AIAA JOURNAL
Vol. 31, No. 1, January 1993

Performance of Compressible Flow Codes at Low Mach Numbers

G. Volpe*
Grumman Corporate Research Center, Bethpage, New York 11714

The accuracy and the performance of three two-dimensional compressible flow codes at freestream Mach numbers as low as 0.001 are examined. Two of the codes employ a finite volume discretization scheme along with a multistage time-stepping algorithm to solve the Euler equations. The two codes differ in their respective use of cell-centered and node-centered differencing schemes. The third code uses an implicit finite difference procedure to solve the unsteady Navier-Stokes equations. Computational test cases are the inviscid steady flow over a circular cylinder and the impulsively started viscous flow over a cylinder. Errors in the numerical results are presented as functions of mesh size and computational Mach number. It is shown that for certain classes of problems, the compressible codes can be sufficiently accurate to predict flow features at essentially incompressible speeds and that they can be adequately efficient since there is no need to lower excessively the computational Mach number in order to extract these results.

Introduction

IN recent years, there has been considerable interest in the computation of flowfields at incompressible speeds. To be sure, the need for such computations was always present (e.g., calculation of the low-speed characteristics of airfoils and wings, hydrodynamic flows). With the maturing of the field of computational fluid dynamics and the development of more sophisticated algorithms, the range of application of these numerical techniques has been branching out of the traditional aerodynamic design area into various areas of industrial design, such as ship and automobile design, as well as the design of pumps, ducts, and turbomachinery, in general. In such cases, there is a need for the computation of both inviscid and viscous flows. These new areas of opportunity for CFD applications encompass relatively low-speed regimes that are incompressible for all practical purposes. On the other hand, the driving force behind the development of many widely used codes was the need to compute compressible flows, often in the transonic regime.

The computation of external, incompressible, inviscid flows has been and—to a large extent—is still being performed with panel- or singularities-type codes. These codes solve Laplace's equation in either two or three dimensions. The equation provides an adequate description of the inviscid, irrotational flowfield in the low-speed regime. Many interesting flows include regions where rotational and/or viscous effects are important. The more sophisticated panel codes account for these effects through iterative procedures in which the shape of the object being examined is sequentially modified. Examples of flows where such effects are significant are wake surfaces that issue from a wing's trailing edge and the vortex sheets shed from the sides of a missile or a submarine-type body, or from the sharp edge of a body. These vortex-sheet discontinuities can be modeled in panel codes by having their shape and their position adjusted to the surrounding flow. This already difficult task is complicated by the fact that it requires an a priori knowledge of where the sheets originate. In some cases, such as for wake surfaces or in the case of flow separation at sharp edges, the point, or line, of departure from the body is known. In other cases, where the vortex sheet is a result of separation from the smooth surface of a body, the location might be obtained by coupling the inviscid flow calculation with some procedure that estimates the viscous parameters on the body. Coupling of the inviscid calculation with a viscous one near the body's surface might be desirable in all cases in order to improve the accuracy of the results.

Panel methods had their genesis before the development of techniques and algorithms that could solve the potential (e.g., the Euler, and the Navier-Stokes equations). These later algorithms were developed mainly for applications at transonic and supersonic speeds. In addition to properly accounting for compressibility effects, they offer additional advantages over panel codes. For example, an Euler-type code can track a wake surface or a vortex sheet through the flowfield automatically; a panel code needs to iterate on the shape of the surface or sheet. A Navier-Stokes solver automatically accounts for the viscous effects. Sometimes these advantages translate into lower computing costs and better accuracy. Given these advantages, it is reasonable to ask whether these compressible flow codes could be used in the low Mach number regime, hereto the realm of panel codes. After all, as reported by Van Dyke,[1] slightly compressible flow is a regular perturbation of incompressible flow. It is true that, in this latter case, slightly compressible flows are seen as being constructed up from the incompressible ones, whereas in the question asked earlier the incompressible flows are to be extracted from the compressible ones. The fact remains, however, that the two kinds of flows are differentiated by essentially perturbation-type terms. The attempt to use compressible flow codes at low Mach numbers is thus not an unreasonable one.

Various authors (see Refs. 2 and 3, for example) have reported as inadequate the performance and the accuracy of compressible codes when used in the low Mach number range ($M_\infty < 0.100$). Modifications to the basic techniques (e.g., matrix conditioning) can be applied to better the performance. In fact, codes specifically designed to compute incompressible flow (see Ref. 4, for example) have been developed. However, because of time constraints, organizational attitudes or other factors, in practice engineers routinely use compressible codes to compute incompressible flows. The codes are typically exercised at a Mach number that is sufficiently low so that the difference from the desired zero Mach results can be presumed to be small. An explanation of why this is done can be garnered from the fact that the particular compressible flow code has been in use with the engineer or its organization for a con-

Presented as Paper 91-1662 at the AIAA 22nd Fluid Dynamics, Plasma Dynamics, and Lasers Conference, Honolulu, HI, June 24-26, 1991; received Sept. 3, 1991; revision received April 14, 1992; accepted for publication April 21, 1992.

*Senior Staff Scientist. Associate Fellow AIAA.

siderable length of time. Numerical programs require time to develop from a research phase to a stage where they can be routinely used by an engineer who is trying to solve a problem. During this time, the various parameters needed for its functioning are calibrated, and the code acquires the capability of handling geometrical shapes far more complex than those usually encountered during its early development period. Confidence and familiarity with the code develop during this period, and the code can become part of a library of flow codes. Pre- and postprocessing programs to be used in conjunction with the code are put into place, and the latter is set up to swap information with the former. The newly developed incompressible flow codes still need to go through this maturation period. Given the current situation, it seemed appropriate that the performance and the accuracy of compressible flow codes when used at low Mach numbers should be systematically evaluated. It is the intent of this paper to provide some of this evaluation, so that some quantitative judgments regarding the value of the results obtained under these conditions can be made instead of vague, qualitative statements regarding their worth, whether they would be disparaging or laudative. The paper is not meant to detract from the value of developing techniques and codes that specifically address incompressible flows. It is examing the validity of a widely shared practice; namely, the use of familiar codes outside the parameter space that they were originally designed for.

The present investigation covers three widely used two-dimensional programs: FLO52, FLO62, and ARC2D. The first two are Euler codes, and the third is a Navier-Stokes solver. The discretization schemes (finite volume in the case of the first two, finite differences in the case of the third) and integration techniques (implicit approximate factorization by ARC2D, multistage time stepping by the others) used are shared by other programs; thus, the results of this study should apply to those codes as well. The evaluation of the codes was carried out using geometrically simple flows that have been widely reported on in the literature as test cases. These examples allow one to identify the numerical errors. All three codes were used to compute the steady inviscid flow over a circular cylinder at various low Mach numbers, and the results are compared with the analytical incompressible solution for the flow over the circle in the next section. Computations by ARC2D for the laminar flow over an impulsively started circular cylinder are compared with an approximate solution for the problem and with experimental data in a subsequent section. The last section contains the conclusions that may be drawn from this study.

Inviscid Steady Flow over a Circular Cylinder

In this section, the surface pressure coefficient will be used as the error indicator. There are several reasons for using this as the measure. One is the fact that, in practice, engineers need precisely such a quantity for their studies. In addition, errors in any one of the fundamental variables of the numerical scheme (density, velocity components, energy) would show up as errors in the pressure coefficient.

The solution for the incompressible, noncirculatory, inviscid flow over a circular cylinder is well known and is given in any textbook on fluid dynamics. The pressure coefficient is defined as

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty q_\infty^2} \tag{1}$$

where p is the local static pressure, and p_∞, ρ_∞, and q_∞ represent the freestream pressure, density, and speed, respectively. The density in incompressible flow is constant, of course. With the use of Bernoulli's equation for incompressible flow, the pressure coefficient can be expressed as

$$C_p = 1 - \frac{q^2}{q_\infty^2} \tag{2}$$

where q is the local speed. On the surface of the cylinder, the solution is given as

$$C_p = 1 - 4 \sin^2 \theta \tag{3}$$

Here, θ denotes the azimuthal location of a point on the surface as measured from the direction of the freestream (assumed to be along a Cartesian horizontal axis). The test problem consists in reproducing this result numerically using the algorithms mentioned earlier.

FLO52 Computations for Low Mach Numbers

FLO52, an Euler code developed by Jameson et al.,[5] is widely used and contains numerical features that are shared by many similar codes. Here, the two-dimensional, unsteady Euler equations in primitive variables are discretized spatially using a finite volume scheme in which the variables are stored at the centers of the cells of the computational mesh that surrounds the body in question. The discretized equations are integrated to steady state using a multistage Runge-Kutta scheme. Attainment of the steady state can be accelerated by the use of a variable time step with the algorithm remaining stable for relatively high values of the Courant number. Convergence can also be accelerated by the use of a multigrid technique, as well as by enthalpy damping. This latter feature is superfluous for the present study. Residual smoothing, another acceleration device of the standard code, is also useful in the low-speed regime. A damping term, which goes to zero in the limit of zero mesh width, is introduced to prevent odd-even point decoupling and the appearance of oscillations in the numerical solution near shock waves. The latter feature, which is controlled by a user-specified parameter multiplying a second-order difference of the variables, is not necessary at low speeds and has, therefore, been kept at very low values in the computations that will be shown. The fourth-order difference term, which controls point decoupling, has been retained, but was assigned a value smaller than the value typically used for transonic calculations. The scheme is formally second-order accurate in space. Near shock waves the accuracy drops to first order, but this is of no concern here. By switching to a constant time step, time-accurate calculations are possible. Boundary conditions to be satisfied are tangency at the body surface (the circular cylinder in the present case) and freestream in the far field. Details of the numerical method can be found in the original referenced work.

The code was used to compute the flow over the circular cylinder at Mach numbers of 0.100 and smaller. It was implemented in a straightforward manner as it would be in the computation of a transonic flow. The only exception to this was the adoption of Eq. (2) for the computation of the pressure coefficient rather than the more general Eq. (1). The latter expression can be derived from the first through a formal expansion in M_∞^2, and the discrepancy between the two forms for C_p is proportional to the square of the Mach number. The advantage in using Eq. (2), of course, lies in the fact that this relation is independent of Mach number.

In FLO52, as in all codes including the others that will be discussed later, the flow quantities that are computed are scaled with their reference values. Here the reference values for the pressure and the speed are assumed to be the freestream pressure p_∞ and the freestream speed of sound a_∞, respectively. Thus, the pressure will be of the order of p_∞, which is assigned a value of unity. With a_∞ also given a value of 1, the velocity components will be of the order of the Mach number. Their numerical values will be dropping with decreasing Mach number. In order to get meaningful results (e.g., the third decimal place in the pressure coefficient), calculations have to be done on machines with large precision. All of the present computations were done on a Cray-XMP machine. They would not have been possible on a single-precision device.

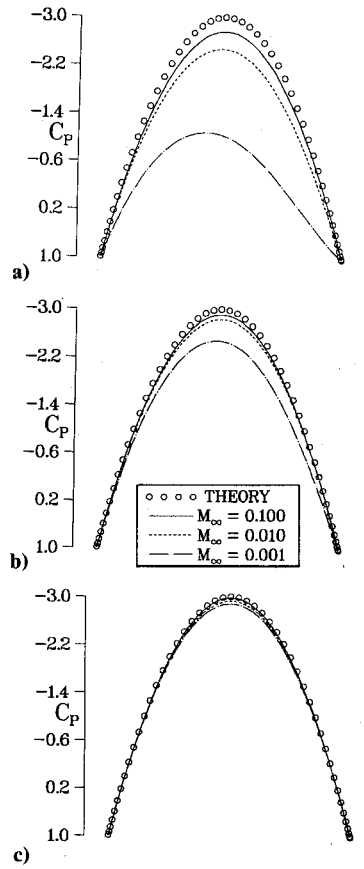

Fig. 1 Pressure distribution for inviscid steady flow on circular cylinder as computed by FLO52 and analytical solution: a) 96 × 16 mesh, b) 192 × 32 mesh, and c) 384 × 64 mesh.

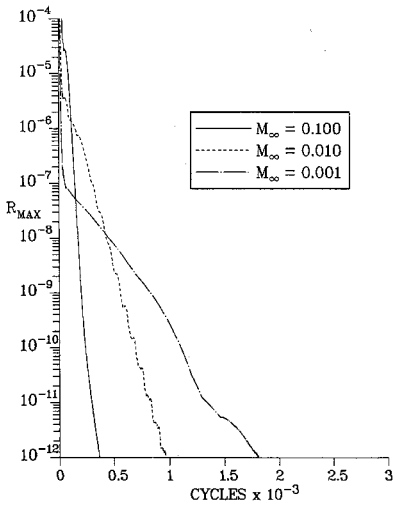

Fig. 2 FLO52 convergence histories of maximum error in density; 96 × 16 mesh.

Figures 1a–c depict results of computations of the flow over the circular cylinder for various meshes and for Mach numbers of 0.100, 0.010, and 0.001. This last value is notably lower than any reported by other investigators. All of the calculations were done on polar-type meshes with a uniform point distribution in the azimuthal direction and a stretched distribution in the radial direction to a distance of 100 radii. In each figure, the analytical solution is also given in addition to the numerical results for the three different Mach numbers. As expected, the error in the numerical solution increased as the Mach number was dropped. This is readily discerned by looking at Fig. 1a, which shows the results computed on a mesh containing 96 points in the circumferential direction and 16 points in the radial direction. In fact, the solution for the $M_\infty = 0.001$ case is so poor that the solution does not display the expected fore and aft symmetry, even though the calculation had reached a steady state and the maximum residual in the density field had been reduced to machine zero. The convergence history for this calculation is given in Fig. 2, along with the histories corresponding to the other Mach numbers. In this figure, the horizontal axis denotes the number of multigrid cycles performed. The deterioration of the solution with decreasing Mach number has been noted by others. However, a remarkable finding, which has not been reported, is that the discrepancy between computed and analytical solutions can be made smaller by refining the grid. In Fig. 1b, the numerical solutions obtained on a 192 × 32 mesh for the various Mach numbers are depicted along with the analytical solution. On comparing the results in this figure with the corresponding results in Fig. 1a, it is clear that the errors in the numerical solutions are smaller for all Mach numbers. On further refinement to a 384 × 64 computational mesh, the discrepancies in the solutions due to freestream Mach number become even smaller. These results should not be surprising since the accuracy of the underlying numerical scheme depends on the square of the mesh width and, thus, it should improve with decreasing mesh spacing regardless of Mach number. Even the solution obtained for $M_\infty = 0.001$ on the finest mesh might be acceptable, although the solution computed for $M_\infty = 0.100$ is within 4% of the analytical result.

The convergence rate of the numerical scheme deteriorates noticeably as the Mach number is decreased, as can be seen from Fig. 2, which shows results for the 96 × 16 mesh. The relative slowdown is present on all computational meshes. Similar trends are observed with the 192 × 32 and with the 384 × 64 grids. In all cases, identical numerical parameters were used. It is likely that the convergence rate of any one case could be changed by a different choice of parameters. In Fig. 3, the convergence histories for the $M_\infty = 0.010$ cases experienced on the various meshes are shown. A slowdown with decreasing mesh width is apparent, even though the horizontal axis should be interpreted with reservations. In each case the maximum error is reported as a function of iteration cycle. The time by which the solution is advanced in one cycle is different in each case and different for each point in the flowfield. Determining the relative times needed to arrive to the steady state is difficult, and the job is complicated by the fact that the multigrid scheme was used in each case to accelerate convergence. Four multigrid levels were used for the 96 × 16 case; five and six, respectively, for the successively finer meshes. The point expressed by this figure is that more cycles were needed to arrive at a solution. For the finest

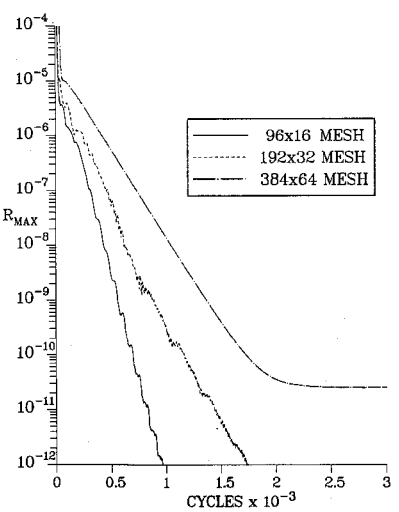

Fig. 3 FLO52 convergence histories of maximum error in density; $M_\infty = 0.010$.

(384 × 64) mesh, a plateau is eventually reached at a level higher than the one attained by the coarser grids. It is most likely due to the fact that the precision limit is reached earlier.

FLO62 Computations for Low Mach Numbers

FLO62 is very similar to FLO52 since both codes use the same basic techniques for spatial discretization and time integration. The difference between them consists of the assigned location of the flow variables on the grid. In FLO62, the variables are assigned at the nodes of the mesh in contrast to the mesh cell center option followed in FLO52. As in the other code, the unsteady compressible Euler equations are discretized using a second-order accurate central differencing scheme. Dissipation terms are constructed in a consistent and similar fashion, and all of the acceleration techniques mentioned earlier for FLO52 can be implemented also. In the field, the two codes are formally identical to second order (in the absence of shock waves) since this is the order of accuracy to which they are discretized. The significant difference occurs at a solid boundary. In the case of FLO62, boundary conditions can be implemented using values of the flow variables defined right on the boundary, rather than one-half a mesh

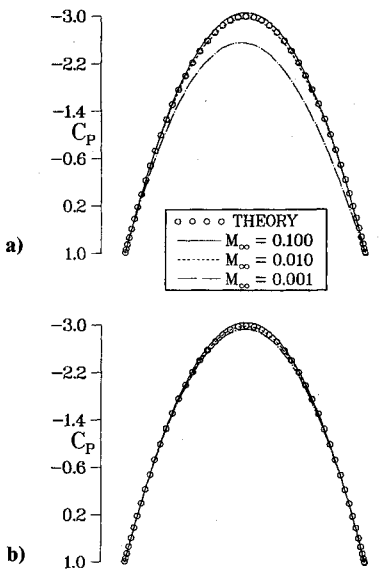

Fig. 4 Pressure distribution for inviscid steady flow on circular cylinder as computed by FLO62 and analytical solution: a) 192 × 32 mesh, and b) 384 × 64 mesh.

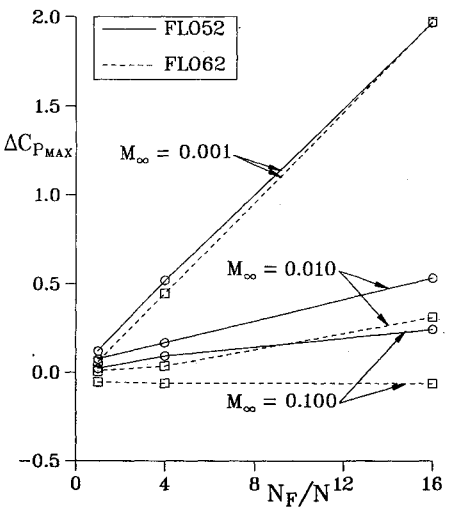

Fig. 5 Effect of Mach number on peak error in FLO52 and FLO62 numerical solutions for inviscid steady flow over circular cylinder as function of relative mesh size.

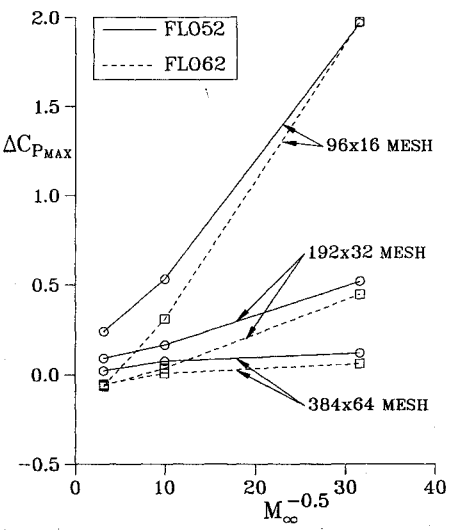

Fig. 6 Effect of relative mesh size on peak error in FLO52 and FLO62 numerical solutions for inviscid steady flow over circular cylinder as function of Mach number.

width away as in the previous case. Both implementations of boundary conditions are accurate to second order, nonetheless.

Results for the computation of the flow over the circular cylinder for various assigned freestream Mach numbers and computational meshes are similar to those obtained with FLO52. This was not unexpected. The trends observed in the earlier case in regard to solution accuracy and convergence rates were evident again. The numerical solutions for the various Mach numbers used are given in Fig. 4a for the 192 × 32 mesh and in Fig. 4b for the 384 × 64 mesh. The results for the FLO52 and FLO62 computations are summarized in Fig. 5, which depicts the error in the peak value of the numerically computed pressure distribution as a function of mesh size for each of the Mach numbers mentioned earlier. The label of the abscissa in Fig. 5 (N_F/N) refers to the ratio of the mesh size of the finest mesh considered (N_F) to a particular mesh (N). The nearly linear variation of the error with the square of the mesh size attests to the second-order accuracy of the algorithms. To be noted in Fig. 5 is the slightly smaller error in the solution computed by FLO62 as compared to the FLO52 result for each of the computed cases. The different implementations of the boundary conditions might explain some of this difference. However, it should be mentioned that the absolute levels of the errors for all cases can be altered by a different choice of the dissipation parameter previously mentioned. All runs with FLO62 were done with the identical set of parameters. The same is true of the runs done with FLO52. In each case, the dissipation parameter is near the minimum needed to maintain stability. The relative errors between the two codes are not important. What is important is the behavior of the error with mesh size and Mach number for each of the codes. The behavior of the error with Mach number at a constant mesh is shown in Fig. 6. It appears that the error is, roughly, inversely proportional to the square root of the Mach number, an interesting observation that might have a logical explanation.

ARC2D Computations for Low Mach Numbers

ARC2D is a well-known and widely distributed code for the solution of the compressible Euler and the Navier-Stokes equations developed by Pulliam and Steger.[6] It is built around an implicit finite difference scheme originally introduced by Beam and Warming. The equations are discretized in a generalized coordinate space and are integrated using an implicit approximate factorization technique. Damping terms consisting of a combination of second- and fourth-order differences similar to those mentioned earlier for FLO52 are introduced

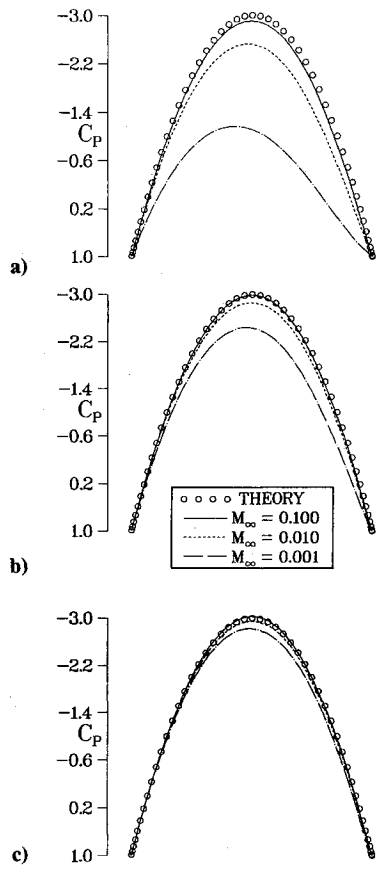

Fig. 7 Pressure distribution on circular cylinder as computed by ARC2D and analytical solution: a) 96 × 16 mesh, b) 192 × 32 mesh, and c) 384 × 64 mesh.

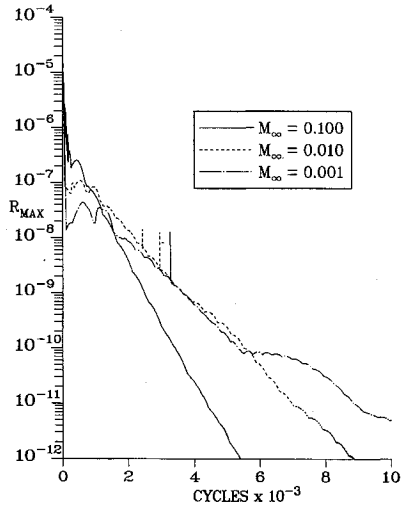

Fig. 8 ARC2D convergence histories of $L2$ norm; 192 × 32 mesh.

for inviscid calculations. Viscous effects can be introduced as an option with a thin-layer approximation of the Navier-Stokes equations. Additional viscous terms can also be retained, if desired. Calculations can be done for either laminar or turbulent flow. The original version of the code contained the Baldwin-Lomax turbulent model, although a more recent development of the code incorporates a newer model due to Baldwin and Barth. Nothing more will be said in this regard since in this investigation no turbulent flow cases will be discussed. The method can be run in either a time-accurate mode or, if a steady-state solution is the objective, in a non-time-accurate manner. In the second case, various techniques such as local time stepping, matrix reduction, and matrix diagonalization, can be invoked to accelerate convergence. The method is very stable even on meshes with the small cell spacings of the size needed for viscous flow calculations. Details of the method can be found in Ref. 6.

The accuracy of ARC2D in computing low Mach number inviscid flows was checked as in the previous cases against the analytical solution of incompressible flow over the circular cylinder. The computational meshes used were identical to the grids employed for the corresponding FLO52 cases. In Fig. 7a, the numerical solutions for $M_\infty = 0.100$, 0.010, and 0.001 on a 96 × 16 mesh are shown, as well as the analytical solution. Figures 7b and 7c contain the results for the 192 × 32 and the 384 × 64 meshes, respectively. The trends in the computed solution with mesh cell width and freestream Mach number retrace the results experienced with FLO52, despite the fact that the differencing scheme and the solution procedure are dissimilar. The trends in the convergence rates are also very similar. In Fig. 8, the convergence histories for cases computed on the 192 × 32 mesh for the various Mach numbers are shown. Figure 9 depicts the iteration histories for the M_∞ = 0.010 cases on the various meshes. Slowdowns in convergence are again evident with both decreasing Mach number and with decreasing mesh width. It is interesting that, again, a plateau

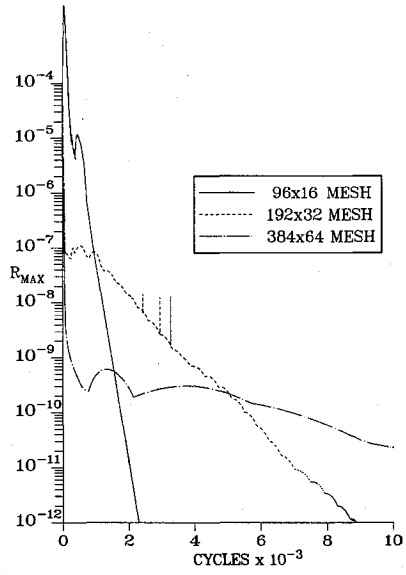

Fig. 9 ARC2D convergence histories of $L2$ norm; $M_\infty = 0.010$.

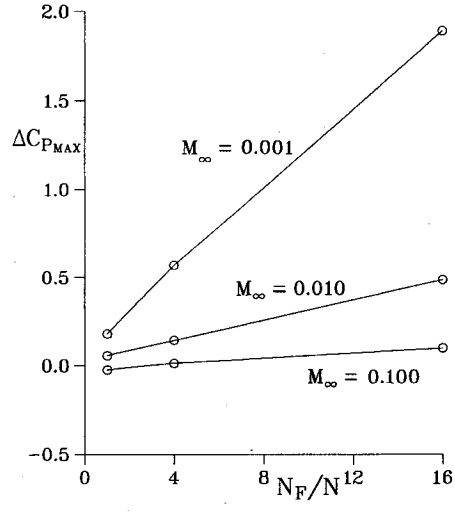

Fig. 10 Effect of Mach number on peak error in ARC2D numerical solutions for inviscid steady flow over circular cylinder as function of relative mesh size.

is eventually reached for the $M_\infty = 0.010$ case on the finest mesh, indicating that indeed the precision limit of the machine might be approached. The discrepancies between the numerical results and the analytical solution at the peak of the pressure distribution is depicted in Fig. 10 as a function of mesh width and in Fig. 11 as a function of Mach number. The former attests to the second-order accuracy of the method, and the other confirms the trend observed for the other codes. Table 1 gives a summary of the peak error in the numerical solutions for all of the three codes investigated.

The conclusion to be drawn from these results is that the errors encountered in running these compressible flow codes at low Mach numbers are numerical errors due to discretization, since they can be reduced by refining the computational mesh, and, apparently, are not due to the approach of a singularity.

Laminar Flow over an Impulsively Started Cylinder

In this section, ARC2D will be used to compute the evolution in time of the flow over a circular cylinder impulsively started from rest. Given that a particular level of accuracy can be obtained with a sufficiently fine mesh for Mach numbers as small as 0.001, as was shown in the preceding section, computations in this section will be done with $M_\infty = 0.100$. The accuracy of the code for tracking the evolution of an incompressible unsteady flow is to be assessed. Of particular interest in this flow are the computation of the time when separation of the flow from the surface first occurs and the growth of the separation region. This is a problem for which considerable experimental data exist for essentially incompressible flow. Furthermore, due to the simple geometry, approximate analytical solutions for this flow have been developed. One such analytical solution has been found by Wang.[7] Wang used the

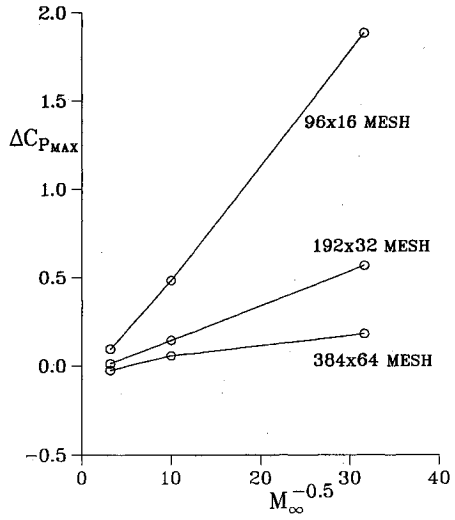

Fig. 11 Effect of relative mesh size on peak error in ARC2D numerical solutions for inviscid steady flow over circular cylinder as function of Mach number.

Table 1 Error in computed value of pressure coefficient at top of circular cylinder

Mesh	M_∞	FLO52	FLO62	ARC2D
96 × 16	0.100	0.24194	− 0.06461	− 0.09520
	0.010	0.53227	0.31048	0.48349
	0.001	1.96456	1.97055	1.88577
192 × 32	0.100	0.09227	− 0.06102	− 0.01217
	0.010	0.16738	0.03572	0.14313
	0.001	0.51980	0.44728	0.56850
384 × 64	0.100	0.02218	− 0.05437	− 0.02439
	0.010	0.07569	0.00772	0.05631
	0.001	0.12066	0.06165	0.18046

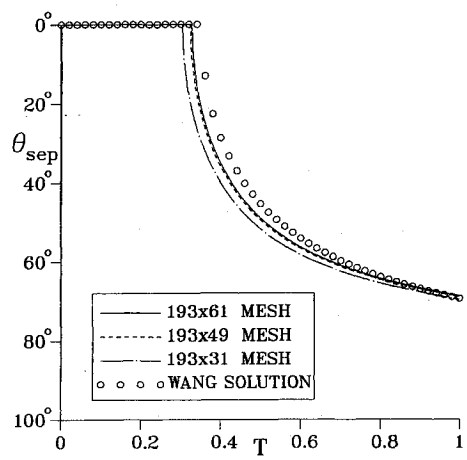

Fig. 12 Separation point on impulsively started circular cylinder; effect of mesh size.

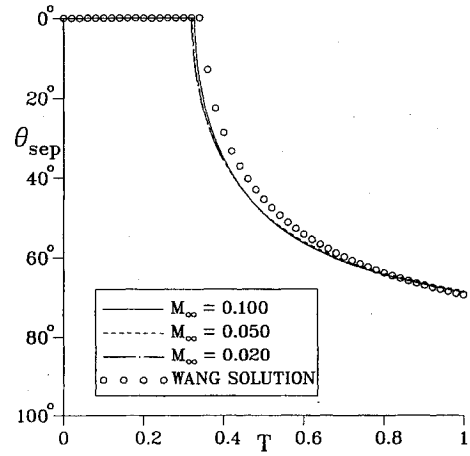

Fig. 13 Separation point on impulsively started circular cylinder; effect of freestream Mach number.

method of inner and outer expansions to construct a composite solution that is valid in the limit of large Reynolds numbers Re and small times during which the cylinder has traveled a distance that is only a small fraction of its radius r. In Wang's solution, the angular position of the separation point, as measured from the rear stagnation point, is given by

$$AT\cos\theta - \sqrt{\frac{\pi T}{Re}} = 2 \tag{4}$$

where $A = 4[1 + (4/3\pi)]$ and T is a nondimensional time related real time t by

$$t = \frac{Tr}{q_\infty} \tag{5}$$

Here q_∞ is the freestream speed. Separation first develops at the rear stagnation point, $\theta = 0$. For a large Reynolds number ($Re > 10^5$), Eq. (4) yields a separation time T equal to 0.351. The movement of the separation point θ_s with time T is depicted by the symbols in Fig. 12. The flat portion of the curve reflects the absence of separation, and the break in the curve marks the time when separation first appears.

ARC2D was used with a polar-type grid to numerically compute this flow. The computations were done in a time-accurate manner, of course. Computational accuracy in this case is heavily dependent on the degree of resolution of the viscous layer near the wall. This, in turn, is determined by the size of the mesh and the clustering of points near the cylinder's wall. The ARC2D results for three different meshes containing an

increasing number of grid points in the radial direction are depicted in Fig. 12 along with Wang's analytical solution. The computations were carried out at a Mach number of 0.100 and a Reynolds number of 6×10^4. All other computational parameters, such as time step size Δt and distance from the wall to the first row of grid points Δs_w, are identical. In ARC2D, time is nondimensionalized by multiplying it by the ratio of the reference length (the diameter d in this case) to the reference speed (the freestream speed of sound a_∞). Thus real time is related to the number of integration steps performed N by

$$t = \frac{Nd\Delta t}{a_\infty} \tag{6}$$

By combining Eqs. (5) and (6), the integration time of the ARC2D computations can be related to Wang's nondimensional time. The relation is

$$T = 2M_\infty N\Delta t \tag{7}$$

In Fig. 12, as in the subsequent figures, the computations are plotted using Wang's nondimensional time as the abscissa. To be noticed in Fig. 12 is the good agreement between the computed results and the analytical solution. There are some differences, not unexpected, in the times denoting the occurrence of separation. All of the computed results are, however, within 10% of Wang's number. These times are also reported in Table 2, along with the times corresponding to other computations made with different sets of various parameters.

The objective of this investigation is to determine how well incompressible results can be predicted using a compressible flow code. In Fig. 13, the movement of the separation point as computed using three different Mach numbers is shown. It is seen that lowering the Mach number below 0.10 does not have

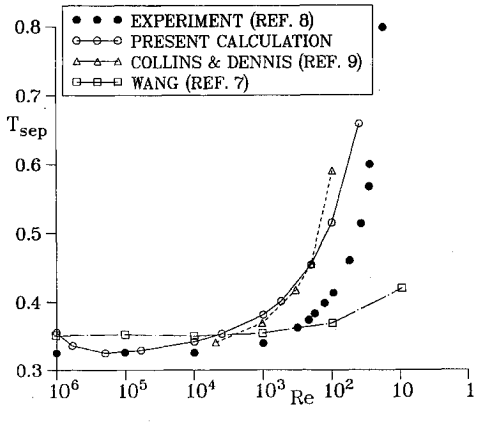

Fig. 14 Dependence on Reynold's number of time corresponding to initial separation of flow on impulsively started circular cylinder.

Table 2 Nondimension time to first occurrence of separation on impulsively started circular cylinder

Mesh	M_∞	Δt	Δs_w	T_{sep}
	0.00	0.005	0.0000	0.000
193 × 31	0.10	0.005	0.0020	0.364
193 × 49	0.10	0.005	0.0020	0.379
193 × 61	0.10	0.005	0.0020	0.329
193 × 61	0.05	0.005	0.0020	0.323
193 × 61	0.02	0.005	0.0020	0.335
193 × 61	0.10	0.010	0.0002	0.336
193 × 61	0.10	0.020	0.0002	0.348
256 × 64	0.10	0.005	0.0001	0.338
	Wang solution			0.351
	Experiment[8]			0.325

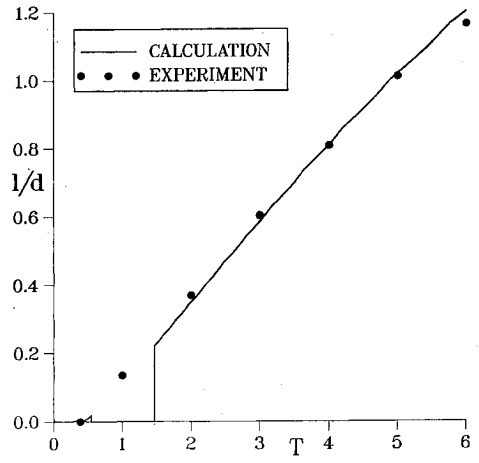

Fig. 15 Evolution with time of closed wake length behind impulsively started circular cylinder; $Re = 2 \times 10^2$.

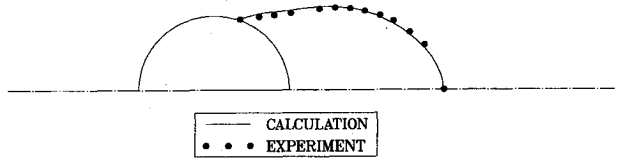

Fig. 16 Computed and experimental wake shapes for impulsively started circular cylinder, $Re = 2 \times 10^2$, $T = 5.0$.

any significant effect on the results. The Mach number has the same effect as the time step size as far as the attainment of a particular time is concerned [see Eq. (7)]. Thus, lower Mach numbers entail longer running times, just as a smaller time step does. This effect mirrors the lower convergence rates reported earlier for steady-state calculations. From the results shown in Fig. 13, it can be concluded that running the code at a Mach number of 0.10 gives results that can be considered incompressible for all practical purposes. In all cases computed, the break time compares well with Wang's analytical value of 0.351. It compares even better with the experimental value of 0.325 reported by Bouard and Coutanceau.[8] The value obtained with a 193 × 61 mesh at a Mach number of 0.100 ($T = 0.329$) is plotted in Fig. 14 along with other experimental and theoretical results (including Wang's). ARC2D results for various other Reynolds numbers as well as computational results due to Collins and Dennis[9] are also given. The dependence on Reynolds number of the time of initial separation is predicted quite well by the code. Even at low Reynolds numbers the present numerical results are no worse than the other computations reported in the figure.

The separation lines from either side of the longitudinal symmetry plane eventually join downstream of the cylinder to form a closed bubble in the wake. The evolution in time of this closed wake bubble for a Reynolds number of 2×10^2 is shown in Fig. 15. The computed results compare quite well with the experimental measurements. Even the initial formation of the wake (i.e., $T < 1$) appears to be predicted correctly. Experimentally, a nonzero wake length is detected only after a time lag with respect to the initial appearance of separation. Numerically, after separation first occurs at the trailing edge, the wake bubble grows to a small size and then it disappears, as shown by the small ramp in the curve in Fig. 15 following separation. Apparently, in this initial stage of evolution, as separation moves up the cylinder's side, the flow reattaches on the cylinder itself before reaching the trailing edge. As a result, there is no bubble along the wake centerline during this time. Only later do the two small bubbles of recirculating flow from either side of the centerline burst and join to form a larger bubble behind the cylinder. The size and shape of the bubble

at $T = 5$ are depicted in Fig. 16. As can be seen, the numerical results compare well with the experimental results of Bouard and Coutanceau.[8]

Concluding Remarks

Properly implemented, compressible codes can be used to simulate flows at low Mach numbers. The present investigation has concentrated on the dynamic features of external flows. As shown, these features can be predicted quite well by a number of existing codes. Indications are that the problem in running compressible codes with small values of the Mach number are associated with accuracy and precision. As the Mach number is lowered, all variables and/or their associated gradients in the discretized equations become progressively smaller. Situations arise where the values of the quantities to be computed are of the same order as the formal accuracy limit of the algorithm, typically the square of the mesh size. Under these conditions, solution accuracy can be increased by decreasing the mesh size, a fact that was clearly demonstrated. Reducing the mesh size to regain accuracy at smaller Mach numbers carries with it a deterioration of the convergence rates. However, the results have also shown that essentially incompressible results are reproduced at a Mach number of 0.100, a value large enough to provide good performance by the compressible codes.

This investigation has concentrated on some flow features commonly needed for external flow studies. Additional numerical tests involving lifting and turbulent flows should be performed. Other flow features, such as heat transfer, should be investigated for accuracy. The reproducibility of the present results in the case of internal flows should also be determined. At this point, however, considering the time and experience gained in developing, evaluating, calibrating, and running the compressible codes, their use for low Mach number studies should not be dismissed.

Acknowledgments

Portions of this investigation are an outgrowth of work performed under the auspices of the DARPA Advanced Submarine Technology Program. Funding was made available by Cortana Corporation, K. J. Moore, president, and J. Dugan, program manager, under a contract issued by G. W. Jones, DARPA program area manager, Hydrodynamics and Ship Control. The support of these individuals is gratefully acknowledged.

References

[1]Van Dyke, M., *Perturbation Methods in Fluid Mechanics,* Academic, New York, 1964, Chap. 2.

[2]Briley, W. R., McDonald, H., and Shamroth, S. J., "A Low Mach Number Euler Formulation and Application to Time-Iterative LBI Schemes," *AIAA Journal,* Vol. 21, No. 10, 1983, pp. 1467–1469.

[3]Merkle, C. L., and Choi, Y. H., "Computation of Low Speed Flow with Heat Addition," *AIAA Journal,* Vol. 25, No. 6, 1987, pp. 831–838.

[4]Kwak, D., and Chakravarty, S. R., "A Three-Dimensional Incompressible Flow Solver Using Primitive Variables," *AIAA Journal,* Vol. 24, No. 3, 1986, pp. 390–396.

[5]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-stepping Schemes," AIAA 14th Fluid and Plasmadynamics Conference, Paper 81-1259, Palo Alto, CA, June 1981.

[6]Pulliam, T. H., and Steger, J., "Implicit Finite-Difference Simulations of Three-Dimensional Compressible Flow," *AIAA Journal,* Vol. 18, No. 2, 1980, pp. 159–167.

[7]Wang, C.-Y., "The Flow Past a Circular Cylinder which is Started Impulsively from Rest," *Journal of Mathematics and Physics,* Vol. XLVI, 1967, pp. 195–202.

[8]Bouard, R., and Coutanceau, M., "Development of the Wake Behind an Impulsively Started Cylinder," *Journal of Fluid Mechanics,* Vol. 101, No. 3, 1980, pp. 583–607.

[9]Collins, W. M, and Dennis, S. C. R., "Flow Past an Impulsively Started Circular Cylinder," *Journal of Fluid Mechanics,* Vol. 60, Pt. 1, 1973, pp. 105–127.