INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 23, 455–466 (1996)

METHOD-OF-LINES SOLUTION OF TIME-DEPENDENT TWO-DIMENSIONAL NAVIER–STOKES EQUATIONS

OLCAY OYMAK AND NEVİN SELÇUK*

Department of Chemical Engineering, Middle East Technical University, 06531 Ankara, Turkey

SUMMARY

A novel approach to the development of a code for the solution of the time-dependent two-dimensional Navier–Stokes equations is described. The code involves coupling between the method of lines (MOL) for the solution of partial differential equations and a parabolic algorithm which removes the necessity of iterative solution on pressure and solution of a Poisson-type equation for the pressure. The code is applied to a test problem involving the solution of transient laminar flow in a short pipe for an incompressible Newtonian fluid. Comparisons show that the MOL solutions are in good agreement with the previously reported values. The proposed method described in this paper demonstrates the ease with which the Navier–Stokes equations can be solved in an accurate manner using sophisticated numerical algorithms for the solution of ordinary differential equations (ODEs).

KEY WORDS: computational fluid dynamics; unsteady incompressible flow; method of lines

1. INTRODUCTION

In the field of computational fluid dynamics (CFD) the advent of computers with their constantly growing processing and storing capabilities has made it possible to compute very complex flow fields by the numerical solution of the Navier–Stokes equations. Most algorithms on the solution of the Navier– Stokes equations are based on the steady state formulation and a considerable manner of finite-difference-, finite-volume- and finite-element-based numerical algorithms are available for the solution of these equations. However, development of a more efficient and accurate method, which this paper introduces, is still needed.

The method proposed in this paper is a numerical solution technique for the solution of partial differential equations (PDEs). The proposed technique, the method of lines (MOL), consists of converting the PDE system into an ordinary differential equation (ODE) initial value problem by discretizing the spatial derivatives together with the boundary conditions via Taylor series, spline or weighted residual techniques and integrating the resulting ODEs using a sophisticated ODE solver which takes the burden of time discretization and chooses the time steps in such a way that maintains

* Corresponding author.

CCC 0271–2091/96/050455–12

Received August 1995
Revised January 1996

the accuracy and stability of the evolving solution. The most important advantage of the MOL approach is that it is has not only the simplicity of explicit methods but also the superiority of implicit methods unless a poor numerical method for the solution of ODEs is used. The computational accuracy and efficiency of this method have previously been reported by the present authors.[1] Detailed information on the MOL can be found elsewhere.[2] In the present study the spatial derivatives of a dependent variable are approximated by using a five-point Lagrange interpolation polynomial.[3] Hence it is possible to investigate the solutions of the Navier–Stokes equations on both uniform and non-uniform grid topologies by a higher-order discretization scheme in which convective terms are discretized using upwindings and diffusive terms are discretized centrally. The structured grid generator based on quadrilateral serendipity elements is used. In the present paper the primitive variables formulation is used for solving the two- dimensional, unsteady Navier–Stokes equations.

The most advantageous feature of the MOL is that the resulting system of ODEs is an initial value problem (mathematically parabolic) and can be solved numerically by any powerful ODE solver. However, the presence of the coupled pressure gradients leads to an ill-posed initial value problem. Computing the pressure is the most difficult and CPU-consuming part of the overall solution of the incompressible Navier–Stokes equations and requires the solution of a Poisson-type equation which introduces an elliptic nature. In order to suppress the ellipticity, a parabolic algorithm in time which does not require the solution of a Poisson-type equation for the pressure is tailored.

The main contribution of this paper is to propose a time-accurate Navier–Stokes code based on the MOL approach with a non-iterative algorithm for the pressure. The predictions are validated against the available numerical solution for laminar developing flow in a circular duct.

2. GOVERNING EQUATIONS

The Navier–Stokes equations for two-dimensional, unsteady, incompressible developing flow in a circular duct can be written in the form

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial z} - v\frac{\partial u}{\partial r} - \frac{1}{\rho}\frac{\partial p}{\partial z} + \nu\left(\frac{\partial^2 u}{\partial z^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial r^2}\right), \tag{1}$$

$$\frac{\partial v}{\partial t} = -u\frac{\partial v}{\partial z} - v\frac{\partial v}{\partial r} - \frac{1}{\rho}\frac{\partial p}{\partial r} + \nu\left(\frac{\partial^2 v}{\partial z^2} + \frac{1}{r}\frac{\partial v}{\partial r} + \frac{\partial^2 v}{\partial r^2} - \frac{v}{r^2}\right), \tag{2}$$

$$\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} + \frac{v}{r} = 0. \tag{3}$$

The initial and boundary conditions for the problem are

$$\text{IC} \quad @\, t = 0, \quad \forall r \wedge \forall z: \quad v = 0, \quad u = 0, \tag{4}$$

$$\text{BC1} \quad @\, r = 0, \quad \forall z \wedge \forall t: \quad v = 0, \quad \partial u/\partial r = 0, \tag{5}$$

$$\text{BC2} \quad @\, r = R, \quad \forall z \wedge \forall t: \quad v = 0, \quad u = 0, \tag{6}$$

$$\text{BC3} \quad @\, z = 0, \quad \forall r \wedge \forall t: \quad v = 0, \quad u = u_{\text{in}}, \tag{7}$$

$$\text{BC4} \quad @\, z = L, \quad \forall r \wedge \forall t: \quad \partial^2 v/\partial z^2 = 0, \quad \partial^2 u/\partial z^2 = 0. \tag{8}$$

3. NUMERICAL SOLUTION TECHNIQUE

3.1. Spatial discretization

In the proposed method the spatial derivatives in the Navier–Stokes equations are approximated by utilizing the general definition of the five-point Lagrange interpolation polynomial

$$y = \sum_{i=1}^{5} \prod_{\substack{j=1 \\ i \neq j}}^{5} \frac{x - x_j}{x_i - x_j} y_i. \tag{9}$$

The discretization procedure is applied in both radial and axial directions after the transformation of the dependent variable, say $\varphi(r, z, t)$, into its pseudo-one- dimensional form, $\varphi(\bar{x}, t)$, in a certain spatial direction $\bar{x}$, for a value of remainder direction, by transforming the two-dimensional array into a one-dimensional array. For the sake of clarity the computer implementation of this transformation can be written as follows for all values along the radial direction at any axial level:

```
do j = 1, jz
   do i = 1,ir
   phi_1d(i) = phi_2d(i, j)
   end do
...
body of the discretization
...
   do i=1,ir
   phi_x_2d(i, j) = phi_x_1d(i)
   end do
end do
```

The pseudo-one-dimensional velocity at any point along the $\bar{x}$-direction can be expressed by writing the Lagrange interpolation formula as

$$\varphi(\bar{x}, t) = \sum_{i=1}^{5} a_i(\bar{x}) \varphi_i(t) \tag{10}$$

where

$$a_i(\bar{x}) = \prod_{\substack{j=1 \\ j \neq i}}^{5} \frac{\bar{x} - \bar{x}_j}{\bar{x}_i - \bar{x}_j}. \tag{11}$$

Differentiation of (10) with respect to $\bar{x}$ leads to

$$\frac{\mathrm{d}}{\mathrm{d}\bar{x}}(\varphi(\bar{x}, t)) = \sum_{i=1}^{5} \frac{\mathrm{d}}{\mathrm{d}\bar{x}}(a_i(\bar{x}))\varphi_i(t) = \sum_{i=1}^{5} b_i(\bar{x})\varphi_i(t) = \varphi_{\bar{x}}(\bar{x}, t). \tag{12}$$

The second derivative of (10) with respect to $\bar{x}$ can be easily obtained from

$$\frac{\mathrm{d}^2}{\mathrm{d}\bar{x}^2}(\varphi(\bar{x}, t)) = \frac{\mathrm{d}}{\mathrm{d}\bar{x}}(\varphi_{\bar{x}}(\bar{x}, t)) = \varphi_{\bar{x}\bar{x}}(\bar{x}, t). \tag{13}$$

Thus the five-point Lagrange interpolation polynomial forms the mathematical basis for the calculation of weighting coefficients of the differentiation formulae. In the case of using equal intervals, the weighting coefficients $b_i(\bar{x})$ becomes the elements of the Bickley matrix[4] with the factor $1/(4!\Delta x)$. Substitution of the spatial derivatives into the Navier–Stokes equations leads to the following coupled system of ODEs in time:

$$\frac{\mathrm{d}\bar{\Omega}}{\mathrm{d}t} = \bar{F}(\bar{\Omega}), \tag{14}$$

where

$$\bar{\Omega} = [u_{1,1}, u_{1,2}, \ldots, u_{IR,JZ}, v_{1,1}, v_{1,2}, \ldots, v_{IR,JZ}]^{\mathrm{T}}, \tag{15}$$

$$\bar{F} = [F_{1,1}, F_{1,2}, \ldots, F_{IR,JZ}]^{\mathrm{T}}. \tag{16}$$

Here IR and JZ are the numbers of grid points in the r- and z-direction respectively.

3.2. *Treatment of pressure gradient*

As already noted, the computation of pressure is the most difficult and most time-consuming part of the overall solution of the Navier–Stokes equations and there are various pressure correction methods which are applicable to both stationary and time-dependent incompressible flow equations. Basically, most of them involve an iterative procedure between the velocity and pressure fields through the solution of a Poisson-type equation for the pressure to satisfy the global mass flow constraint and divergence-free condition for confined flows. Existing methods using the primitive variables approach can be classified into three categories, namely pressure-based finite volume methods,[5,6] projection (fractional step) methods[7,8] and artificial compressibility methods.[9]

Among the pressure-based finite volume methods, the most well-known algorithm is the SIMPLE method of Patankar and Spalding.[5] There are also some variants of the method, namely SIMPLER[5] and SIMPLEC.[6] In this method, for an initial approximation of pressure, the momentum equations are solved to obtain a tentative velocity field which actually does not satisfy the divergence-free condition of velocity. In order to satisfy this condition, the obtained velocity field is inserted into the continuity equation to obtain a Poisson-type equation for the pressure correction and this is followed by the application of a cyclic series of guess-and-correct procedures on pressure and velocity fields.

Fractional step methods, on the other hand, follow the Chorin algorithm.[7] Another widely used algorithm for the solution of the unsteady Navier– Stokes equations is the 'pressure implicit by splitting of operators' (PISO) algorithm proposed by Issa,[8] which is a similar approach to fractional step methods. In this approach a predictor–corrector-type scheme is adopted. In spite of the fact that the PISO algorithm does not require any iteration on pressure, it requires the solution of a Poisson-type equation for the pressure twice at each time step. Therefore this algorithm is not suitable for an efficient and time-accurate Navier–Stokes code.

The artificial compressibility method proposed by Chorin[9] is an alternative non-iterative method which does not require the solution of a Poisson-type equation for the pressure. The principle of the method is to obtain the steady state solution of the Navier–Stokes equations from the transient form of the equations in the limit when $t \to \infty$. It depends upon the use of a fictitious state equation $p/\rho = \beta^2$, where β is an artificial sound speed. This method perturbs the continuity equation as

$$\frac{\partial p}{\partial t} = -\beta^2 \rho \left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r} + \frac{v}{r} \right), \tag{17}$$

which is actually true only for the steady state. Hence the results predicted by the artificial compressibility method have no physical meaning before the steady state is reached. Furthermore, since the pressure is considered as a dependent variable in the artificial compressibility method, it brings a certain amount of burden to the ODE solver; as the number of grid points increases more equations have to be solved. The additional drawbacks of this method are the selection of a proper artificial compressibility factor, which actually varies from case to case, and the associated stability problem.

In the present paper a non-iterative procedure for the calculation of pressure is applied. In the transient solution of the Navier–Stokes equations the streamwise pressure gradient must be known in such a way that the mass conservation at each cross-section is satisfied. In order to accomplish this, the static pressure $p(r, z, t)$ in the Navier–Stokes equations is split into two parts as suggested by Raithby and Schneider:[10]

$$p(r, z, t) = \hat{p}(z, t) + \tilde{p}(r, z, t). \tag{18}$$

The physical assumption in this decoupling procedure is that $\partial\tilde{p}/\partial z$ is very small compared with $\partial\hat{p}/\partial z$. When the pressure field is split into two in this manner, equations (1) and (2) can be written as

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial z} - v\frac{\partial u}{\partial r} - \frac{1}{\rho}\frac{\partial \hat{p}}{\partial z} + \nu\left(\frac{\partial^2 u}{\partial z^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{\partial^2 u}{\partial r^2}\right), \tag{19}$$

$$\frac{\partial v}{\partial t} = -u\frac{\partial v}{\partial z} - v\frac{\partial v}{\partial r} - \frac{1}{\rho}\frac{\partial \tilde{p}}{\partial r} + \nu\left(\frac{\partial^2 v}{\partial z^2} + \frac{1}{r}\frac{\partial v}{\partial r} + \frac{\partial^2 v}{\partial r^2} - \frac{v}{r^2}\right). \tag{20}$$

The pressure gradient in (19), $\partial\hat{p}/\partial z$, is determined with the aid of global mass flow constraint combined with the discretized form of the z-momentum equation as follows:

$$u_{i,j}^{n+1} = \Phi_{i,j}^n + \left(\frac{\partial \hat{p}}{\partial z}\right)_j^n \Psi^n, \tag{21}$$

where

$$\Phi_{i,j}^n = u_{i,j}^n - \Delta t\left\{u_{i,j}^n\left(\frac{\partial u}{\partial z}\right)_{i,j}^n + v_{i,j}^n\left(\frac{\partial u}{\partial r}\right)_{i,j}^n - \nu\left[\left(\frac{\partial^2 u}{\partial z^2}\right)_{i,j}^n + \frac{1}{r}\left(\frac{\partial u}{\partial r}\right)_{i,j}^n + \left(\frac{\partial^2 u}{\partial r^2}\right)_{i,j}^n\right]\right\}, \tag{22}$$

$$\Psi^n = -\frac{\Delta t}{\rho^n}. \tag{23}$$

Equation (21) is then multiplied by the density ρ^{n+1} and the resulting equation is subsequently integrated numerically over the cross- sectional area perpendicular to the streamwise direction. This yields

$$\int_0^{2\pi}\int_0^R \rho^{n+1} u_{i,j}^{n+1} r\, dr\, d\theta = \dot{m} = \int_0^{2\pi}\int_0^R \rho^{n+1}\Phi_{i,j}^n r\, dr\, d\theta + \left(\frac{\partial \hat{p}}{\partial z}\right)_j^n \int_0^{2\pi}\int_0^R \rho^{n+1}\Psi^n r\, dr\, d\theta. \tag{24}$$

Note that the density ρ^{n+1} is not known *a priori*. For strictly incompressible flow it is logical to set $\rho^{n+1} = \rho^n$. However, if the flow is even slightly compressible, then the density ρ^{n+1} should be determined using values ρ^{n-1} and ρ^n by applying an accurate extrapolation method. Since the mass flow is prespecified by the problem inlet boundary condition, the pressure gradient $\partial\hat{p}/\partial z$ can be determined as

$$\left(\frac{\partial \hat{p}}{\partial z}\right)_j^n = \frac{2\pi\rho^n \int_0^R \Phi_{i,j}^n r\, dr - \dot{m}}{\pi R^2 \Delta t}. \tag{25}$$

On the other hand, the pressure gradient in (20), $\partial\tilde{p}/\partial r$, should be determined in such a way that it satisfies the divergence-free condition of velocity for incompressible flow. In order to accomplish this, the general procedure to obtain the Poisson-type equation for the pressure is started. However, since the static pressure is split into parts by (18), the resulting equation, which can be written as follows, is not a Poisson-type equation:

$$\frac{1}{\rho}\left[\frac{\partial^2\hat{p}}{\partial z^2}+\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\tilde{p}}{\partial r}\right)\right]=\Re, \tag{26}$$

where

$$\Re=-\frac{\partial\Gamma}{\partial t}-\kappa+\nu\left(\frac{\partial^2\Gamma}{\partial z^2}+\frac{1}{r}\frac{\partial\Gamma}{\partial r}+\frac{\partial^2\Gamma}{\partial r^2}\right), \tag{27}$$

with

$$\kappa=\frac{\partial^2u^2}{\partial z^2}+\frac{1}{r}\frac{\partial^2}{\partial r^2}(rv^2)+\frac{2}{r}\frac{\partial^2}{\partial r\partial z}(ruv), \tag{28}$$

$$\Gamma=\frac{\partial u}{\partial z}+\frac{1}{r}\frac{\partial}{\partial r}(rv). \tag{29}$$

Note that if the pressure were not split into parts, then the following Poisson-type equation for the pressure would have been obtained which would require solution at each time step, resulting in an expensive procedure:

$$\frac{1}{\rho}\nabla^2p=\Re. \tag{30}$$

Since the pressure gradients rather than the pressures are actually needed in the solution of the incompressible Navier–Stokes equations, equation (26) is rearranged as

$$\left(\frac{\partial\tilde{p}}{\partial r}\right)_{i,j}^{n}=\frac{(1/\Delta r_i)(\partial\tilde{p}/\partial r)_{i-1,j}^{n}+\rho^n\Re_{i,j}^{n}-(\partial^2\hat{p}/\partial z^2)_i^n}{1/\Delta r_i+1/r_i}, \quad i=2,\ldots,IR, \quad j=1,\ldots,JZ, \tag{31}$$

where

$$\Delta r_i=r_i-r_{i-1}, \tag{32}$$

to solve for the pressure gradient $\partial\tilde{p}/\partial r$ to be used in the r-component momentum equation. By this formulation it is possible to investigate even detailed transient solutions of the Navier–Stokes equations. In this study, in order not to bring an extra burden to the ODE solver, the r-direction velocity $v(r, z, t)$ is determined with the direct utilization of the continuity equation by the formula

$$v_{i+1,j}^{n}=\frac{r_i}{r_{i+1}}\left[v_{i,j}^{n}-(r_{i+1}-r_i)\left(\frac{\partial u}{\partial z}\right)_{i,j}^{n}\right], \quad i=1,\ldots,IR-2, \quad j=2,\ldots,JZ. \tag{33}$$

Hence, by this formulation, not only is the r-direction velocity $v(r, z, t)$ computed without bringing an extra burden to the ODE solver, but also the divergence-free condition for incompressible flows is satisfied automatically.

3.3. *Time integration*

The integration of the resulting ODEs derived from the discretization of the Navier–Stokes equations is carried out by an implicit algorithm (Adams–Moulton) embedded in the well-known ODE solver LSODES.[11] The implicit nature of the solution method requires some additional discussion. In order to illustrate this, a typical implicit formulation for the solution of ODEs can be written in the form of the backward Euler method as

$$\bar{\Omega}^{n+1} = \bar{\Omega}^{n} + \bar{F}(\bar{\Omega}^{n+1})\Delta t, \tag{34}$$

where $\bar{\Omega}^{n+1}$ and $\bar{F}(\bar{\Omega}^{n+1})$ are the solution and derivative vectors respectively. As can be seen from (34), the derivative vector is evaluated at the next time level. In other words, equation (34) is implicit in the derivative vector $\bar{F}(\bar{\Omega}^{n+1})$. It is this implicit term that gives the method its good stability properties. Therefore the elegance of the MOL is that it shares the advantages of both explicit and implicit methods. In the MOL the spatial derivatives and source terms are evaluated at the previous time level as applied in the explicit approach, so that no linearization problem arises. Furthermore, the solution of the resulting ODEs is carried out by an implicit algorithm such as the implicit Adams–Moulton method,[11] the backward differentiation formula (BDF) method[11] or the implicit Runge–Kutta method.[12] Hence it can be concluded that the MOL has the simplicity of the explicit approach and the power of the implicit approach unless a poor algorithm for the solution of ODEs is adopted.

4. NUMERICAL SOLUTION PROCEDURE

The general algorithm for the solution of the Navier–Stokes equations by using the MOL approach is based upon the evaluation of the derivative vector by which the solution is advanced from one time step to the next. Once the derivative vector has been obtained, the first step in solving the system is to combine the dependent variables into a one-dimensional array. The evaluation of the derivative vector can be summarized as follows.

The complete velocity field satisfying the continuity equation is known *a priori* at the beginning of each cycle, either as a result of the previous cycle or from the prescribed initial conditions for the dependent variables. Once the spatial derivatives appearing in the governing equations have been evaluated using values from the previous cycle, the corresponding pressure gradients along the axial direction are calculated by using equation (25), which ensures that the mass flow is conserved. Then the radial component of the velocity is calculated by direct utilization of the continuity equation, so the divergence-free condition is ensured automatically. Once these calculations have been settled, the derivative vector is calculated over the spatial domain of interest, then it is sent to the ODE solver in the form of a one-dimensional array to compute the dependent variables at the advanced time level. This completes the progression of the solution to the end of the new cycle having the new values of the velocity field. This cyclic procedure is then continued until the steady state is reached.

5. VALIDATION OF THE CODE

Although there are some experimental data on velocity profiles for laminar flows in the tube entrance, those results are open to question owing to the absence of either original data points or a detailed description of the experimental work. Therefore the proposed code was validated with similar results obtained by Hornbeck[13] in the developing region of a circular duct which is uniform in cross-section over its entire length. The internal diameter of the pipe, its length, the mean velocity at the inlet and the Reynolds number based upon the pipe diameter were taken as 1 cm, 15 cm, 4·3 cm s^{-1} and 426 respectively as reported by Agrawal *et al.*[14] The numerical calculations were performed on an IBM

RISC Sys/6000-590. Calculations were carried out for increasing numbers of grid points clustered near the wall, centreline and inlet and the results were found to be independent of grid size beyond 41 and 101 nodes along the radial and axial directions respectively.

Figure 1 shows a comparison of the developing axial velocities at various radial positions in the entrance region of the pipe with the corresponding results of Hornbeck.[13] As can be seen from the figure, the axial velocities in the regions close to the wall develop far more swiftly than those in the core region of the pipe. The predictions obtained by the present authors are in accord with those obtained by Hornbeck.[13] The axial velocity profile for $r/R = 0{\cdot}8$ shows a concave profile close to the entrance. This concavity in the entrance region is also in good agreement with the findings of other workers.[15]

Figure 2 shows the profiles of the radial velocities along the transverse direction at various axial positions. The radial velocities increase away from the wall, reach maxima and then decrease to zero at the pipe centre, as expected. The maximum values of radial velocities show a tendency to decrease along the downstream direction and almost vanish as the flow reaches its fully develop state.

Figure 3 shows the percentage relative errors in mass flow at the pipe exit with respect to the actual mass flow known *a priori* by the problem inlet boundary condition, at various time intervals. For time-accurate Navier–Stokes codes the mass flow should be conserved at each time step. In order to accomplish this, the static pressure in the governing equations is split into parts and then the axial pressure gradient, which is actually the term responsible for satisfying the mass conservation, is calculated with the aid of mass flow constraint. As can be seen from the figure, the percentage relative error is 100 per cent at the beginning because of the stagnant condition at $t = 0$. Since it takes the fluid molecules at the pipe exit a certain time to be energized by the upstream fluid molecules, the satisfaction of the mass flow at the exit takes the same period of time. In order to take this into consideration, the computations should be carried out at very small time intervals. As can be seen from the figure, as the time interval to update the pressure gradient increases, the percentage relative

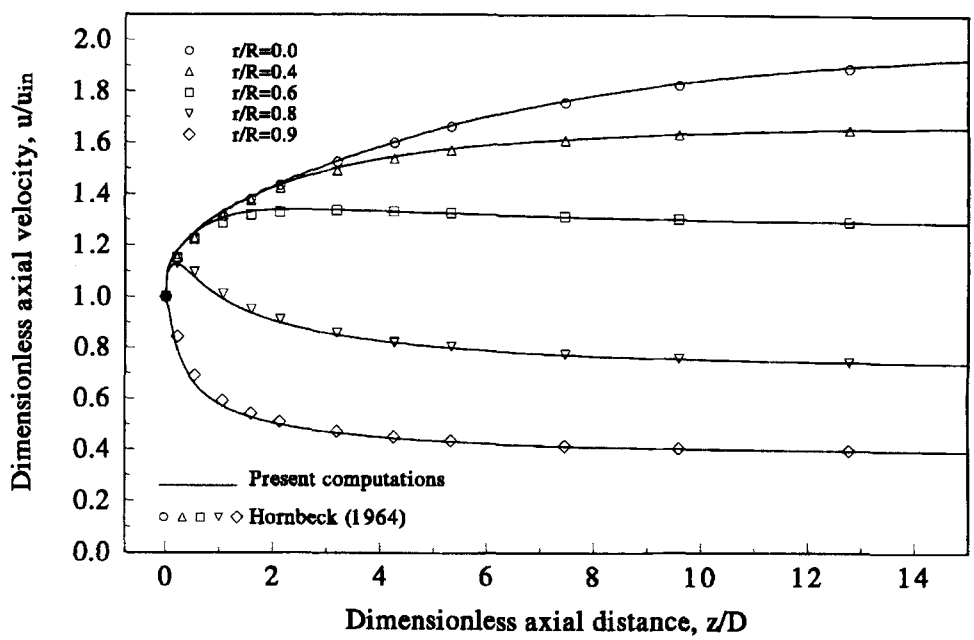

Figure 1. Variation in axial velocity with inlet distance for various radial positions

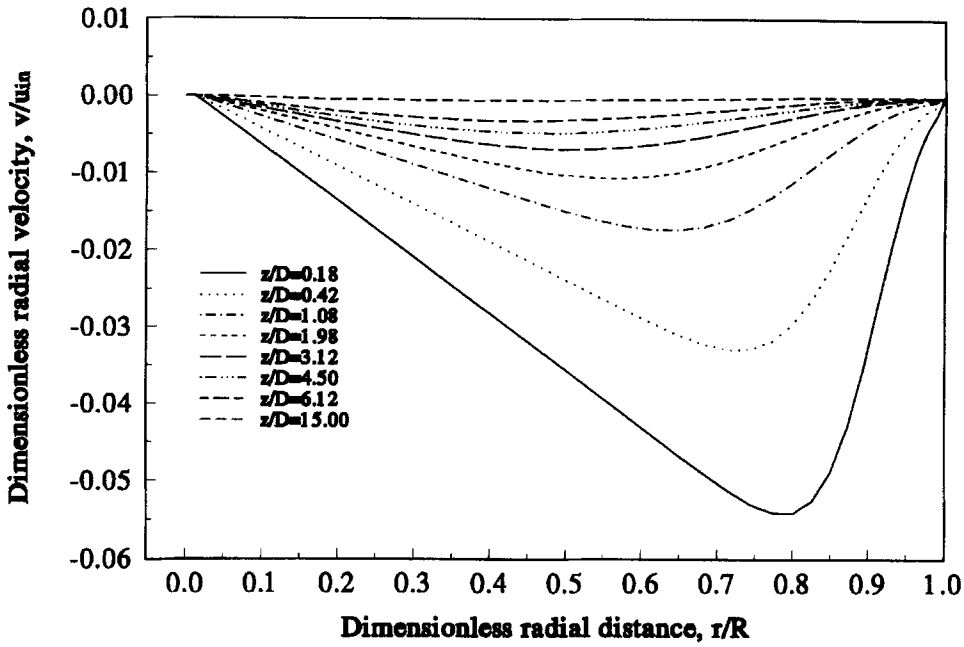

Figure 2. Radial velocity profiles in developing flow

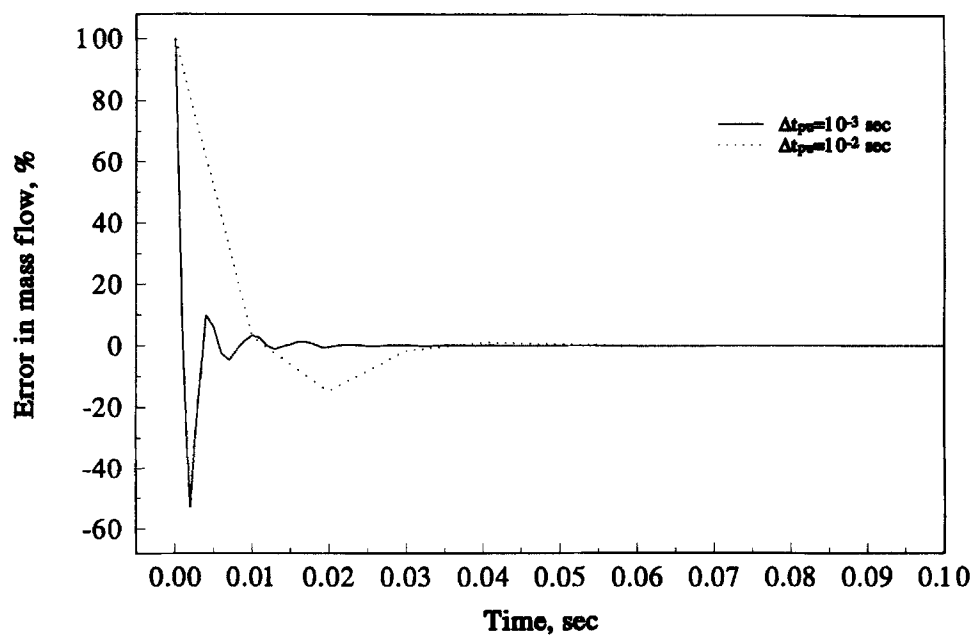

Figure 3. Percentage relative error in mass flow

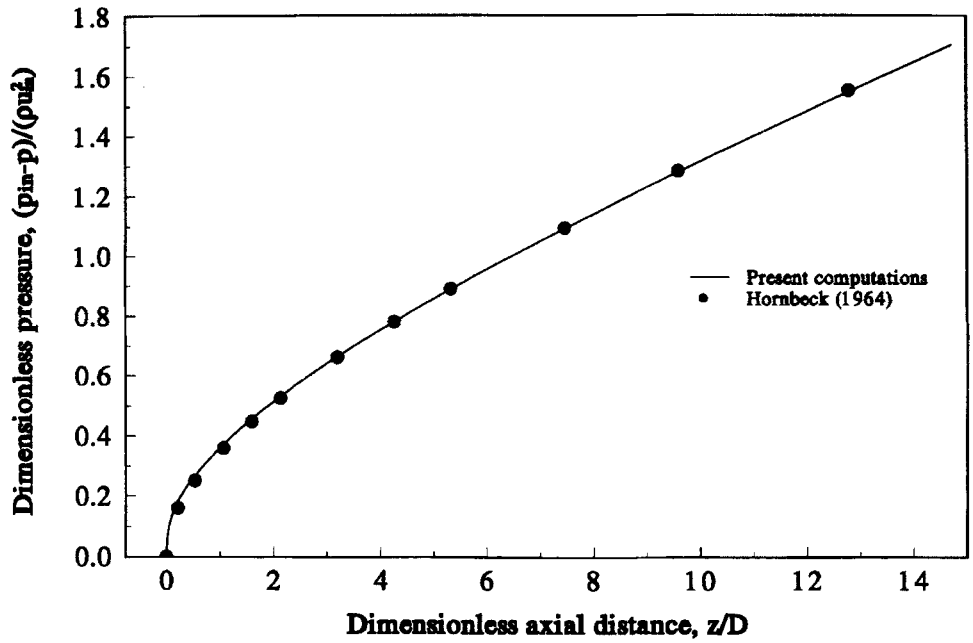

Figure 4. Variation in pressure with axial distance

error takes more time to vanish. However, sophisticated ODE solvers generally adjust the marching step size automatically. Therefore in the present study the output inquiry time interval $\Delta t_{pu} = 10^{-3}$ s was used to update the axial pressure gradient.

Figure 4 shows a comparison of the axial variation in predicted pressure with that obtained by Hornbeck.[13] In the downstream region of the pipe a constant pressure gradient is predicted, whereas in the inlet region of the pipe a higher pressure gradient is obtained to overcome the axial flow development.

As can be seen from Figures 1 and 4, the present results are in good agreement with those obtained by Hornbeck.[13] The small discrepancies between the predictions obtained by the present authors and those obtained by Hornbeck[13] can be attributed to the fact that the results obtained by Hornbeck[13] are based upon boundary layer assumptions.

In order to show the ability of the present code to predict transient solutions, the radial variation in axial velocity at the exit of the pipe at various times is exhibited in Figure 5. As can be seen from the figure, the time development of velocity profiles at the pipe exit shows the expected trend.

6. CONCLUSIONS

The main objective of this study have been to introduce the MOL solution of the time-dependent two-dimensional Navier–Stokes equations and provide a parabolic algorithm which does not require any iteration on the pressure nor solution of a Poisson-type equation for the pressure.

The computational procedure proposed in this study has the simplicity of the explicit approach and the power of the implicit approach and does not require any linearization in the governing equations. The method has been applied to the calculation of transient laminar flow development and pressure drop in the entrance region of a circular duct. In the absence of transient numerical predictions and experimental data, the code has been validated only against steady state numerical predictions

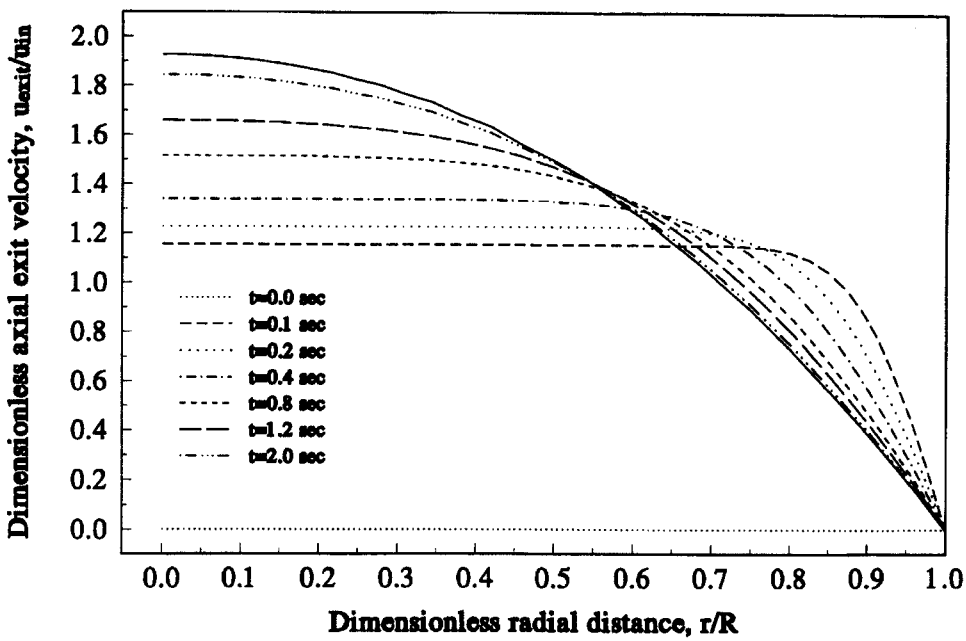

Figure 5. Time development of axial exit velocity profiles (————, steady state)

previously reported in the literature. The good agreement obtained shows that the present code provides a useful tool for the investigation of the time evolution of detailed, unsteady structures in a flow field.

ACKNOWLEDGEMENTS

This study was performed as part of AGARD Project T51/PEP on 'Soot Formation and Radiative Heat Transfer in Combustors'. The support is gratefully acknowledged.

APPENDIX: LIST OF SYMBOLS

a_i	weighting coefficients of dependent variable defined by Lagrange interpolation polynomial
b_i	weighting coefficients of first derivative of dependent variable defined by Lagrange interpolation polynomial
D	diameter
$\bar{F}$	derivative vector of dependent variables
IR	number of nodes in r-direction
IZ	number of nodes in z-direction
$\dot{m}$	mass flow rate
p	static pressure
$\hat{p}$	component of pressure to be used in z-momentum equation
$\tilde{p}$	component of pressure to be used in r-momentum equation
r, z	spatial independent variables in cylindrical co- ordinates
R	radius
t	time

u, v components of velocity vector
$\bar{x}$ generic spatial independent variable
∇^2 Laplacian

Greek letters

β artificial sound speed
Δ increment
ν kinematic viscosity
ρ density
$\underline{\varphi}$ dependent variable transformed into one-dimensional array
$\underline{\Omega}$ solution vector of dependent variables

Subscripts

in inlet
pu pressure update
$\bar{x}$ first derivative of dependent variable
$\bar{x}\bar{x}$ second derivative of dependent variable

Superscripts

n previous time level
$n+1$ next time level

REFERENCES

1. O. Oymak and N. Selçuk, 'MOL vs FDM solutions of an unsteady viscous flow problem', *Proc. Eighth Int. Conf. on Numerical Methods in Laminar and Turbulent Flows*, Swansea 1993, Part 1, pp. 153–162.
2. W. E. Schiesser, *The Numerical Method of Lines: Integration of Partial Differentiation Equations*, Academic, San Diego, CA, 1991.
3. W. E. Schiesser, 'Variable grid spatial differentiator in the numerical method of lines', *DSS/2 Manual No. 6*, Lehigh University, 1988.
4. W. G. Bickley, 'Formulae for numerical differentiation', *Math. Gaz.*, **25**, 19–27 (1941).
5. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, Washington, DC, 1980.
6. J. P. Van Doormaal and G. D. Raithby, 'Enhancements of the SIMPLE method for predicting incompressible fluid flows', *Numer. Heat Transfer*, **7**, 147–163 (1984).
7. A. J. Chorin, 'Numerical solutions of the Navier–Stokes equations', *Math. Comput.*, **22**, 745–762 (1968).
8. R. I. Issa, 'Solution of implicitly discretized fluid flow equation by operator-splitting', *J. Comput. Phys.*, **62**, 40–65 (1985).
9. A. J. Chorin, 'A numerical method for solving incompressible viscous flow problems', *J. Comput. Phys.*, **2**, 12–26 (1967).
10. G. D. Raithby and G. E. Schneider, 'Numerical solution of problems in incompressible fluid flow: treatment of the velocity pressure coupling', *Numer. Heat Transfer*, **2**, 417–440 (1979).
11. A. C. Hindmarsh, 'ODEPACK: a systematized collection of ODE solvers', in *Scientific Computing*, North-Holland, New York, 1983.
12. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II. Stiff and Differential Algebraic Problems*, Springer, New York, 1991.
13. R. W. Hornbeck, 'Laminar flow in the entrance region of a pipe', *Appl. Sci. Res. A*, **13**, 224–232 (1964).
14. A. K. Agrawal, S. Krishnan and T. Yang, 'Use of subdomains for inverse problems in branching flow passages', *J. Fluids Eng.*, **115**, 227–232 (1993).
15. M. Friedmann, J. Gilles and N. Liron, 'Laminar flow in a pipe at low and moderate Reynolds numbers', *Appl. Sci. Res.*, **19**, 426–438 (1968).