JOURNAL OF COMPUTATIONAL PHYSICS 92, 369–379 (1991)

An Improvement of Fractional Step Methods for the Incompressible Navier–Stokes Equations

HUNG LE AND PARVIZ MOIN

Center for Turbulence Research, Stanford University, Stanford, California 94305-3030

Received April 4, 1989; revised January 10, 1990

A numerical method for computing three-dimensional, unsteady incompressible flows is presented. The method is a predictor–corrector technique combined with a fractional step method. Each time step is advanced in three sub-steps. The novel feature of the present scheme is that the Poisson equation for the pressure is solved only at the final sub-step resulting in substantial savings in computing time. It is shown that the method allows a larger CFL number and reduces the computing cost without loss of accuracy by satisfying the continuity equation only at the last sub-step. Numerical solutions for the decaying vortices and flow over a backward-facing step are obtained and compared with analytical and other numerical results.

1. INTRODUCTION

For three-dimensional, time-dependent incompressible Navier–Stokes equations, Kim and Moin [1] developed a fractional step, or time-splitting, scheme in conjunction with the approximate factorization technique. The convective terms were advanced using the second-order-explicit Adams–Bashforth scheme and the second-order-implicit Crank–Nicolson was used for the viscous terms. This method is second-order accurate in space and time.

The objective of this paper is to present a modification to Kim and Moin's (KM) scheme that reduces the restrictions on the time step with simultaneous reduction in computing effort at each time step. The method is based on a predictor–corrector algorithm which is of the Runge–Kutta genre [2, 3]. Every time step is advanced in three sub-steps, each of which uses an explicit treatment in the convective terms and implicit in the viscous terms. The KM time-splitting method is incorporated at each sub-step. It is well known that the predictor–corrector scheme allows a higher stability limit, CFL, based on the full time step. However, the time-splitting method requires that the Poisson equation for the pressure be solved at each sub-step. The main novelty of the present scheme is that the velocity field is advanced through the sub-steps without satisfying the continuity equation. The Poisson equation is used to project the predicted vector field into a divergence-free velocity field only at the final sub-step. *A modification to the terms in the Navier–Stokes equation is necessary because conservation of mass is not satisfied at the sub-steps.* The boundary condi-

0021-9991/91 $3.00

tions for the intermediate velicity field are derived accordingly using the method similar to that of LeVeque and Oliger [4]. The present method is developed for a staggered grid [5].

Section 2 briefly presents the numerical procedures of combined Runge–Kutta and fractional step method. Section 3 shows the modification for the present method. The derivation for the boundary conditions for the intermediate velocity field is presented in Section 4. Section 5 provides the numerical results for two-dimensional decaying vortices in a box and two-dimensional flow over a backward facing step; a summary is presented in Section 6.

2. Algorithms for the Combined Runge–Kutta and Fractional Step

In this section, we incorporate a predictor–corrector scheme (of the Runge–Kutta genre) to the KM time-splitting method. That is, we simply replace the Adams–Bashforth/Crank–Nicolson combination used by KM with a semi-implicit Runge–Kutta type scheme. The three-dimensional Navier–Stokes and continuity equations are non-dimensionalized by a characteristic length and velocity scale:

$$\frac{\partial u_i}{\partial t} = -\frac{\partial}{\partial x_j} u_i u_j + \frac{1}{\mathrm{Re}} \frac{\partial^2 u_i}{\partial x_j \partial x_j} - \frac{\partial P}{\partial x_i}, \qquad i = 1, 2, 3, \tag{2.1}$$

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{2.2}$$

A three-step time advancement scheme for Eqs. (2.1) and (2.2) can be written as [3]:

$$\frac{u_i^k - u_i^{k-1}}{\Delta t} = \alpha_k L(u_i^{k-1}) + \beta_k L(u_i^k) - \gamma_k N(u_i^{k-1}) - \zeta_k N(u_i^{k-2}) - (\alpha_k + \beta_k) \frac{\delta P^k}{\delta x_i}, \qquad i = 1, 2, 3, \tag{2.3}$$

$$\delta u_i^k / \delta x_i = 0, \tag{2.4}$$

where $k = 1, 2, 3$ denotes the sub-step number, $k-2$ is ignored for $k = 1$; u_i^0 and u_i^3 are the velocities at time step n and $n+1$; $\delta/\delta x_i$ is the finite difference operator; and $L(u_i)$ and $N(u_i)$ represent second-order finite difference approximations to the viscous and convective terms, respectively:

$$L(u_i) = \frac{1}{\mathrm{Re}} \frac{\delta^2 u_i}{\delta x_j \delta x_j}$$

$$N(u_i) = \frac{\delta}{\delta x_j} u_i u_j.$$

Equation (2.3) represents the time advancement of the Navier–Stokes equation at each sub-step. The coefficients α_k, β_k, γ_k, and ζ_k, $k = 1, 2, 3$ are constants selected such that the total time advancement between t^n and t^{n+1} is third-order accurate for the convective term and second-order for the viscous term [3]. These coefficients are:

$$\gamma_1 = \tfrac{8}{15} \qquad \gamma_2 = \tfrac{5}{12} \qquad \gamma_3 = \tfrac{3}{4}$$

$$\zeta_1 = 0 \qquad \zeta_2 = -\tfrac{17}{60} \qquad \zeta_3 = -\tfrac{5}{12}$$

$$\alpha_1 = \beta_1 = \tfrac{4}{15}$$

$$\alpha_2 = \beta_2 = \tfrac{1}{15}$$

$$\alpha_3 = \beta_3 = \tfrac{1}{6}$$

$$\sum_{k=1}^{3} (\alpha_k + \beta_k) = \sum_{k=1}^{3} (\gamma_k + \zeta_k) = 1.$$

As shown in [2], this time advancement scheme, which belongs to the family of Runge–Kutta schemes, requires two storage locations per variable at each step. At each sub-step, the convective term (N) is advanced explicitly, and the viscous term (L) advancement is implicit. Strictly, the overall time-advancement is third-order on N and second order on L. The numerical stability is restricted by the explicit treatment of the convective terms. The stability limit, $\mathrm{CFL} = \Delta t(|u(\max)_i|/\Delta x_i)$, is $\sqrt{3}$ based on the total time step Δt. This allows a larger time step than the KM method at the expense of three evaluations of the non-linear and viscous terms per step.

Since the divergence free condition is enforced at all sub-steps by Eq. (2.4), each sub-step requires as much work as one Δt advancement. The effective time step for (2.3) is $(\alpha_k + \beta_k)\,\Delta t$. We now apply the fractional step of KM to (2.3) and (2.4). As shown in [1], the accuracy of the fractional step method is second-order in Δt, provided that the primary time advancement schemes (e.g., Adams–Bashforth and Crank–Nicolson) are second-order accurate. Thus second-order accuracy is retained when this method is used in conjunction with schemes such as Runge–Kutta method. The result is

$$\frac{\hat{u}_i^k - u_i^{k-1}}{\Delta t} = (\alpha_k + \beta_k) L(u_i^{k-1}) + \beta_k L(\hat{u}_i^k - u_i^{k-1}) - \gamma_k N(u_i^{k-1}) - \zeta_k N(u_i^{k-2}) \tag{2.5}$$

$$\frac{u_i^k - \hat{u}_i^k}{\Delta t} = -\frac{\delta \phi^k}{\delta x_i}, \tag{2.6}$$

where ϕ^k and P^k are related by

$$\frac{\delta \phi^k}{\delta x_i} = (\alpha_k + \beta_k) \frac{\delta P^k}{\delta x_i} - \beta_k L(u_i^k - \hat{u}_i^k). \tag{2.7}$$

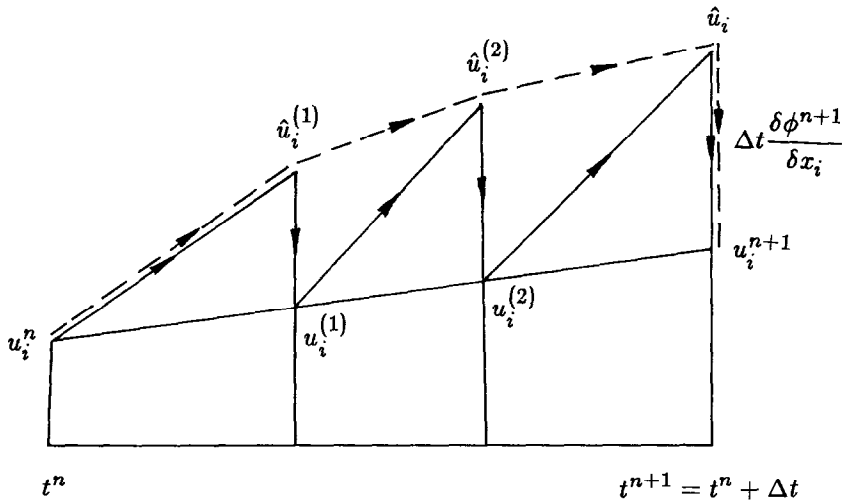

FIG. 1. Combined Runge–Kutta and fractional-step time advancement: ——— original scheme; – – – present scheme.

Equation (2.6) combined with the divergence free constraint (2.4) gives the discrete Poisson equation:

$$\frac{1}{\Delta t}\frac{\delta \hat{u}_i^k}{\delta x_i} = \frac{\delta^2 \phi^k}{\delta x_i \, \delta x_i}. \tag{2.8}$$

Equations (2.5) and (2.8) are solved at each sub-step for u_i^k and ϕ^k. The schematic diagram showing the steps taken with the above scheme is shown in Fig. 1. Note that the vertical arrows represent enforcement of the continuity equation at each sub-step.

The advantages of the scheme in Fig. 1 are: (a) allowing relatively large time steps (CFL $= \sqrt{3}$) and (b) requiring low memory storage [2]. However, all calculations in Eqs. (2.5) and (2.8) must be repeated at each sub-step. This causes a significant increase in computing cost, particularly in solving the Poisson equation. We will next modify Eq. (2.5) to eliminate the need for the Poisson equation at the sub-steps.

3. Modification of Velocity Advancement

Equations (2.5) and (2.6) as applied to the first sub-step are

$$\frac{\hat{u}_i^1 - u_i^n}{\Delta t} = (\alpha_1 + \beta_1) L(u_i^n) + \beta_1 L(\hat{u}_i^1 - u_i^n) - \gamma_1 N(u_i^n) \tag{3.1}$$

$$\frac{u_i^1 - \hat{u}_i^1}{\Delta t} = -\frac{\delta \phi^1}{\delta x_i}. \tag{3.2}$$

At the second sub-step, Eq. 3.3 below represents the advancement without time-splitting:

$$\frac{u_i^2 - u_i^1}{\Delta t} = \alpha_2 L(u_i^1) + \beta_2 L(u_i^2) - \gamma_2 N(u_i^1) - \zeta_2 N(u_i^n) - (\alpha_2 + \beta_2)\frac{\delta P^2}{\delta x_i}. \tag{3.3}$$

Substituting for u_i^1 in (3.3) using (3.2), and time-splitting yield the following algorithms for the second substep:

$$\frac{\hat{u}_i^2 - \hat{u}_i^1}{\Delta t} = \frac{\alpha_2 + \beta_2}{\mathrm{Re}}\frac{\delta^2 \hat{u}_i^1}{\delta x_j \delta x_j} - \frac{\alpha_2}{\mathrm{Re}}\frac{\delta}{\delta x_i}\left(\frac{\delta \hat{u}_j^1}{\delta x_j}\right) + \frac{\beta_2}{\mathrm{Re}}\frac{\delta^2}{\delta x_j \delta x_j}(\hat{u}_i^2 - \hat{u}_i^1)$$
$$- \gamma_2 \frac{\delta}{\delta x_j}\left[\left(\hat{u}_i^1 - \Delta t \frac{\delta \phi^1}{\delta x_i}\right)\left(\hat{u}_j^1 - \Delta t \frac{\delta \phi^1}{\delta x_j}\right)\right] - \zeta_2 \frac{\delta}{\delta x_j} u_i^n u_j^n \tag{3.4}$$

and

$$\frac{u_i^2 - \hat{u}_i^2}{\Delta t} = -\frac{\delta \phi^1}{\delta x_i} - (\alpha_2 + \beta_2)\frac{\delta P^2}{\delta x_i} + \frac{\beta_2}{\mathrm{Re}}\frac{\delta^2}{\delta x_j \delta x_j}(u_i^2 - \hat{u}_i^2)$$
$$= -\frac{\delta \phi^2}{\delta x_i}. \tag{3.5}$$

Note that, in Eq. (3.4), $\hat{u}_i$ denotes the velocity field without the pressure correction; whereas u_i in Eq. (3.3) is the divergence-free velocity field.

Without solving the Poisson equation at the first sub-step, $\Delta t\, \delta\phi^1/\delta x_i$ in Eq. (3.4) is unknown. This term is approximated by the gradient of ϕ at the end of the previous time step, $\delta\phi^n/\delta x_i$. It can easily be shown that this substitution maintains the second-order accuracy of the overall scheme. From Eq. (2.7)

$$\Delta t \frac{\delta \phi^1}{\delta x_i} = (\alpha_1 + \beta_1)\Delta t \frac{\delta P^1}{\delta x_i} - \beta_1 \Delta t\, L(u_i^1 - \hat{u}_i^1)$$
$$= (\alpha_1 + \beta_1)\Delta t \frac{\delta P^1}{\delta x_i} + O(\Delta t^2).$$

But,

$$\frac{\delta P^1}{\delta x_i} = \frac{\delta P^n}{\delta x_i} + \Delta t \frac{\partial}{\partial t}\left(\frac{\delta P^n}{\delta x_i}\right) + O(\Delta t^2).$$

Thus,

$$\Delta t \frac{\delta \phi^1}{\delta x_i} = (\alpha_1 + \beta_1)\Delta t \frac{\delta P^n}{\delta x_i} + O(\Delta t^2).$$

Furthermore, $P = \phi + O(\Delta t)$, [1], the resulting approximation of $\Delta t\, \delta\phi^1/\delta x_i$ which maintains second-order accuracy in time is

$$\Delta t \frac{\delta \phi^{*1}}{\delta x_i} = (\alpha_1 + \beta_1)\, \Delta t \frac{\delta \phi^n}{\delta x_i} + O(\Delta t^2). \tag{3.6}$$

The same procedure is repeated for the final sub-step. The generalization of Eqs. (3.4) and (3.5) for all three sub-steps is

$$\frac{\hat{u}_i^k - \hat{u}_i^{k-1}}{\Delta t} = (\alpha_k + \beta_k) L(\hat{u}_i^{k-1}) - \frac{\alpha_k}{Re} \frac{\delta}{\delta x_i} \left(\frac{\delta \hat{u}_j^{k-1}}{\delta x_j} \right) + \beta_k L(\hat{u}_i^k - \hat{u}_i^{k-1})$$
$$- \gamma_k N(u_i^{*k-1}) - \zeta_k N(u_i^{*k-2}), \qquad k = 1, 2, 3, \tag{3.7}$$

$$\frac{u_i^{*k} - \hat{u}_i^k}{\Delta t} = -\frac{\delta \phi^{*k}}{\delta x_i}, \tag{3.8}$$

where

$$\hat{u}_i^0 \equiv u_i^n, \qquad \hat{u}_i^3 \equiv \hat{u}_i$$

and

$$u_i^{*m} = \hat{u}_i^m - \Delta t \sum_{l=1}^{m} (\alpha_l + \beta_l) \frac{\delta \phi^n}{\delta x_i}, \qquad m = 1, 2,$$

$$u_i^{*0} = u_i^n.$$

The approximation for $\delta\phi^k/\delta x_i$ is

$$\frac{\delta \phi^{*k}}{\delta x_i} = \sum_{l=1}^{k} (\alpha_l + \beta_l) \frac{\delta \phi^n}{\delta x_i} + O(\Delta t), \qquad k = 1, 2, 3. \tag{3.9}$$

As illustrated in Fig. 1, Eq. (3.7) allows calculating $\hat{u}_i$ at every sub-step without projecting into the divergence free velocity field. The projection into the divergence free field, and hence the solution of the Poisson equation, is required only at the last sub-step. This results in a significant reduction in computational cost. The time advancement accuracy remains second-order. Note that, compared to Eq. (2.5), Eq. (3.7) appears to require additional computations due to the terms $\delta\hat{u}_j^{k-1}/\delta x_j$ and $\delta\phi^n/\delta x_i$. However, these calculations are also required by the original method at every sub-step.

4. Boundary Conditions

In the previous scheme described in Section 2, each sub-step is considered as one complete time advancement of u_i with an effective time step $(\alpha_k + \beta_k)\, \Delta t$. The KM

method [1] of calculating boundary conditions for $\hat{u}_i$ can then be applied at each substep:

$$\hat{u}_i^k = u_i^k + (\alpha_k + \beta_k)\,\Delta t \frac{\delta P^{k-1}}{\delta x_i}, \qquad k = 1, 2, 3. \tag{4.1}$$

However, this procedure cannot be used in the present scheme because without solving the Poisson equation, the sub-step advancement of u_i is not complete. The boundary conditions for $\hat{u}_i^k$ are thus calculated by applying Eq. (3.8) directly at the boundaries:

$$\hat{u}_i^k = u_i^{*k} + \Delta t \frac{\delta \phi^{*k}}{\delta x_i}, \qquad k = 1, 2, 3. \tag{4.2}$$

Using (4.2) at the last sub-step, $k = 3$, and substituting Eq. (3.9) for $\delta\phi^{n+1}/\delta x_i$ gives

$$\hat{u}_i = u_i^{n+1} + \Delta t \frac{\delta \phi^n}{\delta x_i} + O(\Delta t^2). \tag{4.3}$$

Now $\hat{u}_i^1$ and $\hat{u}_i^2$ can be calculated by linear interpolation without loss of accuracy. The algorithm for $\hat{u}_i^k$ is

$$\hat{u}_i^k = u_i^n + \left(u_i^{n+1} - u_i^n + \Delta t \frac{\delta \phi^n}{\delta x_i}\right) \sum_{l=1}^{k} (\alpha_l + \beta_l), \quad k = 1, 2, 3. \tag{4.4}$$

Equation (4.4) gives the boundary conditions for $\hat{u}_i^k$ accurate to $O(\Delta t^2)$.

5. Numerical Examples

The proposed scheme is independent of the number of spatial dimensions and can be used for two- and three-dimensional flows. In the numerical examples presented below, the flows considered are two-dimensional. In all subsequent comparisons, the original Runge–Kutta scheme in which the divergence free velocity field is enforced at every sub-step (Eqs. (2.5), (2.6), and (2.8)) will be referred to as scheme A; the present scheme with the Poisson equation solved only at the last sub-step will be referred to as scheme B; and the KM scheme denotes the Kim and Moin method [1].

5.1. *Decaying Vortices*

The proposed scheme and the boundary conditions derived above are tested in computing the following two-dimensional unsteady flow [6]:

$$u(x, y, t) = -\cos x \sin y e^{-2t} \tag{5.1}$$

$$v(x, y, t) = \sin x \cos y e^{-2t} \tag{5.2}$$

$$P(x, y, t) = -\tfrac{1}{4}(\cos 2x + \cos 2y) e^{-4t} \tag{5.3}$$

$$0 \leqslant x, y \leqslant \pi.$$

Computations were carried out for both schemes A and B. The same Courant number is used for both schemes. Table I shows the maximum error in u for the two schemes as compared to the exact solution. Comparison is at $t = 0.35$ at which the vortices decay to half their original strength.

The variation of the maximum error in u with mesh refinement is also plotted in Fig. 2. Keeping the Courant number constant, three different mesh sizes are used to determine the overall accuracy of the scheme. Figure 2 shows that both schemes are second-order accurate. Similar calculations for v yield the same results. The comparison in Table I and Fig. 2 indicates that, with proper modification to the Runge–Kutta type scheme, the Poisson equation can be eliminated from the sub-step time advancement without reducing the accuracy.

5.2. *Flow over a Backward-Facing Step*

The flow over a backward-facing step in a channel provides another comparison among the three schemes. Computations are performed on a two-dimensional laminar flow. The inflow boundary condition is a parabolic profile. The outflow boundary is located at $30h$ downstream from the step, where h is the step height. Convective bounary condition is applied at the exit. Results are obtained using a staggered grid with 128×32 cells. The reattachment length is plotted as a function of Reynolds number in Fig. 3 for all three schemes. The Reynolds number is based on the step height and the mean inlet velocity.

Figure 3 shows the agreement of all three schemes in predicting the reattachment length of the laminar flow over a backward-facing step. Indistinguishable results are also obtained by these schemes with half the time step, a further confirmation that the time accuracy of scheme B is not reduced.

TABLE I

Maximum Error at $t = 0.35$: $\varepsilon_{\max}$

Grid	Scheme A	Scheme B
16	1.9539×10^{-4}	1.9769×10^{-4}
32	3.0052×10^{-5}	3.0399×10^{-5}
64	6.0019×10^{-6}	5.9337×10^{-6}

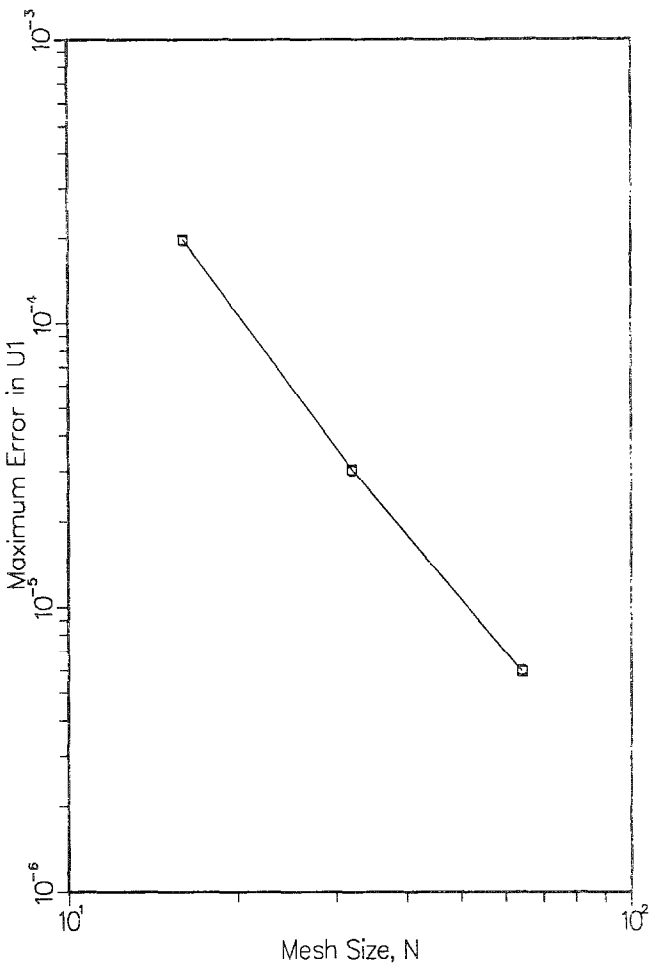

FIG. 2. Maximum error as a function of mesh refinement.

5.3. *CPU Comparison*

Comparison was made of the computational effort required for each of the numerical schemes for a given integration time T. The backward facing step calculations were used for this purpose.

By eliminating the Poisson equation at the sub-steps, the required CPU time per time step, Δt, for scheme B is 62% of scheme A. Thus, a reduction of 38% in computation time is achieved.

Comparison with the KM scheme is more subjective because the KM scheme is a one-step method whereas schemes A and B require three sub-steps. For a given time step Δt, scheme B requires 1.75 times the CPU time of the KM scheme due to repeated calculations in three sub-steps. However, a higher stability limit, CFL,

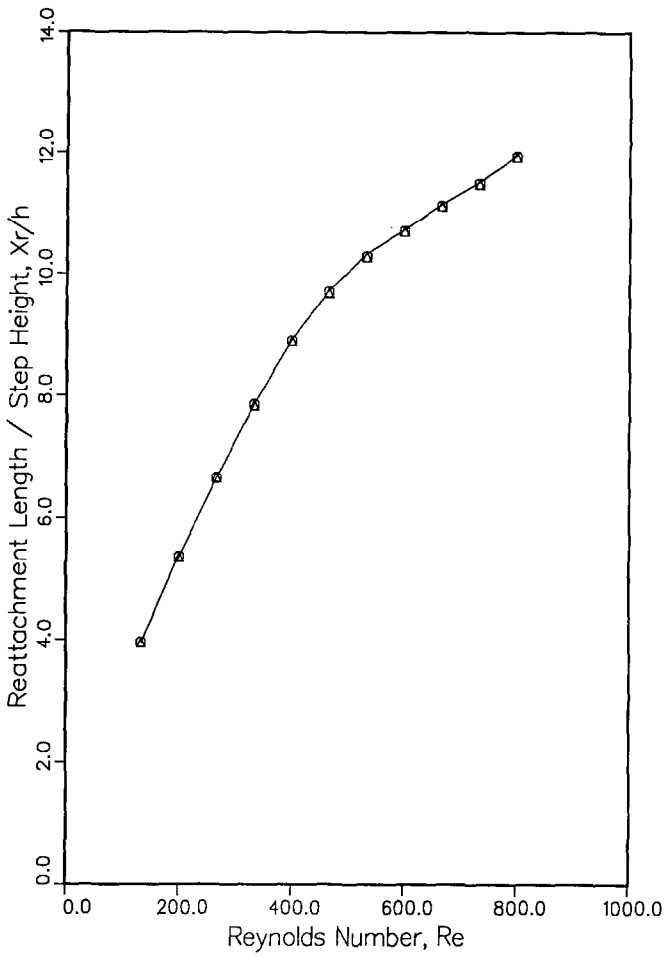

FIG. 3. Reattachment Length vs Reynolds Number.

in schemes A and B allows a larger Δt than that of the KM scheme. For schemes A and B, the CFL limit is $\sqrt{3}$. It is difficult to set the CFL limit for the KM scheme because it involves the Adams–Bashforth method which is strictly unstable for the linear convective model problem. The stability of the KM scheme is, therefore, dependent on the presence of the viscous terms and on the Reynolds number. For comparison purposes in this study, a nominal CFL limit of 0.5 was used for the KM scheme. Based on these CFL limits, the Δt for scheme B can be 3.46 times larger than that of the KM scheme. Therefore, for a given integration time T, scheme B achieves an overall CPU time saving of 49% over the KM scheme.

Our implementation of scheme B on the CRAY-YMP was executed at 175 million floating point operations per second.

6. SUMMARY

A numerical method is presented for solving three-dimensional, unsteady flows. The method is the fractional-step method combined with a three-step Runge–Kutta type scheme. In the first two sub-steps, the velocity field is advanced without the divergence free constraint; the Poisson equation is solved directly by a transform method only at the last sub-step to satisfy the continuity equation. The method is second-order accurate in space and time. The results show that, with proper modification of the non-linear terms, the Poisson equation does not need to be solved at every sub-step which significantly reduces computing cost while still maintains the desired accuracy. The concept imbedded in this method can in principle be applied to higher order Runge–Kutta type advancement schemes.

ACKNOWLEDGMENTS

The authors express their thanks to Dr. John Kim for useful discussions during the course of this work. Computations were carried out at the NAS Facility at NASA-Ames Research Center.

REFERENCES

1. J. KIM AND P. MOIN, *J. Comput. Phys.* **59**, 308 (1985).
2. A. A. WRAY, NASA-Ames Research Center, Moffett Field, CA, private communication (1986).
3. P. R. SPALART, NASA-Ames Research Center, Moffett Field, CA, private communication (1987).
4. R. L. LEVEQUE AND J. OLIGER, *Math. Comput.* **40**, 469 (1983).
5. F. H. HARLOW AD J. E. WELCH, *Phys. Fluids* **8**, 2182 (1965).
6. G. I. TAYLOR, *Philos. Mag.* **46**, 671 (1923).