A New and Improved Computational Technique for Two-Dimensional, Unsteady, Compressible Flows

Gino Moretti*
GMAF, Inc., Freeport, New York
and
Luca Zannetti†
Politecnico di Torino, Torino, Italy

An integration scheme for two-dimensional, time-dependent, inviscid Euler equations extends the concept of Riemann variables whose efficiency is acknowledged for one-dimensional, transonic flow problems to two-dimensional problems. The Riemann variables are obtained from a vector combination of velocity and sound speed. Space derivatives are approximated by two-point differences. Each term contributing to updatings is assigned its own domain of dependence. The technique is very accurate and simplifies the treatment of boundary conditions. Numerous examples are given and discussed.

Introduction

In recent years, strong emphasis has been put on the concept of characteristics in creating integration schemes for Euler equations and, occasionally, Navier-Stokes equations. Some of the schemes are based on the equations of motion in conservation form[1-4] and some are based on nonconservative forms.[5,6] The latter have been applied successfully to two-dimensional, time-dependent flows and three-dimensional, steady, supersonic flows.[7-9] Nevertheless, we consider it proper to recast them in a different form in order to give them a stronger theoretical background, increase their accuracy, simplify the calculation at boundary points, and provide a more efficient extension to flows with imbedded shocks.

For the first goal we will use the concept of Riemann variables as first proposed in Ref. 10 and later used in Ref. 11. In these papers, the extension is limited to isentropic flows; the restriction, however, can be lifted easily. We will maintain it in the present paper for the sake of simplicity.

As far as efficiency, accuracy, and simplicity of coding are concerned, we may note that in all works reported in Refs. 5 and 8-11, two-level schemes were used; at one of the levels one-sided derivatives were approximated by a three-point operator. In Ref. 6, however, second-order accuracy (including nonlinear effects) was reached using two-point operators at both levels. We considered the scheme very appealing for two reasons: the code is more symmetric and concise, and it decreases the labor at boundary points (when three-point operators are used, not only the boundary points, but the row of points next to the boundary must be treated in a special way, as shown, for example, in Ref. 11).

It has been pointed out[12] that, for one-dimensional problems, a slight modification of the Chinese scheme could improve its accuracy further, and make the code even more symmetric. The aim of the present paper is to show that a similar approach can be used in two-dimensional time-dependent problems.

Presented as Paper 82-0168 at the AIAA 20th Aerospace Sciences Meeting, Orlando, Fla., Jan. 11-14, 1982; submitted Jan. 22, 1982; revision received Aug. 9, 1983.

*Consultant. Fellow AIAA.
†Associate Professor.

One-Dimensional Problems

The genesis and meaning of the two-dimensional code are understood better after studying the one-dimensional approach in detail. Because of space limitations, the reader is referred to the first nine pages of Ref. 13. The definitions of the terms denoted by f_i in Eqs. (12), their use in the equations of motion [Eqs. (13)], and their manipulation in the two-level scheme [Eqs. (14-23)], are the necessary background for the two-dimensional extension. In particular, it has to be noted that in both stages the indices j and m, which specify the domains of dependence of f_i in Eqs. (19) and (23), are defined *separately and independently* for each f_i.

The important feature of the method is that quantities such as the speed of sound a and the velocity u which do not have a one-sided domain of dependence, such as the Riemann variables R_1 and R_2, can be updated with an unequivocal choice of building elements, f_i, both at the predictor and corrector levels.

As far as boundary points are concerned, enforcement of the boundary conditions is straightforward and is briefly discussed in Ref. 13 (pp. 28 and 29). More on the philosophy of boundary point calculation can be found in Ref. 14.

Two-Dimensional Problems

In dealing with two-dimensional problems, a straightforward unambiguous definition of characteristics is no longer possible. We will attempt a definition of new variables to generalize the application of the concept of characteristics. Let the physical plane be defined by Cartesian coordinates x and y. Let $\boldsymbol{n}$ be a unit vector, generally variable in space, which forms an angle α_0 with a fixed direction. Let $\boldsymbol{k}$ be a unit vector normal to the x,y plane, and $\boldsymbol{\tau}=\boldsymbol{k}\times\boldsymbol{n}$. Therefore, for any derivative,

$$\boldsymbol{n}'=\alpha_0'\boldsymbol{\tau}$$

Let $\boldsymbol{q}$ be the velocity vector and $\boldsymbol{w}$ any vector, a the speed of sound, γ the ratio of specific heats, and $\delta=(\gamma-1)/2$. The following vector identities are proven easily,

$$\nabla\cdot\boldsymbol{q}=\boldsymbol{n}\cdot\nabla(\boldsymbol{n}\cdot\boldsymbol{q})+\boldsymbol{\tau}\cdot\nabla(\boldsymbol{\tau}\cdot\boldsymbol{q})+\boldsymbol{k}\times\boldsymbol{q}\cdot\nabla\alpha_0 \tag{1}$$

$$\boldsymbol{w}\cdot(\boldsymbol{q}\cdot\nabla)\boldsymbol{q}=\boldsymbol{q}\cdot\nabla(\boldsymbol{w}\cdot\boldsymbol{q})-\boldsymbol{q}\cdot[(\boldsymbol{q}\cdot\nabla)\boldsymbol{w}] \tag{2}$$

$$(q\cdot\nabla)n=\tau(q\cdot\nabla\alpha_0),\quad (q\cdot\nabla)\tau=-n(q\cdot\nabla\alpha_0) \tag{3}$$

We will limit the present study to the case of isentropic flows. Thus, the Euler equations can be written in the form

$$(a_t/\delta)+q\cdot\nabla(a/\delta)+a\nabla\cdot q=0$$

$$q_t+(q\cdot\nabla)q+a\nabla(a/\delta)=0 \tag{4}$$

If the second equation is dot-multiplied by a unit vector, w, and added to the first, a single equation is obtained:

$$(a_t/\delta)+w\cdot q_t+q\cdot\nabla(a/\delta)+w\cdot(q\cdot\nabla)q$$

$$+a[w\cdot\nabla(a/\delta)+\nabla\cdot q]=0 \tag{5}$$

which, using Eqs. (1) and (2), can be written in the form:

$$(a_t/\delta)+w\cdot q_t+q\cdot\nabla[(a/\delta)+w\cdot q]+aw\cdot\nabla(a/\delta)$$

$$+an\cdot\nabla(n\cdot q)+a\tau\cdot\nabla(\tau\cdot q)$$

$$-q\cdot[(q\cdot\nabla)w]+ak\times q\cdot\nabla\alpha_0=0 \tag{6}$$

In the spirit of Refs. 10 and 11, four equations are obtained by letting $w=n,-n,\ \tau$, and $-\tau$ successively. Using the notation

$$\rho_1=(a/\delta)+n\cdot q,\qquad \Lambda_1=q+an$$

$$\rho_2=(a/\delta)-n\cdot q,\qquad \Lambda_2=q-an$$

$$\rho_3=(a/\delta)+\tau\cdot q,\qquad \Lambda_3=q+a\tau$$

$$\rho_4=(a/\delta)-\tau\cdot q,\qquad \Lambda_4=q-a\tau$$

$$\beta=q\cdot\nabla\alpha_0,\qquad F=ak\times q\cdot\nabla\alpha_0 \tag{7}$$

the equations are

$$(a_t/\delta)+n\cdot q_t+\Lambda_1\cdot\nabla\rho_1+a\tau\cdot\nabla(\tau\cdot q)-\beta q\cdot\tau+F=0$$

$$(a_t/\delta)-n\cdot q_t+\Lambda_2\cdot\nabla\rho_2+a\tau\cdot\nabla(\tau\cdot q)+\beta q\cdot\tau+F=0$$

$$(a_t/\delta)+\tau\cdot q_t+\Lambda_3\cdot\nabla\rho_3+an\cdot\nabla(n\cdot q)+\beta q\cdot n+F=0$$

$$(a_t/\delta)-\tau\cdot q_t+\Lambda_4\cdot\nabla\rho_4+an\cdot\nabla(n\cdot q)-\beta q\cdot n+F=0 \tag{8}$$

By adding the four equations, we obtain

$$(4/\delta)a_t+\Sigma\Lambda_1\cdot\nabla\rho_1+2a[\tau\cdot\nabla(\tau\cdot q+n\cdot\nabla(n\cdot q)]+4F=0$$

or, if Eq. (1) is used,

$$(4/\delta)a_t+\Sigma\Lambda_1\cdot\nabla\rho_1+2a\nabla\cdot q+2F=0$$

Finally, the equation is simplified, taking the first of Eqs. (4) into account.

$$(2/\delta)a_t+\Sigma\Lambda_1\cdot\nabla\rho_1-(2/\delta)q\cdot\nabla a+2F=0 \tag{9}$$

By subtracting the second of Eqs. (8) from the first, we obtain

$$2n\cdot q_t+\Lambda_1\cdot\nabla\rho_1-\Lambda_2\cdot\nabla\rho_2-2\beta q\cdot\tau=0 \tag{10}$$

and, similarly, by subtracting the fourth of Eqs. (8) from the third,

$$2\tau\cdot q_t+\Lambda_3\cdot\nabla\rho_3-\Lambda_4\cdot\nabla\rho_4+2\beta q\cdot n=0 \tag{11}$$

We will use Eqs. (9-11) as our basic equations of motion.

Different scalar equations are obtained from Eqs. (9-11) according to the choice of frame of reference and the vector $n(x,y)$. We will work out some examples explicitly and, for each of them, we will make appropriate choices.

At this point, however, it should be noted that the procedure described above restates, with some formal differences, the one introduced in Ref. 10, Eqs. (6-9). The restriction imposed to n in the rest of Ref. 10 is lifted here, since we want to examine whether a proper choice of n could increase accuracy. Regions where such a choice may be critical are boundary and shock neighborhoods. Here we limit ourselves to the former.

Subsonic Source

The exact solution for the subsonic "source" flow (that is, a steady subsonic flow with streamlines emanating from a single point, and all variables functions of the distance from that point alone) is

$$(1+\delta M^2)^{(\gamma+1)/2(\gamma-1)}=Mr \tag{12}$$

where the radial coordinate r can be scaled so that, at $r=1$, the Mach number M has a given "inflow" value, M_0.

We have computed the flow in a straight-walled divergent duct, with a semiaperture equal to ϵ, as in Ref. 15, imposing initial and boundary conditions from which the subsonic source flow with a prescribed value of M_0 should result as an asymptotic steady state. The computation in the duct is limited to a portion comprised between $x=1$ and $x=2$ and a stretched Cartesian grid is used (Fig. 1). The computational variables X and Y are defined by

$$X=x,\qquad Y=y/\sigma_0 x,\qquad \sigma_0=\tan\epsilon \tag{13}$$

The computational boundaries are thus defined by $Y=0,1$ (rigid walls) and $X=1,2$ (subsonic inflow and outflow artificial boundaries). We impose that the total temperature is

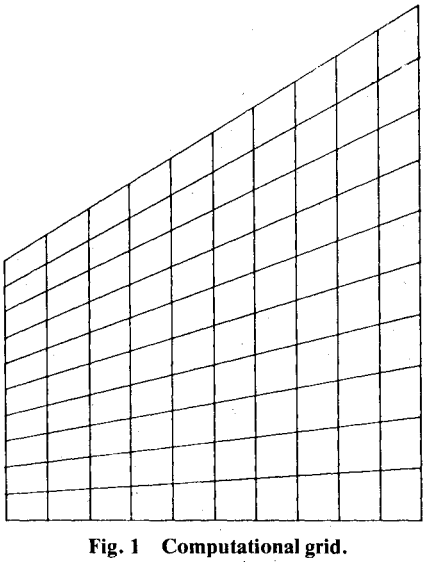

Fig. 1 Computational grid.

constant at the inflow boundary and we prescribe the speed of sound at the outflow boundary consistent with a steady state having $M=M_0$ at $r=1$. We also prescribe that the inflow velocity is, at all inflow boundary points, directed radially.

As a consequence of the normalization of the y coordinate, defined by Eqs. (13), the x and y derivatives in Eqs. (9-11) must be replaced by

$$\frac{\partial}{\partial x}=\frac{\partial}{\partial X}+Y_x\frac{\partial}{\partial Y},\qquad \frac{\partial}{\partial y}=Y_y\frac{\partial}{\partial Y} \tag{14}$$

In the present case, $Y_x=-Y/x$, $Y_y=1/(\sigma_0 X)$.

To simplify the notation let U and V be the Cartesian velocity components, $c=\cos\alpha_0$, $s=\sin\alpha_0$, and

$$u=Uc+Vs,\qquad v=-Us+Vc \tag{15}$$

We stipulate that α_0 may be a function of x and y, but not a function of t. After some tedious but obvious manipulations, Eqs. (9-11) can be recast in the form

$$a_t=(\delta/2)\,(f_1+2f_2+f_3+f_4+f_5+f_6$$
$$+2f_7+f_8+f_9+f_{10}+f_{11}) \tag{16}$$

$$u_t=\tfrac{1}{2}\,(f_1-f_3+f_6-f_8+f_{12}) \tag{17}$$

$$v_t=\tfrac{1}{2}\,(f_4-f_5+f_9-f_{10}+f_{13}) \tag{18}$$

where

$$R_1=a/\delta+u,\quad R_2=-a/\delta,\quad R_3=a/\delta-u$$
$$R_4=a/\delta+v,\quad R_5=a/\delta-v \tag{19}$$

$$\lambda_1=U+ac,\ \lambda_2=U\ \lambda_3=U-ac,\ \lambda_4=U-as,\ \lambda_5=U+as$$
$$\lambda_6=(V+as)\,Y_y+\lambda_1 Y_x,\ \lambda_7=VY_y+\lambda_2 Y_x$$
$$\lambda_8=(V-as)\,Y_y+\lambda_3 Y_x$$
$$\lambda_9=(V+ac)\,Y_y+\lambda_4 Y_x,\ \lambda_{10}=(V-ac)\,Y_y+\lambda_5 Y_x \tag{20}$$

$$f_i=-\lambda_i R_{iX},\ f_j=-\lambda_j R_{iY}\quad (i=1,5;\ j=i+5) \tag{21}$$

$$f_{11}=-2a\,[\,(UY_y-VY_x)\,\alpha_{0Y}-V\alpha_{0X}\,]$$

$$f_{12}=2v(\lambda_7\alpha_{0Y}+U\alpha_{0X}),\ f_{13}=-u(\lambda_7\alpha_{0Y}+U\alpha_{0X}) \tag{22}$$

We have experimented with two choices of α_0.

First Choice

First, we choose $\alpha_0=0$. The calculation is highly simplified since $c=1$, $s=0$, $u=U$, $v=V$, $\lambda_4=\lambda_5=\lambda_2$, $f_{11}=f_{12}=f_{13}=0$, and $f_4+f_5+2f_2=0$. Consequently, Eqs. (16-18) can be written in a simpler form as

$$a_t=(\delta/2)\,(f_1+f_3+f_6+2f_7+f_8+f_9+f_{10})$$

$$u_t=\tfrac{1}{2}\,(f_1-f_3+f_6-f_8)$$

$$v_t=\tfrac{1}{2}\,(f_4-f_5+f_9-f_{10}) \tag{23}$$

In Fig. 2 we show an interior point to be computed, and the trace of the pertinent Mach cone and the four bicharacteristics which must be used according to the choice of α_0. They have been numbered consistently with the definition of R_i above. The derivatives appearing in terms f_i and f_j must be approximated by differences taken between the point to be computed and another point, belonging to the domain of dependence defined by the pertinent λ_i. In Fig. 3 we have indicated such points for the situation depicted in Fig. 2. The updating of interior points will be accomplished following the same guidelines as in the one-dimensional case. Having separated the contributions to a_t, u_t, and v_t into terms whose domains of dependence are defined unequivocally, the points from which the value of each $f^k_{i,j}$ must be chosen at the corrector level are also found without ambiguities.

All calculations necessary for a proper enforcement of boundary conditions are performed at the end of both predictor and corrector levels just before updating a, u, and v. All boundary points are actually computed using the same routine as for interior points but, before updating, all f_i that

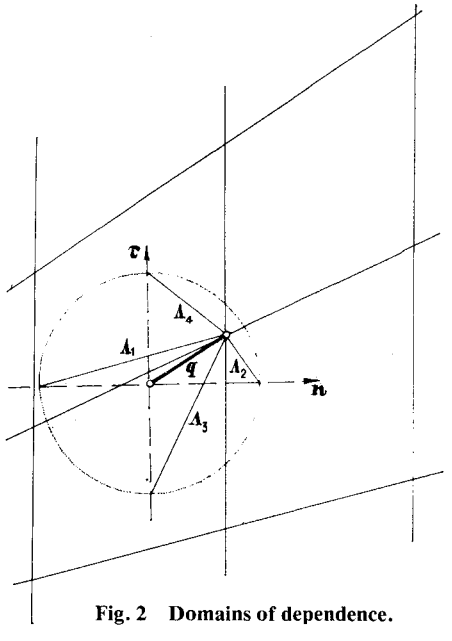

Fig. 2 Domains of dependence.

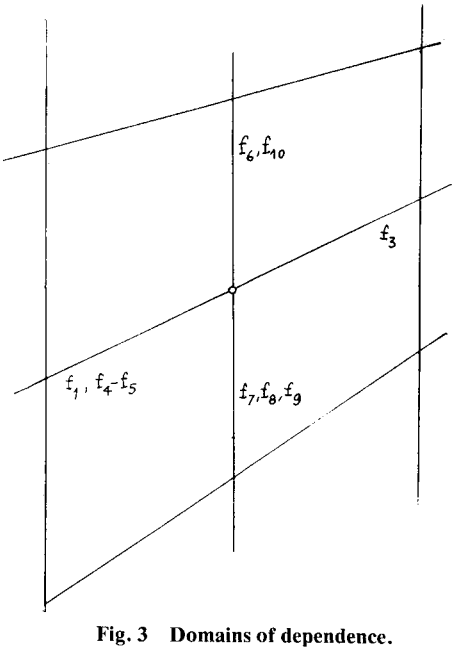

Fig. 3 Domains of dependence.

depend on exterior information are re-evaluated making use of the boundary conditions.

At the inflow boundary, it appears from Fig. 3 that f_1, f_4, and f_5 must be re-evaluated. In reality, however, only f_1 and $f_4 - f_5$ are needed. Since v is obtained directly from u, as we will see below, one can skip the calculation of v_t. The value of $f_4 - f_5$ is needed at the corrector level to compute v_t along the line next to the boundary. As was stated above, we choose to prescribe the direction of the velocity vector; at each inflow boundary point, we assume that

$$v = \sigma u \quad \text{where} \quad \sigma = y \tag{24}$$

We also prescribe that the total temperature (and, consequently, the stagnation speed of sound) remain constant. Since the stagnation speed of sound a_{st} is given by

$$a_{st}^2 = a^2 + \delta u^2 (1 + \sigma^2) \tag{25}$$

the second boundary condition follows

$$a_t = -\delta(1 + \sigma^2)(u/a) u_t \tag{26}$$

If the first and second of Eqs. (23) are substituted into Eq. (26), f_1 can be obtained using information obtained along the boundary or from the interior of the computational region. Subsequently, $f_4 - f_5$ can be obtained by substituting the second and third of Eqs. (23) into Eq. (24) differentiated in time.

At the outflow boundary only f_3 must be re-evaluated, and this can be done by prescribing a_t and using the first of Eqs. (23). At the lower boundary, which is a symmetry line, $V = 0$ and $Y_x = 0$. Therefore, λ_6, λ_7, and λ_8 all vanish, as well as f_6, f_7, and f_8; f_9 is the only term to be re-evaluated, but clearly $f_9 = f_{10}$ since both equal $-aY_y v_Y$.

The upper boundary, in this case, requires more careful consideration. It is evident, indeed, that our current choice of α_0 is poorly suited for a straightforward treatment of such a boundary. We note that we dispose of only one boundary condition (the vanishing of the normal component of the velocity) but we have two terms to re-evaluate: f_6 and f_{10}. Since we want to abide by the rule that no arbitrary elements should be introduced into the numerical treatment of the boundaries (that is, no manipulations should be performed at boundary points which are not physically justified; see Ref. 14), we must find an additional equation, valid at the upper boundary, which is a necessary consequence of the equations of motion. Such an equation is, indeed, easy to find. The upper boundary is a particle path. Along a particle path the following momentum equation is valid:

$$q_t + q\, q_X + (a/\delta) a_X = 0 \tag{27}$$

Note that here X is consistent with the definition of X given above. Clearly, Eq. (27) does not make use of any information other than from the boundary itself. If we take into account that along the boundary $v = \sigma_0 u$ and define $\nu^2 = 1 + \sigma_0^2$ we see that $\nu q = u + \sigma_0 v$. Consequently,

$$\nu q_t = \tfrac{1}{2} [f_1 - f_3 + f_6 - f_8 + \sigma_0 (f_4 - f_5 + f_9 - f_{10})]$$

Comparison of this equation with Eq. (27) shows that

$$f_6 - f_8 + \sigma_0 (f_9 - f_{10}) = 0 \tag{28}$$

identically. The boundary condition differentiated in time yields

$$v_t = \sigma_0 u_t \tag{29}$$

Making use of the second and third of Eqs. (23), we obtain

$$f_4 - f_5 + f_9 - f_{10} = \sigma_0 (f_1 - f_3 + f_6 - f_8) \tag{30}$$

Now, Eqs. (28) and (30) are the necessary and sufficient equations to determine f_6 and f_{10}, and they are physically correct.

Numerical experiments were made with different values of ϵ, M_0, and different mesh refinements. In addition, two sets of initial conditions were used. In the first, the duct contains a gas at rest, and a diaphragm is suddenly broken at the outflow section letting the gas in the duct communicate with an infinite cavity at a lower pressure. In the second, the initial conditions are the exact steady source flow. In the latter case, we cannot expect the numerical steady solution to coincide with the analytical solution to within any number of significant digits. Therefore, as the computation is started, small perturbations appear and grow until a steady state is reached, compatible with the truncation errors of the scheme and the fineness of the mesh.

Here we report typical evolutions of errors in u at two points on the inflow boundary (Fig. 4) when the initial

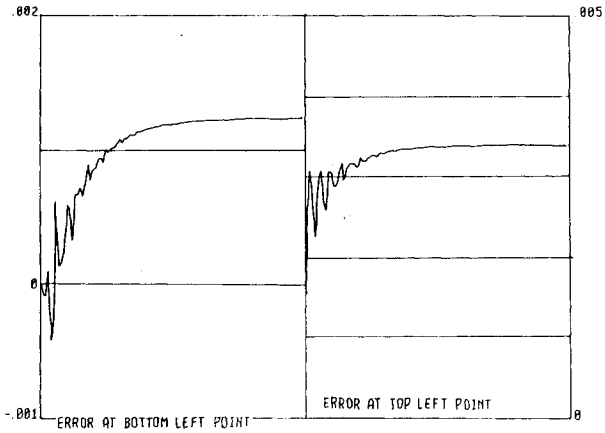

Fig. 4 Computational errors.

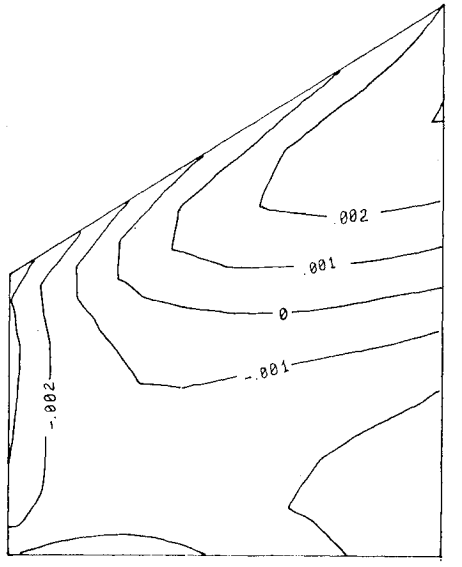

Fig. 5 Level lines for errors in M, 10×10 mesh.

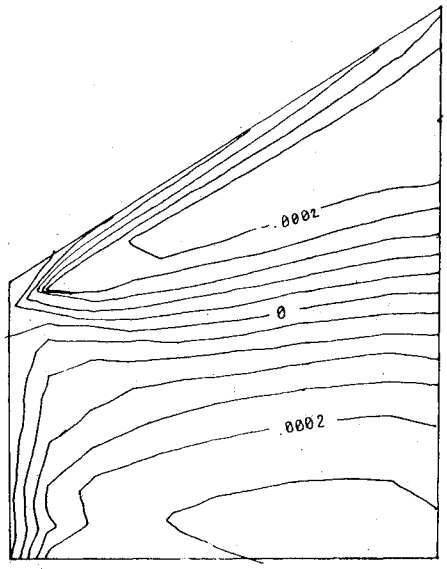

Fig. 6 Level lines for errors in T_{st}, 10×10 mesh.

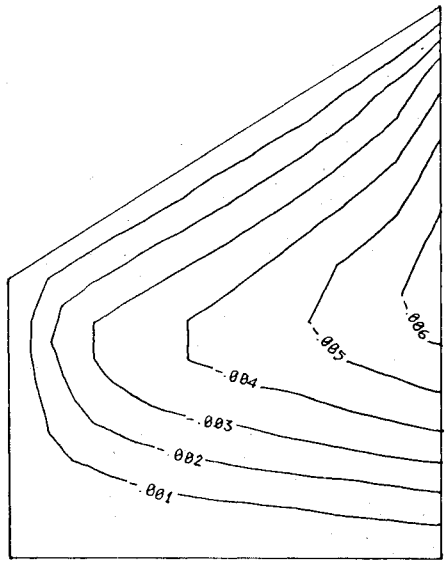

Fig. 7 Level lines for errors in v/u, 10×10 mesh.

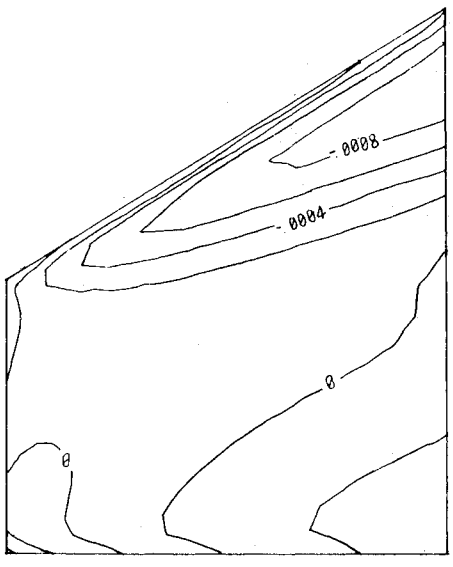

Fig. 8 Level lines for errors in M, 30×20 mesh.

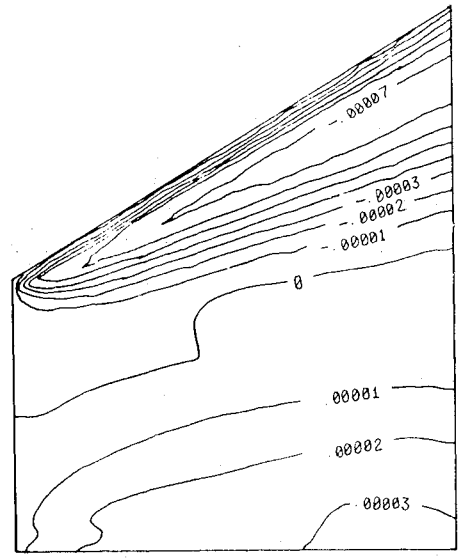

Fig. 9 Level lines for errors in T_{st}, 30×20 mesh.

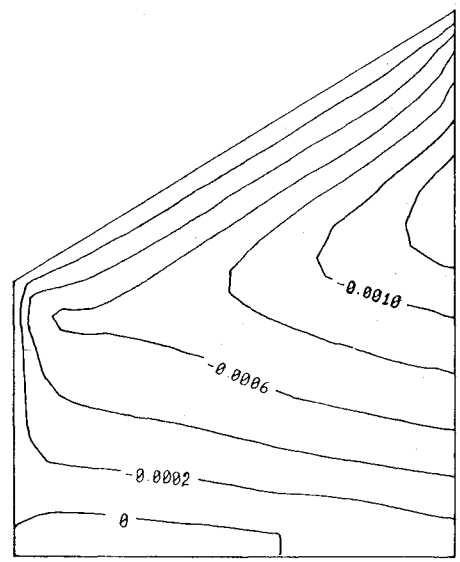

Fig. 10 Level lines for errors in v/u, 30×20 mesh.

conditions are given by the exact analytical solution, $\epsilon = 30$ deg and $M_0 = 0.5$. The computational mesh has 10 intervals in each direction. One can see that the errors reach a steady value in a stable way. The same values are reached if the computation is started from a state of rest. This is true for all the combinations of parameters tested so far. Therefore, the errors in the steady-state configurations depend on the fineness of the mesh, but not on the initial values. Distributions of errors in M, T_{st}, and v/u after the steady state is reached are also shown in Figs. 5-7, respectively.

Similar results are presented in Figs. 8-10 for a calculation which differs from the previous one only in the fineness of the mesh. In this case, there are 30 intervals in the x direction and 20 intervals in the y direction. Clearly, the accuracy of the calculation has improved.

In two-dimensional problems, where the mesh can be refined in two directions separately, it is not easy to see how accuracy improves with mesh refinements. An example of parametric analysis of mesh-accuracy relationship is shown in Fig. 11, where level lines of the logarithm (base 10) of the mean square error in a over the entire computational region are plotted vs the number of intervals in the x and y directions (abscissas and ordinates, respectively). It is evident that

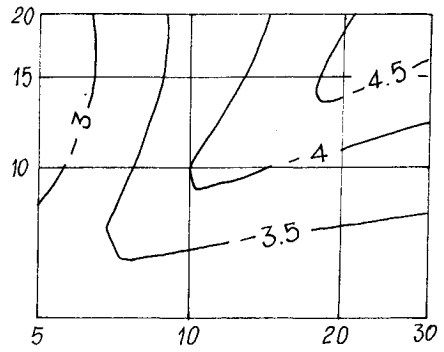

Fig. 11 Logarithms of errors as functions of the mesh.

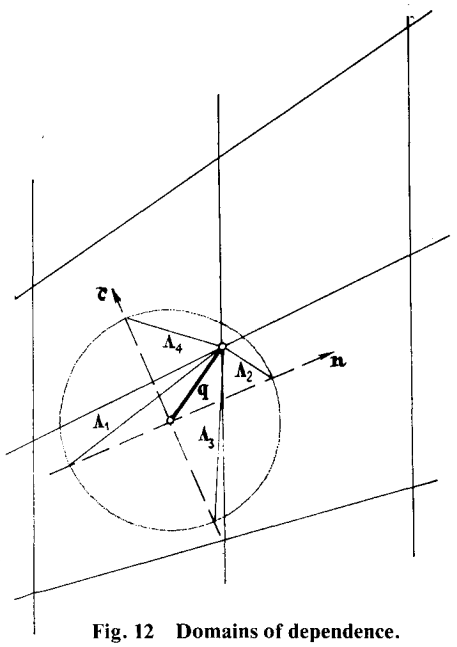

Fig. 12 Domains of dependence.

refinement in one direction only is not efficient (at times, it can even lower the overall accuracy).

Second Choice

Now we choose $\alpha_0 = \arctan(\sigma_0 Y)$. The X derivative of α_0 is still equal to zero, but the Y derivative is

$$\alpha_Y = \sigma_0 / (1 + \sigma_0^2 Y^2)$$

We also have,

$$c = (1 + \sigma_0^2 Y^2)^{-1/2}, \qquad s = \sigma_0 Yc$$

The situation described by Figs. 2 and 3 is now modified as shown in Figs. 12 and 13. Equations (16-18) now must be used.

At the symmetry boundary, $Y=0$, $Y_x=0$, $\alpha_0=0$, $\alpha_{0Y}=\sigma_0$, $c=1$, $s=0$, $V=0$, $a_Y=0$, $U_Y=0$, $u=U$, $v=0$, $\lambda_4=\lambda_5=\lambda_2$, $\lambda_6=\lambda_7=\lambda_8=0$, $\lambda_9=aY_y$, $\lambda_{10}=-\lambda_9$, $f_4=f_5=-f_2$, $f_6=f_7=f_8=0$, $f_{12}=f_{13}=0$. According to Fig. 13 only f_9 must be re-evaluated, and the boundary condition is again $f_9=f_{10}$.

At the upper boundary, the current choice of α_0 makes enforcement of the boundary condition much easier than in the previous choice. From Fig. 13, indeed, we see that f_{10} only has to be re-evaluated. Since $s=\sigma_0 c$, $V=\sigma_0 U$, $v=0$, $\lambda_6=\lambda_7=\lambda_8=0$, $\lambda_9=a/sx$, $\lambda_{10}=-\lambda_9$, $f_6=f_7=f_8=0$, $f_{11}=-2U/x$, $f_{12}=f_{13}=0$, the boundary condition $v_t=0$ directly

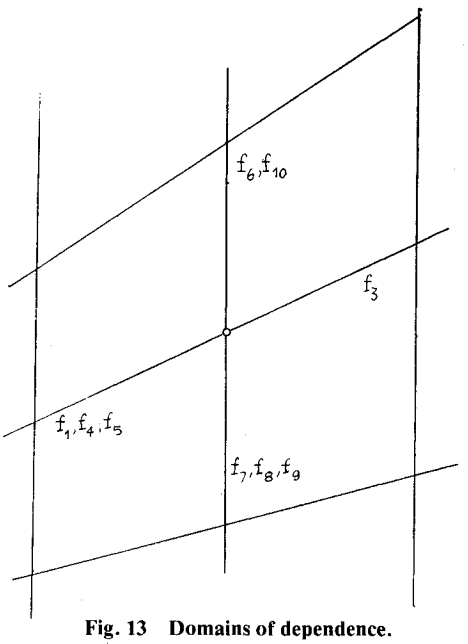

Fig. 13 Domains of dependence.

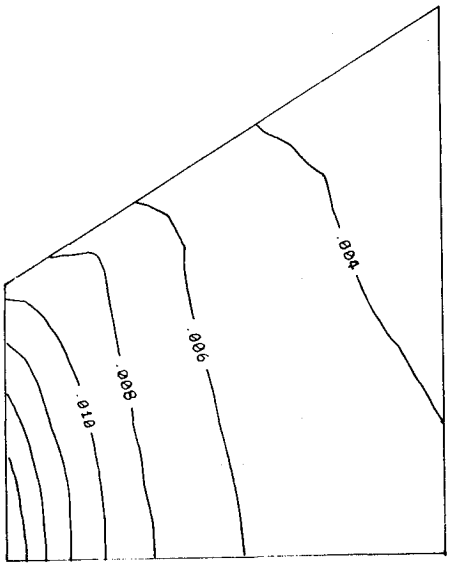

Fig. 14 Level lines for errors in M, 10×10 mesh.

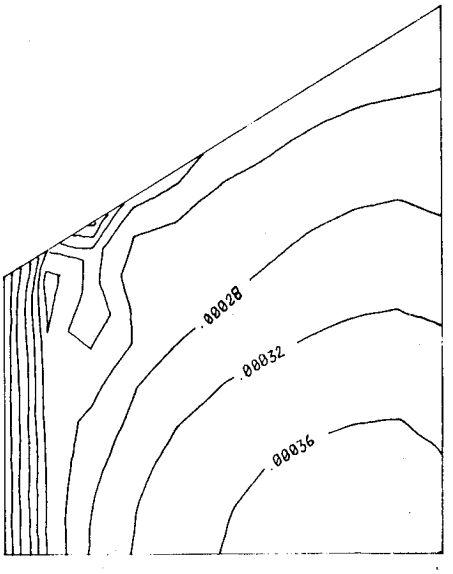

Fig. 15 Level lines for errors in T_{st}, 10×10 mesh.

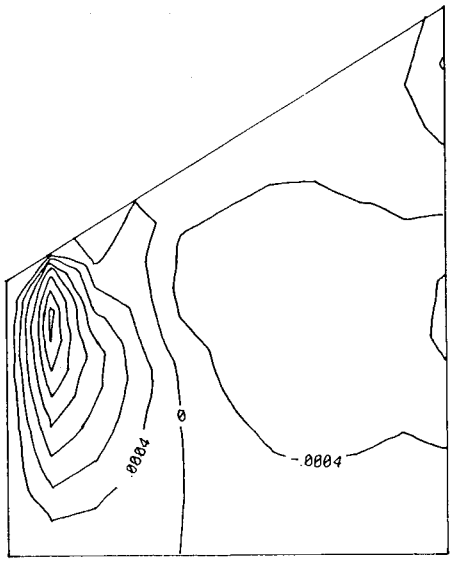

Fig. 16 Level lines for errors in v/u, 10×10 mesh.

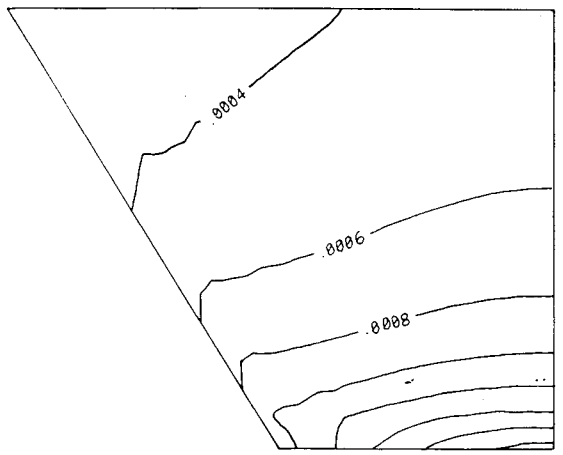

Fig. 17 Level lines for errors in M, 30×20 mesh.

yields

$$f_{10} = f_9 + f_4 - f_5 \tag{31}$$

Such a simple result is due to the fact that two of the bicharacteristics are along the boundary and two are orthogonal to it.

At the inflow boundary, instead, the bicharacteristic system is oblique to the boundary, and we may anticipate some difficulty. The unknown quantities are now $f_4 - f_5$, $f_4 + f_5 - 2f_2$, and f_1. Since f_4 and f_5 are not needed separately, f_4 can be computed from the interior, and f_5 obtained from the condition expressing the vanishing of v_t. The guiding vane model is compatible with the assumption that $v_X = 0$ outside the boundary. Then, we can postulate that $2f_2 + f_4 + f_5 = 0$ on the boundary, and get f_2. Finally, Eq. (26) allows f_1 to be determined.

At the outflow boundary, instead, the condition $a_t = 0$ yields directly,

$$f_3 = -(f_1 + 2f_2 + f_4 + f_5 + f_6 + 2f_7 + f_8 + f_9 + f_{10} + f_{11})$$

The plots presented in Figs. 14-16 and 17-19 parallel the ones shown in Figs. 5-7 and 8-10, respectively. Similar considerations can be made. In the present cases, the overall accuracy is slightly worse than in the previous ones. The

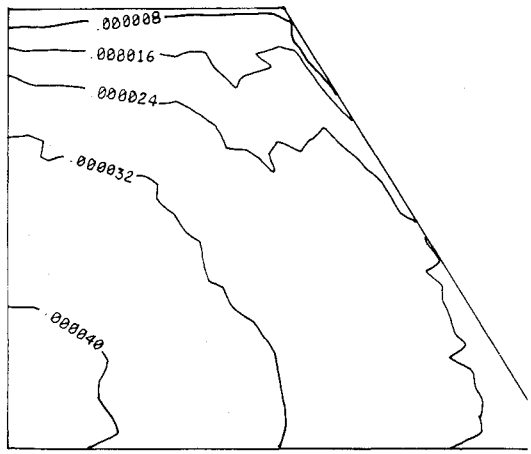

Fig. 18 Level lines for errors in T_{st}, 30×20 mesh.

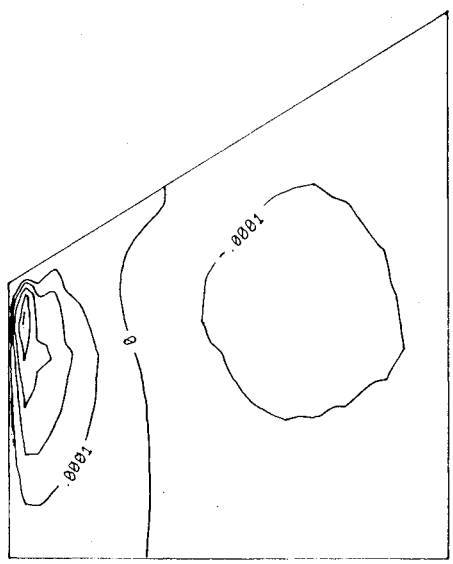

Fig. 19 Level lines for errors in v/u, 30×20 mesh.

errors seem to emanate from the inflow boundary; this is, indeed, the most difficult boundary to evaluate. It is interesting to note, however, that the errors seem to spread out in roughly concentric waves. This seems to indicate that the present choice of α_0 properly reflects the real one-dimensional nature of the problem.

Conclusions

We have also studied in detail three other problems. The first is the subsonic vortex, which we have analyzed in a stretched Cartesian mesh, assuming $\alpha_0 = 0$. This is a particularly exacting problem due to the presence of sharp corners at the inflow and outflow boundaries. The second is the Ringleb flow, which is almost one-dimensional if a stretched polar mesh is used,[16] but becomes substantially more difficult if a stretched Cartesian mesh is used. The advantage of the Ringleb flow is that, as the two previous examples, it provides an exact solution for comparison and evaluation of accuracy. The third is the subsonic flow around a circle, computed in an unstretched (and, therefore, orthogonal) polar frame. The calculation is performed within a ring. Since reliable numerical solutions to this problem exist, the relationship between accuracy and size of the computational ring can be analyzed. In this case, one important feature to be pointed out is that the total temperature remains constant throughout the computed region when a steady state is achieved, despite the coarseness of the mesh. The large amount of material justifies the issue of a separate paper on the subject.

In conclusion, we wish to point out that the arbitrariness in the choice of α_0 is, in reality, weaker than the arbitrariness intrinsic to all schemes for the solution of two-dimensional time-dependent problems based on approximate factorization. The latter, indeed, split the equations according to the coordinate directions; the integration scheme is thus controlled by the mesh but not related directly to any physical property. In the present scheme, instead, the freedom in the choice of α_0 (which is not necessarily a constant throughout) can be used to make the numerical procedure more consistent with the physical structure of the flow. We have seen above how a simple choice of α_0 can help improve the treatment of boundary conditions. The most important application of the method, however, seems to come about in the treatment of flows with imbedded shocks. The direction normal to the shock at every point is a preferential direction. Along the normal, indeed, the treatment of the shock is strictly one-dimensional; therefore, it is easy and extremely accurate.[13] In a forthcoming paper, we will show how the use of the present technique will allow a simple one-dimensional treatment of the shock to be used, by orienting the vector $\boldsymbol{n}$ in the direction of the normal to the shock, without having to introduce a shock-oriented grid.

References

[1]Warming, R. F. and Beam, R. M., "Upwind Second-Order Difference Schemes and Application in Aerodynamic Flows," *AIAA Journal,* Vol. 14, Sept. 1976, pp. 1241-1249.

[2]Engquist, B. and Osher, S., "One-Sided Difference Approximations for Nonlinear Conservation Laws," *Mathematics of Computation,* Vol. 36, 1981, pp. 321-351.

[3]van Leer, B., "On the Relation Between the Upwind-Differencing Schemes of Godunov, Engquist-Osher and Roe," ICASE Rept. 81-11, 1982.

[4]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors and Difference Schemes," *Journal of Computational Physics,* Vol. 43, 1981, pp. 357-372.

[5]Moretti, G., "The λ Scheme," *Computers and Fluids,* Vol. 7, 1979, pp. 191-205.

[6]Zhu, Y-L. and Chen, B-M., "Difference Methods for Initial-Boundary-Value Problems and Computation of Flow Around Bodies," *Computers and Fluids,* Vol. 9, 1981, pp. 339-364.

[7]Zhong, X-C. and Moretti, G., "Comparison of Different Integration Schemes Based on the Concept of Characteristics as Applied to the Ablated Blunt Body Problem," *Computers and Fluid,* Vol. 10, 1982, pp. 277-294.

[8]Moretti, G., "A Numerical Analysis of Muzzle Blast—Precursor Flow," *Computers and Fluids,* Vol. 10, 1982, pp. 51-86.

[9]Moretti, G., "Calculation of the Three-Dimensional, Supersonic, Inviscid, Steady Flow Past an Arrow-Winged Airframe," NASA CR 3573, 1982.

[10]Zannetti, L. and Colasurdo, G., "Unsteady Compressible Flow: A Computational Method Consistent with the Physical Phenomena," *AIAA Journal,* Vol. 19, July 1981, pp. 852-856.

[11]Zannetti, L. and Moretti, G., "Numerical Experiments on the Leading Edge Flowfield," *AIAA Journal,* Vol. 20, Dec. 1982, pp. 1668-1673.

[12]Colasurdo, G., private communication, Politecnico di Torino, Italy, 1981.

[13]Moretti, G., "An Improved λ Scheme for One-Dimensional Flows," NASA CR 3712, 1983.

[14]Moretti, G., "A Physical Approach to the Numerical Treatment of Boundaries in Gas Dynamics," NASA CP 2201, 1981, pp. 73-95.

[15]Moretti, G. and Pandolfi, M., "Critical Study of Calculations of Subsonic Flows in Ducts," *AIAA Journal,* Vol. 19, April 1981, pp. 449-457.

[16]Förster, K., ed., *Boundary Algorithms for Multidimensional Inviscid Hyperbolic Flows,* Friedr. Vieweg & Sohn, Braunschweig, Wiesbaden, 1978.