THE NUMERICAL SOLUTION OF THE NAVIER-STOKES EQUATIONS FOR AN INCOMPRESSIBLE FLUID[1]

BY ALEXANDRE JOEL CHORIN

Communicated by E. Isaacson, June 21, 1967

A finite difference method for solving the Navier-Stokes equations for an incompressible fluid has been developed. This method uses the primitive variables, i.e. the velocities and the pressure, and is equally applicable to problems in two and three space dimensions. Essentially it constitutes an extension to time dependent problems of the artificial compressibility method introduced in [1] for steady flow problems.

The equations to be solved can be written in the dimensionless form

$$\partial_t u_i + R u_j \partial_j u_i = -\partial_i p + \Delta u_i + E_i, \qquad \Delta \equiv \sum \partial_j^2, \tag{1}$$

$$\partial_j u_j = 0 \tag{2}$$

where u_i are the velocity components, E_i the external forces, p is the pressure, and R the Reynolds number. ∂_i denotes differentiation with respect to the space variable x_i, and ∂_t differentiation with respect to the time t. The summation convention is used in (1) and (2). The equations are to be solved in a domain $\mathfrak{D}$. Problems where the velocities or their derivatives are prescribed at the boundary or where some boundary conditions are replaced by periodicity conditions have been investigated. For simplicity, we shall describe the case where the velocities are given at the boundary and shall assume that a cartesian coordinate system is used. An important feature of the method is the use of equation (2) rather than a derived equation (see e.g. [2]) for determining the pressure. This makes it possible to satisfy the equation of continuity at the boundary and determine the pressure in a natural way.

Let $Du=0$ be a difference approximation to $\partial_j u_j = 0$. Du takes a different form in the interior of $\mathfrak{D}$ where first order centered differences are used, and at the boundary where one-sided differences up to second order are used to insure second order accuracy. It is assumed that at the time $t = n\Delta t$, velocity and pressure fields u_i^n and p^n, where $u^n \equiv u(n\Delta t)$ and $p^n \equiv (n\Delta t)$, are given, satisfying $Du^n = 0$. The task at hand is to compute u^{n+1}, p^{n+1} from equation (1), so that $Du^{n+1} = 0$.

Auxiliary fields u_i^{aux} are first computed through

[1] This work was partially supported by the U. S. Atomic Energy Commission, Contract No. AT(30-1)-1480.

$$\overset{\text{aux}}{u_i} = \overset{n}{u_i} + \Delta t F_i u$$

where $F_i u$ approximates

$$-Ru_j\partial_j u_i + \Delta u_i + E_i, \qquad \Delta \equiv \sum \partial_j^2.$$

$F_i u$ may depend on u_i^{aux}, u_i^n, and intermediate fields, say u^*, u^{**}, etc. In general, an implicit alternating direction scheme is used to find the fields u^{aux}, u^*, etc. One of several versions of that method is the following (see Samarski [3]): Let Δx_i, $i=1, 2, 3$, be the space variable increments, let $u_{i(q,r,s)}$ denote $u_i(q\Delta x_1, r\Delta x_2, s\Delta x_3)$, and let $E_{i(q,r,s)}$ denote $E_i(q\Delta x_1, r\Delta x_2, s\Delta x_3)$. u_1^{aux} is then computed through

$$
\begin{aligned}
\overset{*}{u}_{1(q,r,s)} &= \overset{n}{u}_{1(q,r,s)} - R\frac{\Delta t}{2\Delta x_1}\overset{n}{u}_{1(q,r,s)}(\overset{*}{u}_{1(q+1,r,s)} - \overset{*}{u}_{1(q-1,r,s)}) \\
&\quad + \frac{\Delta t}{\Delta x_1^2}(\overset{*}{u}_{1(q+1,r,s)} + \overset{*}{u}_{1(q-1,r,s)} - 2\overset{*}{u}_{1(q,r,s)}), \\
\overset{**}{u}_{1(q,r,s)} &= \overset{*}{u}_{1(q,r,s)} - R\frac{\Delta t}{2\Delta x_2}\overset{*}{u}_{2(q,r,s)}(\overset{**}{u}_{1(q,r+1,s)} - \overset{**}{u}_{1(q,r-1,s)}) \\
&\quad + \frac{\Delta t}{\Delta x_2^2}(\overset{**}{u}_{1(q,r+1,s)} + \overset{**}{u}_{1(q,r-1,s)} - 2\overset{**}{u}_{1(q,r,s)}), \\
\overset{\text{aux}}{u}_{1(q,r,s)} &= \overset{**}{u}_{1(q,r,s)} - R\frac{\Delta t}{2\Delta x_3}\overset{**}{u}_{3(q,r,s)}(\overset{\text{aux}}{u}_{1(q,r,s+1)} - \overset{\text{aux}}{u}_{1(q,r,s-1)}) \\
&\quad + \frac{\Delta t}{\Delta x_3^2}(\overset{\text{aux}}{u}_{1(q,r,s+1)} + \overset{\text{aux}}{u}_{1(q,r,s-1)} - 2\overset{\text{aux}}{u}_{1(q,r,s)}) \\
&\quad + \Delta t E_{1(q,r,s)},
\end{aligned} \tag{3}
$$

with similar expressions for u_2^{aux}, u_3^{aux}.

So far the pressure term in (1) and equation (2) have not been taken into account. An iteration procedure is now introduced to find u_i^{n+1} inside $\mathfrak{D}$ and p^{n+1} in $\mathfrak{D}$ and on its boundary by setting

$$\overset{n+1,1}{p} = \overset{n}{p}, \tag{4a}$$

$$\overset{n+1,m+1}{p} - \overset{n+1,m}{p} = -\lambda D\overset{n+1,m+1}{u}, \tag{4b}$$

$$\overset{n+1,m+1}{u_i} = \overset{\text{aux}}{u_i} - \Delta t G_i^m p \qquad (m \geqq 1) \tag{4c}$$

where λ is a parameter, the quantities $u_i^{n+1,m}$ and $p^{n+1,m}$ are successive approximations to u_i^{n+1} and p^{n+1}, and $G_i^m p$ is a function of $p^{n+1,m+1}$ and $p^{n+1,m}$ which tends to a difference form of $\partial_i p^{n+1}$ as $|p^{n+1,m+1} - p^{n+1,m}|$ tends to zero. The form of $G_i^m p$ is crucial for the accuracy and rapid convergence of the method and will be defined below in equations (5a) and (5b).

When for some l and some small predetermined constant ϵ

$$\max_{\mathfrak{D}} \left| p^{n+1,l+1} - p^{n+1,l} \right| \leqq \epsilon,$$

we define

$$u_i^{n+1} = u_i^{n+1,l+1}, \qquad p^{n+1} = p^{n+1,l+1}.$$

If the velocity component u_i is prescribed at the boundary, $u_i^{n+1,m+1}$ and u_i^{aux} in (4b) and (4c) are replaced by the given value of u_i. The iterations (4b) and (4c) insure that equation (1) is satisfied inside the domain $\mathfrak{D}$ and equation (2) is satisfied in $\mathfrak{D}$ and at the boundary.

The iteration (4) is carried out as follows: Let $p_{(q,r,s)}$ denote $p(q\Delta x_1, r\Delta x_2, s\Delta x_3)$.

$$p_{(q,r,s)}^{n+1,m+1}, \quad u_{1(q\pm 1,r,s)}^{n+1,m+1}, \quad u_{2(q,r\pm 1,s)}^{n+1,m+1}, \quad u_{3(q,r,s\pm 1)}^{n+1,m+1}$$

are to be evaluated simultaneously; $p_{q,r,s}^{n+1,m+1}$ using equation (4b); $u_{1(q\pm 1,r,s)}^{n+1,m+1}$ using the formulas

$$\text{(5a)} \quad u_{1(q+1,r,s)}^{n+1,m+1} = u_{1(q+1,r,s)}^{\text{aux}} - \frac{\Delta t}{2\Delta x_1}\left(p_{(q+2,r,s)}^{n+1,m} - \frac{1}{2}\left(p_{(q,r,s)}^{n+1,m+1} + p_{(q,r,s)}^{n+1,m}\right)\right),$$

$$\text{(5b)} \quad u_{1(q-1,r,s)}^{n+1,m+1} = u_{1(q-1,r,s)}^{\text{aux}} - \frac{\Delta t}{2\Delta x_1}\left(\frac{1}{2}\left(p_{(q,r,s)}^{n+1,m+1} + p_{(q,r,s)}^{n+1,m}\right) - p_{(q-2,r,s)}^{n+1,m}\right),$$

with similar expressions for u_2, u_3. Natural modifications of these expressions are introduced near the boundary, where some of the u_i are prescribed and where Du employs noncentered differences. From (4b), (5a), (5b) and the analogous expressions for u_2 and u_3, we obtain a system of seven equations in the seven unknowns

$$u_{1(q\pm 1,r,s)}^{n+1,m+1}, \quad u_{2(q,r\pm 1,s)}^{n+1,m+1}, \quad u_{3(q,r,s\pm 1)}^{n+1,m+1} \quad \text{and} \quad p_{(q,r,s)}^{n+1,m+1}.$$

which can be solved explicitly. (There is no need to evaluate the $u_i^{n+1,m+1}$ until after the $p^{n+1,m}$ have converged, since the $u_i^{n+1,m+1}$ can be eliminated from the seven equations.)

The resulting iteration procedure converges for all $\lambda > 0$ under natural restrictions on the domain $\mathfrak{D}$, and a best value of λ, say λ_{opt}, can be determined. For this iteration scheme it can be expected that $\epsilon = O(\Delta t)$ is sufficient to insure the overall accuracy of the scheme to

$O(\Delta t)$, $0 \leqq t \leqq T$. This results in a reduction of the amount of computational labor. It is assumed that $\Delta t = O(\Delta x^2)$.

That the scheme is indeed accurate to $O(\Delta t)$ has been verified on certain simple test problems for which an analytical solution can be obtained. In one such problem, used as test problem by Pearson [4] for a vorticity-stream function method, $\mathfrak{D}$ is a two dimensional square $0 \leqq x_1 \leqq \pi$, $0 \leqq x_2 \leqq \pi$. $R = 1$, $E_1 = E_2 = 0$, and the exact solution is

$$u_1 = -\cos x_1 \sin x_2 e^{-2t}, \qquad u_2 = \sin x_1 \cos x_2 e^{-2t},$$
$$p = -\tfrac{1}{4}(\cos 2x_1 + \cos 2x_2)e^{-4t}$$

where appropriate boundary conditions and initial data are used. Accurate results have been obtained at the price of a modest amount of computing effort. For $\Delta x_1 = \Delta x_2 = \pi/19 = \Delta x$, $\Delta t/\Delta x^2 = 2$, the maximum relative error in u_1, u_2 over $\mathfrak{D}$ after one step is less than 0.08%; and after sixteen steps, less than 0.02%, using $\lambda = \lambda_{\text{opt}} = \Delta x^2/(\Delta t \sin 2\Delta x)$, $\epsilon = \Delta x^3$; the largest l needed is 6.

The method is presently being applied to the problem of wave number selection and finite amplitude stability in a convective layer (the Benard problem, see e.g. [5]) and to a numerical study of energy cascading between large and small eddies in a model of turbulence (the Green-Taylor problem [6]). The results of these studies and further details on the numerical method will be presented in a forthcoming paper and will point out crucial differences between the behavior of solutions of the Navier-Stokes equations in two and three space dimensions.

Bibliography

1. A. J. Chorin, *A numerical method for solving incompressible viscous flow problems*, J. Comp. Physics (to appear).

2. F. H. Harlow and J. E. Welch, *Numerical calculations of time dependent viscous incompressible flow of fluid with a free surface*, Phys. Fluids **8** (1965), 2182.

3. A. A. Samarski, *On an economical difference method for the solution of a multi-dimensional parabolic problem in an arbitrary region*, U.S.S.R. Comput. Math. and Math. Phys. **5** (1963), 894.

4. C. E. Pearson, *A computational method for time dependent two dimensional incompressible viscous flow problems*, Sperry-Rand Research Center, Sudbury, Mass., Report No. SRRC-RR-64-17 (1964).

5. A. J. Chorin, *Numerical study of thermal convection in a fluid layer heated from below*, AEC Report No. NYO-1480-61, New York University (1966).

6. G. I. Taylor and A. E. Green, *Mechanism of the production of small eddies from large ones*, Proc. Roy. Soc. A **158** (1937), 499.

Courant Institute of Mathematical Sciences, New York University