknown in this unsteady convection-diffusion equation with source term  $f$ , constant convective speeds  $u$  and  $v$ , and constant diffusion coefficients  $v_x$  and  $v_y$  is the function  $\Phi$ .

We would like to find the steady-state solution  $\Phi^*(x,y)$  of the differential equation (4.1) with appropriate boundary conditions. To this end, the equation is discretized implicitly in time so as to construct a first-order implicit relaxation scheme, with which  $\Phi^*$  can be determined iteratively. For reasons of efficiency, the two space dimensions in the implicit part are uncoupled through the AF:

$$[1 + \Delta t(u \frac{\partial}{\partial x} - v_x \frac{\partial^2}{\partial x^2})] [1 + \Delta t(v \frac{\partial}{\partial y} - v_y \frac{\partial^2}{\partial y^2})] \Delta\Phi^t = \Delta t \text{RHS}(\Phi^{t-1}) \quad (4.2)$$

$$t = 1, 2, 3, \dots$$

The unknown  $\Delta\Phi^t$  of this equation denotes the *correction* or *increment*  $\Phi^t - \Phi^{t-1}$ . The addition of  $\Delta\Phi^t$  to the previously determined approximation  $\Phi^{t-1}$  yields the next "improved" approximation  $\Phi^t$  of  $\Phi^*$ . The iterative process is initialized by means of some suitably chosen  $\Phi^0$ . The symbol *RHS* at the right-hand side of (4.2) stands for the differential operator:

$$\text{RHS}(\Phi) = -u \frac{\partial \Phi}{\partial x} - v_x \frac{\partial \Phi}{\partial y} + v_x \frac{\partial^2 \Phi}{\partial x^2} + v_y \frac{\partial^2 \Phi}{\partial y^2} + f.$$

The factorization error of (4.2) is  $O(\Delta t^2)$ , indicating that for large time steps, the factorization error becomes very large. As we mentioned already before, this is known to have a negative effect on the convergence properties of the scheme. For very small  $\Delta t$ , on the contrary, the factorization error will be very small, but the speed of convergence will be small also. It is clear that there must be a finite optimal time step, for which value the positive effect of increasing  $\Delta t$  for still better convergence properties, is balanced by the negative effect of an increasing factorization error. The subject of this chapter is to determine that value, by means of a Fourier mode analysis.

But first of all, we will derive the recursive expression that is satisfied at every iteration step by the discretized steady-state error  $\Delta\Phi^{*t}$ . Here,  $\Delta\Phi^{*t}$  denotes the difference between the approximation  $\underline{\Phi}^t$  of the vector of nodal-point values  $\Phi^*$ , and  $\Phi^*$  itself, where the vector  $\underline{\Phi}^t$  is the solution of the system of algebraic equations  $\text{RHS}^h(\underline{\Phi}) = 0$ , obtained upon the discretization in space of the steady-state differential equation  $\text{RHS}(\Phi) = 0$ .

The equation for  $\Delta\Phi^{*t}$  is obtained upon subtracting  $0 = \text{RHS}^h(\underline{\Phi}^*)$  from the discretized equation (4.2). Since  $\text{RHS}^h$  determines the accuracy of

the numerical approximation  $\Phi^{th}$ , the operator  $RHS$  will be discretized using the finite volume discretization techniques that have been developed in CHAPTER III. As the left-hand side of (4.2) vanishes at steady state, its discretization does not affect the accuracy of the numerical approximation  $\Phi^{th}$ , and can therefore be realized in principle in any convenient way. However, the discretization of this operator is obviously of great influence on the convergence properties of the scheme.

On a uniform cartesian grid of  $N$  times  $M$  grid cells, of size  $\Delta x$  times  $\Delta y$ , the algebraic operator  $RHS^h$  for the inner points, obtained upon the discretization of the integral  $\iint RHS dx dy$  over a finite volume of  $2\Delta x$  times  $2\Delta y$ , becomes:

$$\begin{aligned} 4\Delta x \Delta y RHS^h = & - 2\Delta y u \left(1 + \frac{1}{6}\delta_y^2\right) (\mu\delta)_x - 2\Delta x v \left(1 + \frac{1}{6}\delta_x^2\right) (\mu\delta)_y + \\ & + 4\Delta y v_x \left(1 + \frac{1}{6}\delta_y^2\right) \frac{\delta_x^2}{\Delta x} + 4\Delta x v_y \left(1 + \frac{1}{6}\delta_x^2\right) \frac{\delta_y^2}{\Delta y} + f, \end{aligned} \quad (4.3)$$

where  $\delta_x$  and  $\mu_x$  denote the central difference operators:

$$\delta_x \Phi_{ij} = \Phi_{i+1/2,j} - \Phi_{i-1/2,j} \quad \text{and} \quad \mu_x \Phi_{ij} = \Phi_{i+1/2,j} + \Phi_{i-1/2,j},$$

and where  $\delta_y$  and  $\mu_y$  denote similar operators in the  $y$ -direction, with  $i$  and  $j$  the indices of the nodal points in  $x$ - and  $y$ -direction respectively.

The numerical approximation  $RHS^h$  of  $RHS$  on a three-times-three-point computational molecule is illustrated in a schematic way in FIGURE 11. It shows more clearly the relation that is established between the unknowns at the different nodal points, when the discretization techniques of CHAPTER III are used in combination with this specific grid. The discretization weights for the different terms of  $RHS$  are exactly the same as the coefficients that one would have obtained by applying a finite element Galerkin technique with piecewise bilinear elements in combination with the same regular grid. On an irregular grid, both methods would have led to different discretizations.

The equation (4.2) is discretized in two parts as follows:

$$\begin{aligned} [(1 + \alpha\delta_x^2) \frac{4\Delta x \Delta y}{\Delta t} + 2\Delta y u \left(\frac{2}{3} + \frac{\beta}{3}\right) (\mu\delta)_x - 4\Delta y v_x \left(\frac{2}{3} + \frac{\gamma}{3}\right) \frac{\delta_x^2}{\Delta x}] \tilde{\Delta\Phi}_{ij}^t = \\ = 4\Delta x \Delta y RHS^h (\Phi_{ij}^{t-1}), \end{aligned} \quad (4.4a)$$

$$[(1 + \alpha \delta_y^2) \frac{4 \Delta x \Delta y}{\Delta t} + 2 \Delta x v \left( \frac{2}{3} + \frac{\beta}{3} \right) (\mu \delta)_y - 4 \Delta x v_y \left( \frac{2}{3} + \frac{\gamma}{3} \right) \frac{\delta^2}{\Delta y}] \Delta \Phi_{ij}^t = 4 \Delta x \Delta y \tilde{\Delta \Phi}_{ij}^t / \Delta t , \quad (4.4b)$$

for  $0 < i < N$  and  $0 < j < M$ .

Expression (4.4a), when completed in a suitable way by  $2M-4$  equations at the boundaries, represents a system of equations, where the unknowns are uncoupled in  $j$ -direction. In this way, the intermediate correction  $\tilde{\Delta \Phi}_{ij}^t$  can indeed be calculated very efficiently from the right-hand side vector  $RHS^h(\Phi_{ij}^{t-1})$ , by solving one system of equations per  $j$ -line. The calculation of the correction  $\Delta \Phi_{ij}^t$  to the approximation  $\Phi_{ij}^{t-1}$  becomes equally efficient, by solving systems of equations that are associated with the  $i$ -lines.

In this realization of the implicit relaxation scheme, the parameter  $\alpha$  determines the way the discretized time derivative is distributed over the nodal points of the computational molecule. This molecule is composed of four grid cells with in total three times three nodal points, while the equations (4.4a) and (4.4b) couple essentially only the five nodal points of the molecule on the  $i$ -line and on the  $j$ -line. So the parameters  $\beta$  and  $\gamma$  have been introduced to be able to compensate for this loss of information.

This is explained in more detail in FIGURE 12. The factors  $1+\delta_y^2/6$  and  $1+\delta_x^2/6$  in expression (4.3) indicate that on a uniform grid the optimal finite volume discretizations of CHAPTER III differ from the classical central finite difference discretizations  $\mu \delta/2h$  and  $\delta^2/h^2$  for first and second derivatives respectively, in that they distribute these differences over three grid lines with the weights  $1/6$ ,  $2/3$  and  $1/6$ . So two thirds of the discretization coefficients are related to unknowns at nodal points on the two center lines of the molecule (the  $i$ - and  $j$ -line), and one third to unknowns at the six nodal points on the off-center lines.

Only the unknowns on the two center lines of the molecule appear in the implicit part (the left-hand sides of the two expressions (4.4)) of the relaxation scheme. This is due to the restriction that we have put on the type of system of equations that may be used for the scheme (of tridiagonal structure only).

We have investigated the possibility whether the explicit treatment of the unknowns from the corner nodes may be compensated for to some

extend by redistributing their weight factors over the other five nodal points of the molecule (FIGURE 12). When different redistributions are used for the convective terms and for the diffusion terms, it can easily be verified that this redistribution boils down to replacing the operators  $1+\delta_x^2/6$  and  $1+\delta_y^2/6$  by the factor  $2/3+\beta/3$  for the first derivative terms, and by  $2/3+\gamma/3$  for the second derivatives. With  $\beta$  and  $\gamma$  equal to zero, or one, the weights for the corner points are not at all, or fully taken into account. Clearly, the optimal value for these two parameters must be somewhere in between these extrema.

The convergence properties of the scheme can be determined by the study of the equation that expresses the next steady-state error  $\Delta\Phi_{ij}^{*t}$  in terms of the previous one  $\Delta\Phi_{ij}^{*t-1}$ . This equation is obtained upon subtracting  $\Delta t R H S(\Phi_{ij}^*)$ , which expression is equal to zero, from the right-hand side of (4.4a), and upon the substitution of the intermediate correction  $\tilde{\Delta\Phi}_{ij}^t$  in (4.4a) by the left-hand side of (4.4b), where the correction  $\Delta\Phi_{ij}^t$  has been written as the difference between  $\Delta\Phi_{ij}^{*t}$  and  $\Delta\Phi_{ij}^{*t-1}$ . For convergence, the sequence of successive corrections should satisfy the relation:

$$|\Delta\Phi^{*t}| \leq K e^{-\epsilon t} |\Delta\Phi^{*0}|, \quad t = 1, 2, 3, \dots, \quad (4.5)$$

for any  $\Phi^0$  that is sufficiently close to  $\Phi^*$ , i.e. satisfies  $|\Delta\Phi^{*0}| \leq L$ , where  $\epsilon$  and  $L$  are positive constants, and where  $K$  is a polynomial in  $t$  of finite degree with positive coefficients. Notice that the variable  $t$  now indicates the iteration level, and can therefore only take positive integer values.

The maximum possible value of  $\epsilon$  in (4.5),  $\epsilon_{max}$ , indicates the asymptotic speed of convergence. In order to verify whether inequality (4.5) holds for relaxation scheme (4.4), we will derive the expression for its amplification matrix  $G^h$ , defined as  $\Delta\Phi^{*t}/\Delta\Phi^{*t-1}$ , by means of a Fourier mode analysis.

Such an analysis does not allow to include many important features, such as the effect of boundary conditions and irregular grids, in the study of the convergence properties of the scheme, but it is the only practical way that exists to optimize (although to a rather limited extend) the relaxation parameters  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\Delta t$ . We will suppose therefore that the steady-state error of (4.1) and of (4.2) is periodic in space, over a length  $L_x$  in  $x$ -direction, and over a length  $L_y$  in  $y$ -

direction. Normally,  $L_x$  and  $L_y$  are multiples of  $N\Delta x$  and  $M\Delta y$ , the size of the domain that we consider, and depend on the type of boundary conditions, as we will see in the next chapter. The prolongation  $\Delta\Phi^{*ht}$  of  $\Delta\Phi^{*t}$  can then be written as the infinite sum of trigonometric functions:

$$\Delta\Phi^{*ht}(x, y) = \sum_{p, q > 0} \Delta\Phi_{pq}^{*ht},$$

where the  $\Delta\Phi_{pq}^{*ht}$  denote the Fourier modes:

$$\Delta\Phi_{pq}^{*ht} = [A^t \cos(\omega_p \frac{x}{\Delta x}) + B^t \sin(\omega_p \frac{x}{\Delta x})] [C^t \cos(\omega_q \frac{y}{\Delta y}) + D^t \sin(\omega_q \frac{y}{\Delta y})], \quad (4.6)$$

with the frequencies:

$$\omega_p = p \frac{2\pi\Delta x}{L_x}, \quad \text{and} \quad \omega_q = q \frac{2\pi\Delta y}{L_y}, \quad p, q = 1, 2, \dots.$$

For the convergence analysis, it is sufficient to consider only the frequencies lower than or equal to  $\pi$ , since this is the highest frequency that can be represented unambiguously on the grid.

The amplification factor per Fourier mode is found immediately from the relation between  $\Delta\Phi^{*t}$  and  $\Delta\Phi^{*t-1}$ , by using expression (4.6) to replace the value of the unknowns at the nodal points. This yields the rather complex expression:

$$G_{pq}^h = \frac{\sqrt{\text{numer}}}{\sqrt{\text{denom}}}, \quad (4.7)$$

where the numerator and the denominator are respectively:

$$\begin{aligned} \text{numer} = & [(1 - \alpha_p + \gamma_p)(1 - \alpha_q + \gamma_q) - \gamma_p \frac{2 + \cos \omega_q}{2 + \gamma} - \gamma_q \frac{2 + \cos \omega_p}{2 + \gamma}]^2 + \beta_p^2 \beta_q^2 + \\ & + \beta_p^2 [1 - \alpha_q - \frac{2 + \cos \omega_q}{2 + \beta} + \gamma_q]^2 + \beta_q^2 [1 - \alpha_p - \frac{2 + \cos \omega_p}{2 + \beta} + \gamma_p]^2, \end{aligned}$$

$$\begin{aligned} \text{denom} = & [(1 - \alpha_p + \gamma_p)(1 - \alpha_q + \gamma_q)]^2 + \beta_p^2 \beta_q^2 + \\ & + \beta_p^2 [1 - \alpha_q + \gamma_q]^2 + \beta_q^2 [1 - \alpha_p + \gamma_p]^2, \end{aligned}$$

$$\text{with: } \alpha_p = \alpha(2 - 2 \cos \omega_p), \quad \alpha_q = \alpha(2 - 2 \cos \omega_q),$$

$$\beta_p = u \frac{\Delta t}{\Delta x} \left( \frac{2 + \beta}{3} \right) \sin \omega_p, \quad \beta_q = v \frac{\Delta t}{\Delta y} \left( \frac{2 + \beta}{3} \right) \sin \omega_q,$$

$$\gamma_p = v_x \frac{\Delta t}{\Delta x^2} \left( \frac{2+\gamma}{3} \right) (2 - 2 \cos \omega_p), \quad \gamma_q = v_y \frac{\Delta t}{\Delta y^2} \left( \frac{2+\gamma}{3} \right) (2 - 2 \cos \omega_q).$$

A sufficient condition for unconditional or A-stability is that for all Fourier modes, every term in the numerator is smaller than or equal to the corresponding term in the denominator, independent of the (non-negative!) time step. A term-by-term comparison shows that the scheme is A-stable under the following restrictions of the parameters:

$$\beta, \gamma \geq -1/2, \quad \alpha \leq \min\left(\frac{3+2\beta}{16+8\beta}, \frac{3+2\gamma}{16+8\gamma}\right).$$

In other words, the scheme is unconditionally stable for a large part of the range of parameter values that we would like to consider, namely for  $0 \leq \beta, \gamma \leq 1$  and  $\alpha \leq 3/16$ .

For optimal convergence, which is after all the purpose of the present analysis,  $\alpha, \beta, \gamma$  and  $\Delta t$  should be taken such that they minimize the maximum of  $G_{pq}^h$  over  $p$  and  $q$ , since it is this value that determines the asymptotic speed of convergence of the scheme. Because of the complexity of expression (4.7), it is impossible to determine the optimal expressions of the relaxation parameters  $\alpha, \beta, \gamma$  and  $\Delta t$  analytically, as a function of  $u, v, v_x, v_y, \Delta x, \Delta y, L_x$  and  $L_y$ . Instead, we will use the computer in order to gain some basic insight in the maximum of the function  $G_{pq}^h$ . This could then lead to some indications about the simplifications that can be made, after which an analytical approach will be feasible.

First of all we have restricted ourselves to the case that the convection and the diffusion is the same in both spatial directions, by introducing  $u = v$ , and  $v_x = v_y$ . We will present the maximum of  $G_{pq}^h$  as a function of the CFL-number  $CFL$  and the cell Reynolds number  $Re_c$ , that we have defined as:

$$CFL = \frac{u \Delta t}{\sqrt{(\Delta x \Delta y)}}, \quad Re_c = \frac{u \sqrt{(\Delta x \Delta y)}}{v_x}.$$

Actually, we will give the value of the natural logarithm of the inverse of  $\max(G_{pq}^h)$ , since this corresponds to the asymptotic speed of convergence  $\epsilon_{\max}$  of the scheme. It is very convenient to consider  $\epsilon_{\max}$  as a function of non-dimensionalized variables only. Hence:

$$\epsilon_{\max} = \epsilon_{\max}(\alpha, \beta, \gamma, CFL, Re_c, \frac{\Delta x}{\Delta y}, \frac{L_x}{\Delta x}, \frac{L_y}{\Delta y}), \quad (4.8)$$

where  $\Delta x / \Delta y$  denotes the grid-cell ratio, and where  $L_x / \Delta x$  and  $L_y / \Delta y$  are the dimensions of the periodicity domain, expressed in number of grid cells. The first four variables in (4.8) should be optimized in terms of the last four variables, in the sense that  $\epsilon_{\max}$  should be as large as possible.

The optimization of  $\alpha$ ,  $\beta$  and  $\gamma$  has primarily been done by means of the computer. It could be shown that the optimal configuration of the scheme is obtained with  $\alpha = \beta = \gamma = 0$  [47]. For this scheme, we have calculated  $\epsilon_{\max}$  as a function of  $CFL$  and  $Re_c$  for several values of the parameters  $\Delta x / \Delta y$ ,  $L_x / \Delta x$  and  $L_y / \Delta y$ , by calculating for every set of parameters the maximum of  $G_{pq}^h$  with respect to  $\omega_p$  and  $\omega_q$ . A typical result is presented in FIGURE 13, together with the two frequencies for which the maximum of  $G_{pq}^h$  is reached. Similar results are obtained for other values of  $\Delta x / \Delta y$ ,  $L_x / \Delta x$  and  $L_y / \Delta y$  [47].

All results show the same trend, in the sense that for any set of parameters, and for any  $Re_c$ , the optimal CFL-number is finite, and that when  $CFL \rightarrow \infty$  the speed of convergence tends to zero. This asymptotic behavior can also be deduced directly from (4.7) and is in agreement with the discussion in the introduction of this chapter. Another observation to be made is that the optimum of the CFL-number which maximizes  $\epsilon_{\max}$  for a certain value of  $Re_c$ , is rather pronounced. This phenomenon can be explained as follows.

For increasing  $CFL$ , starting from zero, each mode is damped out better and better, until the value is reached for which that particular mode is damped out optimally. After this point, the damping starts deteriorating again, due to the increasing factorization error. Of course, the optimal damping of different modes occurs in general at different  $CFL$ . Hence for each CFL-number, error modes are damped out with different speeds.

For a specific set of parameter values, the asymptotic speed of convergence is determined by the worst damped mode, or critical mode, that maximizes  $G_{pq}^h$ . Furthermore, for different parameter values, the critical mode will in general be different. So leaving all the other parameters constant, there must be at least one value of  $CFL$ , that we will indicate by  $CFL^*$ , for which the critical mode for  $CFL < CFL^*$  is equally well damped as the critical mode for the range  $CFL > CFL^*$ . In other words,

for  $CFL = CFL^*$ , two modes are critical. This has been illustrated graphically in FIGURE 14.

Our observations have indicated that for each set of values of  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $Re_c$ ,  $\Delta x/\Delta y$ ,  $L_x/\Delta x$  and  $L_y/\Delta y$ , there is only one such  $CFL^*$ , for which value  $\epsilon_{\max}$  always has its maximum. Therefore  $CFL^*$  is referred to as the *optimal CFL-number*. The two critical modes involved in its determination are essentially a function of  $Re_c$ .

FIGURE 13 shows that at low cell Reynolds numbers, the two critical modes that are damped out with a speed  $\epsilon_{\max}$  when the CFL-number is equal to  $CFL^*$ , are the modes  $(\pi, \pi)$  and  $(2\pi\Delta x/L_x, 2\pi\Delta y/L_y)$ . For medium high cell Reynolds numbers, we have that by approximation the modes  $(\pi/2, \pi/2)$  and  $(2\pi\Delta x/L_x, 2\pi\Delta y/L_y)$  become equally well damped at  $CFL = CFL^*$ , while for high  $Re_c$  the critical modes at  $CFL^*$  are  $(\pi/2, \pi/2)$  and  $(\pi, \pi)$ . We observed that this general behavior of  $CFL^*$  does not change substantially with  $\Delta x/\Delta y$ ,  $L_x/\Delta x$  and  $L_y/\Delta y$ . The optimal CFL-number for each of the three ranges can therefore easily be determined from (4.7), since for that value the amplification factor of both critical modes should be equal. This yields:

$$\begin{aligned} CFL_l^* &\approx \frac{Re_c}{8} \left[ \left( \frac{\Delta x + \Delta y}{\Delta y \Delta x} \right)^2 + 3 \frac{1/\Delta x^2 + 1/\Delta y^2}{\pi^2 / L_x^2 + \pi^2 / L_y^2} \right]^{1/2} - \frac{\Delta x}{\Delta y} - \frac{\Delta y}{\Delta x}, \\ CFL_m^* &\approx \sqrt{3.75} \left[ \left( \frac{\Delta x + \Delta y}{\Delta y \Delta x} \right)^2 + 12 \frac{1/\Delta x^2 + 1/\Delta y^2}{\pi^2 / L_x^2 + \pi^2 / L_y^2} \right]^{1/2} + \frac{\Delta x}{\Delta y} + \frac{\Delta y}{\Delta x} \right]^{1/2}, \text{ and} \\ CFL_h^* &\approx .75 (2Re_c)^{1/3}, \end{aligned} \quad (4.9)$$

where  $CFL_l^*$ ,  $CFL_m^*$  and  $CFL_h^*$  denote the "analytically" predicted optimal value for the CFL-number, for low, medium high and high cell Reynolds number respectively.

The approximation (4.9) of the function  $CFL^*(Re_c, \Delta x/\Delta y, L_x/\Delta x, L_y/\Delta y)$  is shown in FIGURE 13, by means of the dashed line. We see that the prediction is excellent, and certainly good enough to be used in practice. It shows that for optimal convergence of the implicit factorization scheme (4.4), the time step at low  $Re_c$  should be taken proportional to the number of points per direction in the grid, while at medium high Reynolds numbers  $\Delta t$  should be proportional to  $\sqrt{(\Delta x \Delta y)}$ .

The former result has been derived already before in the literature [65], while the latter has been established empirically by Pulliam [19, 41], who obtained a considerable increase in convergence speed in this

way. However, the expressions (4.9) are more complete, and take into account not only the size  $\Delta x \Delta y$  of the grid cell, but also the grid cell ratio and the length and width of the periodicity domain. As far as we know, it is the first time that they have been derived, and they will permit us to use a local time-stepping technique in the implicit relaxation scheme for the Navier-Stokes equations, for which not a single parameter had to be "tuned".

It should be mentioned that the expressions in (4.9) have been obtained by considering the full system of Fourier modes (4.6), assuming that all modes are to be damped out by the relaxation scheme (4.4). However, in practice the computational domain is not periodic but bounded, and different (relaxation) schemes are applied at the boundaries due to the discretization of physical and numerical boundary conditions. This may strongly enhance the damping of certain error modes, which as a consequence do not have to be considered anymore in the derivation of the optimal CFL-number (cf. CHAPTER V).

Also the dissipative properties of physical boundary conditions may improve substantially the convergence properties of the scheme as a whole. As these effects cannot be included in a Fourier mode analysis, only heuristic arguments and numerous testing may decide about how a better prediction of  $CFL^*$  should be obtained in practice.

## CHAPTER V

### NUMERICAL SOLUTION OF THE STEADY-STATE COMPRESSIBLE NAVIER-STOKES EQUATIONS - METHOD AND RESULTS

In this chapter we will present the results that were obtained with the calculation of the numerically approximated steady-state solution of the two-dimensional, compressible Navier-Stokes equations, using the discretization technique and iterative solution method that have been developed. The system of partial differential equations that we have considered can be written as:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{\partial}{\partial x} [V_1 \frac{\partial V_2}{\partial x} + V_3 \frac{\partial V_4}{\partial y}] + \frac{\partial}{\partial y} [W_1 \frac{\partial W_2}{\partial x} + W_3 \frac{\partial W_4}{\partial y}] , \quad (5.1)$$

where the vectors  $U$ ,  $F$ ,  $G$ ,  $V_2$ ,  $V_4$ ,  $W_2$  and  $W_4$ , and the matrices  $V_1$ ,  $V_3$ ,  $W_1$  and  $W_3$  have been given in APPENDIX I. Defined in this way, the equations (5.1) describe the two-dimensional motion of a compressible viscous Newtonian fluid, where the effect of the bulk viscosity and of external forces has been neglected.

The conservation equations for mass, x-momentum, y-momentum and energy have been non-dimensionalized by means of a reference temperature  $T_r$ , a reference pressure  $p_r$ , and a reference density  $\rho_r$ , using  $\sqrt{RT_r}$  as the reference velocity, where  $R$  is the gas constant. For the additional relation for the thermodynamic quantities, necessary to complete the system (5.1), a perfect gas has been assumed (APPENDIX I).

Concerning the boundary conditions, at the outlet of the numerical flow domain the static pressure will be imposed, while at the inlet

either the total conditions and the flow direction, or the static temperature together with the two velocity components will be fixed. The inviscid characteristic theory has been used to determine the equations that describe the transport of information from inside the domain toward the boundaries. They will be discretized at inlet and outlet in order to close the system of algebraic equations at these two boundaries (the numerical boundary procedures).

As for the solid wall boundaries, we will assume that they are adiabatic (i.e. normal derivative of temperature equal to zero), and that the velocity of the flow vanishes at the wall (no-slip condition). The latter approximation is valid if the fluid can be considered as a continuous medium, as we will suppose to be the case. For the closure of the system of equations at the solid wall boundaries, a fourth (numerical) condition has to be imposed. We will use an approximate characteristic equation (i.e. the momentum equation normal to the wall), that models the transport of information from inside the domain toward the wall.

#### V.1 THE FIRST-ORDER IMPLICIT RELAXATION SCHEME FOR THE DISCRETIZED TWO-DIMENSIONAL NAVIER-STOKES EQUATIONS

Since the system of equations (5.1) is hyperbolic/parabolic in time, any unsteady flow motion that is described by these equations will finally damp out and converge to a stationary solution, provided that such a solution exists under the given physical conditions. Iterative solution procedures for the discretized steady-state Navier-Stokes equations may therefore be based upon an (inaccurate) simulation of this unsteady physical process, which is the basic idea of using *time-marching schemes* as iterative solution methods. Optimal convergence may be expected when (5.1) is discretized in time by means of a backward Euler scheme.

This yields a first-order implicit relaxation scheme, that should however be linearized and factorized for reasons of efficiency. The maximum possible speed of convergence will decrease, but this is compensated for by the reduced amount of work and memory that is needed per iteration step.

In order that the implicit discretization in time of the nonlinear equations (5.1) will lead to easily solvable tridiagonal systems of

equations, the implicit part of the system should first of all be linearized. For optimal convergence, the linearized backward Euler scheme should approximate the true (nonlinear) backward Euler discretization as close as possible, so the linearization error should be as small as possible. The most accurate way of linearizing the equations is by means of a second-order accurate Taylor-series expansion in time, which yields the expression:

$$\begin{aligned} \frac{\Delta U^t}{\Delta t} + \frac{\partial}{\partial x}[(A-P-R)^{t-1}\Delta U^t] + \frac{\partial}{\partial y}[(B-Q-S)^{t-1}\Delta U^t] &= \\ = \frac{\partial}{\partial x}[-F+V_1 \frac{\partial V_2}{\partial x}+V_3 \frac{\partial V_4}{\partial y}]^{t-1} + \frac{\partial}{\partial y}[-G+W_1 \frac{\partial W_2}{\partial x}+W_3 \frac{\partial W_4}{\partial y}]^{t-1}, \end{aligned} \quad (5.2)$$

with the matrices  $A$ ,  $B$ ,  $P$ ,  $Q$ ,  $R$  and  $S$  equal to the Jacobians:

$$\begin{aligned} A &= \frac{\partial F}{\partial U}, \quad B = \frac{\partial G}{\partial U}, \\ P &= \frac{\partial V_1}{\partial U} \frac{\partial V_2}{\partial x} + \frac{\partial V_3}{\partial U} \frac{\partial V_4}{\partial y}, \quad Q = \frac{\partial W_1}{\partial U} \frac{\partial W_2}{\partial x} + \frac{\partial W_3}{\partial U} \frac{\partial W_4}{\partial y}, \\ R &= V_1 \frac{\partial}{\partial x} \left( \frac{\partial V_2}{\partial U} \right) + V_3 \frac{\partial}{\partial y} \left( \frac{\partial V_4}{\partial U} \right) \quad \text{and} \quad S = W_1 \frac{\partial}{\partial x} \left( \frac{\partial W_2}{\partial U} \right) + W_3 \frac{\partial}{\partial y} \left( \frac{\partial W_4}{\partial U} \right), \end{aligned}$$

and  $\Delta U^t$  equal to the correction  $U^t - U^{t-1}$ , to be added to the approximation  $U^{t-1}$  of  $U^*$ , in order to obtain the next approximation  $U^t$ .

The exact form of these expressions can be found in APPENDIX II. The notation that we have used in the expressions of  $R$  and  $S$  is to indicate that the product of one of these matrices with the vector of unknowns  $\Delta U$  should be interpreted as:

$$R\Delta U = V_1 \frac{\partial}{\partial y} \left( \frac{\partial V_2}{\partial U} \Delta U \right) + \dots, \quad \text{etcetera.}$$

One may notice that in this formulation the viscosity  $\mu$  and thermal conductivity  $k$  are treated in a purely explicit way. The destabilizing effect of such a procedure can be remedied very well by an explicit underrelaxation procedure for these coefficients [47].

When the grid that will be used is rectangular, the factorization of (5.2) does not pose any problem. First of all, the equations should be

transformed into the coordinate system  $(x', y')$  that is aligned with the grid and obtained upon a rotation of the original coordinate system  $(x, y)$ . The  $x'$ - and  $y'$ -derivatives can then be separated as in (4.2), after which the derivatives in space are put in discretized form as in (4.4).

For arbitrary grids, a similar transformation could for example be applied locally, although in that case the transformation should also take into account the curvature of the grid, which will in general affect (i.e. deteriorate) the accuracy of the space discretizations. However, as the purpose of such a technique is to put (5.2) into factorized form, it would be sufficient to apply the local transformation only to the left-hand side of (5.2). The accuracy of the numerical approximation  $u^{*h}$  of the steady-state solution  $u^*$  of (5.1) would thereby not be affected. In some sense the approximate factorization scheme for arbitrary grids that we will propose here may be interpreted in this way.

A closer look at the implicit relaxation scheme (4.4) shows that during the first series of implicit sweeps (expression (4.4a)) the unknowns are only coupled in the  $x$ -direction, by means of the fluxes in  $x$ -direction. In order to uncouple the unknowns in the  $y$ -direction, the fluxes in the implicit part have been discretized in an approximate way, by using for their numerical approximation the unknowns on the center  $j$ -line only. A similar procedure has been used for the implicit sweeps in  $y$ -direction (expression (4.4b)).

On an arbitrary grid, the implicit sweeps will not be in  $x$ - and  $y$ -direction anymore, but still in  $i$ - and  $j$ -direction, and so the general approximate factorization scheme becomes:

$$\begin{aligned} & \frac{\text{Vol}_c}{\Delta t} \tilde{\Delta U}_{ij}^t + \text{flux}_{i+1}^c(\tilde{\Delta U}_{i+1,j}^t) - \text{flux}_{i-1}^c(\tilde{\Delta U}_{i-1,j}^t) + \\ & + \frac{\text{Vol}_c}{\text{Vol}_v} [\text{flux}_{i+1/2}^v(\tilde{\Delta U}_{ij}, \tilde{\Delta U}_{i+1,j}^t) - \text{flux}_{i-1/2}^v(\tilde{\Delta U}_{i-1,j}^t, \tilde{\Delta U}_{ij}^t)] = \\ & = \text{RHS}_{ij}^{t-1}, \end{aligned} \quad (5.3a)$$

$$\begin{aligned} & \frac{\text{Vol}_c}{\Delta t} \tilde{\Delta U}_{ij}^t + \text{flux}_{j+1}^c(\tilde{\Delta U}_{ij+1}^t) - \text{flux}_{j-1}^c(\tilde{\Delta U}_{ij-1}^t) + \\ & + \frac{\text{Vol}_c}{\text{Vol}_v} [\text{flux}_{j+1/2}^v(\tilde{\Delta U}_{ij}^t, \tilde{\Delta U}_{ij+1}^t) - \text{flux}_{j-1/2}^v(\tilde{\Delta U}_{ij-1}^t, \tilde{\Delta U}_{ij}^t)] = \\ & = \text{Vol}_c \tilde{\Delta U}_{ij}^t / \Delta t. \end{aligned} \quad (5.3b)$$

In this schematic representation of the general implicit factorization scheme,  $\text{flux}_{i+1}^c$  and  $\text{flux}_{i-1}^c$  denote approximations of the convection fluxes in the  $i$ -direction, while  $\text{flux}_{i+1/2}^v$  and  $\text{flux}_{i-1/2}^v$  denote the approximations of the diffusion fluxes in that direction. Following the conclusions of the optimization analysis of the scheme in CHAPTER IV, they have been obtained upon a finite volume discretization of the left-hand side of (5.2), discarding the fluxes in  $j$ -direction as well as the terms involving the unknown  $\Delta\tilde{U}^t$  (the intermediate correction) at nodal points on the  $j+1$ - and  $j-1$ -grid line. The latter is the generalization of the parameter choice  $\beta = \gamma = 0$ .

A similar procedure is followed for the approximations  $\text{flux}_{j+1}^c$ ,  $\text{flux}_{j-1}^c$ ,  $\text{flux}_{j+1/2}^v$  and  $\text{flux}_{j-1/2}^v$  of the convection and diffusion fluxes in  $j$ -direction. The term  $RHSt^{-1}$  denotes then the finite volume discretization of the right-hand side of (5.2), evaluated by means of the solution that was determined at the previous iteration.

The surfaces of the finite volumes that have been used for the finite volume discretization of the convection terms and the diffusion terms may be different, and have been denoted in (5.3) by  $\text{Vol}_c$  and  $\text{Vol}_v$ , respectively. The above procedure can be used in combination with any finite volume technique, but will be realized here with the discretization methods developed in CHAPTER III.

The whole procedure is illustrated schematically in FIGURE 15. The part of the contour integrals, obtained upon the integration of the space derivatives at the left-hand side of (5.2) over the control volumes, that is evaluated for the implicit coupling of the unknowns in  $i$ -direction, is indicated by the solid lines (cf. the FIGURES 5 and 9). Similarly, the dashed lines in FIGURE 15 indicate those parts of the finite volume contours whose fluxes contribute to the implicit coupling of the unknowns in  $j$ -direction. For their discretization, only the unknowns at the three nodal points on the  $j$ -line, respectively  $i$ -line, are employed. This has the nice advantage that also the Jacobians have to be evaluated at these three points only. However, for the matrices  $P$ ,  $Q$ ,  $R$  and  $S$  this involves still the use of the solution of the previous iteration at all nine nodal points of the computational molecule.

We will proceed with the determination of the value of the relaxation parameter  $\Delta t$ , and will "extrapolate" hereto the results of CHAPTER

IV in a very rough manner. The expressions (4.9) for the optimal value of  $\Delta t$  have been derived for the implicit approximation factorization scheme applied to the single convection-diffusion equation (4.1), with constant and equal convection speeds  $u$  and  $v$ , and diffusion coefficients  $v_x$  and  $v_y$ . The Navier-Stokes equations (5.1) represent a system of convection-diffusion equations with properties that are variable in space, in direction, in time and per equation. So it does not seem to be obvious that the time step optimization of the previous chapter can be applied directly to the implicit relaxation scheme (5.3).

However, it can be shown that for numerical purposes it is reasonable to consider the speed of sound  $a$  as a typical speed of propagation of the worst-damped steady-state perturbations, while in general the dissipative effect of the physical boundary conditions for these perturbations will be rather small [47]. The approximation is especially good in the low subsonic flow regime, i.e. in boundary layers, in separated flow regions, around stagnation points, and in wakes.

As for the viscous part of the equations (5.1), it suffices to mention that, assuming an (almost) constant density, viscosity coefficient and not too large velocity gradients, the system of equations may be approximated by a set of equations for  $u$ ,  $v$  and the temperature  $T$  only. The original diffusion terms reduce for each of these equations to a simple Laplace operator in the unknown, with coefficients  $\mu/\rho$ ,  $\mu/\rho$  and  $\mu/\rho/Pr$  respectively [66,p84,p319].

This means that equation (4.1) models the convective and diffusive properties of the compressible Navier-Stokes equations well enough to be used for the optimization of its relaxation scheme, if  $u$  and  $v$  in this equation are taken equal to  $a$ , and  $v_x$  and  $v_y$  are taken equal to  $\mu/\rho$  (the Prandl-number  $Pr$  is usually about one). So the optimization procedure of the first-order implicit relaxation scheme in CHAPTER IV may indeed be expected to give a reasonable idea of how to choose the optimal values for the parameters of the implicit relaxation scheme (5.3).

The expressions (4.9) may thus be used for the estimation of the optimal value of the time step for (5.3), but then we should first decide about the lengths  $L_i$  and  $L_j$  of the periodicity domain, where  $L_i$  and  $L_j$  denote these lengths in  $i$ - and in  $j$ -direction respectively. On an arbitrary domain, the assumption of a periodic steady-state error will obviously be doubtful, although still reasonable if the domain and the grid are roughly of rectangular shape. This is the main reason why we will

use grids of H-type only. We will use the *i*-index to number the nodal points on the grid lines that run from inlet to outlet, and the *j*-index to number the points on the other set of grid lines, that are oriented in the direction parallel to inlet and outlet.

The least-damped mode, traveling in *i*-direction with a speed of  $u+a$ , will be reflected back completely at the outlet by the imposed static pressure, and will start traveling rearward through the domain in the same *i*-direction with a speed of  $u-a$ , provided that the *i*-direction is normal to the outlet boundary. At the inlet the imposed total temperature will reflect this reflected mode again forward in *i*-direction, if the *i*-direction is also normal to the inlet, whereby the amplitude is attenuated to some extend. Likewise, an imposed velocity profile will reflect the mode completely [47].

For low Mach-number flow ( $|u| \ll a$ ), this situation is completely similar to the behavior of the one-dimensional wave equation ( $\partial\Phi/\partial t + \partial\Psi/\partial x = 0$ , and  $\partial\Psi/\partial t + \partial\Phi/\partial x = 0$ ), with the value of  $\Psi$  and of  $\Phi$  imposed at the left and at the right boundary respectively of the domain of definition of the problem. For this problem, it can be shown easily that the longest wave length to be considered for the analysis of the convergence properties of the scheme equals four times the length of the computational domain (FIGURE 16).

In *j*-direction, supposing that both sides are bounded by a solid wall boundary, the modes propagate essentially with the speed of sound  $a$ , since this direction, due to the fact that it is oriented normal to the solid walls, will be in general approximately normal to the flow direction. At a solid wall boundary, a no-slip condition as well as a zero heat flux will be imposed, while the additional numerical boundary condition that we will apply, often reduces to the well-known boundary layer approximation  $\partial p/\partial n = 0$ , where  $n$  denotes the coordinate direction normal to the wall. So the longest wavelength that a component of the velocity error modes can have is equal to twice the distance between the two walls, since its value is set to zero at both walls. Also the wavelength of an error component of pressure or temperature can be at most of this length, since its derivative is forced to zero at the walls.

Essentially two steady-state error modes propagate in *j*-direction. They can be approximated linearly by  $\Delta p^* + \rho^* a^* \Delta v'^*$  and  $\Delta p^* - \rho^* a^* \Delta v'^*$ , where  $\rho^*$  and  $a^*$  are the density and the speed of sound of the steady-state

solution, and  $\Delta p^*$  and  $\Delta v'^*$  the steady-state error of the pressure and of the velocity component in  $j$ -direction. Because a zero mass flow will be imposed perpendicular to the solid walls, the two modes that propagate with the speed of sound in  $j$ -direction are reflected without any dissipation or amplification at the walls [47]. As FIGURE 17 shows, the component of lowest frequency that is compatible with this system of reflecting modes has a wavelength of twice the length of the domain in  $j$ -direction, which is in agreement with the previous result.

In blade-to-blade calculations, however, due to the periodicity of the flow, each error mode should be periodic over the length of the domain in  $j$ -direction, and can therefore have at most this length as wavelength. Although the periodicity of the flow field is only imposed in a part of the domain, it may be expected that lower frequency modes that may exist in the region in between the solid wall boundaries of the blades will damp out rather fast. Due to the implicit coupling of the unknowns in  $i$ -direction, these modes will be affected by and "feel" the presence of the imposed periodic solution, with which they are incompatible.

At the high-frequency side of the error mode spectrum, we expect that the modes are damped out rather efficiently, due to the strong coupling at the boundaries of the unknowns at even-numbered and at odd-numbered points that is caused by the discretization over half volumes of the equations used in the numerical boundary procedures (cf. [47]). The component of highest frequency still compatible with the discretization of the numerical boundary conditions, is the  $(\pi/2, \pi/2)$ -mode. At the low-frequency side, we have the mode  $(2\pi/N_i, 2\pi/N_j)$ , where  $N_i$  denotes the number of grid intervals in  $i$ -direction times four, and  $N_j$  the number of grid intervals in  $j$ -direction times two or times one, depending on whether the steady-state solution we are looking for is a channel flow or a blade-to-blade flow.

From the results of CHAPTER IV we know that, according to a Fourier mode analysis for model equation (4.1), either one of these two modes will be the worst damped mode of the implicit approximate factorization scheme, independent of the cell Reynolds number  $Re_\epsilon$ . Optimal convergence properties of the relaxation scheme may be expected therefore at the value of the time step  $\Delta t$ , for which both modes are damped out equally well.

A good approximation of this optimal time step  $\Delta t^*$  can be obtained by considering the range of low  $Re_c$  and the range of medium to high  $Re_c$  separately. In the first range the damping of the steady-state error is primarily due to the diffusion terms, while in the latter it is due to the convective terms. The non-dimensionalized approximation of the value of  $\Delta t^*$  becomes then:

$$CFL^* = \min(CFL_1^*, CFL_m^*) , \quad (5.4)$$

where:

$$CFL_1^* \approx \frac{Re_c}{8} \left[ ((\Delta + 1/\Delta)^2 + 12 \frac{\Delta^2 + 1}{\pi^2 (\Delta^2/N_j^2 + 1/N_i^2)})^{1/2} - \Delta - 1/\Delta \right] , \text{ and}$$

$$CFL_m^* \approx \sqrt{3.75} \left[ ((\Delta + 1/\Delta)^2 + 12 \frac{\Delta^2 + 1}{\pi^2 (\Delta^2/N_j^2 + 1/N_i^2)})^{1/2} + \Delta + 1/\Delta \right]^{1/2} .$$

with  $CFL^* = 2a\Delta t^*/\sqrt{Vol_c}$ ,  $Re_c = a\sqrt{Vol_c}/(2\mu/\rho)$ , and grid cell ratio  $\Delta$  equal to the ratio of the grid spacing in  $i$ -direction and  $j$ -direction. The variable  $Vol_c$  denotes the surface of each finite volume that is used for the discretization of the convective terms, and that comprises four grid cells (the large volume indicated with a solid line in FIGURE 5). So  $\sqrt{Vol_c}/2$  corresponds to the average grid size of a computational molecule.

One may notice that these expressions are essentially the same as in (4.9). The difference stems from the fact that instead of frequencies up to  $(\pi, \pi)$  only the modes up to  $(\pi/2, \pi/2)$  have been considered.

## V.2 CALCULATION OF LAMINAR VISCOUS FLOW OVER A BACKWARD FACING STEP

One of the first test cases that we have solved numerically with the developed method was the compressible, low-subsonic flow over a backward facing step for four different combinations of geometry and Reynolds number. They are the same as the four test cases that have been proposed by GAMM for the workshop on numerical simulation of incompressible viscous flows [34], and are defined in FIGURE 18. Also the physical boundary conditions have been indicated in this figure.

The results that will be presented in this section have been obtained with an older version of the Navier-Stokes solver, which differs from the present version of the code primarily because of the explicit

implementation and the simple choice of its numerical boundary procedures at inlet and outlet. It is for this reason that we will not present any convergence history curve for these calculations. The grids that have been used for the calculations are shown in FIGURE 19. The ratio between the largest and the smallest grid size is about 260 for the configuration with the wide inlet channel, and 430 for the other one. The maximum ratio between the largest and smallest side of a grid cell, the *grid cell ratio* is 52 and 84 respectively, and occurs for a cell at the outlet. A more detailed picture of one of the grids has been given in the FIGURES 7, 8 and 10.

We assumed a flow of perfect gas, and took the ratio of specific heats equal to 1.4 and a Prandtl number of 0.72. The Reynolds number  $Re$  was based on the maximum velocity at the inlet  $UMAX$  and the height of the step. The kinematic viscosity  $\nu$  ( $=\mu/\rho$ ) was kept constant over the whole domain. The stream function  $\psi$  was calculated afterward by integrating the mass flow over the vertically oriented grid lines, starting from  $\psi = 0$  at the wall with the step.

For the presentation of the results the non-dimensionalization of the workshop had to be adapted slightly, to exclude the (very small) compressibility effects. We used:

$$\text{for the shear stress values: } \bar{\tau} = \frac{1}{Re \ RHOUMAX} \times \frac{\partial \rho u}{\partial y},$$

$$\text{for the stream function values: } \bar{\psi} = \frac{\psi}{RHOUMAX},$$

$$\text{and for the pressure values: } \bar{p} = Re \frac{p - p_{\text{at step corner}}}{RHOUMAX \times UMAX},$$

with  $RHOUMAX$  the maximum of  $\rho u$  at the inlet.

Some global features of the four cases we calculated are given in TABLE 1. The values of the inlet Mach number, inlet static pressure and maximum temperature variation show that the flow was indeed low subsonic. The mass conservation error, calculated by verifying the mass flow through every cross-section, was found to be of the order of the round-off error of the computer, showing clearly that OLOCS is fully conservative. Further results are presented in the FIGURES 20 to 27. In general our results agreed well with the measurements and the numerical solutions obtained by other investigators [34].

A severe test is the comparison of the calculated shear stresses with the measured ones (FIGURE 24). We see that the comparison for Case 1 (wide inlet channel,  $Re = 50$ ) is rather good, except for the negative peak, where the measurements show three values of about equal magnitude, which may be due to measurement errors. For the other two cases the comparison is reasonable, but not excellent. However, the measured shear stresses do not tend asymptotically to the correct theoretical downstream values, which may indicate that the Reynolds numbers of the experimental flow were actually smaller than indicated.

Another explanation, which seems to be more plausible, can be found in the way the shear stresses were calculated. The experimental values were obtained upon taking the least square straight line fit of the three or four velocities that were measured at points located between 0.3 and 0.8 mm from the wall. Such a procedure inevitably underestimates the real velocity gradient at the wall. On the contrary, the numerical values were obtained by a second-order accurate extrapolation to the wall, corresponding to a parabolic curve fit. The numerical shear-stress prediction changes considerably (and gets much closer to the measured values) if an extrapolation of only first-order accuracy is used, corresponding to a straight line fit over about 1 mm (FIGURE 25). In that case, also the numerical values do not tend to the correct downstream value anymore. So the conclusion reads that the calculated shear stresses agree very well with the experiments.

Another interesting detail to look at, is the flow around the separation corner. A close-up of the FIGURES 22 and 23, given in the FIGURES 26 and 27, shows that the flow separates after the step corner with a slightly downgoing streamline. This is not in agreement with the measurements [34], but it does correspond to the local analytical solution [67].

The code was also able to calculate the very steep pressure gradients that were predicted by the theory. The pressure, as function of  $x$  on the line  $y = 0$ , shows the correct behavior [67], but does not tend to infinity at the step corner because of the strong local compressibility effects. Only a few results that were presented at the GAMM-workshop [34] show a similar pressure behavior, although the steepest gradients were found in the present work. Yet the solution of the incompressible Navier-Stokes equations is known to be singular at the

step corner. Apparently, in all calculations this discontinuity was completely suppressed.

### V.3 CALCULATION OF LAMINAR VISCOSUS FLOW THROUGH A BLADE-TO-BLADE PASSAGE

The latest version of the Navier-Stokes solver has been used for the determination of the numerical solution of the steady-state compressible Navier-Stokes equations describing a viscous subsonic flow through a cascade of NACA0012-profiles. This test case was chosen, in order to make a comparison possible with other numerical results for the same laminar flow problem, at a Reynolds number of 1000 [17, 27]. However, our calculations show that already at a Reynolds number of 500, the wake behind the profiles is unsteady, so we have calculated this flow primarily at a Reynolds number of 200.

The geometry of the computational domain, as well as the grid, is shown in FIGURE 28. It is formed by a cascade of NACA0012 airfoils, with solidity 1 (i.e. pitch length equal to chord length), and a stagger angle of  $30^\circ$ . The inlet boundary is at about 1.1 chord lengths from the leading edges, while the outlet boundary is at a distance of about 3.1 chord lengths away from the trailing edges.

A total of 39 grid lines have been used in streamwise ( $i$ -)direction, whereby the upper side of the profile coincides with the first line, and the lower side with the 39th grid line. In the blade-to-blade ( $j$ -)direction, 78 grid lines were employed, with the two grid points at the leading edge on the line  $i=18$ , and the two points at the trailing edge on the line  $i=53$ .

Some close-ups of the grid show the high concentration of grid points at leading edge and trailing edge (FIGURE 29). The local size of the grid cells is as small as 0.1 % of the chord length, which is five times smaller than in the C-grid of  $133 \times 33$  points, used by Schäfer et al. [17] for the same problem. Apparently, our Navier-Stokes solution method permits a much higher stretching of the grid than theirs.

The longest size of the grid is at the outlet boundary, and measures 75 % of the chord length. So the maximum grid size ratio of this grid is about 750, while the maximum grid cell ratio is about 90, and is reached for a grid cell in the boundary layer at the upper side of the profile.

Reservoir conditions were specified at the inlet, and so the non-dimensionalized total temperature and total pressure were set equal to 1. The third physical condition that was specified at the inlet is the flow angle  $\alpha$ , that was set equal to  $30^\circ$  (zero angle of attack), as well as to  $0^\circ$  (angle of attack equal to  $-30^\circ$ ). At the outlet we fixed the non-dimensionalized static pressure  $p_s$ , both to 0.95 (low subsonic flow) as well as to 0.80 (high subsonic flow). The Reynolds number  $Re$  was based upon the maximum inlet velocity, and on the length of the pitch. As said before, it was set to 200 for all cases, although we will show one (non-converged) result for  $Re=500$ , proving the unsteady character of the flow for this value. The Prandtl-number for all calculations was taken equal to 0.72, except when mentioned.

The steady-state solution for  $\alpha=30^\circ$ ,  $p_s=0.95$ , and  $Re=200$  is shown in FIGURE 30, in the form of streamline plots and static pressure contours. The inlet Mach number of this calculation appeared to be 0.2144, while the total mass flow error was only 0.0068 %.! The plots show the rather thick wake, the very steep gradients at the stagnation point and at the trailing edge, and some small wiggles in the static pressure solution at the trailing edge.

In FIGURE 31, the steady-state solution for  $\alpha=0^\circ$  shows the effect of changing the angle of incidence. We see that the stagnation point has shifted considerably, and that the pressure gradients around the leading edge have increased significantly, although the inlet Mach number decreased slightly to 0.1834. Another noticeable feature is the much thicker wake. The total mass flow error for this case was even smaller than in the previous calculation, and equal to 0.0031 %!

Like in the previous case, again some small wiggles are present in the pressure field at the trailing edge. Although they cannot be seen directly in FIGURE 31b, they influence the pressure solution elsewhere, which explains the very small wiggles in the pressure contours downstream of the trailing edge.

A high-subsonic flow calculation is presented in FIGURE 32, where the results are shown for a steady-state flow, determined by the parameter values  $Re=200$ ,  $\alpha=30^\circ$ , and  $p_s=0.80$ . The Mach number at the inlet was equal to 0.4117. Otherwise the results are similar to the previous ones, although the static pressure wiggles at the trailing edge became

stronger. The total mass flow error for this case was also higher as before (0.027 %).

The convergence history of the two steady-state calculations with  $\alpha=30^\circ$  can be found in FIGURE 33. For the case with  $p_s=0.95$ , we started from a uniform initial pressure and density field, based on the static pressure at the outlet and the total conditions at the inlet. From the same inlet and outlet conditions an estimation of the mass flow follows, that was used to define an initial velocity field that was uniform and had the imposed inlet flow angle in the inlet part of the domain, and that was aligned with the grid in the rest of the domain. At solid walls, the velocity was set equal to zero.

This procedure introduces locally rather abrupt velocity changes in the domain, which resulted in unacceptably high values of  $\Delta U^t$  during the first few iterations. So a very simple smoothing algorithm was developed, in order to smooth out these large and unphysical gradients in the initial solution.

However, it appeared that this initial "flow field" was still too unphysical for the relaxation scheme to work properly, when the outlet static pressure was taken equal to 0.80. So for this case we started from a still flow field, with total conditions as at the inlet. The disadvantage of such an initial condition is that it lengthens the searching phase of the scheme considerably (cf. FIGURE 33), although it is clearly a more robust initial flow field to start from.

Testing the convergence properties of the scheme, it appeared soon that optimal convergence was obtained, when instead of the speed of sound  $a$ , the maximum speed of propagation  $u+a$  was used in the formulas (5.4) of the optimal time step. Apparently, it is the maximum propagation speed that is to be considered as a typical propagation speed of the steady-state errors.

The prediction of the optimal  $\Delta t$  has been verified for the calculation of a low-subsonic flow only, in order to exclude as much as possible the dissipative effect of the boundary conditions. The results in FIGURE 33 show that the optimal value of  $\Delta t$  appears to be really the optimum, since both for lower and for higher  $\Delta t$ , the convergence speed of the scheme decreases.

We were very surprised by this result. It shows the power of a carefully interpreted Fourier mode analysis, since it appears to be possible by this simple means, to predict within about 10 % (which is

much better than we were expecting) an optimal  $\Delta t$  for this very complicated relaxation problem.

We have not verified if the prediction of the optimal  $\Delta t$  is also good for the case with  $p_s = 0.80$ . Normally, one would expect a less satisfactory prediction, due to the increased dissipative effect of the boundary conditions, that was not taken into account in the Fourier-mode analysis. As FIGURE 33 shows, this effect is clearly present, since the relaxation scheme converged faster to the high-subsonic steady-state solution, than to the low-subsonic solution.

The speed of convergence  $\epsilon_{\max}$  was found to be about 0.025. This is in good agreement with the order of magnitude of the values that were determined analytically, taking into account the fact that the lengths of the periodicity domain of the present problem, expressed in number of grid cells, are 308 in  $i$ -direction (4 times the number of grid cells, that equals 77), and 39 in  $j$ -direction.

A comparison of the present result with the speed of convergence of the implicit approximate factorization scheme of Schäfer et al. [17] shows that for the same geometry and inlet Mach number, both schemes reduce the steady-state error by a factor of  $10^{-6}$  in about the same number of iterations: 500 time steps for our scheme, versus 700 steps for their method. However, they measured the steady-state error in the less severe  $L_2$ -norm (our convergence error is  $\Delta U^t$ , measured in the maximum norm), but showed also the convergence history of a steady-state Euler calculation only, on a grid of 51 times 31 points. No convergence history graphs were presented for the steady-state Navier-Stokes calculations on the grid of 133 times 33 points.

The last result that we would like to present, is the non-converged solution of the cascade flow at a Reynolds number of 500, an inlet flow angle of  $30^\circ$ , and an outlet static pressure of 0.95. With these parameters, the relaxation scheme converged to an error level of  $10^{-3}$  in about 150 iterations, after which the error remained at this level for at least 250 iterations more (FIGURE 34). The solution after this total of 400 iterations looks rather smooth, but the streamline plot reveals that the wake of the flow oscillates, indicating that a steady-state solution for this case does not exist.

This can be explained very easily by considering the flow behind the trailing edge as a wake flow behind a blunt body, where the "body"

consists of the thickness of the trailing edge, augmented with the displacement thickness of the two boundary layers. The total thickness of this "body" is approximately 15.6 % of the pitch, and so the Reynolds number based on the size of this body is about 78. It is well known that for such a high Reynolds number, the wake behind a circular cylinder is already completely unsteady. Although the comparison is of course not fully justified, it gives a clear indication that for  $Re=500$  the wake flow behind the profile may already be expected to become unsteady.

In view of our above-mentioned experiences, it is surprising that other workers are capable of calculating for the same geometry a steady, viscous flow at  $Re=1000$  [17, 27]. A closer look at their numerical schemes reveals however, that in both methods artificial dissipation has been introduced, by using a partially corrected first-order upwind scheme [27], or by adding fourth-order dissipation terms [17]. We conclude therefore that their numerical steady-state solutions have been obtained by means of an artificial averaging procedure (modelization) of the unsteady physical Navier-Stokes equations.

## CHAPTER VI

### *DISCUSSION OF RESULTS*

It has been shown that the relative magnitude of the truncation error rather than the order of accuracy of a discretization scheme determines how well a solution procedure will accurately predict the solution of a system of differential equations. Actually, the order of accuracy of a scheme is *only* well defined on *uniform grids*, and for *sufficiently smooth solutions*. In practice this situation is not very likely to occur often.

For the numerical simulation of the Navier-Stokes equations at high Reynolds number, this means in the first place that only a very small amount of numerical dissipation can be tolerated in regions where the viscous effects are relatively important, which can only be realized by using *centered discretization schemes*. Such schemes introduce in principle only error terms consisting of odd derivatives, and are therefore very sensitive to an inaccurate numerical modeling of the original differential problem (cf. the title of [37]: "Don't Suppress the Wiggles - They're Telling You Something!")

It has been proven that for a conservative discretization method, the introduction of numerical diffusion may only be avoided completely by using high-order schemes, based on quadratic (or even higher-order) interpolations of the unknowns. In this work, we have followed a different approach, by trying to minimize rather than eliminate these low-order truncation errors. This has led to the development of an easy-to-program, low-order, conservative discretization scheme that is *almost free of numerical diffusion* and fairly insensitive to the grid design.

However, some restrictions to the grid are still present, and the use of a so-called structured grid, built up with only quadrilateral grid cells, is imperative.

The higher-order terms of the truncation error should also be negligible. They indicate therefore in some sense the size of the grid that should be used. In consequence the computational grid should be sufficiently fine in regions of steep gradients. In case a sufficiently high concentration of grid points cannot be afforded, and *only in that case*, the truncation error may be modified artificially by adding dissipative terms of even derivatives to the differential equations, in order to *model* rather than *calculate* the steep gradients.

The developed discretization scheme combines in an optimal way high accuracy on a wide variety of grid structures with ease of programming, leading to efficient Navier-Stokes codes. This has been demonstrated by the calculation of the flow over a backward facing step and of the flow through a cascade of NACA0012 profiles. In both cases, excellent results were obtained using highly distorted grids.

As for the applied first-order implicit relaxation scheme, an analytical prediction of the optimal time step for the AF has been derived, with which (nearly) optimal convergence properties were obtained. Such a prediction can only be obtained by means of a correctly performed Fourier mode analysis, where the type of boundary conditions determine which range of error modes are to be considered. Otherwise, the effect of boundary conditions is not taken into account in the prediction. This makes that the prediction may fail completely when the implementation of the numerical boundary procedures is too much different from the scheme used inside the domain, as this may change completely the convergence properties of the scheme as a whole.

Notice that  $\Delta t$  can also be optimized for the damping of only a part of the error modes, e.g. for the scheme to be primarily dissipative for the error modes of high frequency. This would make the AF a very useful smoother for a multigrid technique.

The results of the calculations show that depending on the geometry, stationary viscous flows apparently only exist at rather low Reynolds numbers. Very quickly, the flow develops a self-excited oscillation when

the viscous stresses are reduced. For the calculation of flows at higher Reynolds numbers (say, from about 300 for the present flow fields), a true unsteady flow should therefore be simulated, or the unsteadiness of the flow should be modeled, much like an (unsteady!) turbulent flow is modeled by means of turbulent viscosity.

## REFERENCES

- [1] SCHLICHTING H. (1979), *Boundary-Layer Theory*, McGraw-Hill, New York.
- [2] TOPAKOGLU H.C., EBADIAN M.A. (1985), "On the Steady Laminar Flow of an Incompressible Viscous Fluid in a Curved Pipe of Elliptical Cross-Section", *J.Fluid Mech.* 158, 329-340.
- [3] VAN DYKE M. (1964), *Perturbation Methods in Fluid Mechanics*, Academic Press, New York.
- [4] MOFFATT H.K. (1964), "Viscous and Resistive Eddies Near a Sharp Corner", *J.Fluid Mech.* 18, 1-18.
- [5] SMITH F.T. (1979), "Laminar Flow of an Incompressible Fluid Past a Bluff Body: the Separation, Reattachment, Eddy Properties and Drag", *J.Fluid Mech.* 92, 171-205.
- [6] LADYZHENSKAYA O.A. (1963), *The Mathematical Theory of Viscous Incompressible Flow*, Gordon and Breach, New York.
- [7] HOLLANDERS H., LERAT A., PEYRET R. (1983), "3-D Calculation of Transonic Viscous Flows by an Implicit Method", *AIAA Paper 83-1953*.
- [8] GRESHO P.M., CHAN S.T., LEE R.L., UPSON C.D. (1984), "A Modified Finite Element Method for Solving the Time-Dependent, Incompressible Navier-Stokes Equations. Part 2: Applications", *Int.J.Numer. Meth.Fluids* 4, 619-640.
- [9] HUNG C.M., KORDULLA W. (1984), "A Time-Split Finite-Volume Algorithm for Three-Dimensional Flow Field Simulation", *AIAA-J.* 22, 1564-1572.
- [10] DAVIS R.L., NI R.-H., BOWLEY W.W. (1984), "Prediction of Compressible, Laminar Viscous Flows Using a Time-Marching Control Volume and Multiple-Grid Technique", *AIAA-J.* 22, 1573-1581.
- [11] NI R.-H. (1986), "Multigrid Convergence Acceleration Techniques for Explicit Euler Solvers and Applications to Navier-Stokes Calculations", in: *Numerical Techniques for Viscous Flow Calculations in Turbomachinery Bladings*, Von Karman Institute LS1986-02.
- [12] BRILEY W.R., MCDONALD H. (1977), "Solution of the Multidimensional Compressible Navier-Stokes Equations by a Generalized Implicit Method", *J.Comput.Phys.* 24, 372-397.

- [13] BEAM R.M., WARMING R.F. (1982), "Implicit Numerical Methods for the Compressible Navier-Stokes and Euler Equations", in: *Computational Fluid Dynamics*, Von Karman Institute LS1982-04.
- [14] DEGREZ G. (1984), *Etude Théorique et Expérimentale d'Interactions Bi- et Tri-Dimensionnelles entre Ondes de Choc et Couches Limites Laminaires*, Doctoral Thesis, Université Libre de Bruxelles, Belgium.
- [15] DAWES W.N. (1985), "Computation of Off-Design Flows in a Transonic Compressor Rotor", ASME Paper 85-GT-1.
- [16] BRILEY W.R., BUGGELN R.C., MCDONALD H. (1985), "Solution of the Three-Dimensional Navier-Stokes Equations for a Steady Laminar Horseshoe Vortex Flow", AIAA Paper 85-1520.
- [17] SCHAFER O., FRÜHAUF H.-H., BAUER B., GUGGOLZ M. (1985), "Application of a Navier-Stokes Analysis to Flows Through Plane Cascades", ASME Paper 85-GT-56.
- [18] MÜLLER B., RUES D. (1985), "Implicit Finite-Difference Simulation of Separated Hypersonic Flow over an Indented Nosettip", In: *Notes on Numerical Fluid Mechanics*, Vol 13, Rues D., Kordulla W. (eds.), Vieweg, Braunschweig.
- [19] PULLIAM T.H. (1986), "Efficient Solution Methods for the Navier-Stokes Equations", in: *Numerical Techniques for Viscous Flow Calculations in Turbomachinery Bladings*, Von Karman Institute LS1986-02.
- [20] DAWES W.N. (1986), "Application of Full Navier-Stokes Solvers to Turbomachinery Flow Problems", in: *Numerical Techniques for Viscous Flow Calculations in Turbomachinery Bladings*, Von Karman Institute LS1986-02.
- [21] HAASE W., WAGNER B., JAMESON A. (1984), Development of a Navier-Stokes Method Based on a Finite Volume Technique for the Unsteady Euler Equations, in: *Notes on Numerical Fluid Dynamics VII*, Pandolfi M., Piva R. (eds), Vieweg, Braunschweig.
- [22] KORDULLA W. (1987), "Integration of the Navier-Stokes Equations in Finite-Volume Formulation", in: *Computational Fluid Dynamics*, Von Karman Institute LS1987-04.
- [23] HUGHES T.J.R., LIU W.K., BROOKS A. (1979), "Finite Element Analysis of Incompressible Viscous Flows by the Penalty Function Formulation", *J.Comput.Phys.* 30, 1-60.
- [24] REKLIS R.P., THOMAS P.D. (1982), "Shock-Capturing Algorithm for the Navier-Stokes Equations", *AIAA-J.* 20, 1212-1218.

- [25] LOMBARD C.K., BARDINA J., VENKATAPATHY E., OLIGER J. (1983), "Multi-Dimensional Formulation of CSCM - an Upwind Flux Difference Eigen-vector Split Method for the Compressible Navier-Stokes Equations", AIAA Paper 83-1895.
- [26] LECOINTE Y., PIQUET J. (1984), "On the Use of Several Compact Methods for the Study of Unsteady Incompressible Viscous Flow Round a Circular Cylinder", Computers & Fluids 12, 255-280.
- [27] ROSENFIELD M., WOLFSHTEIN M. (1984), "Numerical Calculation of a Laminar Two-Dimensional Straight Cascade Flow", Computers & Fluids 12, 293-310.
- [28] HAH C. (1984), "A Navier-Stokes Analysis of Three-Dimensional Turbulent Flows Inside Turbine Blade Rows at Design and Off-Design Conditions", ASME J.Eng.Power 106, 421-429.
- [29] COAKLEY T.J. (1984), "Implicit Upwind Methods for the Compressible Navier-Stokes Equations", AIAA-J. 23, 374-380.
- [30] MACCORMACK R.W. (1985), "Current Status of Numerical Solutions of the Navier-Stokes Equations", AIAA Paper 85-0032.
- [31] MOORE J.G., MOORE J. (1985), "Performance Evaluation of Linear Turbine Cascades Using Three-Dimensional Viscous Flow Calculations", ASME Paper 85-GT-65.
- [32] CHAKRAVARTHY S.R., SZEMA K.-Y., GOLDBERG U.C., GORSKI J.J., OSHER S. (1985), "Application of a New Class of High Accuracy TVD Schemes to the Navier-Stokes Equations", AIAA Paper 85-0165.
- [33] CULLEN M.J.P. (1983), "Experiments with Some Low-Order Finite Element Schemes for the Navier-Stokes Equations", J.Comput.Phys. 51, 291-312.
- [34] GAMM-workshop on Analysis of Laminar Flow Over a Backward Facing Step (1984), eds. K. Morgan, J. Periaux, F. Thomasset, Vieweg, Braunschweig.
- [35] PEYRET R., VIVIAND H. (1975), "Computation of Viscous Compressible Flows Based on the Navier-Stokes Equations", AGARDograph AG-212.
- [36] THOMPSON J.F. (1984), "Grid Generation Techniques in Computational Fluid Dynamics", AIAA-J. 22, 1505-1523.
- [37] GRESHO P.M., LEE R.L. (1981), "Don't Suppress The Wiggles - They're Telling You Something!", Computers & Fluids 9, 223-253.
- [38] MASTIN C.W. (1982), "Error Induced by Coordinate Systems", in: *Numerical Grid Generation*, Thompson J.F. ed., North-Holland.

- [39] THOMPSON J.F., WARSI Z.U.A., MASTIN C.W. (1982), "Boundary-Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations - A Review", *J.Comput.Phys.* 47, 1-108.
- [40] GRESHO P.M., CHAN S.T., LEE R.L., UPSON C.D. (1984), "A Modified Finite Element Method for Solving the Time-Dependent, Incompressible Navier-Stokes Equations. Part 1: Theory", *Int.J.Numer.Meth. Fluids* 4, 557-598.
- [41] PULLIAM T.H., STEGER J.L. (1985), "Recent Improvements in Efficiency, Accuracy, and Convergence for Implicit Approximate Factorization Algorithms", AIAA Paper 85-0360.
- [42] BEAM R.M., WARMING R.F. (1978), "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations", *AIAA-J.* 16, 393-402.
- [43] WARMING R.F., BEAM R.M. (1978), "On the Construction and Application of Implicit Factored Schemes for Conservation Laws", *SIAM-AMS Proc.* 11, 85-129.
- [44] PULLIAM T.H., STEGER J.L. (1980), "Implicit Finite-Difference Simulations of Three-Dimensional Compressible Flow", *AIAA-J.* 18, 159-167.
- [45] KLOKER M. (1986), "A Fully Implicit Scheme for Unsteady Flow Calculations Solved by the Approximate Factorization Technique", VKI Project Report 1986-16.
- [46] BORSBOOM M. (1986), "Numerical Modeling of Navier-Stokes Equations", in: *Numerical Techniques for Viscous Flow Calculations in Turbomachinery Bladings*, Von Karman Institute LS1986-02.
- [47] BORSBOOM M. (1987), *A Numerical Solution Method for the Steady-State Compressible Navier-Stokes Equations with Application to Channel and Blade-to-Blade Flow*, Doctoral Thesis, Université de Liège, Belgium.
- [48] SHYY W. (1985), "A Study of Finite Difference Approximations to Steady-State, Convection-Dominated Flow Problems", *J.Comput.Phys.* 57, 415-438.
- [49] PUDDU P.P. (1987), "Solution of Navier-Stokes Equations for Compressible Flow at High Reynolds Numbers", VKI Project Report 1987-07.
- [50] ROE P.L. (1983), "An Introduction to Numerical Methods Suitable for the Euler Equations", in: *Introduction to Computational Fluid Dynamics*, Von Karman Institute LS1983-01.
- [51] LAX P.D. (1954), "Weak Solutions of Nonlinear Hyperbolic Equations and Their Numerical Computation", *Comm.Pure.Appl.Math.* 7, 159-193.

- [52] CULLEN M.J.P. (1982), "The Use of Quadratic Finite Element Methods and Irregular Grids in the Solution of Hyperbolic Problems", *J.Comput.Phys.* 45, 221-245.
- [53] CULLEN M.J.P. (1983), "Analysis of Some Low-Order Finite Element Schemes for the Navier-Stokes Equations", *J.Comput.Phys.* 51, 273-290.
- [54] BLOTTNER F.G. (1980), "Variable Grid Scheme for Discontinuous Grid Spacing and Derivatives", *Computers and Fluids* 8, 421-434.
- [55] SOLIMAN M., BAKER A. (1981), "Accuracy and Convergence of a Finite Element Algorithm for Laminar Boundary Layer Flow", *Computer and Fluids* 9, 43-62.
- [56] BORSBOOM M. (1986), "About the (In)Accuracy of Low-Order Conservative Discretization Schemes", in: *Notes on Numerical Fluid Dynamics*, Volume 13, proceedings of the Sixth GAMM-Conference on Numerical Methods in Fluid Mechanics, Rues D., Kordulla W., eds., Vieweg, Braunschweig.
- [57] DEKKER K. (1980), "Semi-Discretization Methods for Partial Differential Equations on Non-Rectangular Grids", *Int.J.Numer.Meth.Engin.* 15, 405-419.
- [58] STRANG G., FIX G. (1973), *An Analysis of the Finite Element Method*, Prentice-Hall, Englewood Cliffs, New Jersey.
- [59] ROSSOW C. (1987), "Comparison of Cell Centered and Cell Vertex Finite Volume Schemes", Paper presented at the 7th GAMM Conference on Numerical Methods in Fluid Mechanics, Louvain-La-Neuve, Belgium.
- [60] DONEA J. (1984), "Recent Advances in Computational Methods for Steady and Transient Transport Problems", *Nucl.Engin.and Design* 80, 141-162.
- [61] MORGAN K., PERAIRE J. (1987), "Finite Element Methods for Compressible Flows", in: *Computational Fluid Dynamics*, Von Karman Institute LS1987-04.
- [62] JESPERSEN D.C. (1984), "Recent Developments In Multigrid Methods For The Steady Euler Equations", in: *Computational Fluid Dynamics*, Von Karman Institute LS1984-04.
- [63] DAHLQUIST G., BJØRCK A., ANDERSON N. (1974), *Numerical Methods*, Prentice-Hall, Englewood Cliffs, New Jersey.

- [64] VAN LEER B., MULDER W.A. (1985), "Relaxation Methods for Hyperbolic Conservation Laws", in *Numerical Methods for the Euler Equations of Fluid Dynamics*, Angrand F., Dervieux A., Desideri J.A., Glowinsky R., eds., SIAM, Philadelphia, 312-333.
- [65] ABARBANEL S.S., DWOYER D.L., GOTTLIEB D. (1986), "Improving the Convergence Rate to Steady State of Parabolic ADI Methods", *J.Comput.Phys.* 67, 1-18.
- [66] BIRD R.B., STEWARD W.E., LIGHTFOOD E.N. (1960), *Transport Phenomena*, Wiley, New York.
- [67] LADEVÈZE J., PEYRET R. (1974), "Calcul Numérique d'une Solution Avec Singularité des Equations de Navier-Stokes: Ecoulement Dans un Canal Avec Variation Brusque de Section", *Journal de Mécanique* 13, 367-396.

## APPENDIX I

For the numerical solution method of this thesis, the Navier-Stokes equations have been written as:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{\partial}{\partial x} [V_1 \frac{\partial V_2}{\partial x} + V_3 \frac{\partial V_4}{\partial y}] + \frac{\partial}{\partial y} [W_1 \frac{\partial W_2}{\partial x} + W_3 \frac{\partial W_4}{\partial y}],$$

with:

$$U = \begin{vmatrix} \rho \\ \rho u \\ \rho v \\ e \end{vmatrix}, \quad F = \begin{vmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (\epsilon + p)u \end{vmatrix}, \quad G = \begin{vmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (\epsilon + p)v \end{vmatrix}, \quad V_2 = V_4 = W_2 = W_4 = \begin{vmatrix} 0 \\ u \\ v \\ T \end{vmatrix},$$

and:

$$V_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{4}{3}\mu & 0 & 0 \\ 0 & 0 & \mu & 0 \\ 0 & \frac{4}{3}\mu u & \mu v & k \end{bmatrix}, \quad V_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3}\mu & 0 \\ 0 & \mu & 0 & 0 \\ 0 & \mu v & -\frac{2}{3}\mu u & 0 \end{bmatrix},$$

$$W_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 \\ 0 & -\frac{2}{3}\mu & 0 & 0 \\ 0 & -\frac{2}{3}\mu v & \mu u & 0 \end{bmatrix}, \quad W_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 \\ 0 & 0 & \frac{4}{3}\mu & 0 \\ 0 & \mu u & \frac{4}{3}\mu v & k \end{bmatrix},$$

where  $\rho$  denotes the density;  $u, v$  the velocity components;  $e$  the total energy per unit volume;  $p$  the pressure;  $T$  the temperature;  $\mu$  the viscosity; and  $k$  the thermal conductivity.

The non-dimensionalized equation of state reads:

$$\rho = \rho T .$$

Furthermore we have the relations:

$$p = (\gamma - 1) (e - \rho \frac{u^2 + v^2}{2}) , \quad a^2 = \gamma T ,$$

$$\text{and } Pr = \frac{\mu}{k} \frac{\gamma}{\gamma - 1} ,$$

with ratio of specific heat  $\gamma$ , speed of sound  $a$ , and Prandtl number  $Pr$ .

## APPENDIX II

The Jacobians of the implicit linearized discretization in time of the Navier-Stokes equations:

$$\begin{aligned}\frac{\Delta U^t}{\Delta t} + \frac{\partial}{\partial x}[(A-P-R)^{t-1} \Delta U^t] + \frac{\partial}{\partial y}[(B-Q-S)^{t-1} \Delta U^t] = \\ = \frac{\partial}{\partial x}[-F+V_1 \frac{\partial V_2}{\partial x}+V_3 \frac{\partial V_4}{\partial y}]^{t-1} + \frac{\partial}{\partial y}[-G+W_1 \frac{\partial W_2}{\partial x}+W_3 \frac{\partial W_4}{\partial y}]^{t-1}\end{aligned}$$

are:

$$A = \frac{\partial F}{\partial U} =$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\gamma-3}{2}u^2 + \frac{\gamma-1}{2}v^2 & -(\gamma-3)u & -(\gamma-1)v & \gamma-1 \\ -uv & v & u & 0 \\ -\gamma \frac{u^e}{\rho} + (\gamma-1)u(u^2+v^2) & \gamma \frac{e}{\rho} - \frac{\gamma-1}{2}(3u^2+v^2) & -(\gamma-1)uv & \gamma u \end{bmatrix},$$

$$B = \frac{\partial G}{\partial U} =$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ \frac{\gamma-3}{2}v^2 + \frac{\gamma-1}{2}u^2 & -(\gamma-1)u & -(\gamma-3)v & \gamma-1 \\ -\gamma \frac{v^e}{\rho} + (\gamma-1)v(u^2+v^2) & -(\gamma-1)uv & \gamma \frac{e}{\rho} - \frac{\gamma-1}{2}(3v^2+u^2) & \gamma v \end{bmatrix},$$

$$P = \frac{\partial v_1}{\partial u} \frac{\partial v_2}{\partial x} + \frac{\partial v_3}{\partial u} \frac{\partial v_4}{\partial y} =$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -u \frac{\mu}{\rho} \left( \frac{4}{3} u_x - \frac{2}{3} v_y \right) - v \frac{\mu}{\rho} (u_y + v_x) & \frac{\mu}{\rho} \left( \frac{4}{3} u_x - \frac{2}{3} v_y \right) & \frac{\mu}{\rho} (u_y + v_x) & 0 \end{bmatrix}$$

$$Q = \frac{\partial w_1}{\partial u} \frac{\partial w_2}{\partial x} + \frac{\partial w_3}{\partial u} \frac{\partial w_4}{\partial y} =$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -u \frac{\mu}{\rho} (u_y + v_x) - v \frac{\mu}{\rho} \left( -\frac{2}{3} u_x + \frac{4}{3} v_y \right) & \frac{\mu}{\rho} (u_y + v_x) & \frac{\mu}{\rho} \left( -\frac{2}{3} u_x + \frac{4}{3} v_y \right) & 0 \end{bmatrix}$$

$$R = v_1 \frac{\partial}{\partial x} \left( \frac{\partial v_2}{\partial u} \right) + v_3 \frac{\partial}{\partial y} \left( \frac{\partial v_4}{\partial u} \right) =$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\frac{4}{3} \mu \frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) + \frac{2}{3} \mu \frac{\partial}{\partial y} \left( \frac{v}{\rho} \right) & \frac{4}{3} \mu \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & -\frac{2}{3} \mu \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & 0 \\ -\mu \frac{\partial}{\partial y} \left( \frac{u}{\rho} \right) - \mu \frac{\partial}{\partial x} \left( \frac{v}{\rho} \right) & \mu \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & \mu \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & 0 \\ -\frac{4}{3} \mu u \frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) + \frac{2}{3} \mu u \frac{\partial}{\partial y} \left( \frac{v}{\rho} \right) & \frac{4}{3} \mu u \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & -\frac{2}{3} \mu u \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & \\ -\mu v \frac{\partial}{\partial y} \left( \frac{u}{\rho} \right) - \mu v \frac{\partial}{\partial x} \left( \frac{v}{\rho} \right) & + \mu v \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & + \mu v \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & k(\gamma-1) \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) \\ -k(\gamma-1) \frac{\partial}{\partial x} \left[ \left( \frac{u}{\rho^2} - \frac{u^2 + v^2}{\rho} \right) \right] & -k(\gamma-1) \frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) & -k(\gamma-1) \frac{\partial}{\partial x} \left( \frac{v}{\rho} \right) & \end{bmatrix}$$

$$S = w_1 \frac{\partial}{\partial x} \left( \frac{\partial w_2}{\partial U} \right) + w_3 \frac{\partial}{\partial y} \left( \frac{\partial w_4}{\partial U} \right) =$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\mu \frac{\partial}{\partial y} \left( \frac{u}{\rho} \right) - \mu \frac{\partial}{\partial x} \left( \frac{v}{\rho} \right) & \mu \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & \mu \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & 0 \\ \frac{2}{3} \mu \frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) - \frac{4}{3} \mu \frac{\partial}{\partial y} \left( \frac{v}{\rho} \right) & -\frac{2}{3} \mu \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) & \frac{4}{3} \mu \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) & 0 \\ -\mu u \frac{\partial}{\partial y} \left( \frac{u}{\rho} \right) - \mu u \frac{\partial}{\partial x} \left( \frac{v}{\rho} \right) + & \mu u \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) + & \mu u \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) + & \\ + \frac{2}{3} \mu v \frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) - \frac{4}{3} \mu v \frac{\partial}{\partial y} \left( \frac{v}{\rho} \right) + & -\frac{2}{3} \mu v \frac{\partial}{\partial x} \left( \frac{1}{\rho} \right) + & + \frac{4}{3} \mu v \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) + & k(\gamma-1) \frac{\partial}{\partial y} \left( \frac{1}{\rho} \right) \\ -k(\gamma-1) \frac{\partial}{\partial y} \left[ \left( \frac{e}{\rho^2} - \frac{u^2+v^2}{\rho} \right) \right] & -k(\gamma-1) \frac{\partial}{\partial y} \left( \frac{u}{\rho} \right) & -k(\gamma-1) \frac{\partial}{\partial y} \left( \frac{v}{\rho} \right) & \end{bmatrix}$$

where the product  $\frac{\partial}{\partial x} \left( \frac{u}{\rho} \right) \Delta \rho$  denotes  $\frac{\partial}{\partial x} \left( \frac{u}{\Delta \rho} \right)$ , etcetera.

	Case 1, Re = 50 wide inlet channel	Case 2, Re = 50 narrow inlet channel	Case 3, Re = 150 wide inlet channel	Case 4, Re = 150 narrow inlet channel
imposed UMAX	.2	.125	.4	.3
inlet Mach number	.169	.106	.338	.254
inlet p-static	1.0232	1.0260	1.0146	1.0418
maximum temperature variation	$4.34 \times 10^{-3}$	$1.86 \times 10^{-3}$	$1.54 \times 10^{-2}$	$1.10 \times 10^{-2}$
maximum mass conservation error	$3.15 \times 10^{-7}$	$1.21 \times 10^{-6}$	$3.18 \times 10^{-6}$	$1.43 \times 10^{-6}$

TABLE 1 - NUMERICAL RESULTS FOR THE BACKWARD FACING STEP PROBLEM OF FIGURE 18

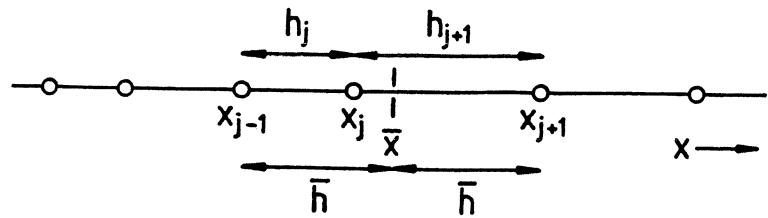


FIGURE 1 - A THREE-POINT COMPUTATIONAL MOLECULE  
ON A ONE-DIMENSIONAL, NON-UNIFORM GRID

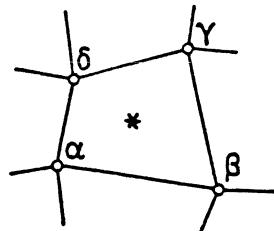


FIGURE 2 -  
THE TWO-DIMENSIONAL  
LOW-ORDER FINITE  
VOLUME  $V_{\alpha\beta\gamma\delta}$

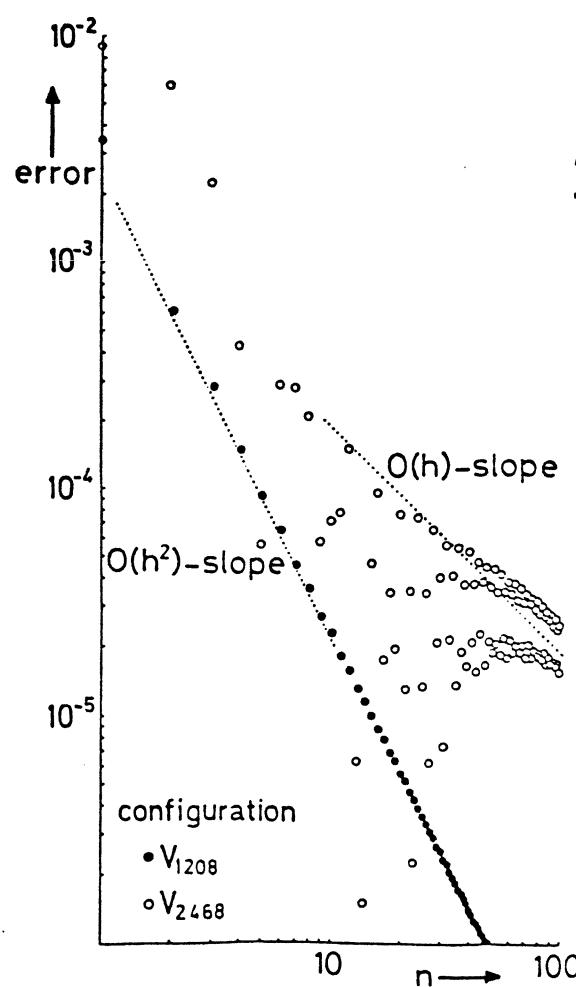


FIGURE 3 -  
THE GRID MODULE REPEATED  $n$  TIMES  $n$  TIME  
( $i = 1, \dots, n$ ;  $n = 1, \dots, 100$ )

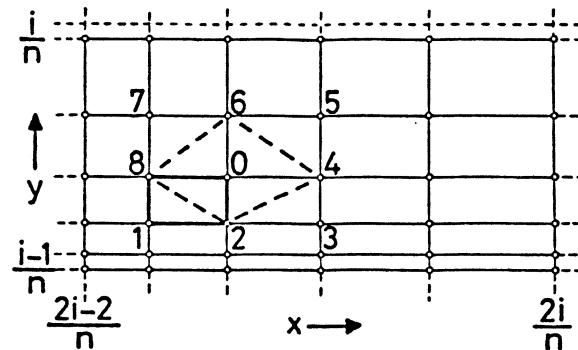
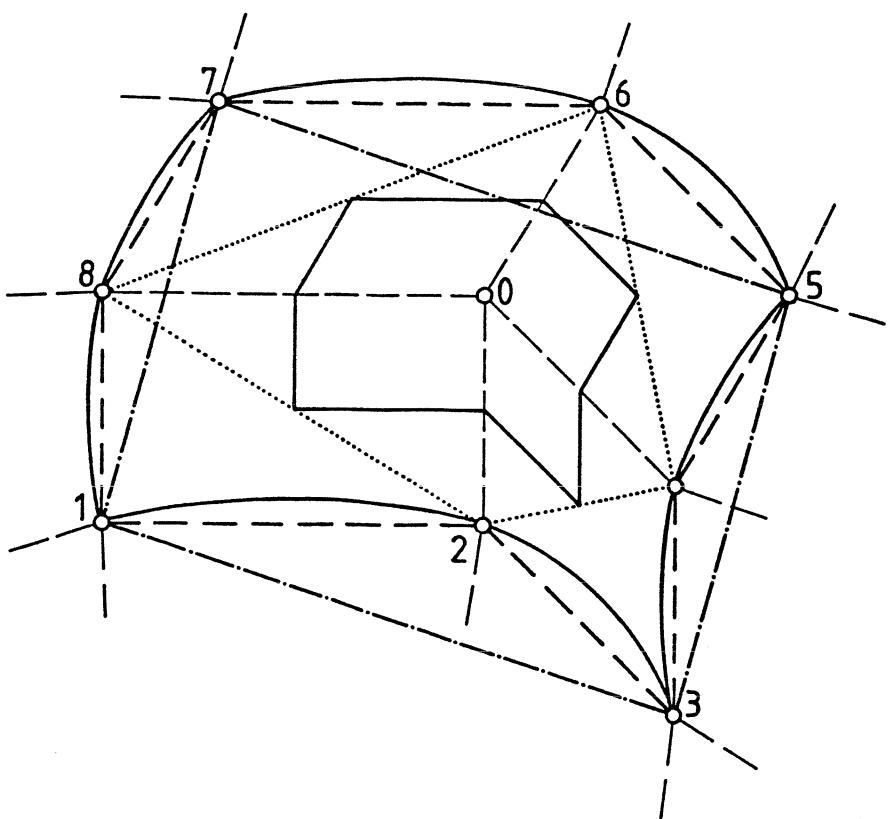
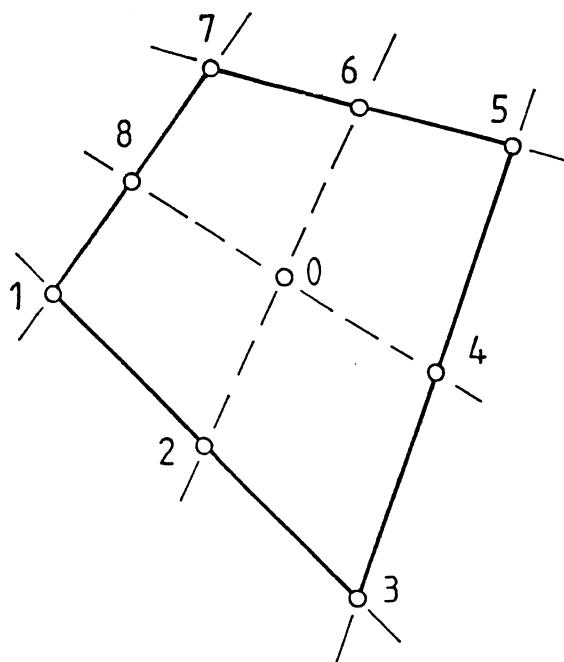


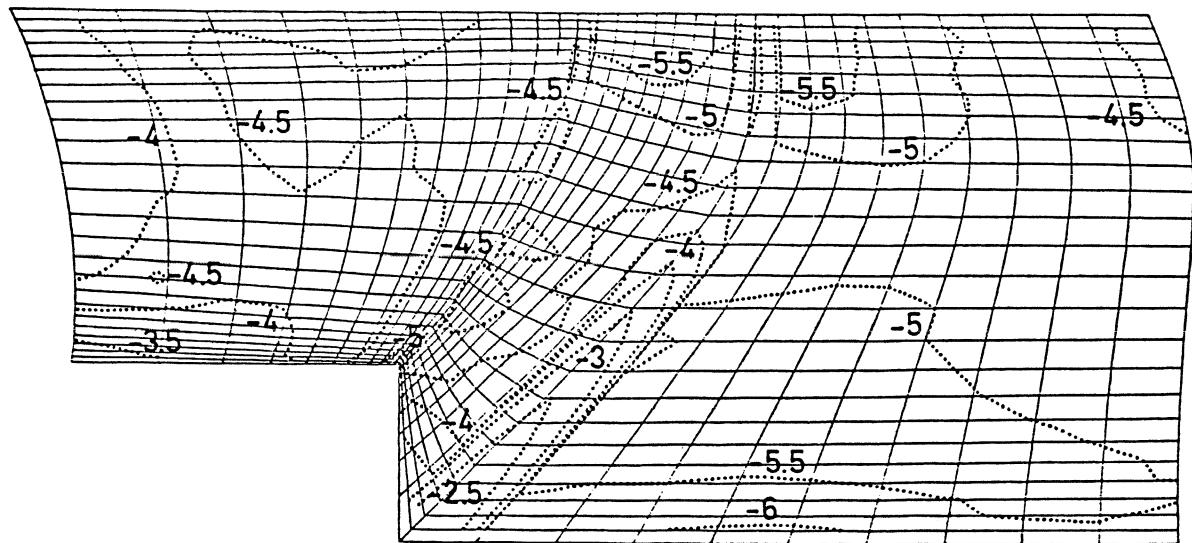
FIGURE 4 -  
CONVERGENCE RATE  
OF  $V_{\alpha\beta\gamma\delta}$



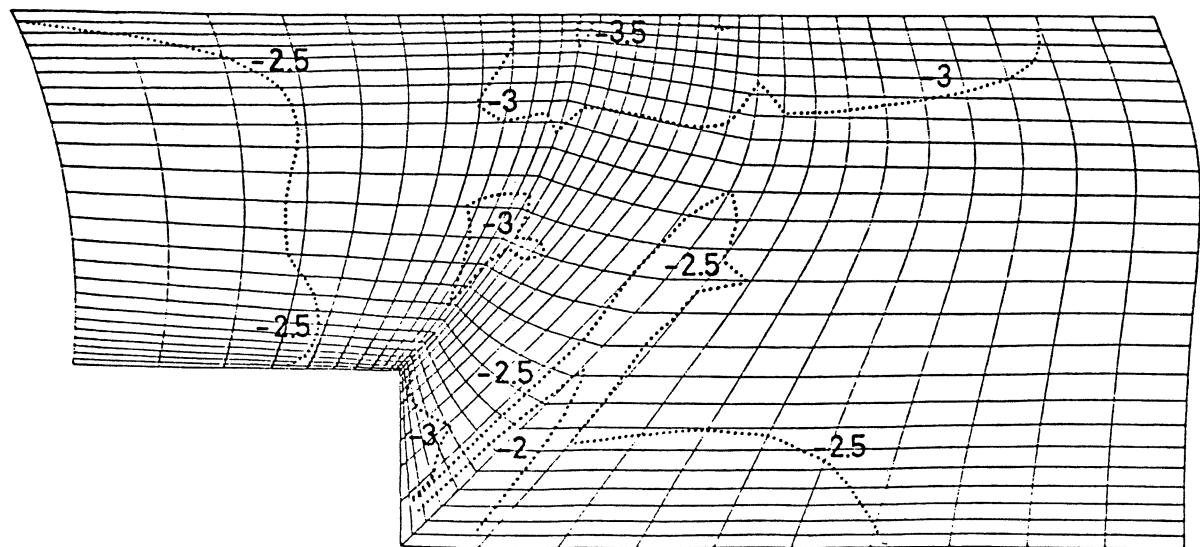
**FIGURE 5 - A THREE-TIMES-THREE POINT COMPUTATIONAL MOLECULE, WITH  
SOME POSSIBLE FINITE VOLUME SHAPES FOR CONVECTION TERMS**



**FIGURE 6 - A BIQUADRATIC FINITE VOLUME**



**FIGURE 7 - DETAIL OF BACKWARD FACING STEP GRID WITH ISO-ERROR LINES FOR CONVECTION TERMS;  $\log(\max|\tilde{a}_{1m}|)$  OF OLOCS (cf. Eq.(3.3))**



**FIGURE 8 - DETAIL OF BACKWARD FACING STEP GRID WITH ISO-ERROR LINES FOR CONVECTION TERMS;  $\log(\max|\tilde{a}_{1m}|)$  OF LOW-ORDER GFEM (cf. Eq.(3.3))**

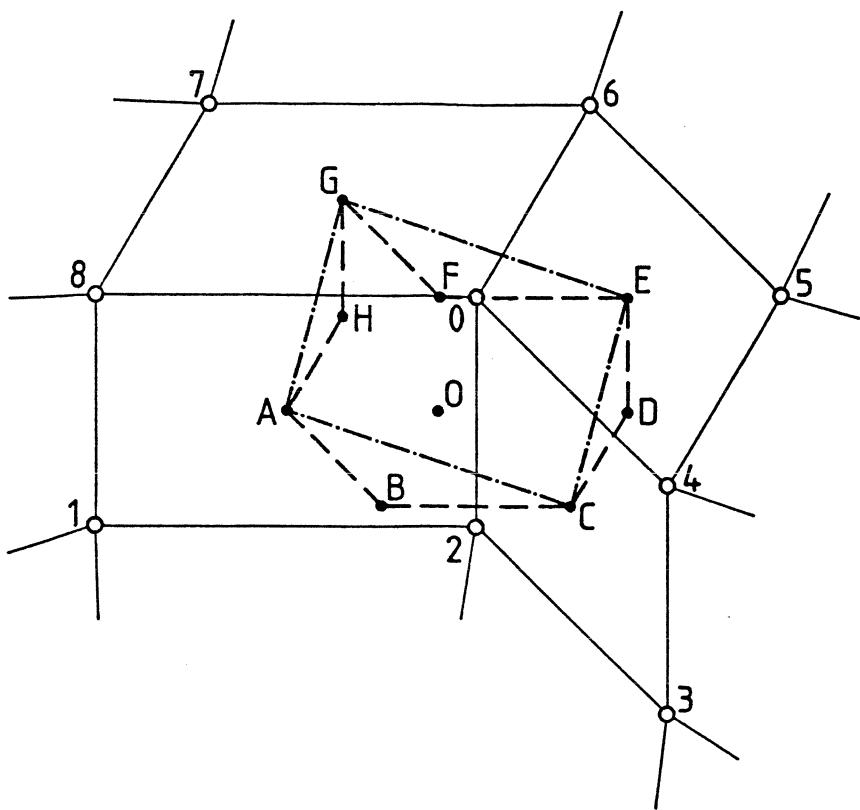


FIGURE 9 - A THREE-TIMES-THREE POINT COMPUTATIONAL MOLECULE  
WITH THE TWO FINITE VOLUMES FOR DIFFUSION TERMS

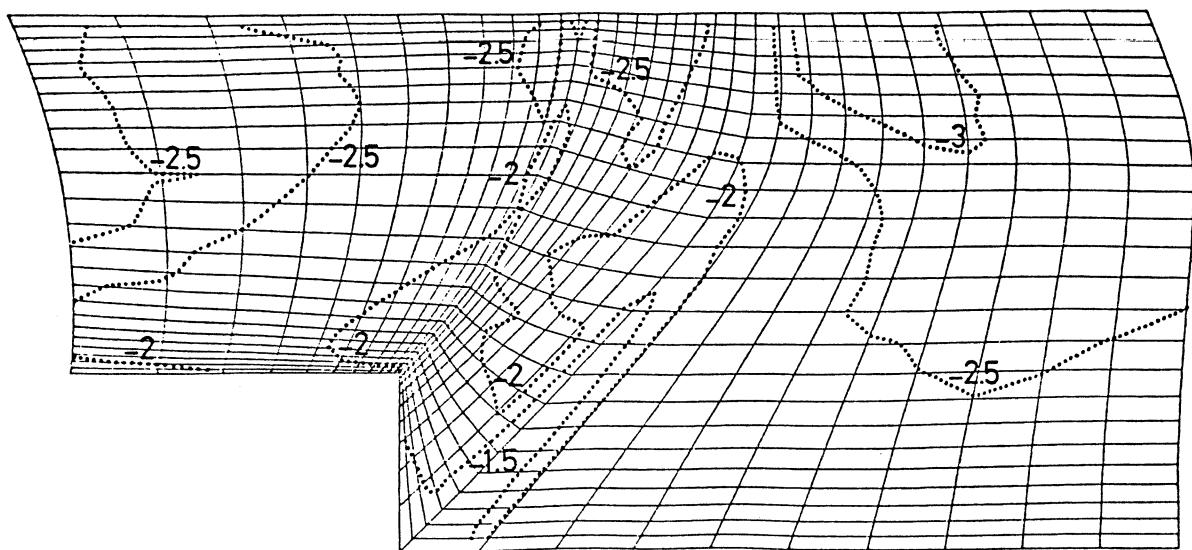


FIGURE 10 - DETAIL OF BACKWARD FACING STEP GRID WITH ISO-ERROR LINES FOR DIFFUSION TERMS;  $\log(\max |\tilde{a}_{0m}|)$  OF SCHEME VII -  $\frac{2}{5}$  SCHEME VIII (cf. Eq. (3.3))

$$4\Delta x \Delta y \text{RHS}^b =$$

$$\begin{array}{c}
 \begin{array}{ccccc}
 & -1/6 & & 1/6 & \\
 & \boxed{-2/3} & & 2/3 & \\
 \begin{array}{l} = -2\Delta y u \\ \hline \end{array} & \boxed{-1/6} & & 1/6 & \\
 & & & & \\
 & & & -2\Delta x v & \\
 & & & & \\
 & & & \boxed{1/6} & \boxed{2/3} \boxed{1/6} \\
 & & & \boxed{-1/6} & \boxed{-2/3} \boxed{-1/6} \\
 & & & & + \\
 \end{array} \\
 \\
 \begin{array}{ccccc}
 & 1/6 & -1/3 & 1/6 & \\
 & \boxed{2/3} & \boxed{-4/3} & 2/3 & \\
 \begin{array}{l} + 4 \frac{\Delta y}{\Delta x} v_x \\ \hline \end{array} & \boxed{1/6} & \boxed{-1/3} & 1/6 & \\
 & & & + 4 \frac{\Delta x}{\Delta y} v_y & \\
 & & & & \\
 & & & \boxed{1/6} & \boxed{2/3} \boxed{1/6} \\
 & & & \boxed{-1/3} & \boxed{-4/3} \boxed{-1/3} \\
 & & & & \\
 \end{array}
 \end{array}$$

FIGURE 11 - SCHEMATIC OF DISCRETIZATION OF RIGHT-HAND SIDE OF (4.3)

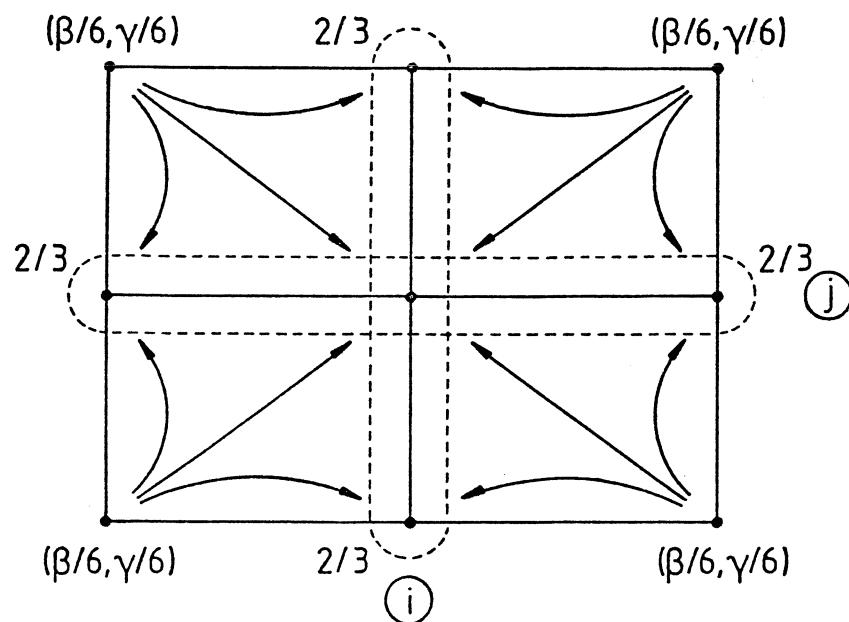
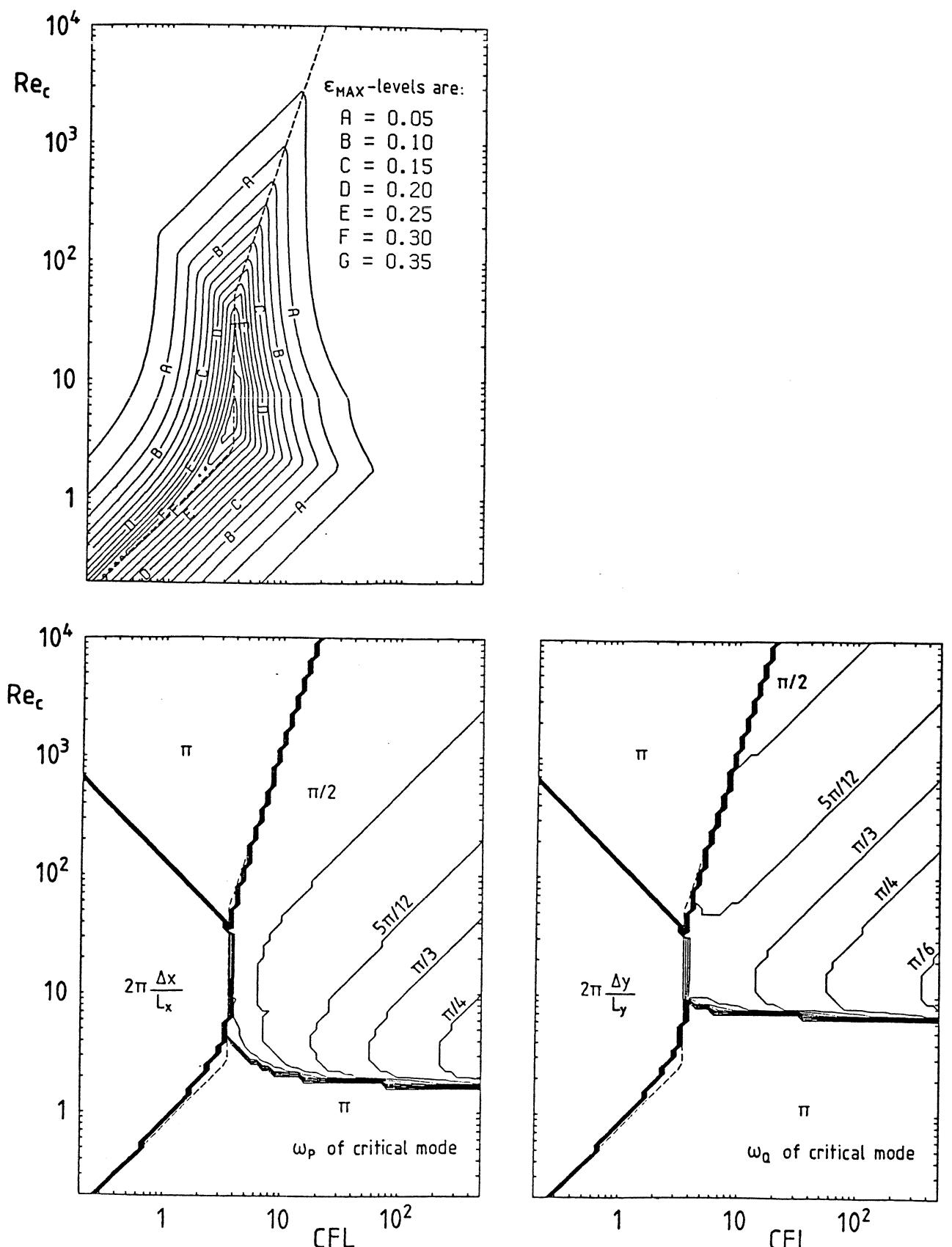
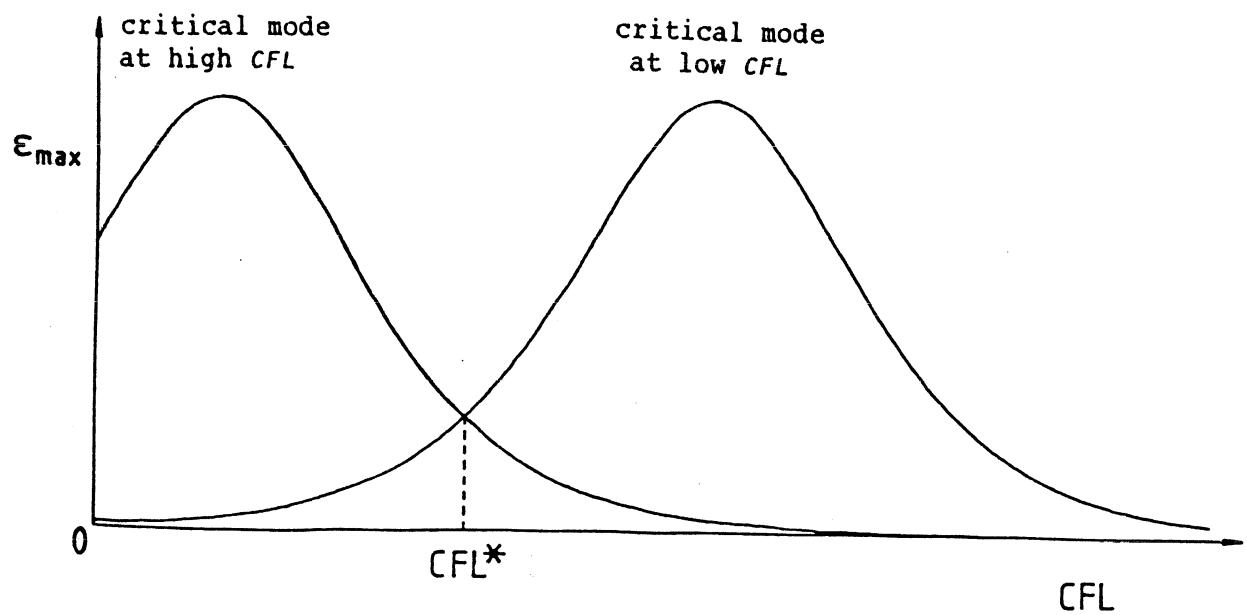


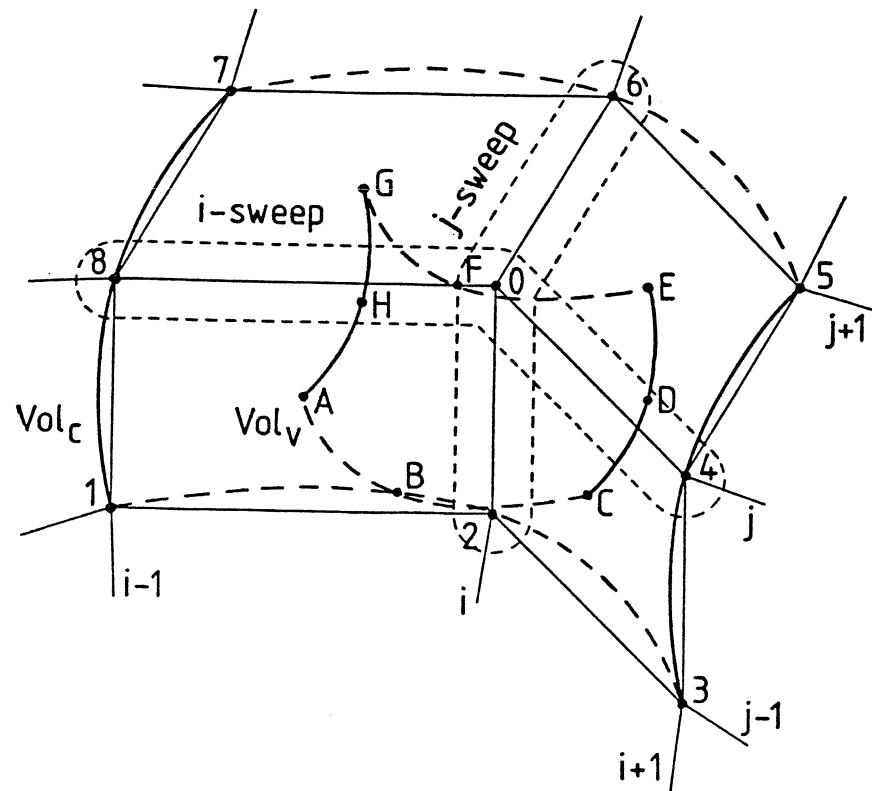
FIGURE 12 - REDISTRIBUTION OF COEFFICIENTS OF CORNER POINTS OVER THE FIVE NODAL POINTS ON THE  $i$ -LINE AND THE  $j$ -LINE



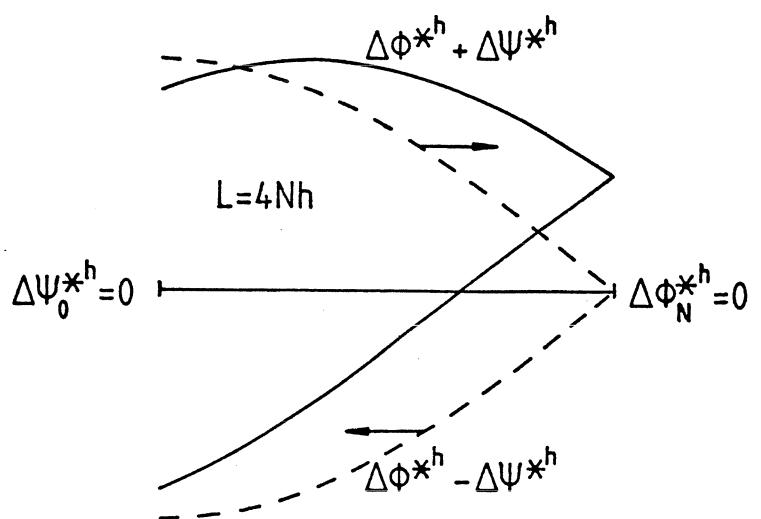
**FIGURE 13 -  $\epsilon_{\text{max}}$  AND THE TWO FREQUENCIES OF THE CRITICAL MODE AS FUNCTIONS OF  $Re_c$  AND CFL, WITH  $\alpha = \beta = \gamma = 0$ ,  $\Delta x / \Delta y = 4$ ,  $L_x / \Delta x = 100$ , AND  $L_y / \Delta y = 25$**   
**PREDICTION OF OPTIMAL CFL-NUMBER INDICATED BY DASHED LINE**



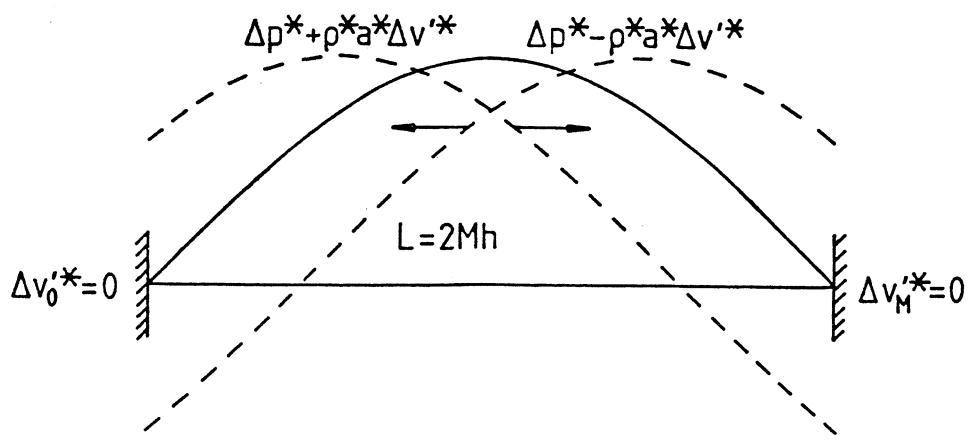
**FIGURE 14 - PRINCIPLE OF THE EXISTENCE OF AN OPTIMAL CFL-NUMBER FOR IMPLICIT APPROXIMATE FACTORIZATION SCHEMES**



**FIGURE 15 - THE IMPLICIT APPROXIMATE FACTORIZATION SCHEME ON A THREE-TIMES-THREE POINT COMPUTATIONAL MOLECULE**



**FIGURE 16 - STEADY-STATE ERROR MODE OF LOWEST POSSIBLE FREQUENCY OF THE ONE-DIMENSIONAL WAVE EQUATION**



**FIGURE 17 - THE STEADY-STATE ERROR MODE OF LOWEST POSSIBLE FREQUENCY IN BETWEEN TWO SOLID WALLS**

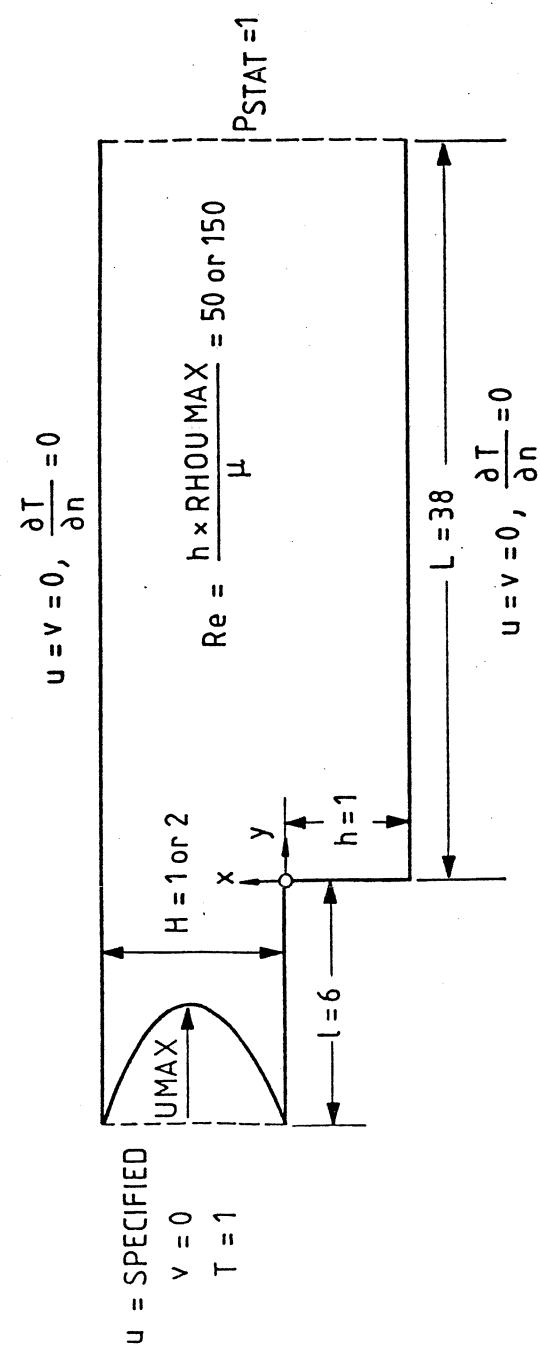
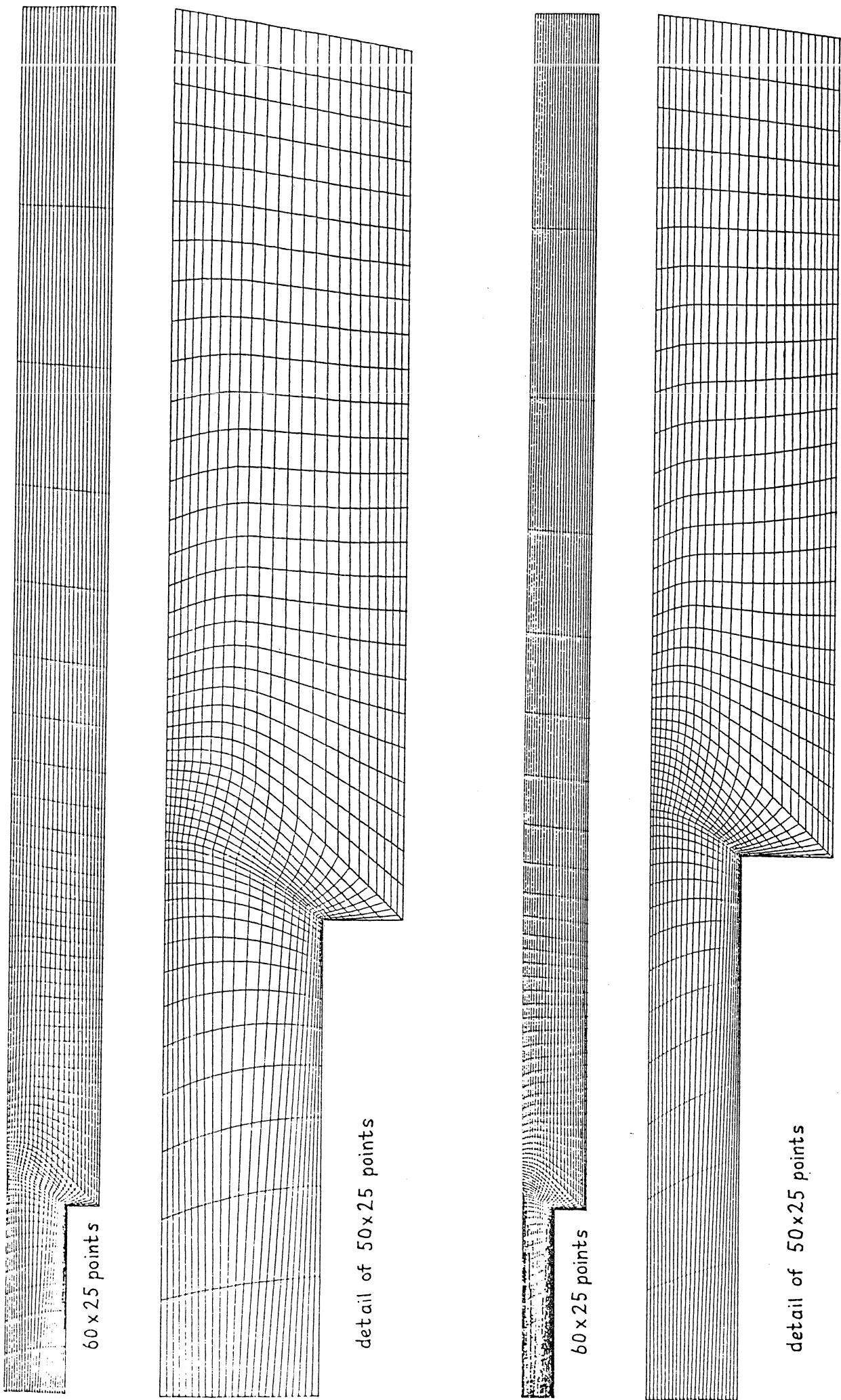
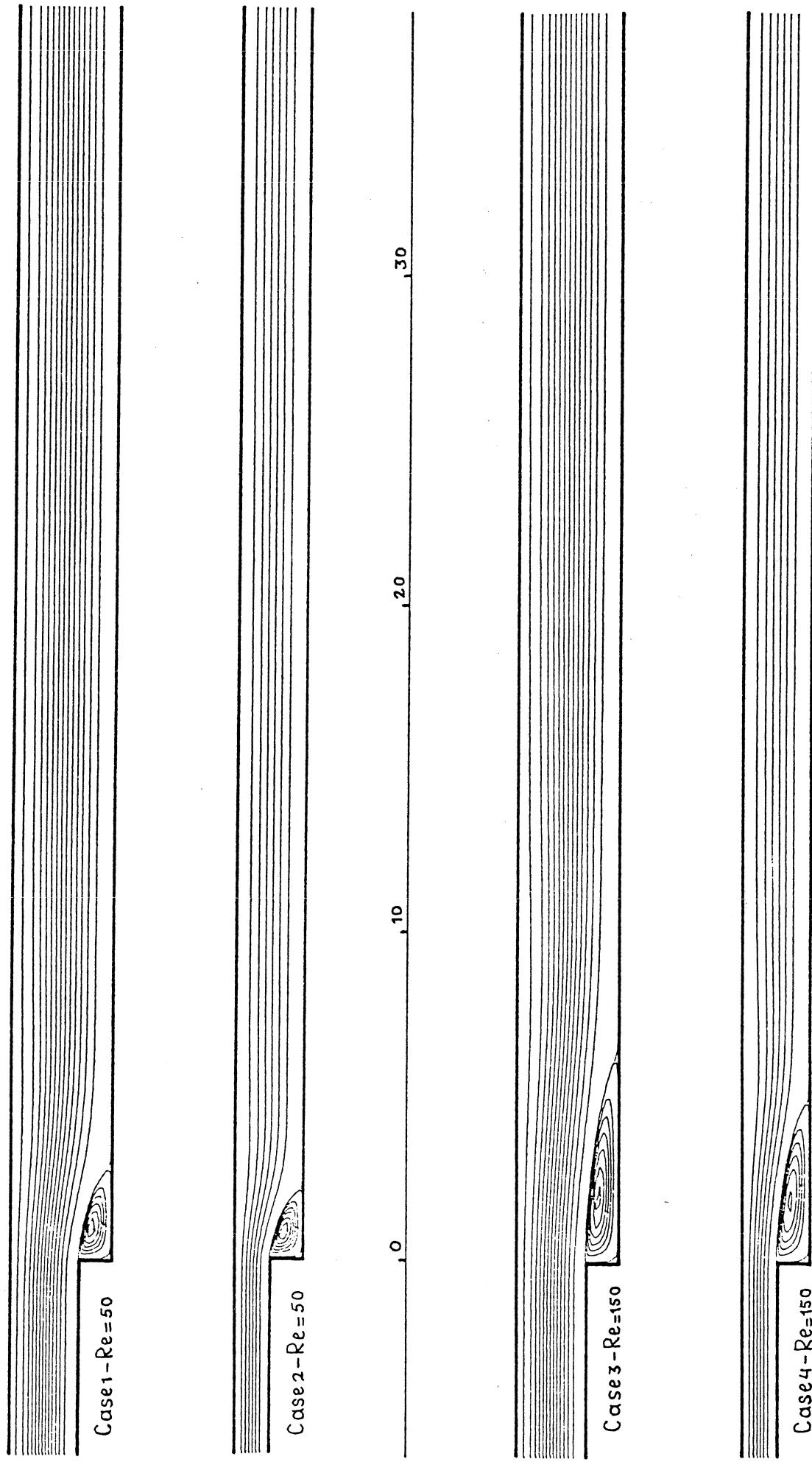


FIGURE 18 - THE BACKWARD FACING STEP PROBLEM



**FIGURE 19 - THE BACKWARD FACING STEP GRIDS**

FIGURE 20 - NORMALIZED STREAMLINES



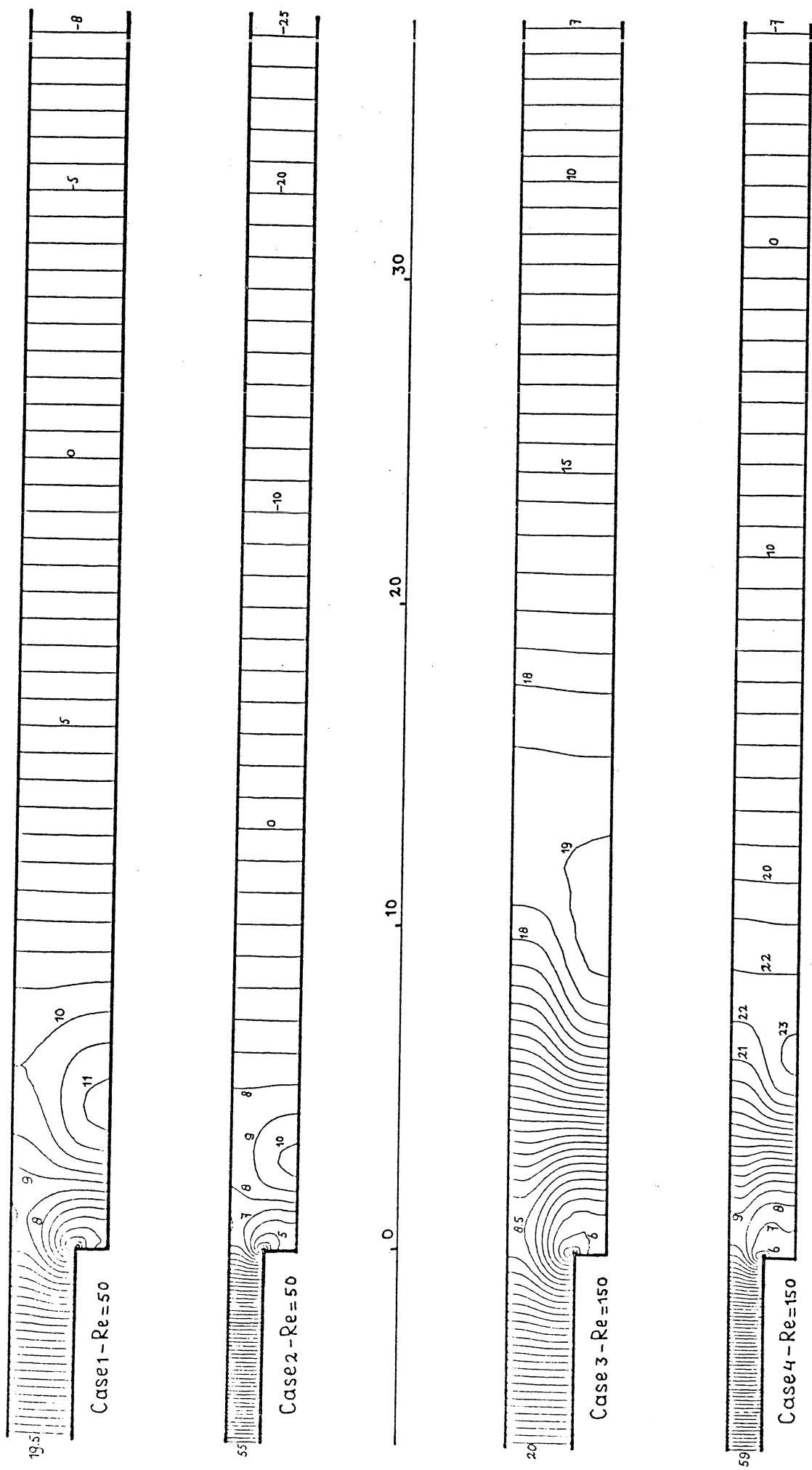
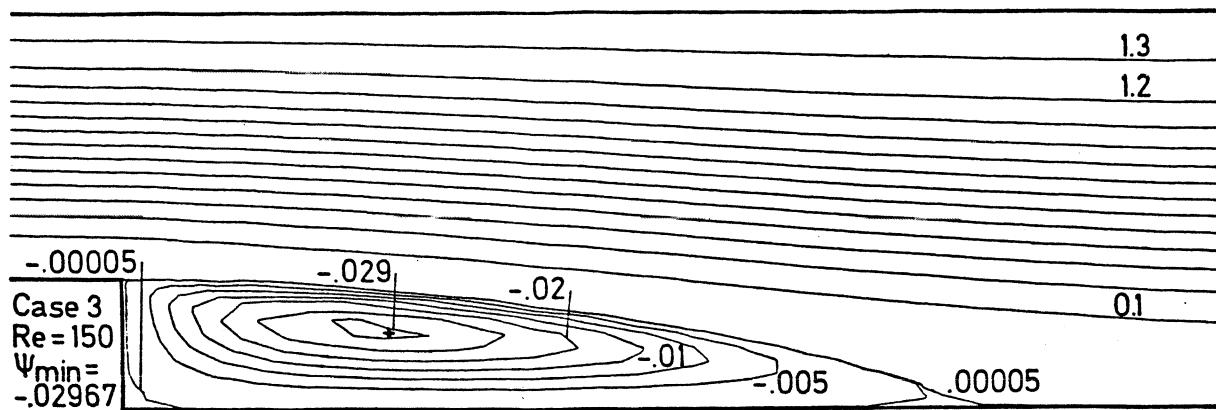
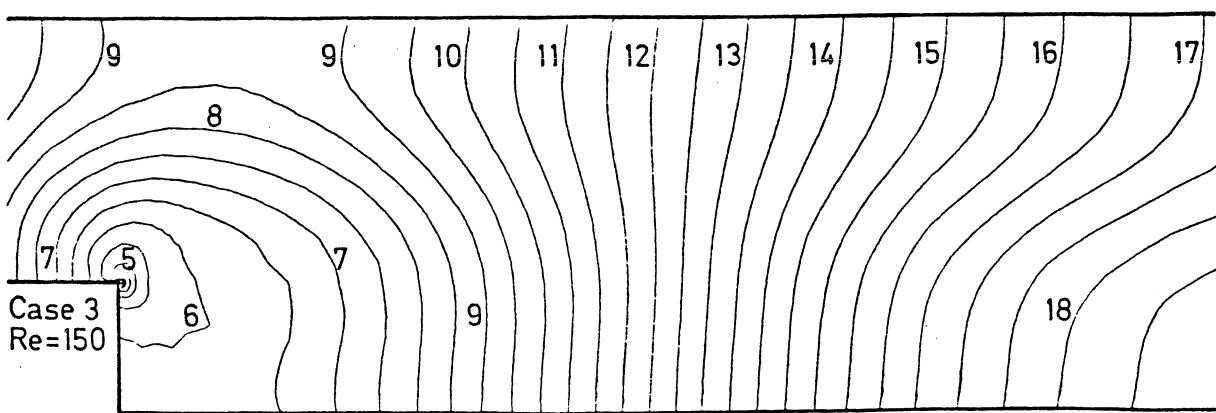


FIGURE 21 - NORMALIZED PRESSURE CONTOURS



**FIGURE 22 - NORMALIZED STREAMLINES OF COMPRESSIBLE NAVIER-STOKES CALCULATIONS AT AN INLET MACH NUMBER OF 0.34**



**FIGURE 23 - NORMALIZED PRESSURE CONTOURS**

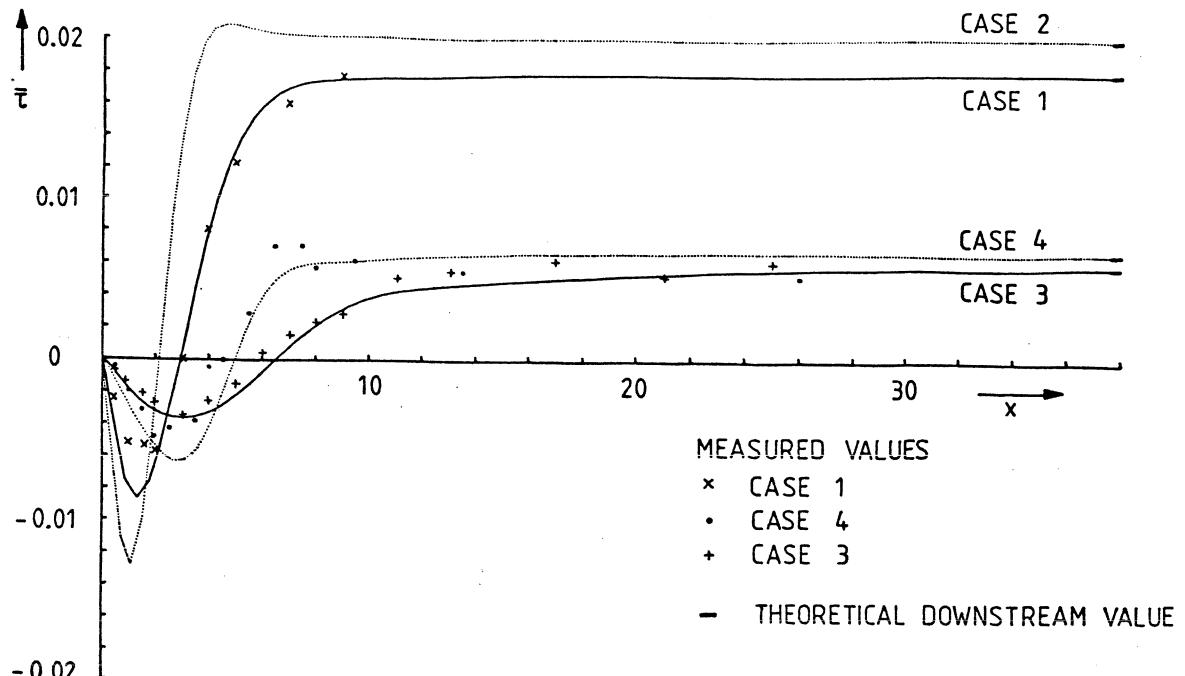


FIGURE 24 - SECOND-ORDER ACCURATE CALCULATION OF SHEAR STRESSES  
AT LOWER WALL OF BACKWARD FACING STEP

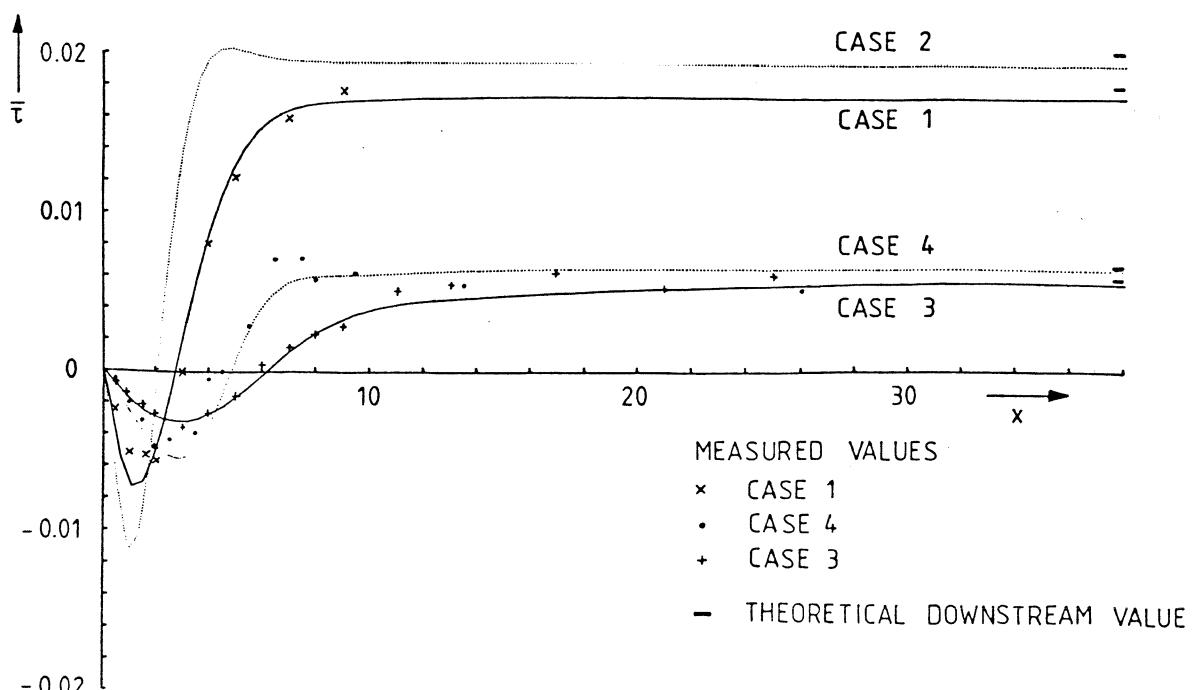


FIGURE 25 - FIRST-ORDER ACCURATE CALCULATION OF SHEAR STRESSES  
AT LOWER WALL OF BACKWARD FACING STEP

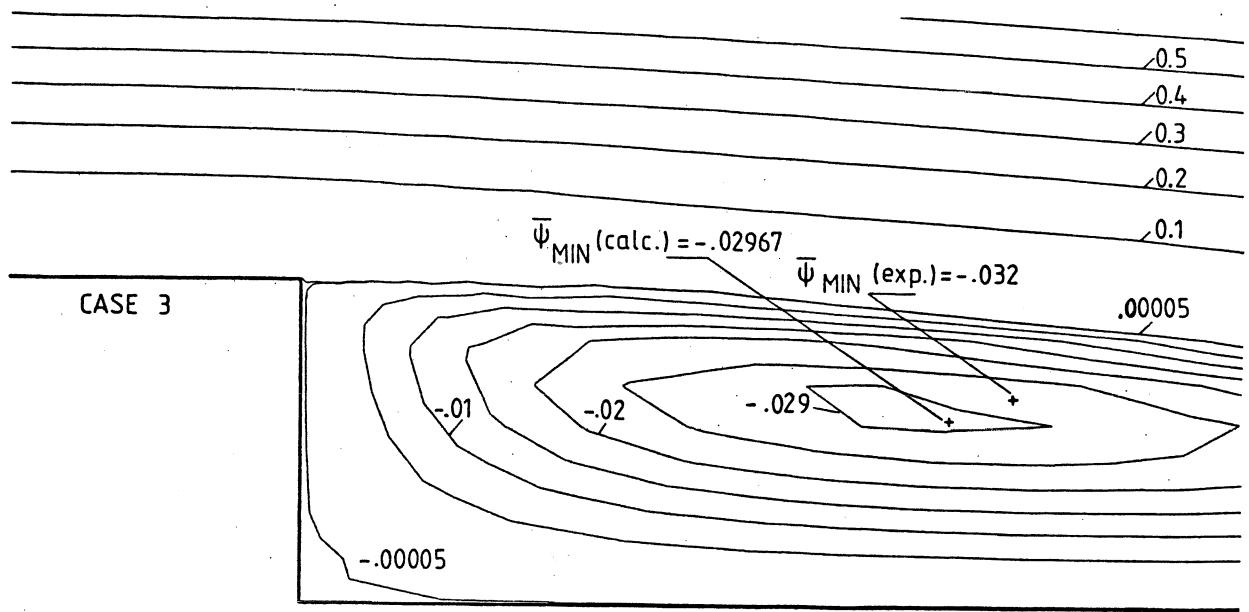


FIGURE 26 - CLOSE-UP OF FIGURE 22

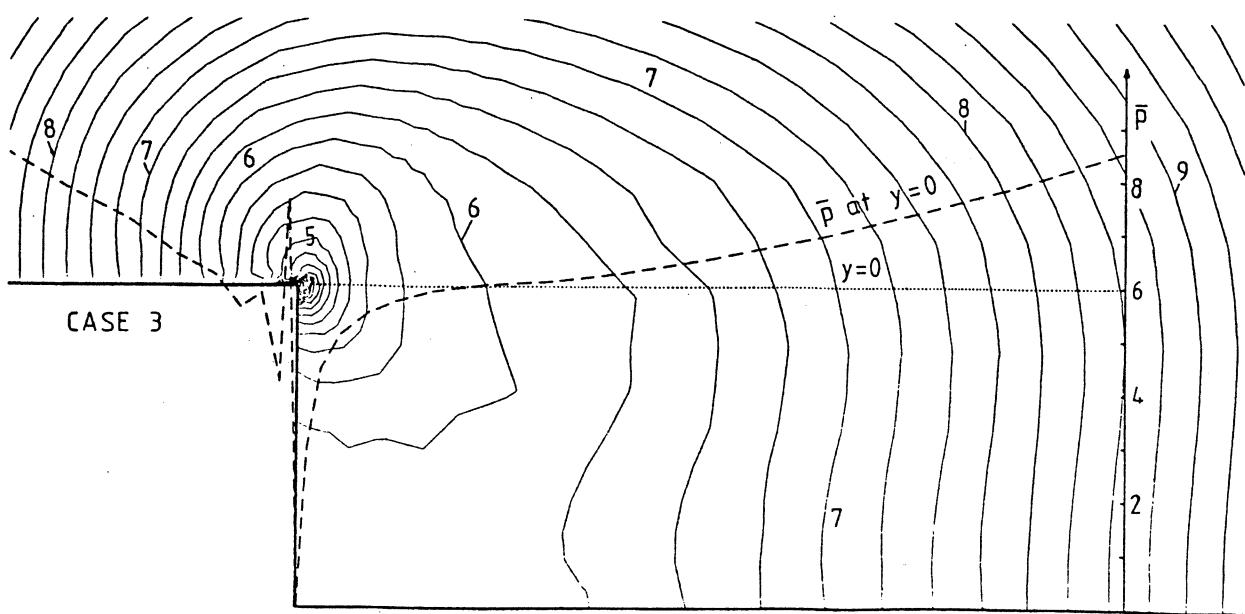


FIGURE 27 - CLOSE UP OF FIGURE 23

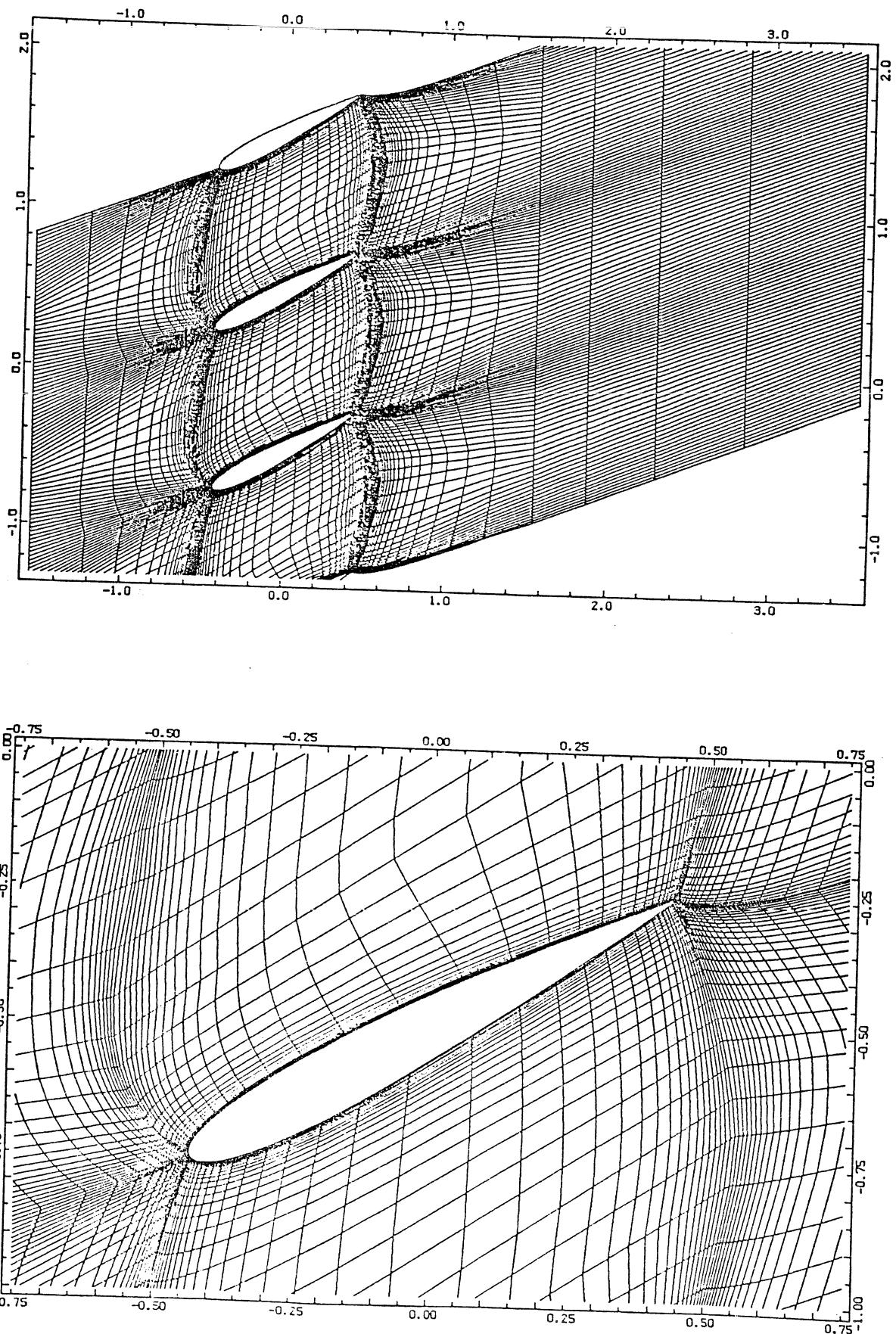


FIGURE 28 - THE 39 TIMES 78 POINT GRID FOR THE  
BLADE-TO-BLADE FLOW CALCULATIONS

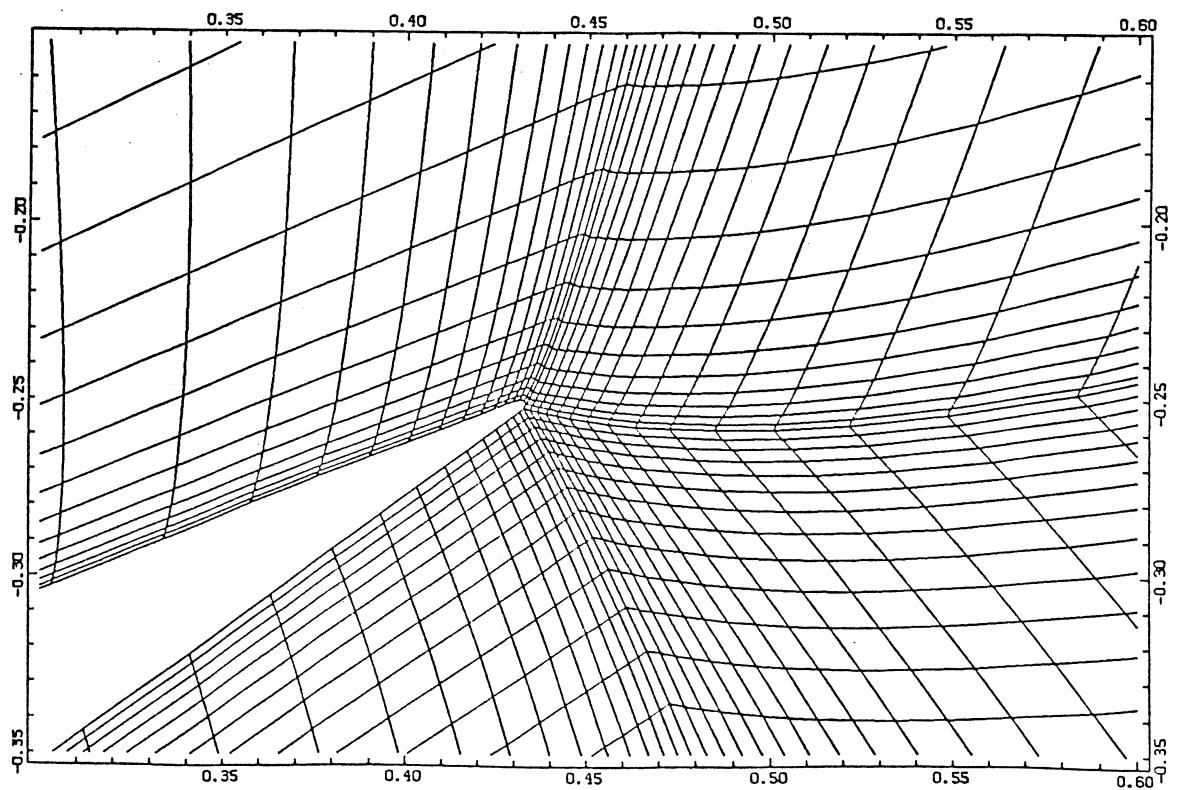
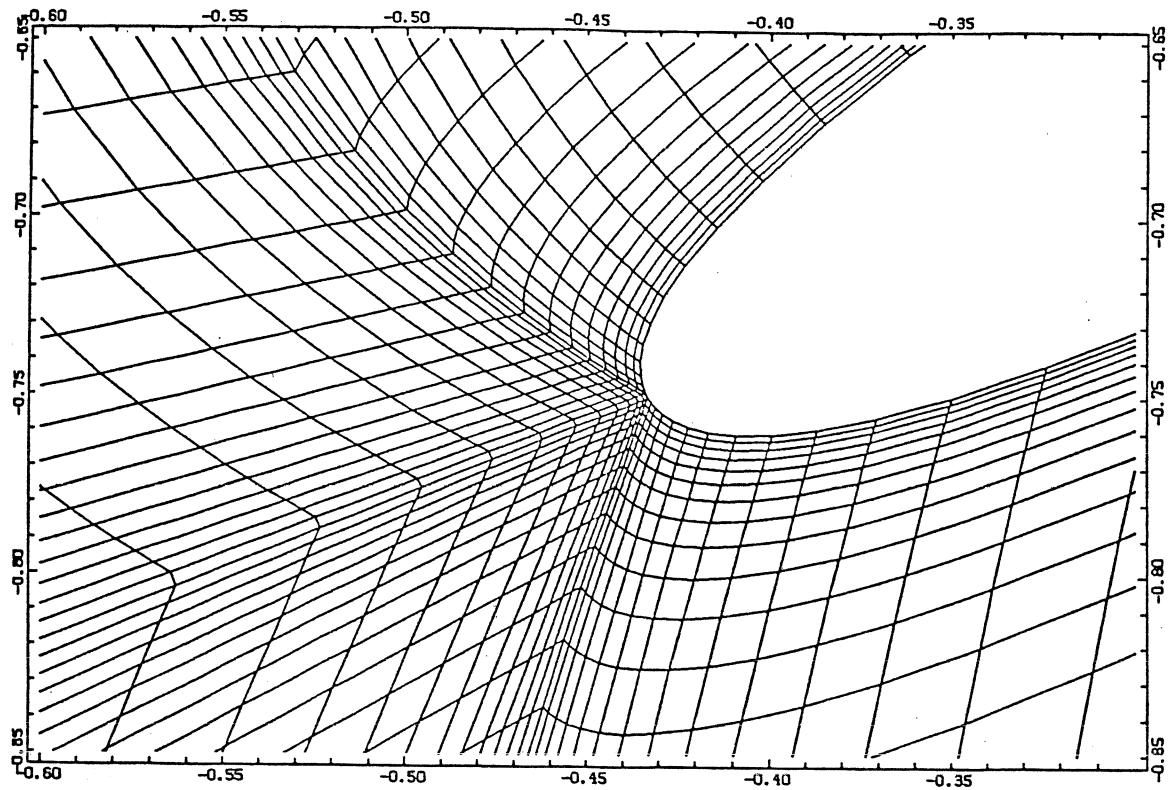


FIGURE 29 - SOME DETAILS OF THE GRID

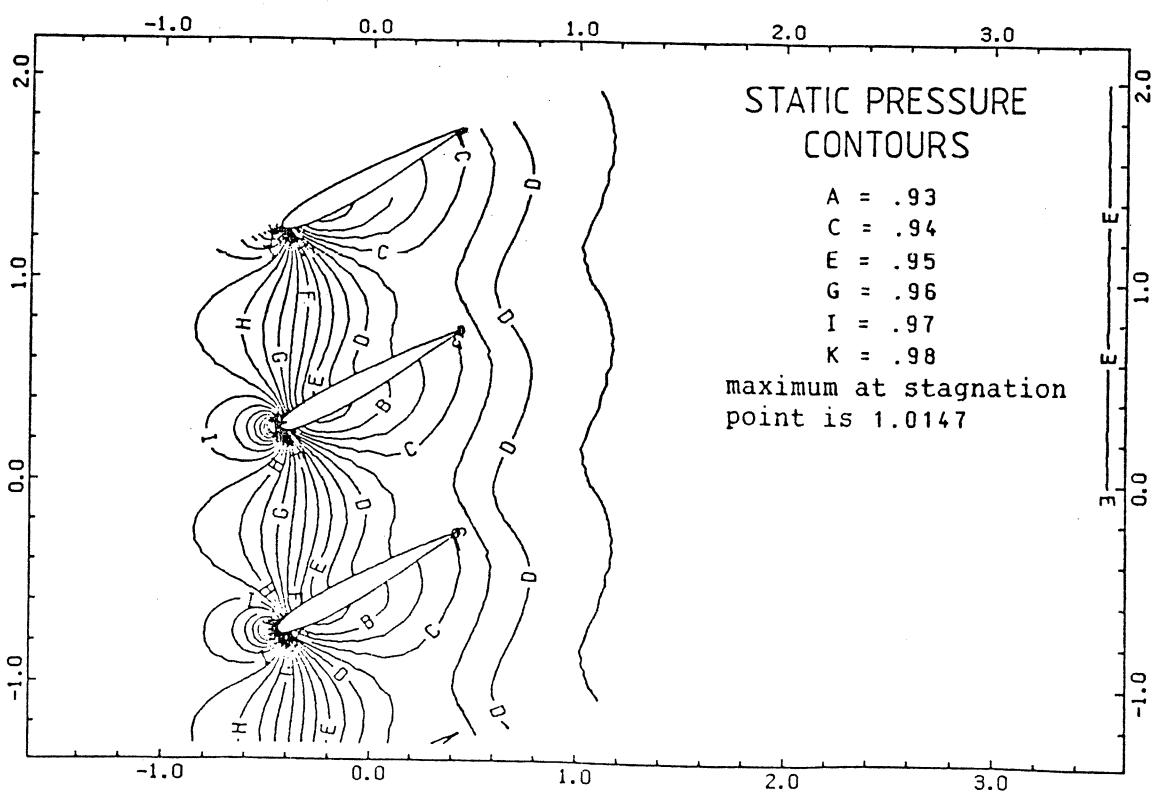
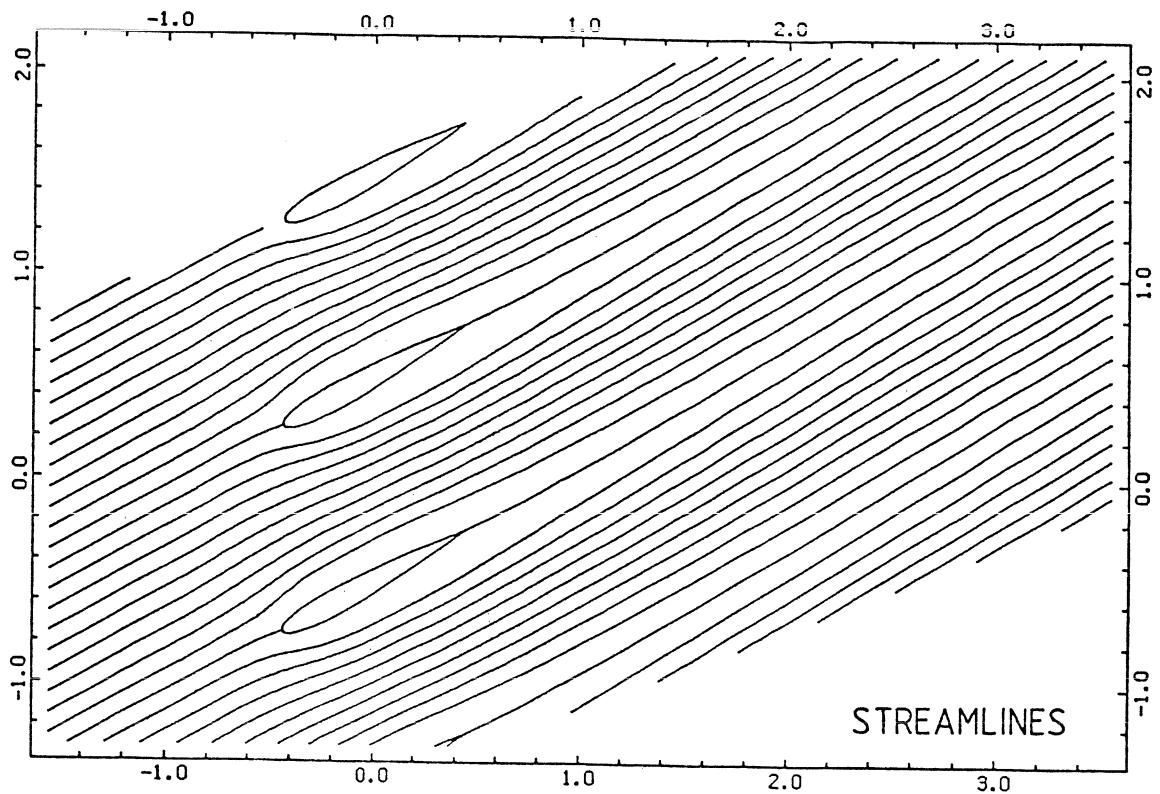


FIGURE 30 - VISCOUS STEADY-STATE SOLUTION

AT  $Re = 200$ ,  $\alpha = 30^\circ$  AND  $p_s = .95$

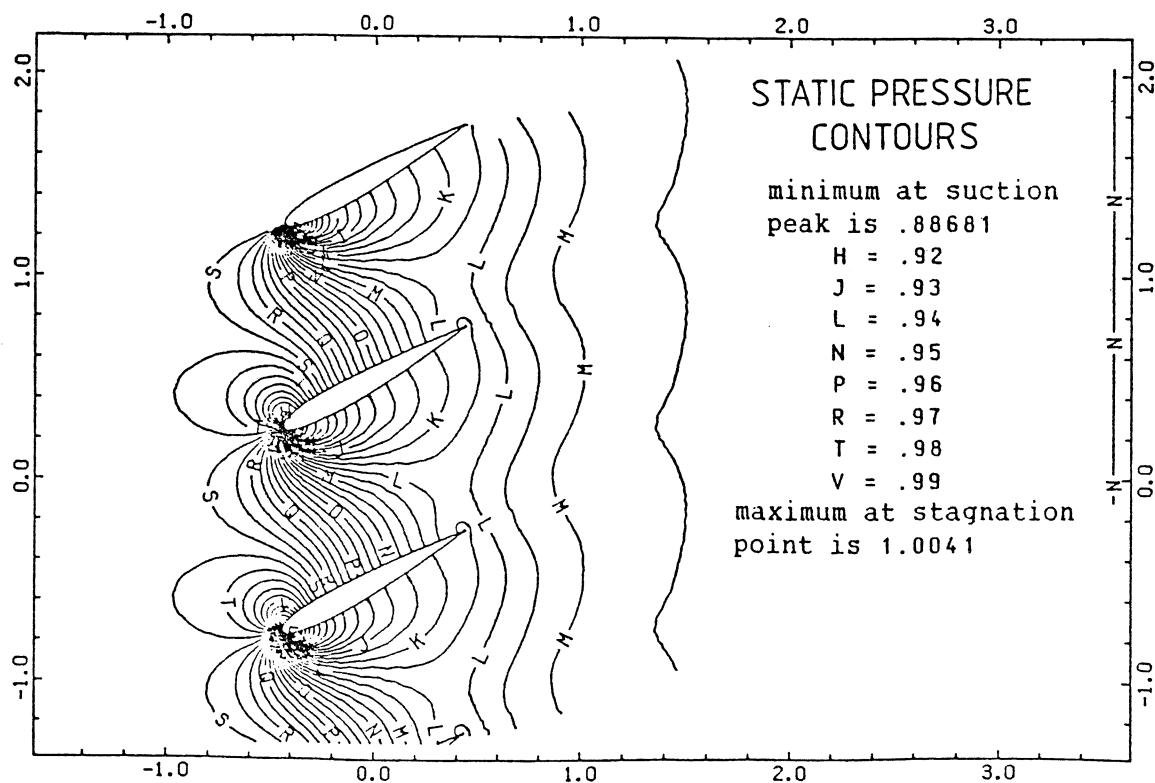
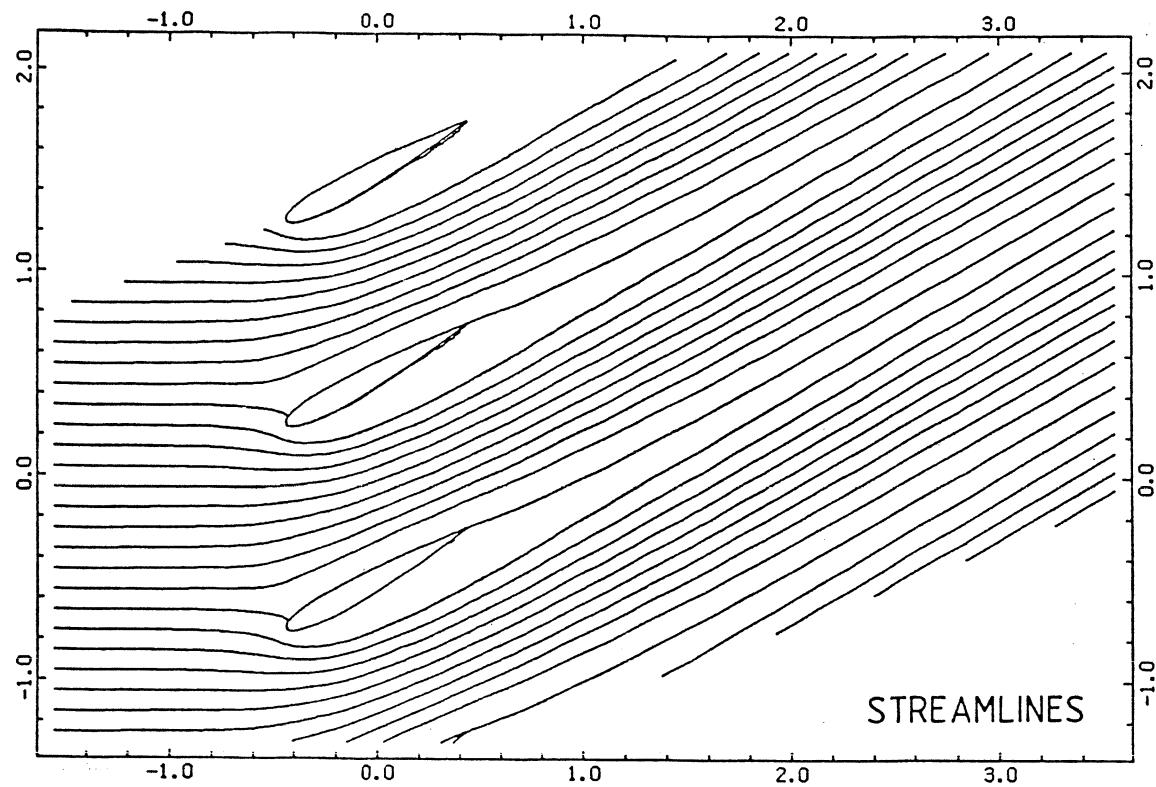


FIGURE 31 - VISCOUS STEADY-STATE SOLUTION

AT  $Re = 200$ ,  $\alpha = 0^\circ$  AND  $p_s = .95$

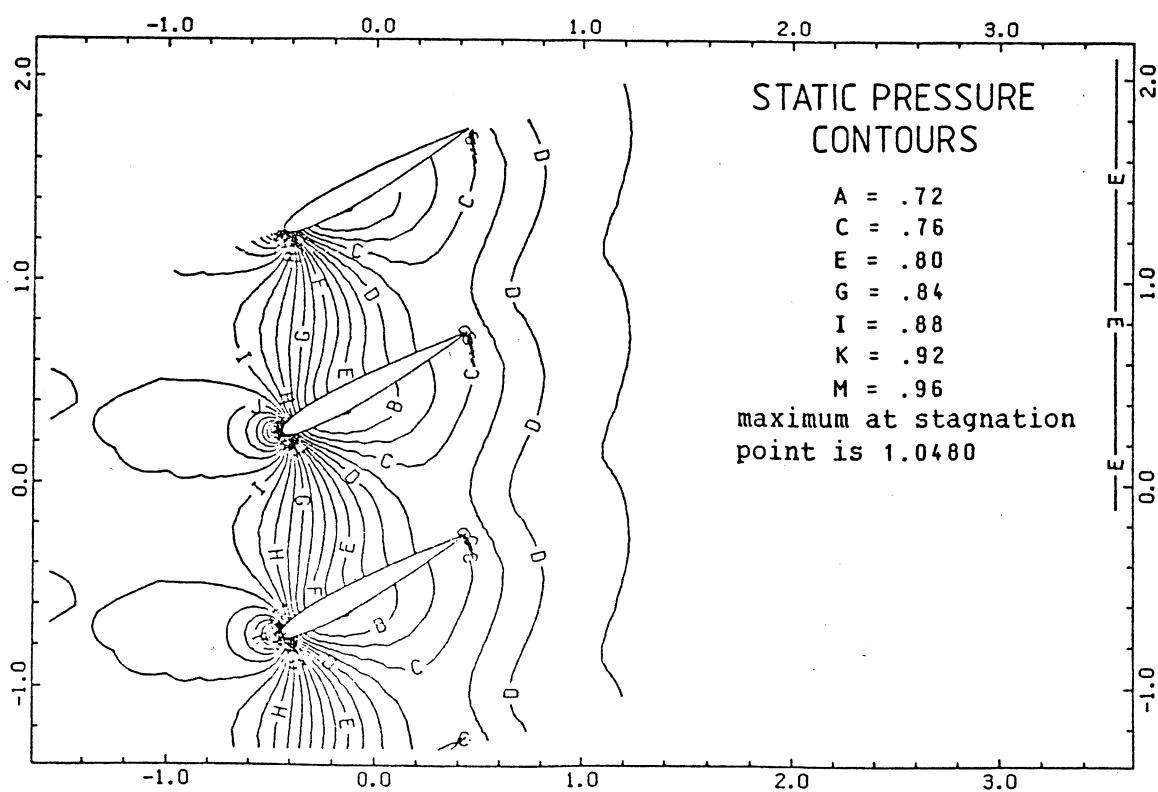
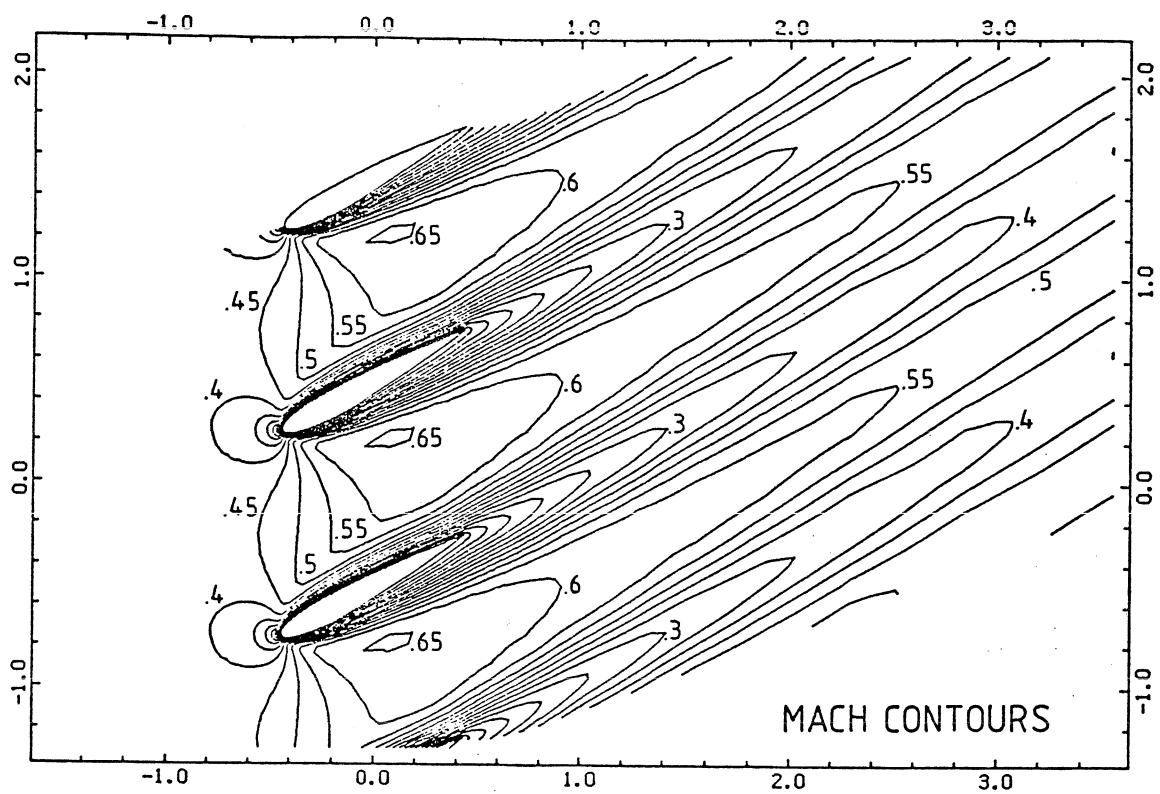


FIGURE 32 - VISCOUS STEADY-STATE SOLUTION

AT  $Re = 200$ ,  $\alpha = 30^\circ$  AND  $p_s = .80$

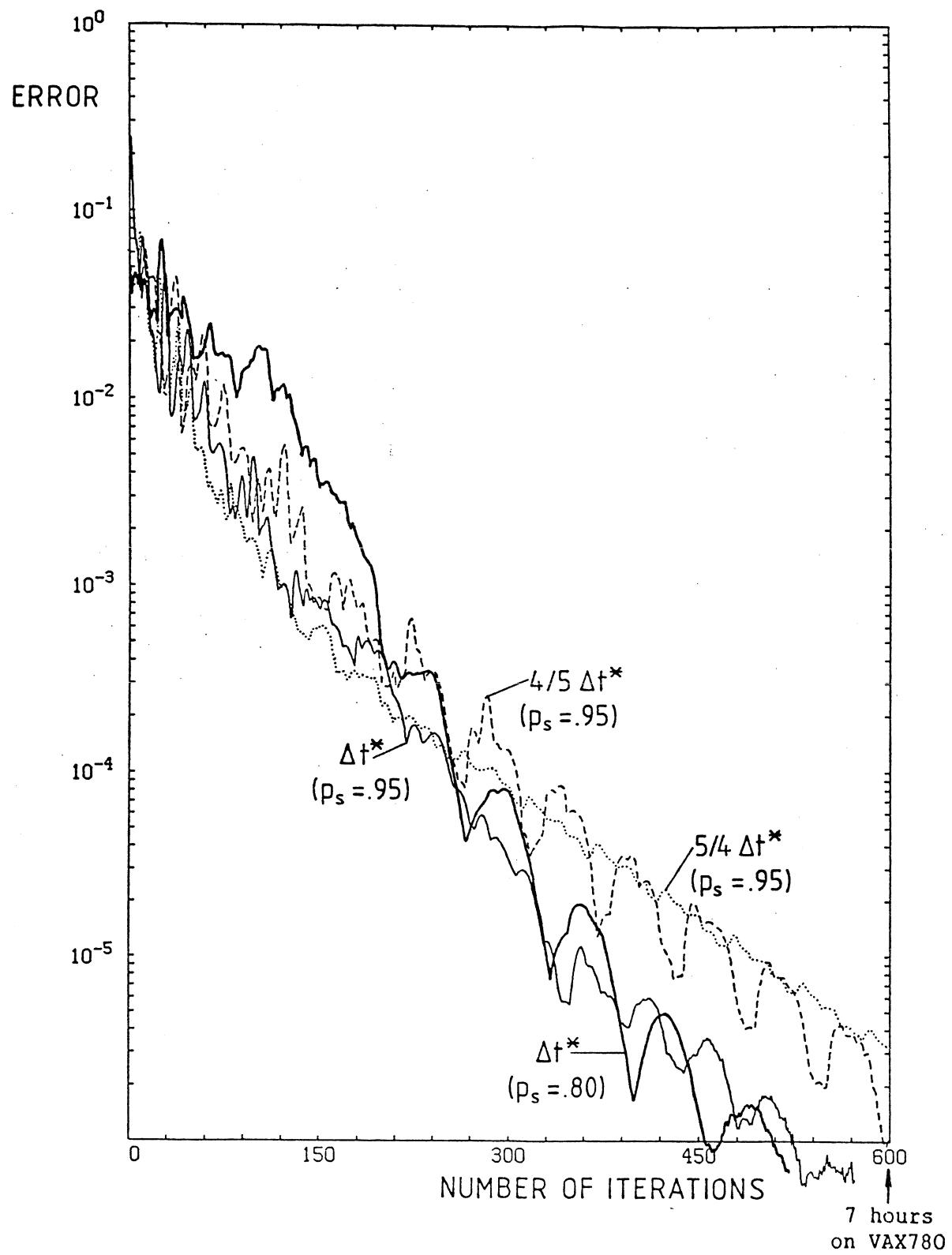
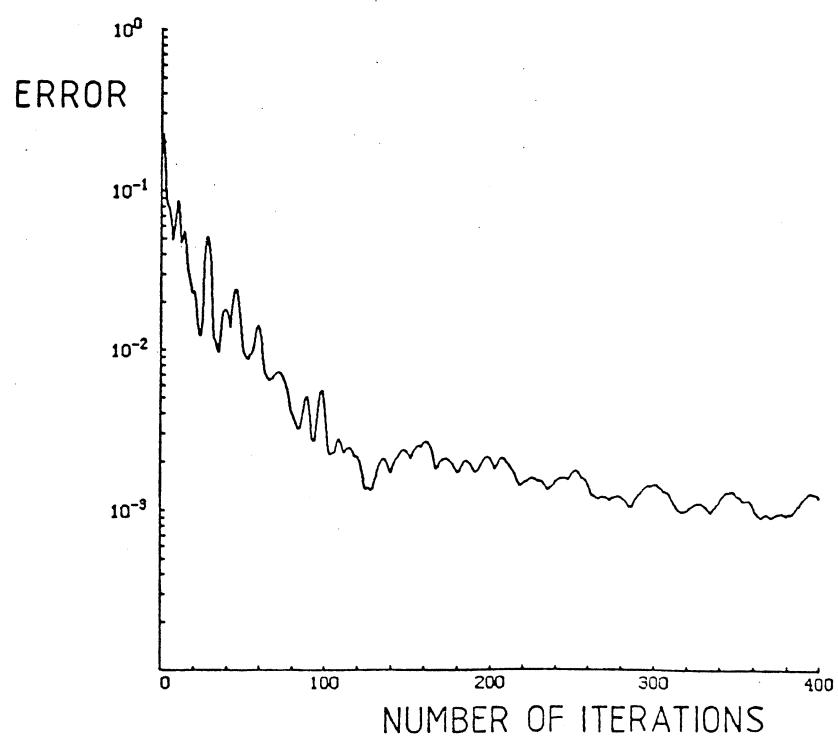
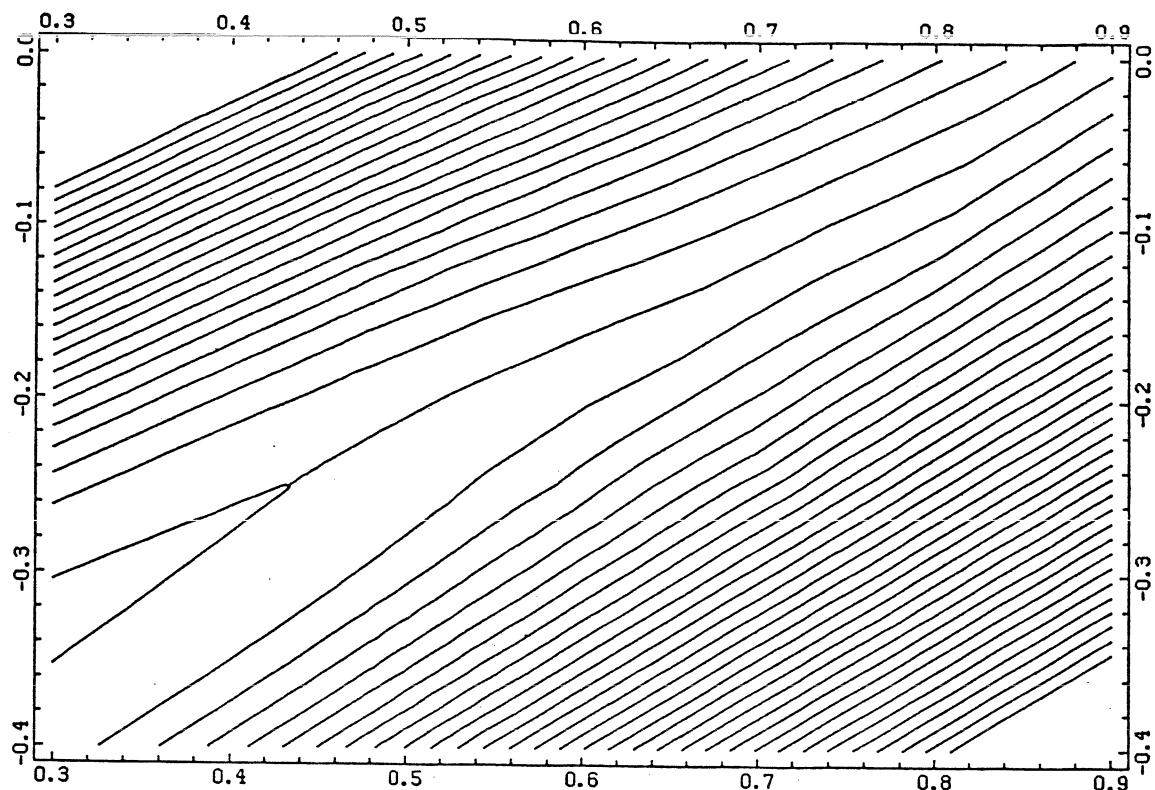


FIGURE 33 - CONVERGENCE HISTORIES OF DEVELOPED NAVIER-STOKES RELAXATION SCHEME, FOR NACA0012 BLADE-TO-BLADE FLOW AT  $Re = 200$  AND  $\alpha = 30^\circ$



**FIGURE 34 - NON-CONVERGED SOLUTION AT  $Re = 500$ ,  $\alpha = 30^\circ$  AND  
 $p_s = .95$ ; STREAMLINES AND CONVERGENCE HISTORY**

VON KARMAN INSTITUTE FOR FLUID DYNAMICS

LECTURE SERIES 1988-05

COMPUTATIONAL FLUID DYNAMICS

MARCH 7 - 11, 1988

A FULLY IMPLICIT SCHEME FOR UNSTEADY FLOW CALCULATIONS  
SOLVED BY THE APPROXIMATE FACTORIZATION TECHNIQUE

M. Kloker & M. Borsboom  
VKI

This report was made by M. Kloker under the supervision of  
M. Borsboom in fulfillment of the VKI Diploma requirements  
during the academic year 1985-86

## 1. INTRODUCTION

On the way to calculate flows at higher and higher Reynolds numbers one has to face very quickly unsteady flow phenomena. Large scale unsteady behaviour like vortex streets can in principle be modelled by creating artificially a steady state, but then important information of the real flow is lost. Hence it is desired to calculate the flow directly by solving the unsteady Navier-Stokes equations. This set of equations is parabolic in time, so the unsteady behaviour of the flow can be simulated numerically by means of a time-accurate time-marching procedure. For that, there are in principle two possibilities:

- first, an explicit method, but then one is obliged to take very small time steps due to stability reasons. The time-step limit is especially strong in the case of the Navier-Stokes equations because of the small grid cells that have to be used in regions of steep gradients (e.g. boundary layers). Stability/accuracy analyses show that the cell-Reynolds number should be smaller than two in those regions;
- the second possibility is to use an accurate implicit time-marching procedure, which is however rather costly. On the one hand the time step can be slightly increased<sup>1</sup> due to improved stability, on the other hand the calculation time per time step is higher in comparison to explicit methods.

From there, both methods have the drawback of a relatively large calculation time. This turns out to be especially evident in cases where the typical time scale of the unsteady flow is much higher than the time step imposed by the numerics. McCroskey of the NASA Ames research center, who is working in the field of wing flutter phenomena, is cited here [1] to pick out one typical example:

"Unfortunately, the current computational aerodynamics codes that might be capable of capturing the low frequency complexity of the flutter problem (...and other ones, M. Kloker) tend to have severe stability limitations with respect to the maximum time step that can be used. This translates into undesired long computational times per cycle of oscillation. A great gain in reducing the computational time will come from removing the stability limit on the time step."

Hence, efforts to overcome the mentioned problem can be said to be remunerative. The present paper deals with the development of an accurate and efficient technique for solving the unsteady flow equations.

<sup>1</sup>) This is valid for the current time-linearized, non-iterative (and therefore virtually semi-)implicit schemes; the fully implicit schemes (fully in the sense of fully nonlinear), characterized by no stability limit on the time step, have been disclaimed up to date, due to the inherently iterative procedure that was supposed to be inefficient.

## 2. DEVELOPMENT OF THE SCHEME

The unsteady fluid-dynamics equations (Euler, Navier-Stokes) can be written in the form

$$\frac{\partial}{\partial t} \bar{U} + \{\text{space operators}\} \bar{U} = 0 \quad (1)$$

with  $\bar{U}$  the vector of unknowns, where  $\{\text{space operators}\}$  denotes the spatial derivatives, i.e. convection, diffusion and source terms (see [2] for the detailed set of equations). For a time-accurate numerical simulation, (1) can best<sup>1</sup> be discretized as

$$\frac{\bar{U}^{t+1} - \bar{U}^t}{\Delta t} + 0.5 \left[ \{\text{sp.op}\} \bar{U}^{t+1} + \{\text{sp.op}\} \bar{U}^t \right] = 0 \quad (2)$$

with  $\bar{U}^t = \bar{U}(t) = \bar{U}(n\Delta t)$ . This second-order accurate time-differencing formula leads upon a spatial discretisation to a large nonlinear system of algebraic equations. By the introduction of a second time derivative solutions of (2) can be found by means of a time-marching technique that should converge fast but does not need to be accurate:

$$\boxed{\frac{\bar{U}^{n+1} - \bar{U}^n}{\Delta t_n}} + \boxed{\frac{\bar{U}^{n+1} - \bar{U}^t}{\Delta t_p}} + 0.5 \left[ \{\text{sp.op}\} \bar{U}^{n+1} + \{\text{sp.op}\} \bar{U}^t \right] = 0, \quad (3)$$

where the additional time derivative, indicated by the dashed square, has been already discretized<sup>2</sup> and where the subscript n denotes the numerical and p the physical time step.  $\bar{U}^{n+1} - \bar{U}^n$  expresses the difference between two successive iteration levels, with  $\bar{U}^n = \bar{U}(n\Delta t_n)$ .

In the case of convergence,  $\bar{U}^{n+1} - \bar{U}^n$  tends to zero and  $\bar{U}^{n+1}$  to  $\bar{U}^{t+1}$ , the solution at the new time level ( $t+1$ ).

Upon the application of the delta formulation and the Approximate Factorization technique [2,3], the equations become in the two-dimensional case

$$\left[ I + 0.5 \frac{\Delta t_n \Delta t_p}{\Delta t_n + \Delta t_p} \{\text{sp.op}\}_X \right] \Delta \bar{U}^* = \text{RHS} \quad (4a)$$

$$\left[ I + 0.5 \frac{\Delta t_n \Delta t_p}{\Delta t_n + \Delta t_p} \{\text{sp.op}\}_Y \right] \Delta \bar{U}^n = \Delta \bar{U}^* \quad (4b)$$

with I the identity matrix and

$$\Delta \bar{U}^n = \bar{U}^{n+1} - \bar{U}^n. \quad (4c)$$

Furthermore, RHS denotes the right-hand side of (4a):

$$\text{RHS} = - \frac{\Delta t_n \Delta t_p}{\Delta t_n + \Delta t_p} \left[ \frac{\bar{U}^n - \bar{U}^t}{\Delta t_p} + 0.5 \left[ (\text{sp.op}) \bar{U}^n + (\text{sp.op}) \bar{U}^t \right] \right] \quad (4d)$$

As can be easily seen, convergence yields  $\Delta \bar{U}^n = 0$  respectively  $\text{RHS} = 0$ , and as a consequence the solution of the discretized unsteady, fully nonlinear system of equations (2), that couples the time levels ( $t$ ) and ( $t+1$ ).

For  $\Delta t_n \rightarrow \infty$ , ( $n=t$ ) and ( $n+1=t+1$ ) the equations (4) degenerate to the classical Trapezoidal Formula of the Approximate Factorization scheme.

FIGURE 1 shows the main difference of the presented method to solve the unsteady flow equations in comparison to a classical time-accurate time-marching technique: the solution at time level  $t_1$  is found by advancing the solution at  $t_0$  in some physical time steps  $\Delta t_p$ , with controllable iteration procedures in between.<sup>3</sup> The physical time step can be chosen to achieve a reasonable resolution of the physical unsteadiness, whereas the numerical time step can be treated in any convenient way to advance the solution of level  $t$  to level  $t+\Delta t_p$  as fast as possible. Hence all current techniques that increase the convergence speed of a time-inaccurate time-marching procedure (i.e. in general the calculation of a steady flow) can be applied to minimize the computational effort for unsteady flow calculations.

<sup>1</sup>) Best at least for a two-level scheme.

<sup>2</sup>) The discretisation in numerical time corresponds to the backward Euler scheme, that is known to be an excellent choice for marching to a steady state.

<sup>3</sup>) In fact, an implicit "steady-state" time-marching method in the numerical time for each physical time level.

### 3. TEST OF THE SCHEME BY MEANS OF A LINEAR PROBLEM

The developed scheme can be tested with low expense of time and costs by applying it to the linear scalar convection-diffusion equation, for which also analytical solutions exist:

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} + v \frac{\partial c}{\partial y} = v_c \left[ \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right]. \quad (5)$$

$c$  can be considered as a kind of concentration,  $u_c$  and  $v_c$  are the convective transport speeds in  $x$ - and  $y$ -direction, and  $v_c$  is the diffusion coefficient.

In order to assess the numerical simulation the following analytical solution has been constructed:

$$c(x, y, t) = c_1(x, y) \cdot c_2(x, y, t) \quad (6)$$

where  $c_1$  is the amplitude and  $c_2$  a sinusoidal function in space and time!

For the test, the equation (5) has been discretized temporally according to scheme (4) and spatially upon the application of centered finite differences. The computational domain is a rectangle, represented by a uniform grid of IL times JH nodal points, with spacing  $\Delta x$  and  $\Delta y$ . It is shown together with the chosen data in FIGURE 2. The unsteadiness is introduced by unsteady boundary conditions at the left and lower boundary. The exact solution at each level  $t + \Delta t_p$  is imposed there during the corresponding iteration procedure between  $t$  and  $t + \Delta t_p$ . At the upper and right boundary the values of  $c$  are calculated explicitly at each iteration step after the implicit sweeps over the inner field, using the temporally and spatially discretized equation (5). The temporal discretisation is done according to equation (2), apart from the viscous terms (right hand side of (5)), which are taken at the old iteration level  $n \Delta t_n$  for simplicity reasons. This has an influence on the error behaviour as we will see later.

FIGURE 3 summarizes the obtained results for the calculation of one half of a sine wave in time. On a log-log scale the error, that has been integrated over the half cycle according to the L1-norm, is shown over the number of needed calculation steps. The calculations performed with the developed scheme are compared with the classical implicit Approximate Factorization method for three different frequencies. For the new scheme the physical time step has been chosen as a twentieth of the time of one cycle, which allows a reasonable resolution of the unsteadiness. The different number of calculation steps stem from a different number of iterations in between the physical time steps. As for the classical method, the curves were obtained by taking different CFL-numbers (s.a. eq. 13). The results show clearly that the needed computational time with the developed scheme is for all the three frequencies less in comparison with the classical method, especially in the range of reasonably low error levels. The lower the error level, the higher is the gain which is partially due to the finally dominant first-order error behaviour of

the classical scheme, since a first-order accurate discretisation in time has been used for the boundary conditions (viscous terms, as mentioned before). The new scheme is not concerned by that because this approach is here only used within the iteration procedures and by converging it drops out. This is in fact an additional advantage of the developed method, since the explicit treatment of the viscous terms is very often applied to avoid excessive programming efforts. Regarding the different frequencies, it can be seen that the lower the frequency, the higher the gain is. The minimum achievable error with the developed scheme depends of course on the size of the chosen physical time step (resolution error!). The general validity of the discussed results is restricted due to the linear feature of the test equation?

Finally it is noticed here, that for the classical method the error increases principally with the square of the time step size due to the second-order discretisation in time and the second-order error of the approximate factorization<sup>3</sup>. Because of the latter one, the absolute error level is increased. It should be kept in mind that the statement "z-order accurate" (z for 'first', 'second' etc.) alone is not sufficiently informative with respect to the absolute error level [4]. In [3, p.399] it is claimed for a simulation of a periodically unsteady Couette flow with the classical implicit Approximate Factorization technique that even for a CFL-number of 10 the results are in very good agreement with the analytical solution. However, it is not mentioned that due to the one-dimensional character of the test problem this test is not able to show the influence of any approximate factorization error. It can be stated that in order to obtain reasonably accurate results for simulations of spatially at least two-dimensional unsteady flow problems the CFL-number should not be taken much higher than one, if the classical Approximate Factorization scheme is applied. This is in agreement with [1].

<sup>1)</sup> Comparable with the propagation of two-dimensional Tollmien-Schlichting waves in a nongrowing boundary layer, as predicted by linear stability theory.

<sup>2)</sup> In real flows, where the effects of nonlinearity often play a dominant role, the classical scheme may lose accuracy because of the used linearization.

<sup>3)</sup> And the linearization error in time, of course.

#### 4. IMPLEMENTATION OF THE SCHEME IN THE EXISTING 2D-NAVIER-STOKES SOLVER "FINNIK"

The Navier-Stokes solver FINNIK, that has been developed during the last years at the von Karman Institute by M.Borsboom, uses the Approximate Factorization technique to solve the steady flow equations by means of an iterative time-marching procedure.

##### 4.1 Principal changes

The set of equations (4), derived in chapter two, can be written in a general form, valid both for steady and unsteady flow calculations:

$$\left[ I + \tilde{\Delta t} \{sp.op\}_x \right] \Delta \bar{U}^* = RHS \quad (7a)$$

$$\left[ I + \tilde{\Delta t} \{sp.op\}_y \right] \Delta \bar{U}^n = \Delta \bar{U}^* \quad (7b)$$

$$RHS = - \tilde{\Delta t} \left[ 2 \frac{\bar{U}^n - \bar{U}^t}{\Delta t_p} + \{sp.op\} \bar{U}^n + \alpha \{sp.op\} \bar{U}^t \right] \quad (7c)$$

with

$$\tilde{\Delta t} = \Delta t$$

$$\bar{U}^t = \bar{U}^n \quad (8)$$

$$\alpha = 0$$

for the implemented scheme in FINNIK, and

$$\tilde{\Delta t} = 0.5 \frac{\Delta t_n \Delta t_p}{\Delta t_n + \Delta t_p} = \Delta t_m \quad (9)$$

$$\bar{U}^t = \bar{U}^{n+1}$$

$$\alpha = 1$$

for the developed scheme.

The modifications for the sweeps over the inner field of the computational domain can now be summarized:

- a.  $\Delta t$  has to be replaced by  $\Delta t_m$  ;
- b. for each first iteration step within the iteration procedures the source term

$$-2 \frac{\bar{U}^t}{\Delta t_p} + \{sp.op\} \bar{U}^t$$

has to be calculated and added to the RHS of system (7/8) ;

- c. furthermore

$$2 \frac{\bar{U}^n}{\Delta t_p}$$

has to be calculated for every iteration step and to be added to the RHS of system (7/8).

#### 4.2 Spatial discretisation of the time derivatives

The spatial discretisation in the FINNIK code is based on the Finite Volume technique. Therefore the equations are transformed into their integral form:

$$\boxed{\int_V \frac{\partial \bar{U}}{\partial t} dV} + \int_V \frac{\partial \bar{F}}{\partial x} dV + \int_V \frac{\partial \bar{G}}{\partial y} dV - \int_V \frac{\partial (\bar{V}_1 + \bar{V}_2)}{\partial x} dV - \int_V \frac{\partial (\bar{W}_1 + \bar{W}_2)}{\partial y} dV = 0 \quad (10)$$

where  $\bar{F}$ ,  $\bar{G}$  respectively  $\bar{V}_1$ ,  $\bar{V}_2$ ,  $\bar{W}_1$ ,  $\bar{W}_2$  are defined vector functions that represent the convective respectively viscous fluxes [2,3]. The spatial derivatives are treated in the original code [4]. Hence, the time derivatives, marked by the dashed square, have to be evaluated additionally.

The finite volume approach gives:

$$\int_{VOL} \frac{\partial \bar{U}}{\partial t} dVOL = \frac{\Delta \bar{U}}{\Delta t} \Big|_C * VOL + \underbrace{\sigma(h) + \sigma(h^2) + \dots}_{\text{error terms}} \quad (11)$$

evaluated at point C  
in the finite volume VOL (FIGURE 4)

If the point C is identical to the Gauss quadrature point  $C^*$ [4], then the error terms of order  $h$  drop out and the approach is second-order accurate in space. The coordinates of point  $C^*$  can be found upon a Taylor-series expansion of the convective flux terms of VOL around an arbitrary point and a minimization of the error terms. This is done once for every volume VOL after the generation of the grid. The values of  $\bar{U}$  at  $C^*$  are then calculated by a biquadratic interpolation from the values at the nine nodal points of VOL.

#### 4.3 Boundary conditions

The characteristic approach is used [5, chapter 9]. That means for

- a. inlet/outlet:    - specification of three respectively one variable(s) for subsonic flow;  
 - the other variables are calculated by combinations of the discretized equations into the one respectively three characteristic informations that leave the computational domain;
- b. solid walls:    - no-slip condition,  $u=v=0$  ;  
 - adiab. or isothermal wall condition;  
 - combination of the two discretized momentum equations into the outgoing characteristic equation normal to the wall.

The main changes consist in the replacement of the implemented time-inaccurate equations by the discretisation (3). This is similar to the previously described modifications for the inner field of the computational domain.

## 5. SIMULATION OF THE 2D UNSTEADY LAMINAR FLOW BEHIND A CASCADE OF CIRCULAR CYLINDERS

This type of flow allows to assess the capability of the out-worked program pack with respect to imposed spatial periodicity (turbomachinery cascades) and the ability to capture self-induced unsteadiness. Additionally, it is possible to compare the results, although to a limited extent, with calculations of the flow around a single cylinder [6...10].

FIGURE 5 shows the computational domain together with the imposed physical conditions. A uniform velocity profile was specified at the inlet, with the nondimensionalized velocity components  $u=1$  and  $v=0$ . This corresponds, together with a fixed static temperature  $T=1$ , at the inlet, to an inlet Mach number of .0845 and leads to a maximum Mach number of less than .3 in the whole flow field. Hence, the flow is approximately incompressible. At the outlet, a constant static pressure  $p=1$  was specified what is actually not in agreement with the real flow conditions there. But since the outlet boundary is placed at a distance of seven cylinder diameters behind the cylinders, this approach should have only a minor influence on the vortex shedding. The Reynolds number, based on the inlet velocity and the diameter of the cylinders, is hundred. The used grid is shown in FIGURE 6. It measures 29 times 81 nodal points in the widthwise respectively lengthwise direction and is refined in the vicinity of the solid walls and the expected separation zones!

The frequency of the self-induced vortex shedding is given by [1]:

$$f = So * \frac{u}{d} \quad , \quad (12)$$

where  $So$  is the Strouhal number,  $u$  a typical velocity (undisturbed upstream velocity) and  $d$  the diameter of the cylinder. For the flow around a single cylinder at  $Re=100$ , the Strouhal number is about .17 and so the nondimensionalized period of one cycle  $t_o = 120$ . For the considered cascade of cylinders the typical velocity is approximately twice as high due to the contraction of the flow in between the cylinders to the half of the inlet area. This yields  $t_o = 60$  and a still reasonable resolution in time is obtained by taking  $\Delta t_p = 6$ . In the case of the classical method, the time step is limited by

$$\Delta t = CFL * \frac{\min(\Delta x_{i,j}; \Delta y_{i,j})}{(u + a)} \quad . \quad (13)$$

<sup>1</sup>) The chosen grid resolves the boundary-layer thickness at the points of maximum velocity on the cylinder surface with about 8 to 10 points.

Because of already mentioned reasons the CFL-number should not be taken much higher than one, what results in  $\Delta t = .016$  for the used grid. From there, the ratio of the time steps is

$$\frac{\Delta t_p}{\Delta t} = 375 \quad . \quad (14)$$

As the first part of the results the convergence history of the performed calculations is shown in FIGURE 7. The maximum error is drawn there over the number of calculation steps. The attempt to calculate a steady state (time inaccurate calculations), starting from an unphysical, symmetrical initial flow field, ended in a "blown up" state. This asymmetrical configuration (FIGURE 8b) was obtained after 400 iterations, with a symmetrical pseudo steady state flow field after 230 steps (FIGURE 8a). It is emphasized that no perturbation had to be introduced to trigger the vortex shedding that can be seen already in FIGURE 8b. One vortex is separating there from the recirculation bubble in the upper part of the wake, and another one, shed by the lower bubble before, is moving towards the outlet boundary. This flow pattern was taken as the initial flow field to start the time-accurate calculations. An astonishingly fast convergence turned out, with the error reaching machine accuracy ( $10^{-6}$ ) after about 75 iteration steps. This results on the basis of the derived time step ratio (14) in a remarkable gain of factor five in comparison to the classical method.

The picture sequence FIGURES 9a...11b indicates clearly the periodic process of the vortex shedding. From a comparison of the flow fields at  $t=60$  and  $t=120$  it can be seen that an amplification still takes place. The simulation was however stopped after two cycles.

In comparison to the flow field around a single cylinder, the wake close behind the cylinders is compressed in the widthwise direction, shedding vortices with the expected double frequency.

The obtained results are limited in their generality by the already mentioned spurious condition of a uniform static pressure at the outlet and the imposed spatial periodicity of the flow in the widthwise direction, whose validity is only confirmed for a steady state flow.

## 6. SUMMARY

The main drawback in solving the unsteady Navier-Stokes equations by means of a classical time-accurate time-marching technique consists in the severe limitation of the usable time step due to stability reasons. To overcome that, a fully implicit, second-order accurate scheme for unsteady flow calculations has been developed that necessitates however the solution of a large nonlinear system of algebraic equations. This system has been solved by the Approximate Factorization scheme by means of uncoupling the numerics from the physics via the introduction of a second time derivative. So the stability limit on the time step was removed. Hence, it is now possible to minimize the computational effort for solving unsteady flow problems by optimizing the convergence speed of the scheme, being able to use any available method for that. The method is especially advantageous in cases where the time step imposed by the numerics is much smaller than the time step that is required to resolve the physical unsteadiness of the flow problem.

For a first test, the new scheme was applied to a linear scalar equation. A comparison of the results with the ones obtained by using the classical implicit Approximate Factorization scheme showed a remarkable gain with respect to the needed computational time. As expected, the lower the frequency of the unsteadiness, the greater the gain was found.

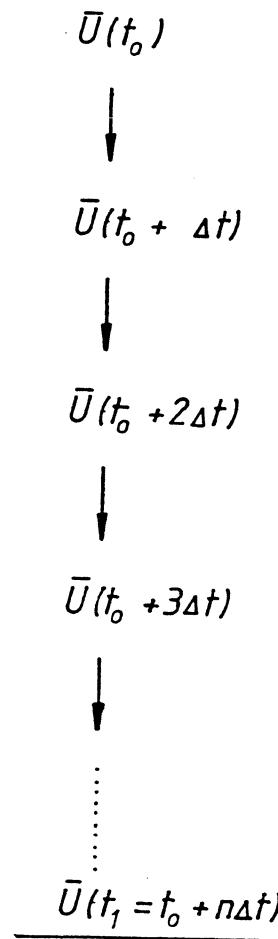
The scheme was implemented in the existing 2D-Navier-Stokes solver "FINNIK" (VKI). The chosen calculation example, the two-dimensional unsteady laminar flow behind a cascade of circular cylinders, confirmed the capability of the developed scheme. The needed computational time was found distinctly less in comparison to the classical scheme. An astonishingly fast convergence was observed, what suggests to apply a similar method to steady state calculations.

A further detailed research of the combined time step that is inherent in the new scheme is recommended as well as an improved tuning of all important features influencing the accuracy and convergence.

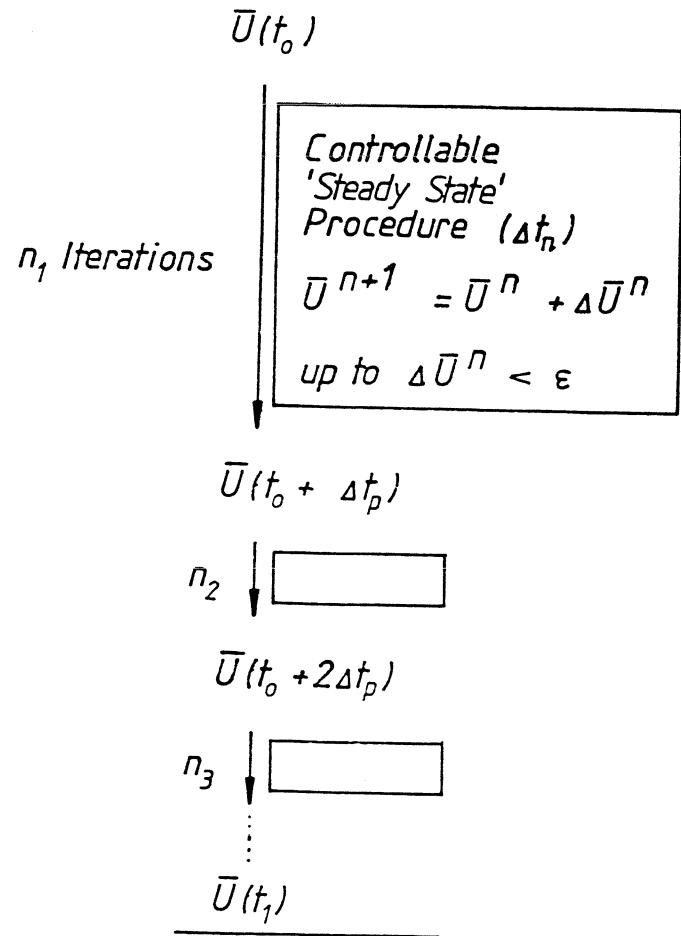
## 7. REFERENCES

- [1] MCCROSKEY,W.J.: Unsteady aerodynamics.  
Von Karman Institute, special lecture, April 1986
- [2] BEAM,R.M. & WARMING,R.F.: Implicit numerical methods for the compressible Navier-Stokes and Euler equations.  
Von Karman Institute, LS 1982-04:  
"Computational Fluid Dynamics", March 1982
- [3] BEAM,R.M. & WARMING,R.F.: An implicit factored scheme for the compressible Navier-Stokes equations.  
AIAA J., Vol. 16, No 4, April 1978, pp 393-401
- [4] BORSBOOM,M.: Numerical modelling of Navier-Stokes equations.  
Von Karman Institute, LS 1986-02:  
"Numerical Techniques For Viscous Flow Calculations In Turbomachinery Bladings", January 1986
- [5] PULLIAM,T.H.: Efficient solution methods for the Navier-Stokes equations.  
Von Karman Institute, LS 1986-02:  
"Numerical Techniques For Viscous Flow Calculations In Turbomachinery Bladings", January 1986
- [6] LECOINTE,Y. & PIQUET,J.: On the use of several compact methods for the study of unsteady incompressible viscous flow around a circular cylinder.  
Computers and Fluids, Vol. 12, No 4, 1984, pp 255-280
- [7] JAIN,P. & RAO,K.: Numerical solution of unsteady incompressible fluid flow past a circular cylinder.  
Physics of Fluids, supp. II, 1969, pp 57-64
- [8] LIN,L. & PEPPER,D.W. & LEE,S.C.: Numerical methods for separated flow solutions around a circular cylinder.  
AIAA J., Vol. 14, 1976, pp 900-907
- [9] JAIN,P.C. & GOEL,B.S.: Shedding of vortices behind a circular cylinder.  
Computers and Fluids, Vol. 4, 1976, pp 137-142
- [10] PATEL,V.A.: Karman vortex street behind a circular cylinder calculated by the series truncation method.  
Journal of Comp. Physics, Vol. 28, 1978, pp 14-42

CONVENTIONAL  
EXPLICIT OR  
LINEARIZED NONITERATIVE  
IMPLICIT PROCEDURE



PROPOSED  
IMPLICIT PROCEDURE



CALCULATION EFFORT:

$\sim n$

$\sim \sum_{\Delta t_p} n \Delta t_p$

- $\Delta t$  limited by numerical accuracy bound (linearization, factorization error)
- $\Delta t_p$  only limited by physical problem to be solved (resolution)
- $\Delta t_n$  optimizable

FIGURE 1 SURVEY: CONVENTIONAL AND PROPOSED TIME-MARCHING PROCEDURE

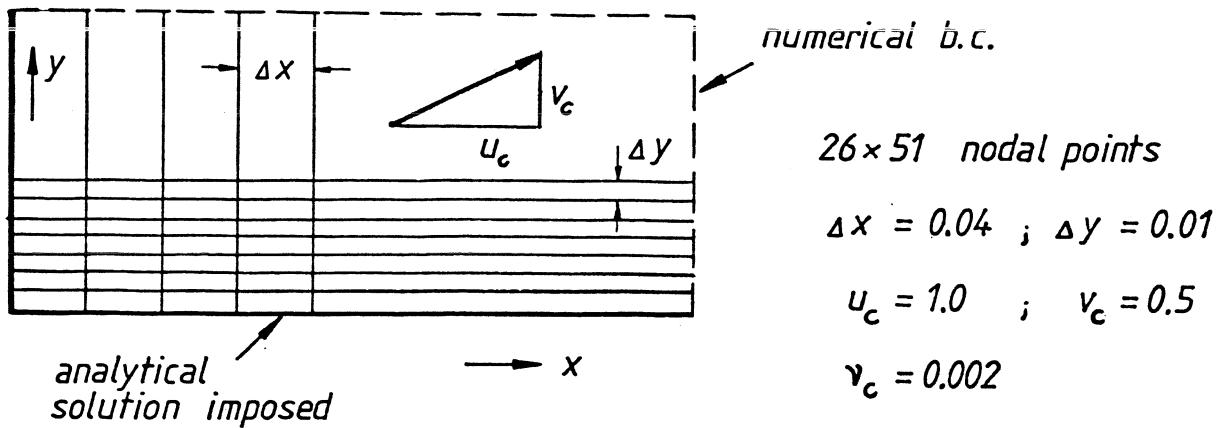


FIGURE 2 COMPUTATIONAL DOMAIN AND DATA FOR THE TEST PROBLEM

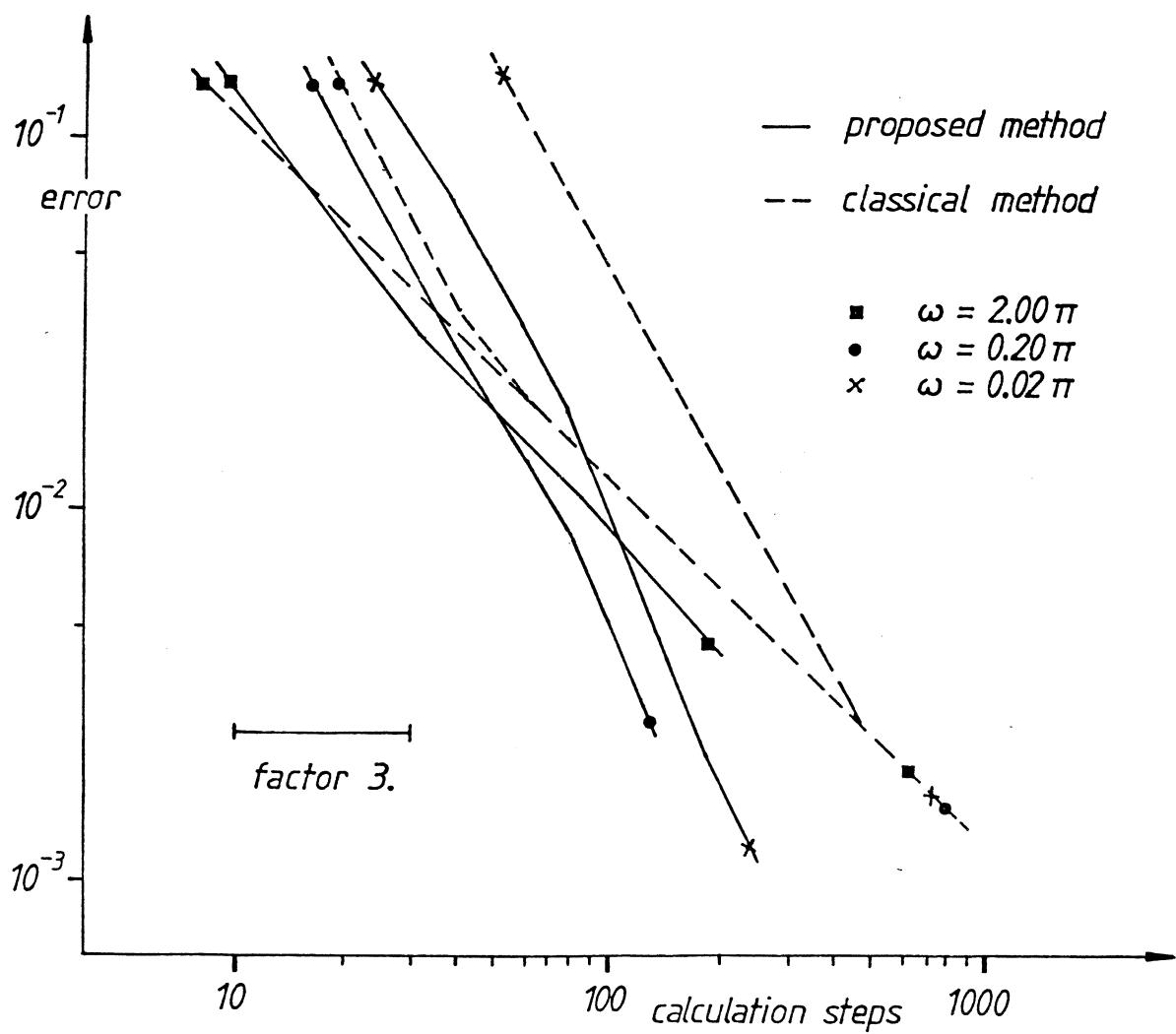


FIGURE 3 COMPARISON OF THE CLASSICAL AND PROPOSED METHOD:  
ERROR BEHAVIOUR FOR THE LINEAR TEST PROBLEM

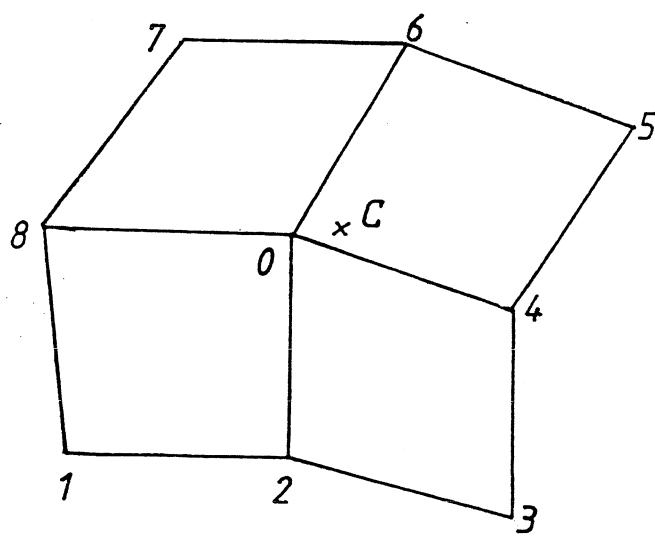


FIGURE 4 COMPUTATIONAL MOLECULE USED IN THE 'FINNIK' CODE

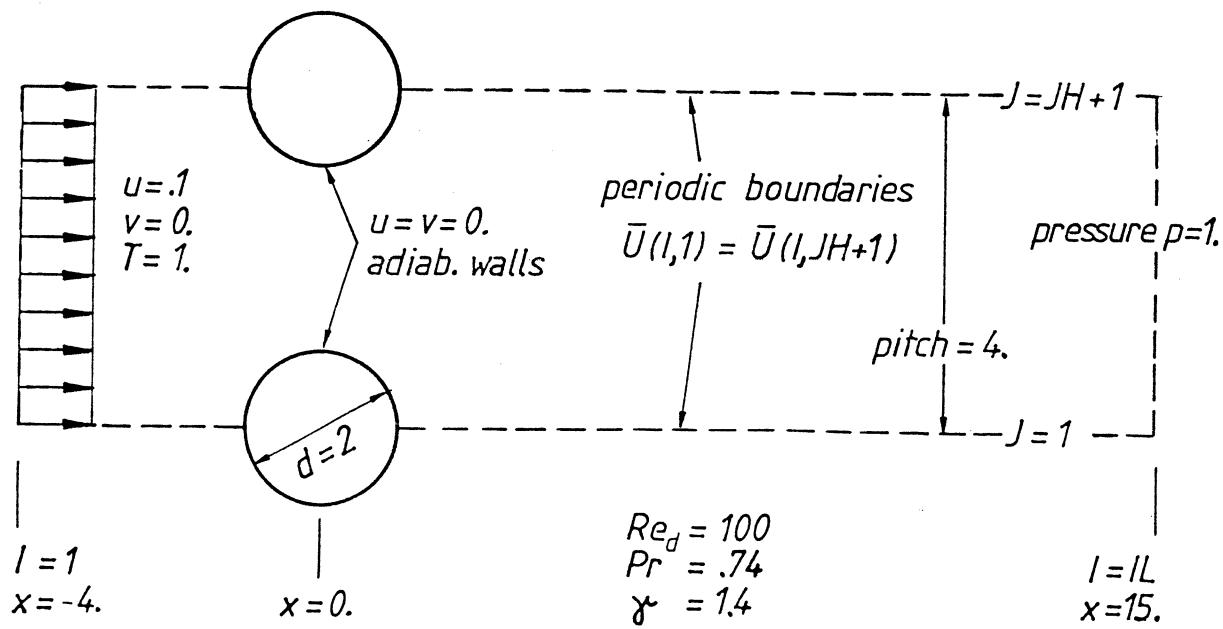
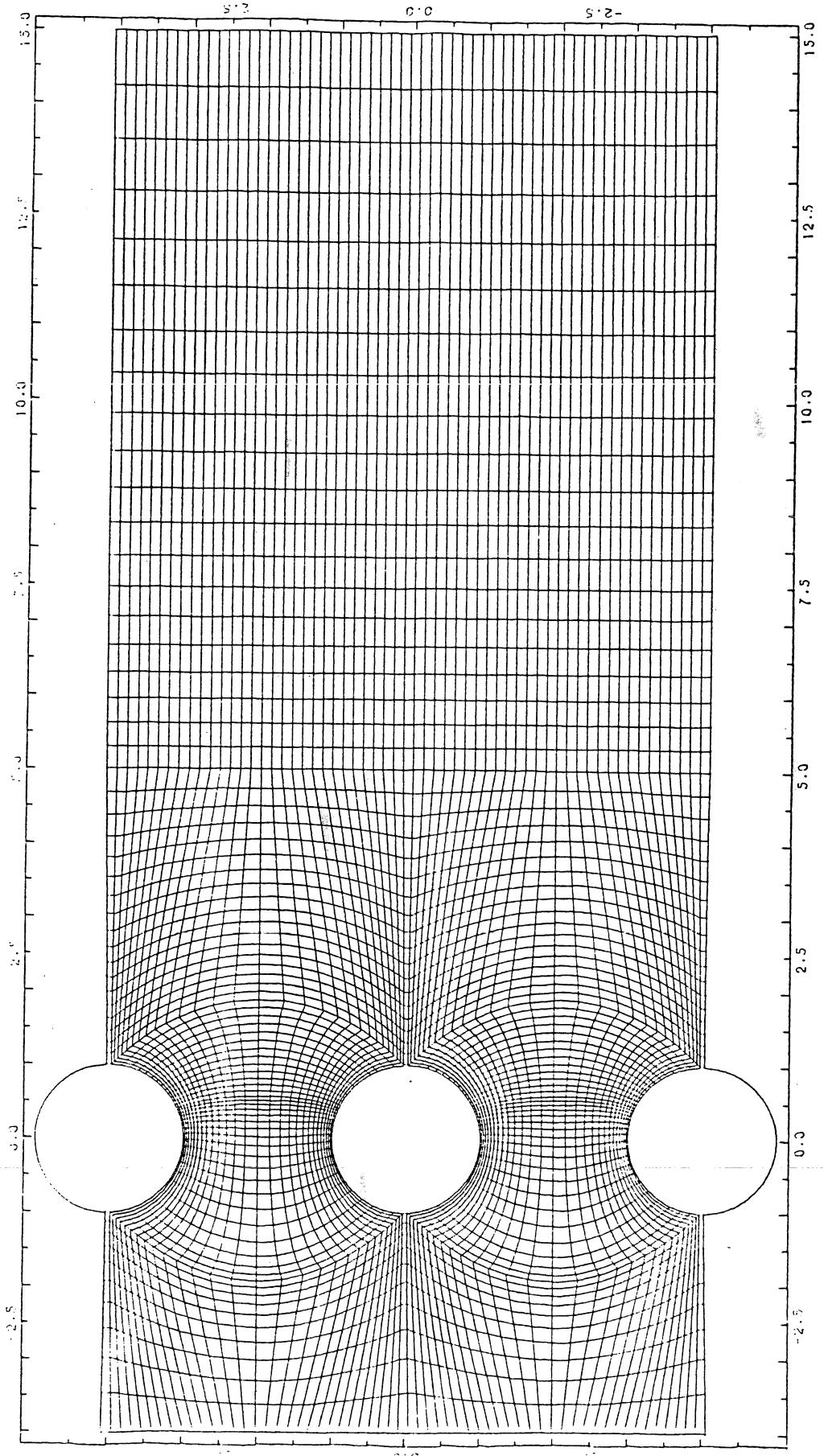


FIGURE 5 COMPUTATIONAL DOMAIN AND PHYSICAL CONDITIONS FOR THE CASCADE FLOW PROBLEM



**FIGURE 6** 29x81-GRID FOR THE CASCADE FLOW SIMULATION

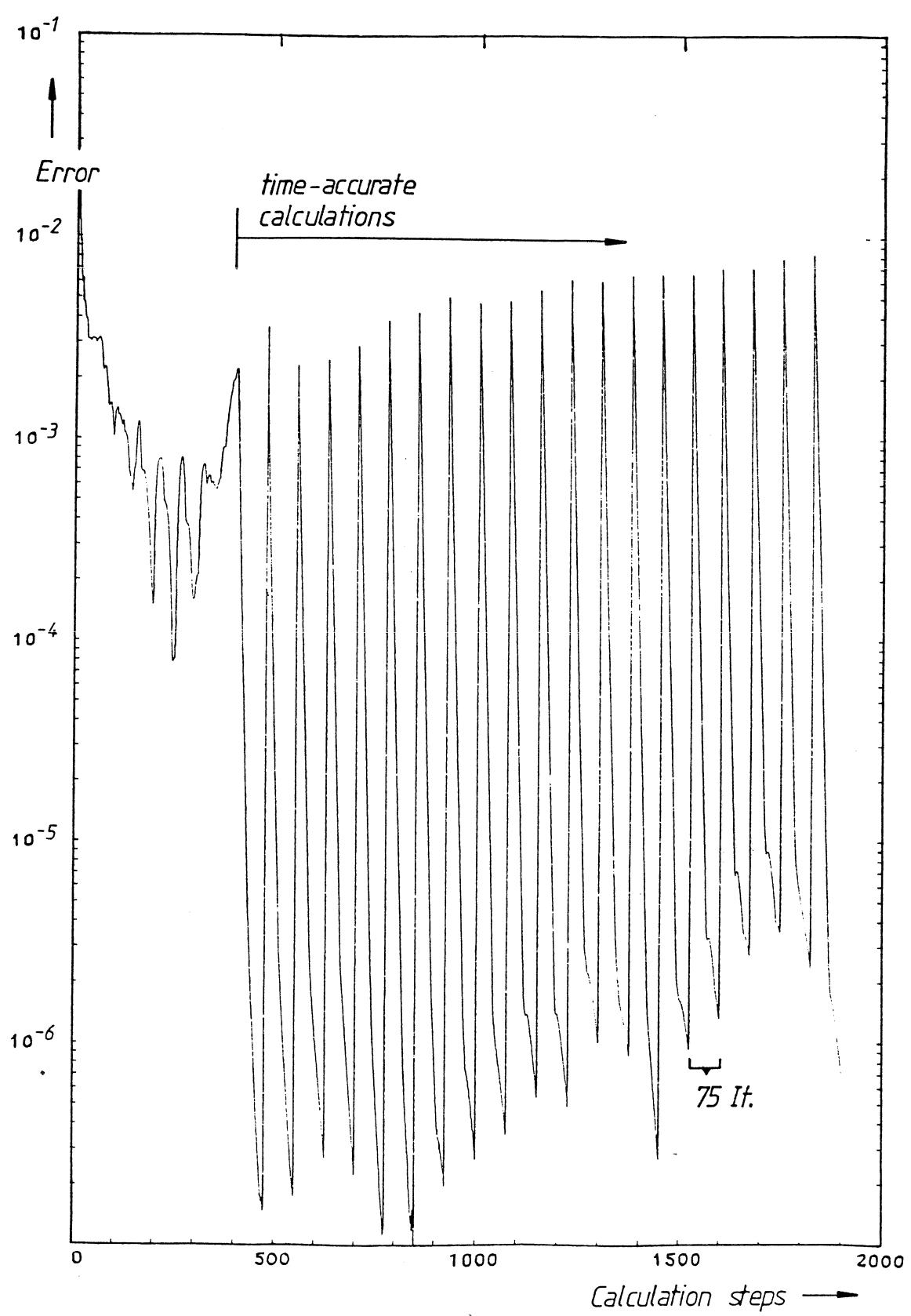


FIGURE 7 CONVERGENCE HISTORY OF THE CASCADE FLOW CALCULATIONS

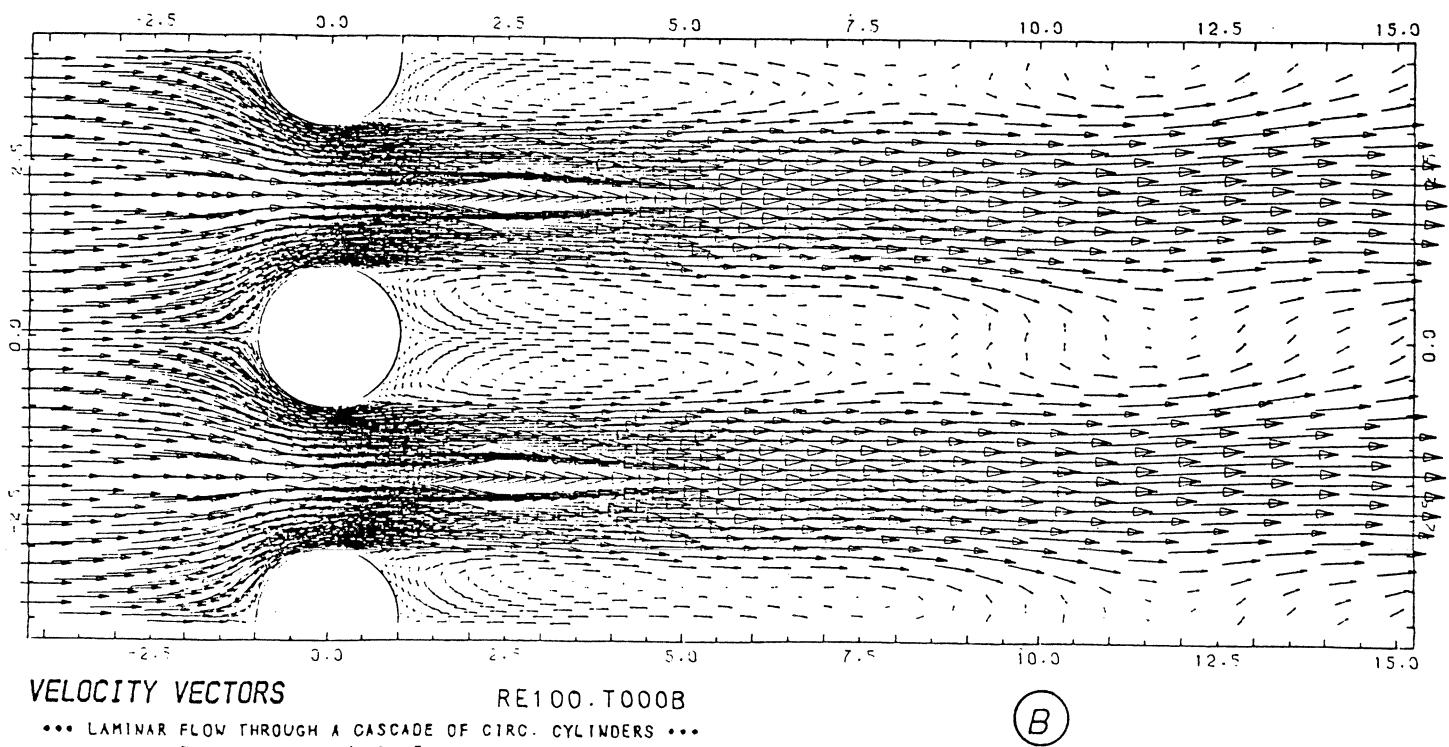
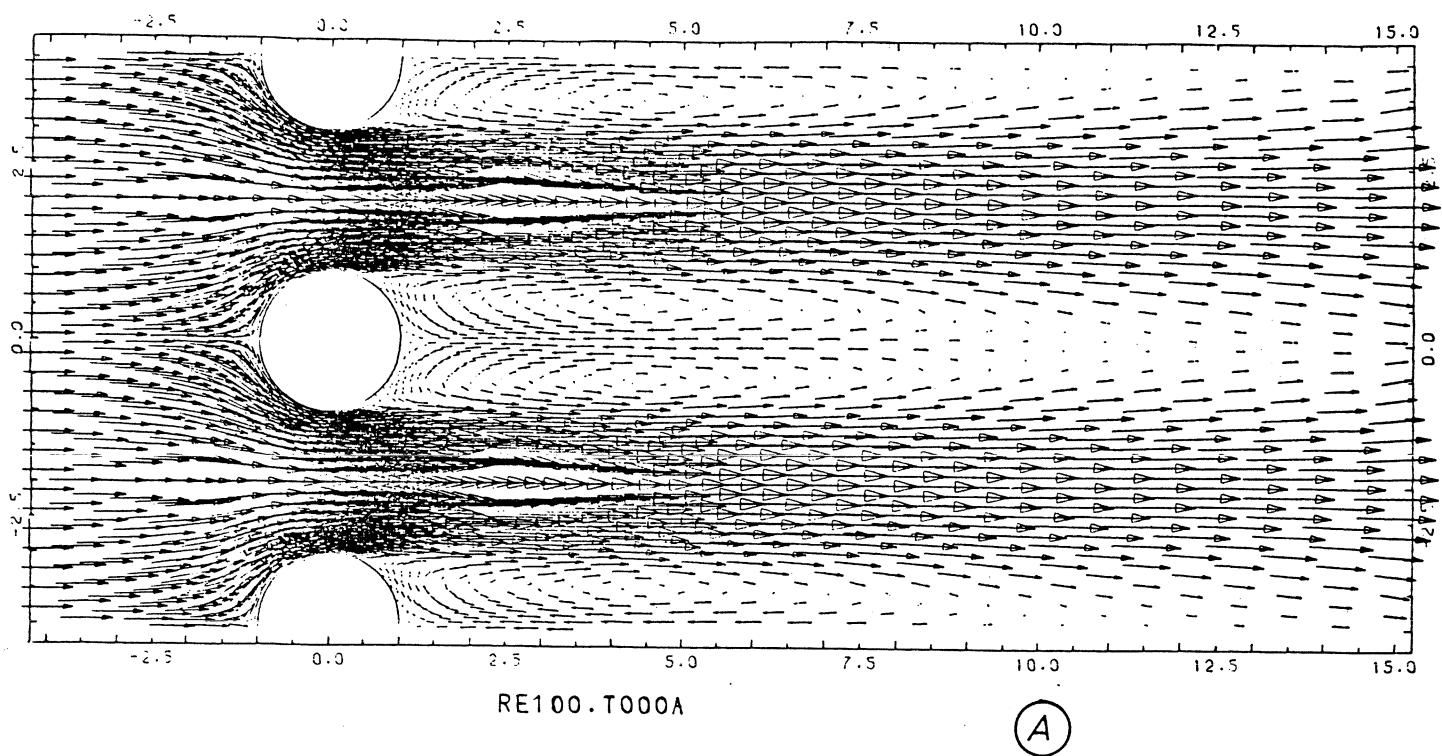
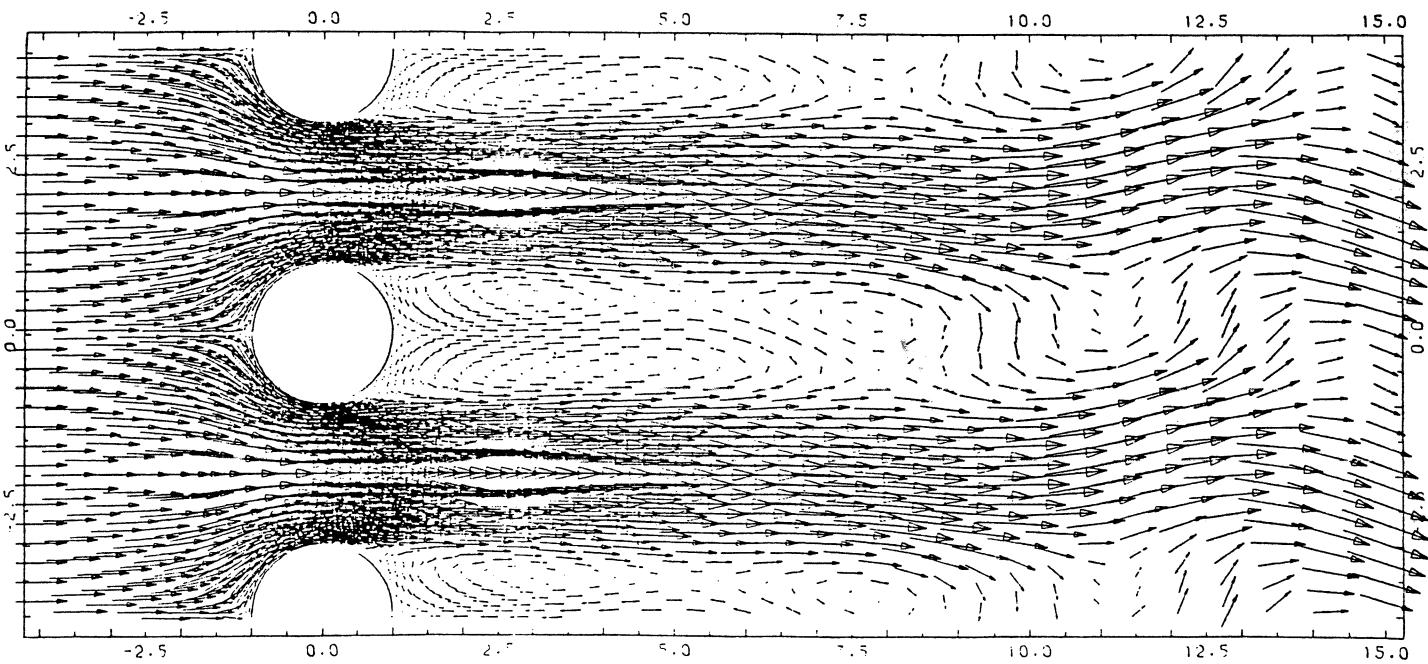
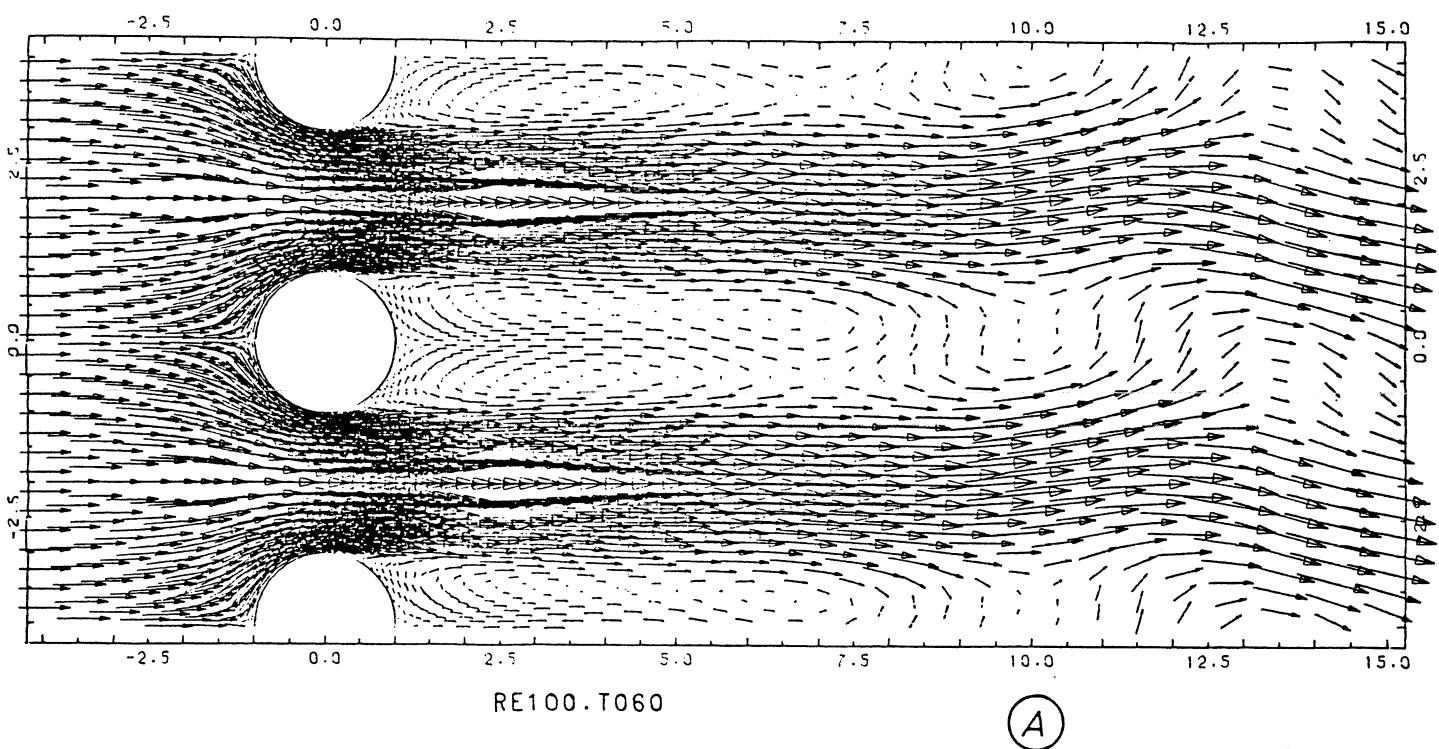
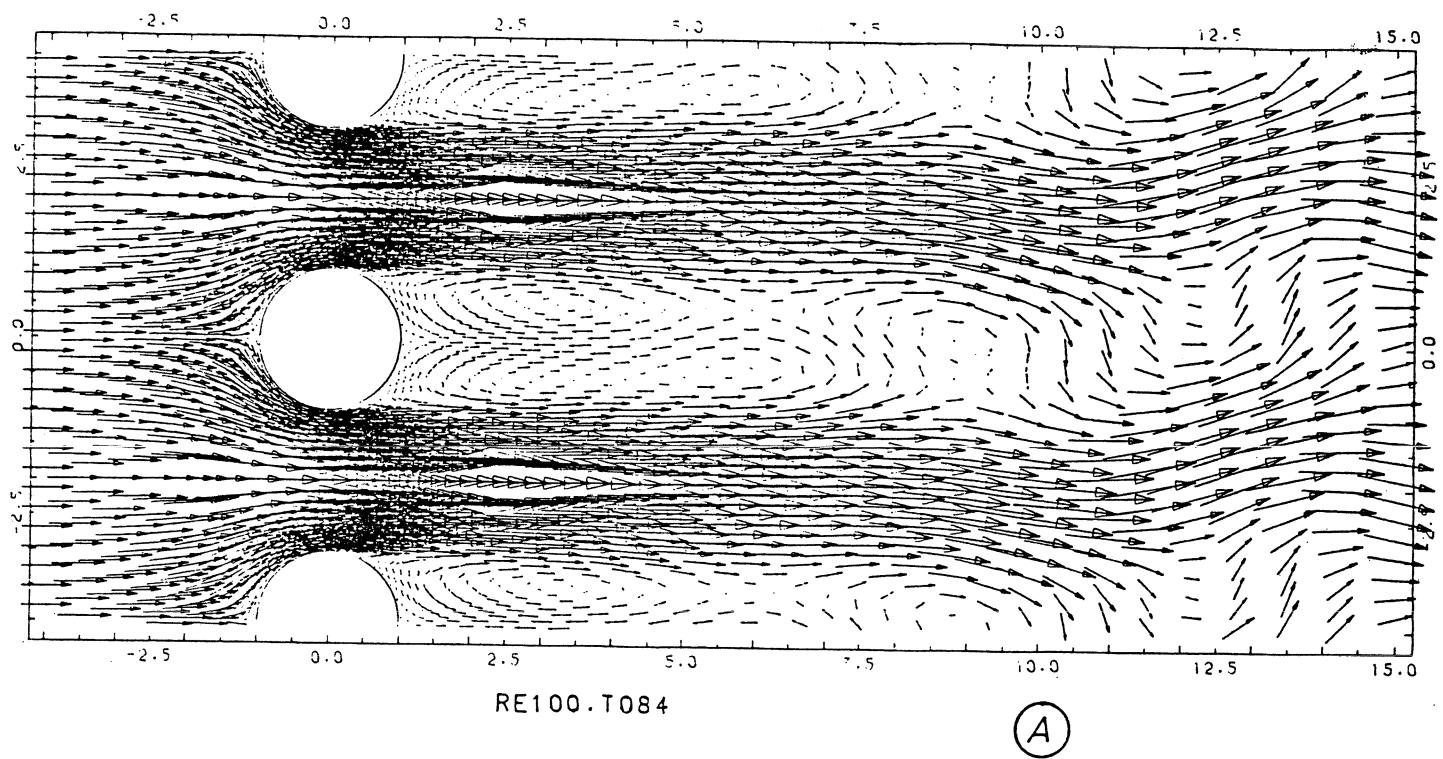


FIGURE 8(A,B) FLOW FIELD AFTER 230 AND 400 ITERATIONS  
( TIME-INACCURATE CALCULATIONS )

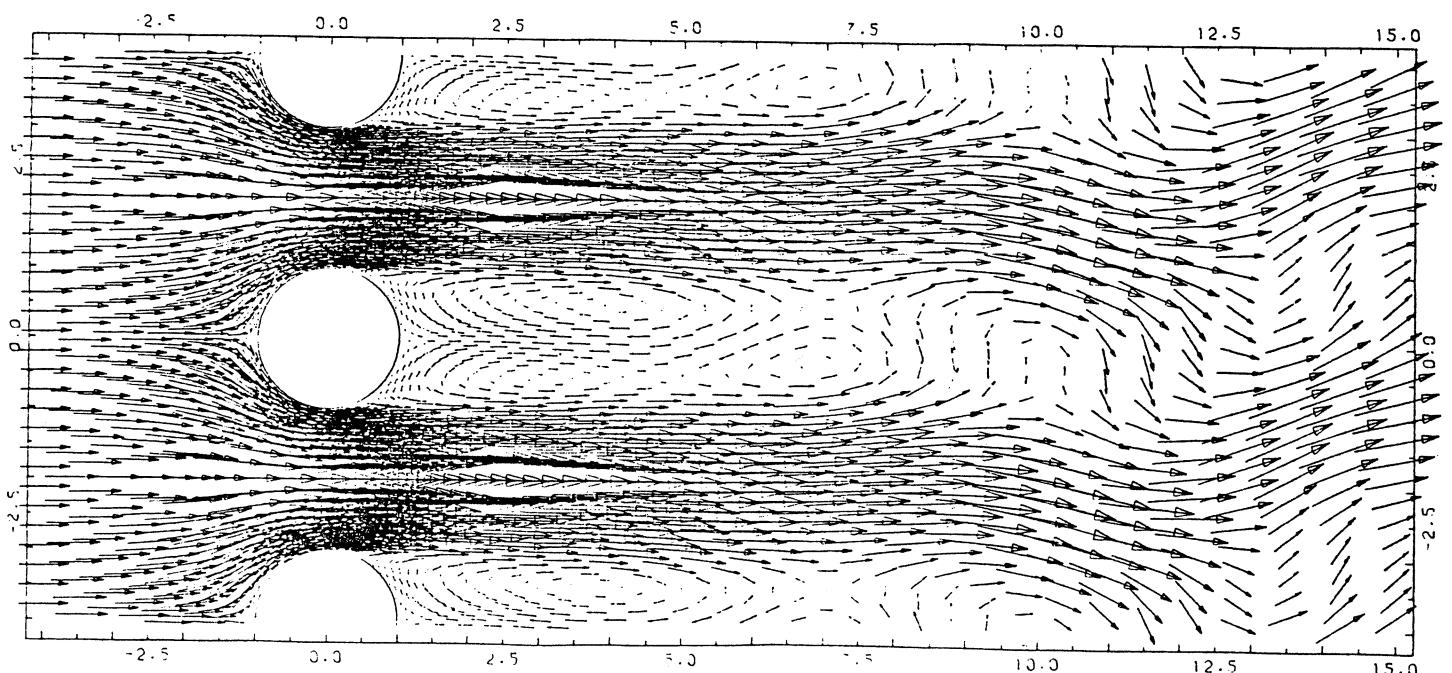


VELOCITY VECTORS  
 ... LAMINAR FLOW THROUGH A CASCADE OF CIRC. CYLINDERS ...  
 TU-DEPARTMENT / PROJECT BOKA-MK '86

FIGURE 9(A,B) FLOW FIELD AT  $t=60$  AND  $t=72$



(A)



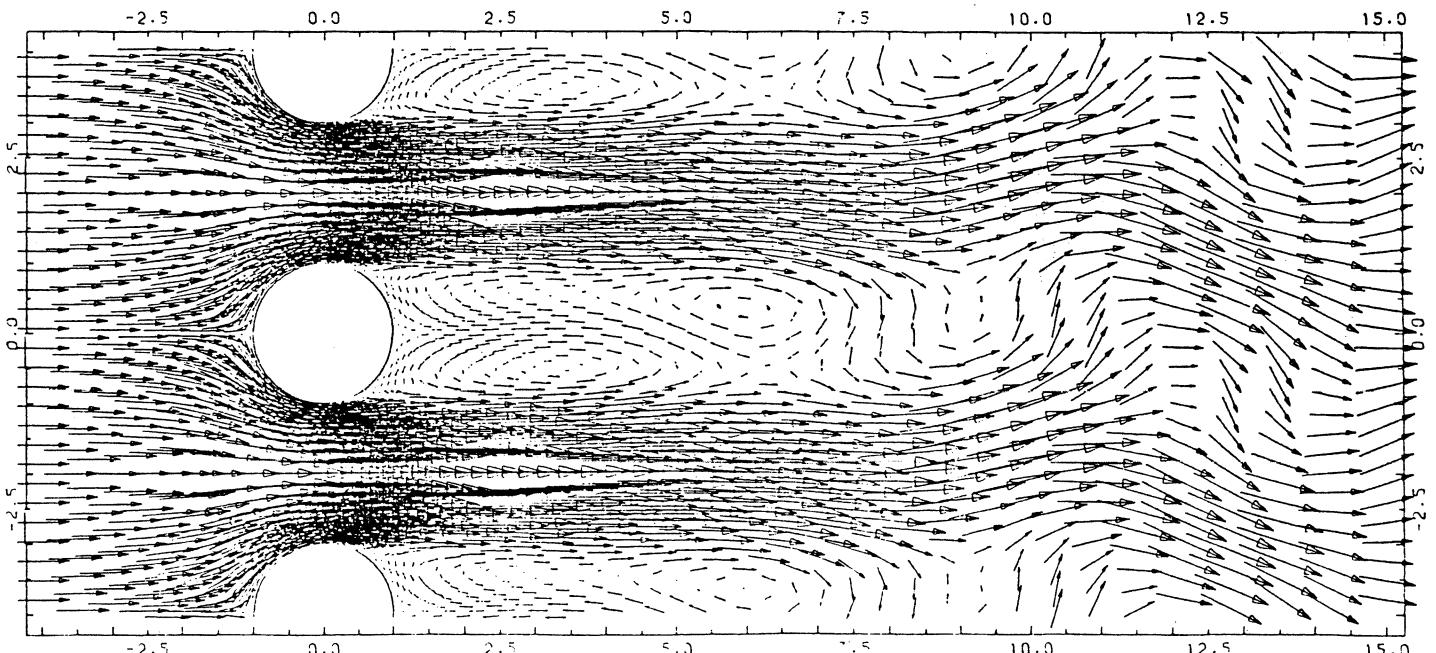
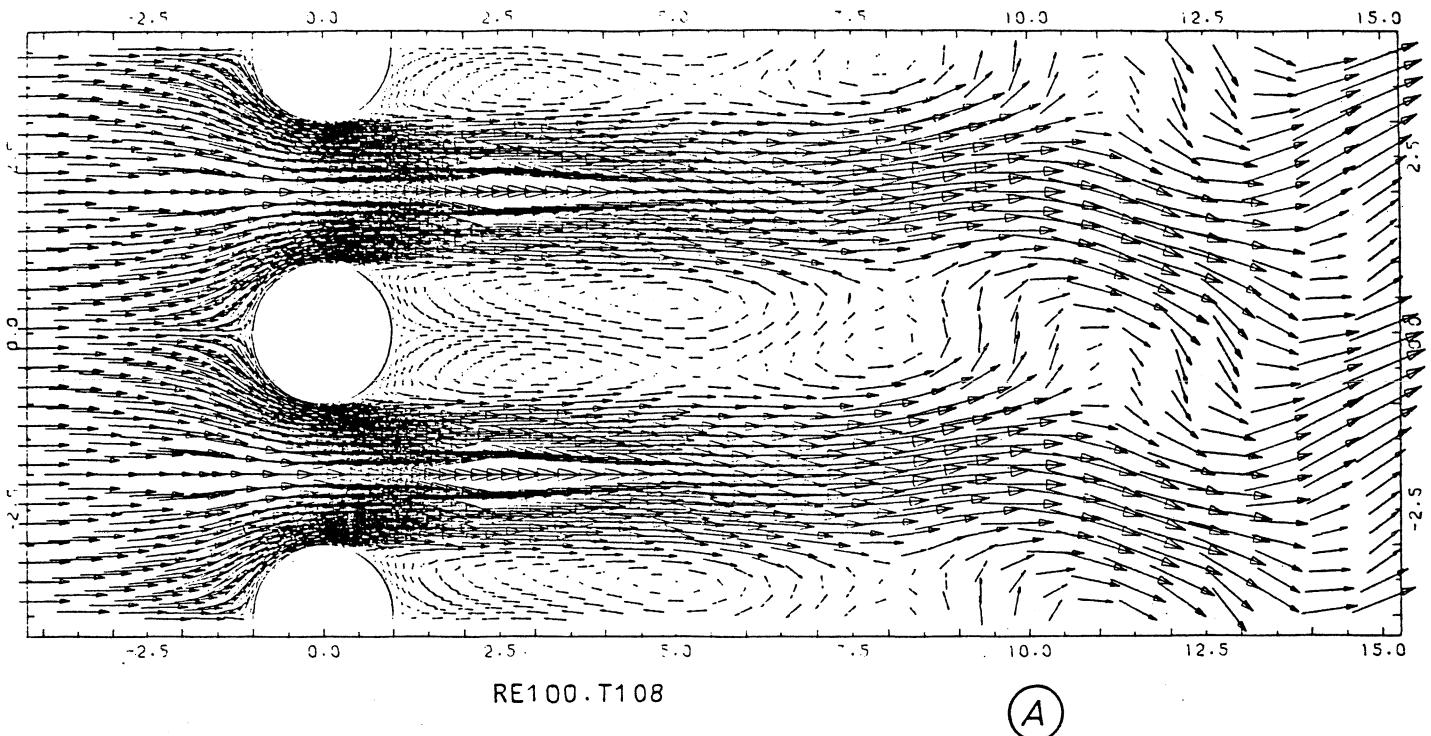
(B)

VELOCITY VECTORS

RE100.T096

... LAMINAR FLOW THROUGH A CASCADE OF CIRC. CYLINDERS ...  
TU-DEPARTMENT / PROJECT BOKA-MK '86

FIGURE 10(A,B) FLOW FIELD AT  $t=84$  AND  $t=96$



**VELOCITY VECTORS**  
**RE100.T120**  
 ••• LAMINAR FLOW THROUGH A CASCADE OF CIRC. CYLINDERS •••  
 TU-DEPARTMENT / PROJECT BOKA-MK '86

**FIGURE 11(A,B) FLOW FIELD AT  $t=108$  AND  $t=120$**

