INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 10, 607–621 (1990)

A STABLE FAST MARCHING SCHEME FOR COMPUTATIONAL FLUID MECHANICS

M. BEHNIA

School of Mechanical and Industrial Engineering, University of New South Wales, PO Box 1, Kensington, NSW, Australia 2033

M. WOLFSHTEIN

Department of Aeronautical Engineering; The Technion, Haifa, Israel

AND

G. DE VAHL DAVIS

School of Mechanical and Industrial Engineering, University of New South Wales, PO Box 1, Kensington, NSW, Australia 2033

SUMMARY

A stable fast marching scheme has been developed for the solution of coupled parabolic partial differential equations such as the Navier–Stokes equations. The scheme was developed with the aid of a stability analysis. The implementation of the method in standard alternating direction implicit algorithms is straightforward. The scheme was tested on the problem of natural convection in a square cavity. The number of iterations required for convergence was significantly reduced compared to conventional methods.

1. INTRODUCTION

Rapid development of digital computers and numerical methods in the last 20 years has made it possible to solve the Navier–Stokes equations for increasingly complicated situations. Today it is possible to obtain numerical solutions for a wide range of problems of engineering significance by numerical methods. However, the solution is often obtained at the cost of a large amount of computer time.

A usual reason for the inefficiency of Navier–Stokes solvers is the lack of stability of the solution procedure, which limits the size of the time step (or the number of time steps required to reach a steady state). Rigorous stability analysis is not possible since convergence and stability cannot be proven for most of these more complicated non-linear coupled problems. In fact, one is often forced to study stability empirically, with the numerical methods playing the role of an experimental tool.

When marching—in time or space—it is sometimes possible to use linear stability theory to predict the upper stability bound for the marching step size. However, in practice it is usually necessary to use a smaller step size than the one predicted by the analysis. Since it is often only the steady state (or fully developed) condition which is of interest in engineering problems, it is most

0271–2091/90/060607–15$07.50

Received 29 January 1988
Revised 17 April 1989

desirable, in a marching scheme, to save computer time by reaching such a condition in as few steps as possible by using as large a step size as possible.

The work described herein was developed during a study of the buoyancy-driven motion of a viscous incompressible fluid in an enclosure. In this paper we propose to use a three-level marching scheme to improve the stability of parabolic solvers in general and Navier–Stokes solvers in particular. The improved stability permits the use of larger time steps, thus speeding up the convergence to a steady state. The application of the method to existing codes is easy and requires very few changes. The implementation of the scheme in an alternating direction implicit (ADI) method is discussed. The improvement in obtaining results to the problem of natural convection in a square cavity is shown.

2. MATHEMATICAL FORMULATION

Consider a partial differential equation of the form

$$\frac{\partial \phi}{\partial t} = L\phi + S, \tag{1}$$

where t is the time or a spatial marching direction, L is a linear elliptic differential operator and S is the source term. The usual finite difference substitution for (1) without a source is a two-point formula based on upstream and downstream values of the form

$$\alpha\phi^{n+1} + \beta\phi^{n} = \Delta t(\delta L\phi^{n+1} + \varepsilon L\phi^{n}), \tag{2}$$

where Δt is the time step size and n represents the time ($t = n\Delta t$). In two-point schemes we usually use $\alpha = -\beta = 1$. When $\delta = \varepsilon = \frac{1}{2}$ we get the second-order Crank–Nicolson scheme;[1] when $\delta = 1$ and $\varepsilon = 0$ the first-order implicit scheme is obtained; and $\delta = 0$ and $\varepsilon = 1$ leads to the first-order explicit scheme.

It is easy to show that the Crank–Nicolson scheme is the only two-point second-order scheme and therefore it is impossible to devise a second-order scheme which is more stable than the Crank–Nicolson scheme. Enhancement of stability is possible *only* if additional levels are used. The simplest possibility is the addition of a third level by adding ϕ^{n-1} and $L\phi^{n-1}$ to the left and right sides of (2) respectively. In this work we chose not to use $L\phi^{n-1}$ because it would increase the computational load and because the enhancement of stability is easily obtained even when this term is not used. Therefore (2) is replaced by

$$\alpha\phi^{n+1} + \beta\phi^{n} + \gamma\phi^{n-1} = \Delta t\delta L\phi^{n+1} + \Delta t\varepsilon L\phi^{n}. \tag{3}$$

To examine the consistency of (3) we substitute a Taylor series expansion of each term around the nth step, giving the following equation (for convenience the superscript n is dropped from ϕ^{n} and its derivatives):

$$\alpha\left(\phi + \Delta t\phi' + \frac{\Delta t^2}{2}\phi'' + \ldots\right) + \beta\phi + \gamma\left(\phi - \Delta t\phi' + \frac{\Delta t^2}{2}\phi'' - \ldots\right)$$
$$= \Delta t\delta L\left(\phi + \Delta t\phi' + \frac{\Delta t^2}{2}\phi'' + \ldots\right) + \Delta t\varepsilon L\phi, \tag{4}$$

where a prime denotes partial differentiation with respect to time. Collection of the terms into equal orders of Δt shows that

$$\alpha + \beta + \gamma = 0. \tag{5}$$

If

$$\alpha-\gamma=\delta+\varepsilon, \tag{6}$$

then the truncation error (first neglected term) will be of order Δt. By requiring further that

$$\alpha+\gamma=2\delta, \tag{7}$$

the truncation error becomes $O(\Delta t^2)$.

For convenience and without loss of generality we may choose $\alpha=1$. It follows that (5)–(7) can be solved for β, ε and δ in terms of γ:

$$\beta=-1-\gamma, \qquad \varepsilon=(1-3\gamma)/2, \qquad \delta=(1+\gamma)/2. \tag{8}$$

Substitution of (8) into (3), with the definition

$$1/(1-\gamma)=w, \tag{9}$$

yields

$$w(\phi^{n+1}-\phi^n)+(1-w)(\phi^n-\phi^{n-1})=\Delta t(w-\tfrac{1}{2})L\phi^{n+1}+\Delta t(\tfrac{3}{2}-w)L\phi^n, \tag{10}$$

where w is a free parameter which may be so chosen as to enhance stability.

It may be noted that the special case of $w=1$ corresponds to the familiar Crank–Nicolson discretization. For $w=\frac{1}{2}$ the scheme reduces to the unstable Richardson scheme. For $w=\frac{3}{2}$ the scheme becomes a variant of the leap-frog methods.

The stability of (10) may be investigated using the family of solutions of the form $\phi=r^n\phi_0$, where r, the amplification factor, is a constant and ϕ_0 is a spatial function. By substitution we get

$$[w-(w-\tfrac{1}{2})\Delta t\rho]r^2+[-2w+1-(\tfrac{3}{2}-w)\Delta t\rho]r-(1-w)=0, \tag{11}$$

where

$$\rho=L\phi_0/\phi_0. \tag{12}$$

For a parabolic equation it can be shown that the real part of ρ is negative, say $\rho=-k^2$, where k may be regarded as a wave number. The imaginary part is non-zero only when the operator L contains first derivatives.

We want the amplification factor r to be as small as possible and the problem reduces to one of choosing the value of w to achieve this. Differentiation of (11) with respect to w yields

$$2r\frac{\mathrm{d}r}{\mathrm{d}w}[w+(w-\tfrac{1}{2})k^2\Delta t]+r^2(1+k^2\Delta t)+\frac{\mathrm{d}r}{\mathrm{d}w}[(1-2w)+(\tfrac{3}{2}-w)k^2\Delta t]-r(2+k^2\Delta t)+1=0. \tag{13}$$

Equation (13) cannot be used to find the value of w which gives a minimum value of r. Therefore we return to (11) and examine the two limiting cases of very small and very large time steps, i.e. $\Delta t\to 0$ and $\Delta t\to\infty$. When $\Delta t\to 0$ we get

$$r^2w+r(1-2w)+w-1=0, \tag{14}$$

with the roots

$$r=\{1,\ (w-1)/w\}. \tag{15}$$

When $\Delta t\to\infty$ we get

$$r^2(w-\tfrac{1}{2})+r(\tfrac{3}{2}-w)=0, \tag{16}$$

with the roots

$$r=\{0,\ (2w-3)/(2w-1)\}. \tag{17}$$

The roots of these two limiting cases are plotted in Figure 1. It is evident that the magnitude of the amplification factor r for the two limiting solutions is the same when

$$|(2w-3)/(2w-1)|=|(w-1)/w|. \tag{18}$$

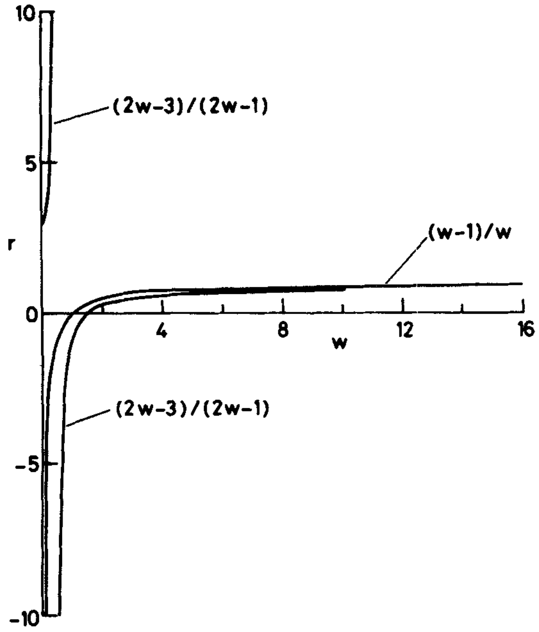

Figure 1. Distribution of the amplification factor r with w

Equation (18) can be solved to yield a compromise value of w for small and large Δt. The equation then reduces to

$$w^2-\tfrac{3}{2}w+\tfrac{1}{4}=0, \tag{19}$$

of which the roots are $w=(3\pm\sqrt{5})/4$ or $w_1=1{\cdot}309$ and $w_2=0{\cdot}191$. It is obvious that the second root is not appropriate (see Figure 1) since it yields an amplification factor of about 4·2. The value of 1·309 for w is therefore chosen. With this value (10) becomes

$$1{\cdot}309\phi^{n+1}-1{\cdot}618\phi^n+0{\cdot}309\phi^{n-1}=0{\cdot}809kL\phi^{n+1}+0{\cdot}191kL\phi^n. \tag{20}$$

3. ADI IMPLEMENTATION

A widely accepted method of solving parabolic and elliptic partial differential equations in both two and three dimensions is by alternating direction implicit (ADI) methods (see e.g. Isenberg and de Vahl Davis[2]). The partial differential equations are approximated by finite differences. With a compromise between accuracy and ease of programming in mind, the finite difference approximations (FDAs) usually utilize first-order forward differences in the marching direction (time) and second-order central differences in space. ADI methods keep the FDAs tridiagonal and hence permit the use of the efficient Thomas algorithm.[3] The first ADI method was introduced by Peaceman and Rachford[4] and for the two-dimensional form of (1) the application of this ADI scheme is as follows:

$$\left(I-\frac{\Delta t}{2}A_x\right)\phi^*=\left(I+\frac{\Delta t}{2}A_y\right)\phi^n, \tag{21}$$

$$\left(I-\frac{\Delta t}{2}A_y\right)\phi^{n+1}=\left(I+\frac{\Delta t}{2}A_x\right)\phi^*, \tag{22}$$

where I denotes the unit matrix and A_x and A_y are one-dimensional matrix operators with $L=[A_x+A_y]$. The symbols ϕ^* and ϕ^{n+1} denote intermediate and updated values of ϕ respectively. Note that the coefficient of A_x and A_y (i.e. $\Delta t/2$) results from the Crank–Nicolson scheme.

Another ADI scheme due to Douglas[5] (sometimes the method is attributed to Samarskii and Andreyev[6]), which was presented independently of the Peaceman–Rachford scheme, can in fact be derived from (21) and (22).

Let

$$f^{n+1}=(\phi^{n+1}-\phi^{n})/\Delta t, \tag{23}$$

so that ϕ^*, which is, in a sense, $\phi^{n+1/2}$, can be evaluated as

$$\phi^*=\phi^n+\frac{\Delta t}{2}f^* \tag{24}$$

and

$$\phi^{n+1}=\phi^n+\Delta t f^{n+1}. \tag{25}$$

Substitution of (24) and (25) into (21) and (22), simplification and inclusion of the source term yields

$$\left(I-\frac{\Delta t}{2}A_x\right)f^*=(A_x+A_y)\phi^n+S^n, \tag{26}$$

$$\left(I-\frac{\Delta t}{2}A_y\right)f^{n+1}=f^*. \tag{27}$$

The Douglas scheme (equations (26) and (27) along with (25)) requires the advancement of the solution over the time step Δt through the use of two intermediate field evaluations (compared with one for Peaceman–Rachford ADI). Yet it is in fact more efficient and requires one less auxiliary array in programming. Furthermore, the Douglas scheme can be extended to three dimensions, whereas the Peaceman–Rachford scheme then becomes unstable. It should be noted that both ADI schemes presented here are based on the Crank–Nicolson scheme ($w=1$). They can be easily modified to accommodate any value of w and in particular the 'optimum' value which leads to (20). For the solution of this equation the Douglas (or Samarskii–Andreyev) ADI scheme corresponds to the following steps. First (20) is further simplified by substitution for ϕ^{n+1} from (25) to give

$$f^{n+1}=0{\cdot}618\Delta t L f^{n+1}+0{\cdot}764\Delta t L\phi^n+0{\cdot}236 f^n. \tag{28}$$

The ADI steps are

$$(I-0{\cdot}618\Delta t A_x)f^*=0{\cdot}764(A_x+A_y)\phi^n+0{\cdot}236\frac{\phi^n-\phi^{n-1}}{\Delta t}, \tag{29}$$

$$(I-0{\cdot}618\Delta t A_y)f^{**}=f^*, \tag{30}$$

$$\phi^{n+1}=\phi^n+\Delta t f^{**}. \tag{31}$$

For implementation, let us suppose that the source term S on the right-hand side of (1) is retained (e.g. the vorticity transport equation in a natural convection problem). Using a similar formulation to (3) but with the source term in the form

$$\Delta t(AS^{n+1}+BS^n+CS^{n-1}),$$

and expanding this equation in a way similar to (4), we find that conditions (5)–(7) for second-order accuracy in time still apply, but now some additional conditions are required. For first-order accuracy

$$A+B+C=\delta+\varepsilon=1/w. \tag{32}$$

An additional condition for second-order accuracy is

$$A-C=(1+\gamma)/2=1-1/2w. \tag{33}$$

Letting $A=0$ in order to eliminate the S^{n+1} term yields, for first-order accuracy,

$$B+C=1/w, \tag{34}$$

and the additional condition for second-order accuracy is

$$C=-1+1/2w. \tag{35}$$

For simplicity and to save on computer memory requirements, it is desirable to retain the source term at only one level (e.g. S^n). Hence we choose $A=C=0$, thus reducing the accuracy in time to first order. In this case (34) becomes

$$B=1/w. \tag{36}$$

For $w=1{\cdot}309$ the coefficient of the source term S^n is 0·764 and (28) becomes

$$f^{n+1}=0{\cdot}618\Delta tLf^{n+1}+0{\cdot}764L\phi^n+0{\cdot}236f^n+0{\cdot}764\Delta tS^n. \tag{37}$$

The new algorithm can be easily implemented in an existing computer program based on the Douglas ADI method. For the first time step ($n=1$) Crank–Nicolson must be used since ϕ^{n-1} in (29) is undefined. After the first step the new scheme can be used; ϕ^{n-1} is an extra array to be saved.

The ease of implementation can be appreciated by noting that (37) is a difference approximation of the differential equation

$$\phi_t=M\phi+R, \tag{38}$$

where $M=1{\cdot}236L$ and $R=-0{\cdot}472L\phi^n+0{\cdot}236f^n+0{\cdot}764\Delta tS^n$.

Suppose an existing computer program has been written for the Crank–Nicolson-type Samarskii–Andreyev ADI. There are three simple changes to be made to convert the program to the proposed scheme.

1. On the left-hand sides of the first- and second-level ADI equations (26) and (27) multiply the coefficients of A_x and A_y by 1·236.
2. On the right-hand side of the first-level equation (26) multiply the coefficient of A_x+A_y by 0·764.
3. On the right-hand side of the first-level equation add the term 0·236 $(\phi^n-\phi^{n-1})/\Delta t$.

Further, if there is a source term in the equation, the term $0{\cdot}764\Delta tS^n$ must replace ΔtS^n on the right-hand side of the first level of ADI.

Other ADI methods in both two and three dimensions can be similarly modified to take advantage of the stability of the proposed scheme.

4. TEST PROBLEM

As described in the Section 1, the new stable fast marching scheme (SFMS) was developed during the study of buoyancy-driven natural convection in a rectangular cavity with different temperatures on the vertical sides. Therefore this problem was used as the test vehicle for the implementation of SFMS and for tests of its efficiency.

The problem specifically considered here is that of the two-dimensional flow of a Boussinesq fluid of Prandtl number 0·71 (i.e. air) in an upright square cavity of side D. The walls are non-slip and impermeable. The horizontal walls are adiabatic and the vertical sides are at temperatures T_h and T_c.

The streamfunction–vorticity $(\psi-\zeta)$ formulation of the Navier–Stokes equations was used. The quantities D, κ (the thermal diffusivity) and D^2/κ were used as the non-dimensionalization scale factors for length, streamfunction and time respectively. The dimensionless temperature was defined as $T=(T'-T_c)/(T_h-T_c)$, where T' is the local dimensional temperature. The horizontal axis is x and the vertical axis is y. The dimensionless governing equations are[7]

$$\frac{\partial}{\partial x}(u\zeta)+\frac{\partial}{\partial y}(v\zeta)=Pr\nabla^2\zeta+RaPr\frac{\partial T}{\partial x}, \tag{39}$$

$$0=\nabla^2\psi+\zeta, \tag{40}$$

$$\frac{\partial}{\partial x}(uT)+\frac{\partial}{\partial y}(vT)=\nabla^2 T. \tag{41}$$

These equations were modified to allow the use of the method of false transient[8] by the inclusion, on the left-hand side of each, of the time derivative terms

$$\frac{1}{\alpha_\zeta}\frac{\partial\zeta}{\partial t},\qquad \frac{1}{\alpha_\psi}\frac{\partial\psi}{\partial t},\qquad \frac{1}{\alpha_T}\frac{\partial T}{\partial t} \tag{42}$$

respectively. The addition of these false time derivatives to the system of coupled non-linear elliptic partial differential equations (39)–(41) converts the system to a parabolic form, thus enabling the utilization of a marching solution, equivalent to an iterative procedure. The transient solution is incorrect but, as $t\rightarrow\infty$ and the steady state is approached, the equations revert to their correct forms and the true steady solution is achieved. The coefficients α_ζ, α_ψ, and α_T modify the transient terms and provide extra degrees of freedom to control stability and the rate of approach to the steady state. A stability ratio was defined as $s=\Delta t/\min(\Delta x^2, \Delta y^2)$ to determine the time step.

The modified equations were solved on a square mesh by a finite difference method; forward differences were used for the time derivatives and second-order central differences for all space derivatives. The resulting finite difference approximations were solved by a Crank–Nicolson-type Samarskii–Andreyev ADI algorithm. The program used (FRECON) was developed by de Vahl Davis[9] and the method of the false transient has proved to be extremely fast and efficient in comparison to other methods.[8] The method is second-order accurate but, with the help of Richardson's extrapolation, solutions with very high accuracy can be obtained.[7]

For comparison purposes the SFMS was implemented in the program FRECON, as an option, in such a fashion that either the Crank–Nicolson or the new marching scheme could be specified. The vorticity transport equation (39) is the least stable of the system (39)–(41). Introduction of SFMS to the streamfunction and energy equations did not improve the speed of convergence significantly; therefore the standard scheme was retained for these two equations to save on computer memory requirements. Further, as mentioned earlier, the false transient coefficients provide extra degrees of freedom to control stability and the rate of approach to the steady state. In order to remove any influence of these factors on the speed of convergence, the αs were set to unity, although this slows the false transient method considerably.

We assumed that the steady state had been reached when f^{**} (the time rate of change of vorticity, streamfunction or temperature) at each mesh point had reached a small prescribed value; the marching process was then terminated.

5. RESULTS

For a Rayleigh number range of 10^4–10^6, solutions were found using uniform meshes of 21×21 and 41×41. The finer mesh was used at the higher values of Ra. The results of the Crank–Nicolson-type marching algorithm (referred to as the 'standard' scheme and designated SS) are compared with the results of the stable fast marching scheme (SFMS). According to the analysis presented here, larger marching steps can be taken with SFMS and a steady state should be reached in fewer steps. The purpose of this comparison exercise was to verify this claim.

Comparisons of the values of vorticity at the mid-point of the hot wall at different times for $Ra = 10^4$ are shown in Figures 2–5. The extrapolated zero-time-step values for both methods at each time differ by less than 0·04%. This indicates, as expected, that for extremely small time steps the schemes yield identical values at all times. Further, it can be seen that SFMS and SS are first-order accurate in time.

The Crank–Nicolson-type Samarskii–Andreyev ADI method can be shown to be unconditionally stable for a single linear equation. Unfortunately, it is not possible to prove (or disprove) the stability of the scheme when applied to coupled non-linear equations. Mallinson and de Vahl Davis[8] in a test calculation on a linear diffusion equation obtained solutions using time steps as large as $160\Delta x^2$. However, they observed that for the non-linear coupled differential equations of natural convection in two or three dimensions there is a much smaller upper limit on the usable time step. They found this limit to be about $0{\cdot}8\Delta x^2$ for false transient coefficients of unity.

In order to determine the optimum value of time step for fastest convergence, a series of numerical experiments for Rayleigh numbers of 10^4, 2×10^4, 3×10^4 and 4×10^4 with $\alpha_\zeta = \alpha_\psi = \alpha_T = 1$ for different values of stability ratio s were performed. In these experiments a

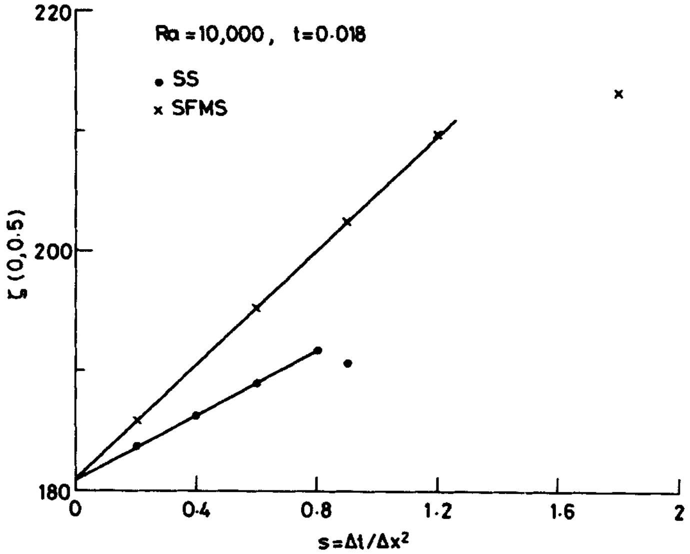

Figure 2. Vorticity at the mid-point of the hot wall at $t = 0{\cdot}018$ as a function of stability ratio

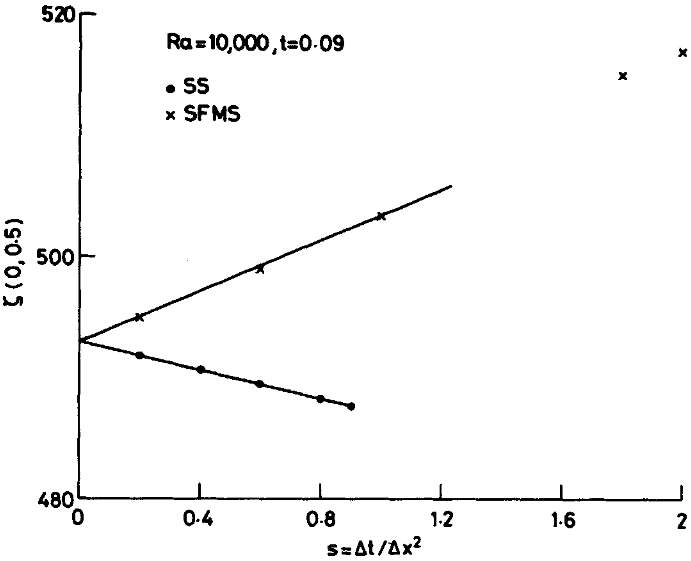

Figure 3. Vorticity at the mid-point of the hot wall at $t=0{\cdot}09$ as a function of stability ratio

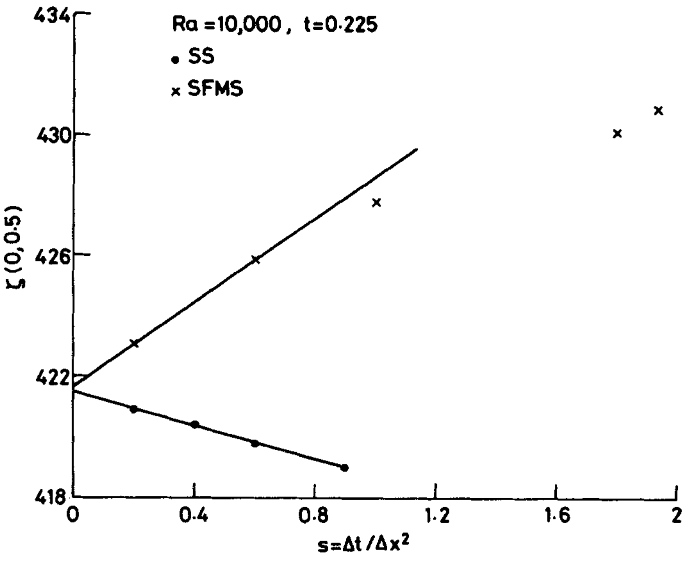

Figure 4. Vorticity at the mid-point of the hot wall at $t=0{\cdot}225$ as a function of stability ratio

convergence criterion of 10^{-3} was used to determine the steady state condition. A summary of the results is presented in Table I.

For a Rayleigh number of 10^4 a stability ratio of 0·2 for the standard scheme requires 2084 iterations to reach steady state. Increasing the time step reduces the number of iterations to a minimum of 427 at an optimum stability ratio of 0·92. Further increase of the stability ratio to 0·94 causes a dramatic rise in the number of time steps to convergence, but the correct steady

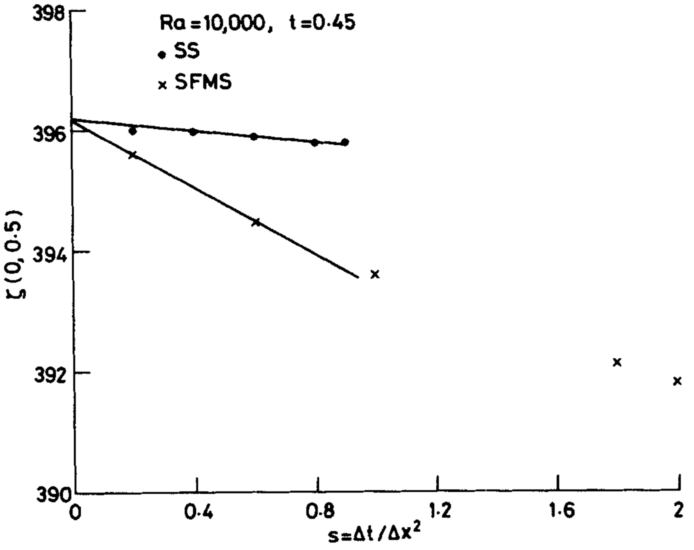

Figure 5. Vorticity at the mid-point of the hot wall at $t=0{\cdot}45$ as a function of stability ratio

Table I. Number of iterations for convergence to 10^{-3}; O and D denote oscillations and divergence respectively

$s=$	$Ra=10\,000$		$Ra=20\,000$		$Ra=30\,000$		$Ra=40\,000$	
$\Delta t/\Delta x^2$	SS	SFMS	SS	SFMS	SS	SFMS	SS	SFMS
0·2	2084	2090	—	—	—	—	—	—
0·4	997	—	—	—	—	—	—	
0·6	662	698	933	980	1402	1586	—	—
0·8	492	—	—	—	—	—	—	—
0·9	437	447	675	654	1000	1039	2160	3069
0·92	427	—	585	—	848	—	—	—
0·94	1269	—	—	—	—	—	—	—
0·95	(O) 2000	—	—	—	—	—	—	—
1·0	(D) 115	402	—	587	—	936	—	2726
1·2	—	335	—	476	—	766	—	2181
1·4	—	286	—	409	—	—	—	1777
1·6	—	250	—	357	—	564	—	1510
1·7	—	237	—	336	—	521	—	1389
1·8	—	(O) 2000	—	(O) 2000	—	(O) 2000	—	(O) 2000
2·0	—	(D) 36	—	—	—	—	—	—

solution is eventually reached after 1269 iterations. A time step corresponding to a stability ratio of 0·95 results in a low-amplitude oscillation which persisted even after 2000 iterations, at which stage the marching was terminated. A stability ratio of 1·0 causes divergence after 115 iterations.

Up to the optimum stability ratio of 0·92 both SS and SFMS require nearly the same number of iterations for convergence. This optimum time step (or stability ratio) for SS compares favourably with the value of $0{\cdot}8\Delta x^2$ proposed by Mallinson and de Vahl Davis. As expected, the SFMS results for $Ra=10^4$ indicate that the stability ratio can be increased far beyond the maximum for SS and hence fewer iterations are required for convergence. In this case the optimum value of s is

1·7 corresponding to 237 iterations, compared to 0·92 and 427 iterations. At this Rayleigh number SFMS diverges for a stability ratio of 2 after 36 iterations.

Results presented in Table I indicate that the optimum values of the time step for both SS and SFMS are almost independent of the Rayleigh number. It should be pointed out that these optimum values are for the case when all the false transient factors are equal to unity. If we define the percentage increase in the required number of iterations (PI) at the optimal stability ratio by using SS instead of SFMS as follows,

$$PI=\frac{N_{\text{SS}}-N_{\text{SFMS}}}{N_{\text{SFMS}}}\times 100, \tag{43}$$

then the results of Table I yield PI values of 80%, 74%, 63%, and 55% for Rayleigh numbers of 10^4, 2×10^4, 3×10^4 and 4×10^4 respectively.

The number of iterations to convergence as a function of the stability ratio for $Ra=10^4$ is plotted in Figure 6. All the points fall on a straight line with a slope of -1 (on a log–log scale), indicating that both methods are first-order accurate in time. In this figure the results for a convergence criterion of 10^{-2} are also shown. They also fall on a straight line with a slope of -1 but, as expected, lie somewhat below the results for the tighter convergence criterion.

In the method of false transient a properly chosen set of false transient factors reduces the required number of iterations considerably. It has been shown that for the natural convection problem the method is at least one order of magnitude faster than a conventional double-iterative procedure.[8] Therefore we tested the method also for an optimal choice of false transient factors.

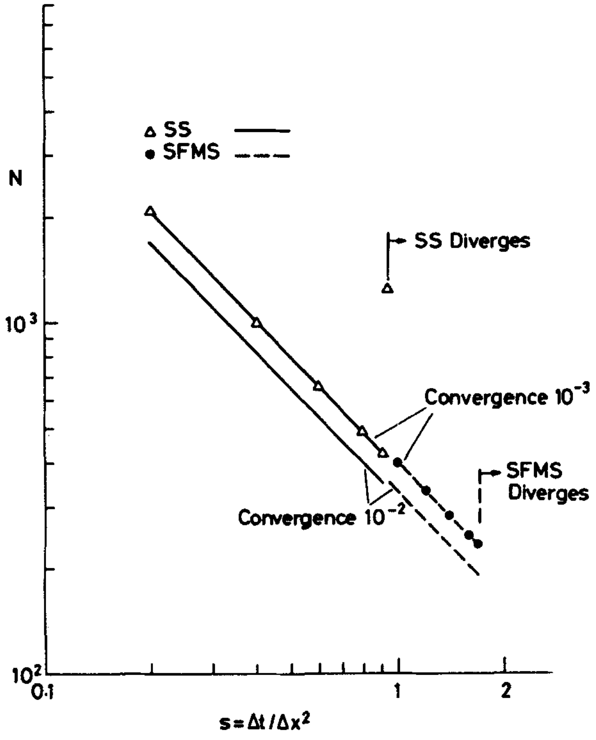

Figure 6. Number of iterations required for convergence as a function of stability ratio; $Ra=10000$

Initially a number of tests were performed and the method was tuned by adjusting the false transient factors α_ψ, α_T and α_ζ. Further numerical experiments were then carried out for different stability ratios (i.e. time steps) to determine the optimum step size (in the computer program the input parameter is s and the time step is internally calculated from $\Delta t = s\Delta x^2$). These optimal values were used then in both SS and SFMS schemes. The optimum stability ratios, giving the fastest possible convergence, for both SS and SFMS runs for Rayleigh numbers in the range 10^4–10^6 are summarized in Table II. The results indicate once again that compared to SS larger step sizes can be used in SFMS and as a result the steady state is reached in fewer steps. It should be pointed out that the starting condition for both schemes was the same (i.e. a linear temperature profile with all the other field variables set to zero). If one were not to use SFMS and instead chose SS, the increase in the number of iterations (PI) would be as shown in Table III.

In other words, the use of the conventional scheme rather than the proposed marching scheme can cost about 50% more computer time for this particular problem.

As mentioned earlier, the value of $w = 1{\cdot}309$ is a good value to be used in the five-term expansion (Crank–Nicolson-type schemes correspond to $w = 1$). The four-term expansion corresponding to $w = \frac{3}{2}$, which was proposed and used by Berger *et al.*[10] in a boundary-layer-type problem, was also implemented and tested. Although the results were somewhat better than the Crank–Nicolson scheme, SFMS proved to be faster.

Table II. Comparison of performance of SS and SFMS

Ra	Mesh	Optimum s	Optimum $\alpha_{\psi,T,\zeta}$	Number of iterations	Marching scheme
10^4	21 × 21	3	1, 1, 0·1	104	SS
10^4	21 × 21	5	1, 1, 0·1	68	SFMS
5×10^4	21 × 21	3	1, 1, 0·1	123	SS
5×10^4	21 × 21	5	1, 1, 0·1	83	SFMS
10^5	21 × 21	3	1, 1, 0·1	151	SS
10^5	21 × 21	5.2	1, 1, 0·1	113	SFMS
10^5	41 × 41	12	1, 0·25, 0·01	314	SS
10^5	41 × 41	20	1, 0·25, 0·01	192	SFMS
5×10^5	41 × 41	12	1, 0·25, 0·01	253	SS
5×10^5	41 × 41	20	1, 0·25, 0·01	156	SFMS
10^6	41 × 41	12	1, 0·25, 0·01	287	SS
10^6	41 × 41	20	1, 0·25, 0·01	186	SFMS

Table III. Percentage increase in the number of iterations between SS and SFMS

Ra	Mesh	PI (%)
10^4	21 × 21	53
5×10^4	21 × 21	48
10^5	21 × 21	34
10^5	41 × 41	64
5×10^5	41 × 41	62
10^6	41 × 41	54

As a final check on the accuracy of the solutions obtained, the SFMS results were compared with a very accurate 'benchmark solution'.[7] Agreement was excellent. Further, a Richardson extrapolation scheme[11] was adopted to determine the truncation convergence of SFMS. Sample results of the extrapolation for $Ra = 10^4$ are presented here. Solutions were obtained for 21 × 21, 41 × 41 and 81 × 81 meshes.

Let Z_t be the true value at a mesh point of a quantity whose computed value using a mesh size h_i is Z_i; then

$$Z_t = Z_i + Ch_i^n + O(h_i^{n+1}),$$

where n is the order of the truncation error and C is taken to be independent of h. Now, if the solution Z_i were known for three different meshes, then the values of Z_t, C and n could be

Table IV. Effect of mesh size on some characteristic values of vorticity

Mesh	$\zeta(0, 0{\cdot}5)$	$\zeta_{\max}$	$\zeta_{\min}$
21 × 21	395·8	431·3	−123·5
41 × 41	393·3	428·3	−124·7
81 × 81	392·7	427·6	−125·0

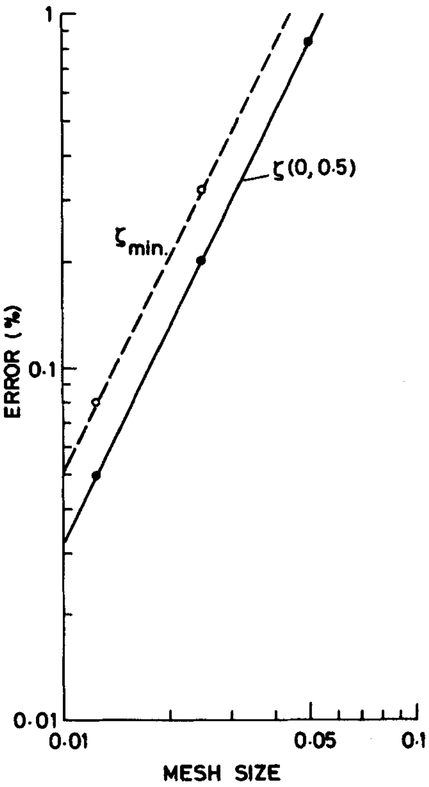

Figure 7. Error in vorticity at the mid-point of the hot wall and at the vorticity minimum as a function of mesh

calculated from the system of three equations with the three unknowns. Alternatively, the three equations can be reduced to one by eliminating Z_t and C. It follows that the order of the truncation error, n, is given by the solution of

$$\frac{Z_1-Z_2}{Z_2-Z_3}=\frac{h_1^n-h_2^n}{h_2^n-h_3^n}$$

at each mesh point. If $h_1/h_2=h_2/h_3=R$, say, this expression can be further simplified to give

$$n=\ln\frac{Z_1-Z_2}{Z_2-Z_3}\Big/\ln R.$$

The 'constant' C can then be found at each mesh point and hence the 'true' value Z_t can be obtained. Since the finite difference appoximations used were second order (in space), the expected value of n was 2.

The values of vorticity are usually very sensitive to the accuracy of convergence. As mentioned, for a Rayleigh number of 10^4 solutions were obtained using three mesh sizes ($h_1=0{\cdot}05$, $h_2=0{\cdot}025$ and $h_3=0{\cdot}0125$). The vorticity values at the mid-point of the hot wall, $\zeta(0, 0{\cdot}5)$, as well as the maximum and minimum vorticities are given in Table IV.

The value of n calculated from each set of quantities given above is very close to 2 (in fact 2·06, 2·09 and 2·00 respectively). Considering the number of significant figures that were used to print

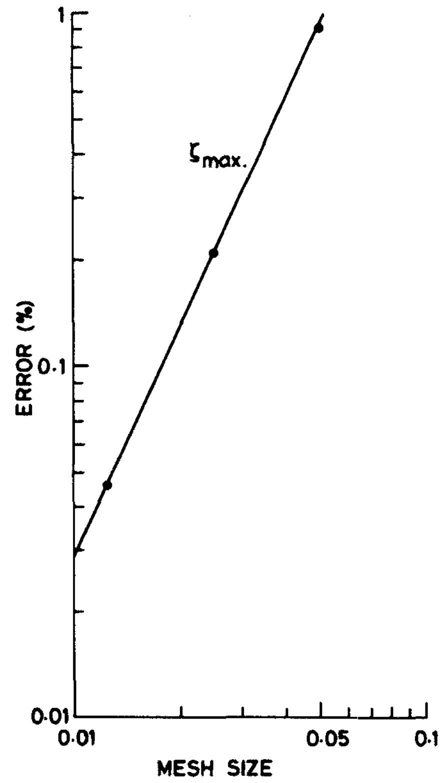

Figure 8. Error in vorticity at the vorticity minimum as a function of mesh

these quantities, the obtained values of n are very good. Similar calculations were performed for other quantities and n was determined to be quite close to the expected value of 2.

The extrapolated zero-mesh ('true' value Z_t) vorticities are

$$\zeta(0,\ 0{\cdot}5)=392{\cdot}5, \qquad \zeta_{\max}=427{\cdot}4, \qquad \zeta_{\min}=-125{\cdot}1.$$

The errors for different mesh sizes are plotted in Figures 7 and 8. The error percentages of each quantity for different mesh sizes fall on a straight line (on a log–log scale) with the slope of each line being very close to 2. Therefore it can be concluded that the method is indeed second-order accurate in space, i.e. that it possesses a rate of convergence of 2 in accordance with the theoretical asymptotic error estimate.

6. CONCLUSIONS

The stable fast marching scheme has been tested and found to converge faster than the conventional Crank–Nicolson method. The number of iterations required for convergence is about two-thirds that of the conventional scheme. Its application to existing codes is easy and involves only a few minor changes. The application of the scheme does not increase the computer time required per iteration, and only one more array of the size of the main variables is required because it was found to be sufficient to apply the method only to the least stable equation (i.e. vorticity).

We have illustrated the use of the method in a natural convection problem. In forced convection it may prove advantageous to apply SFMS to the energy equation as well as the vorticity equation.

ACKNOWLEDGEMENT

This work has been performed with the partial support of the Australian Research Grants Scheme, for which the authors are grateful.

REFERENCES

1. J. Crank and P. Nicolson, *Proc. Camb. Phil. Soc.*, **43,** 50–67 (1947).
2. J. Isenberg and G. de Vahl Davis, in C. Gutfinger (ed.), *Topics in Transport Phenomena: Bioprocesses, Mathematical Treatment, and Mechanism*, Hemisphere, Washington DC, 1975, Chap. 5, pp. 457–553.
3. L. H. Thomas, *Elliptic Problems in Linear Difference Equations over a Network*, Watson Scientific Computing Laboratory, Columbia University, New York, 1949.
4. D. W. Peaceman and H. H. Rachford, *J. SIAM*, **3,** 28 (1955).
5. J. Douglas, Jr., *Numer. Math.*, **4,** 41 (1961).
6. A. A. Samarskii and V. B. Andreyev, *USSR Comput. Math. Math. Phys.*, **3,** 1373 (1963).
7. G. de Vahl Davis, *Int. j. numer. methods fluids*, **3,** 249–264 (1983).
8. G. D. Mallinson and G. de Vahl Davis, *J. Comput. Phys.*, **12,** 435–461 (1973).
9. G. de Vahl Davis, Report *1976/FMT/1*, School of Mechanical and Industrial Engineering, University of New South Wales, 1976.
10. M. Berger, M. Israeli and M. Wolfshtein, in L. Dekker, G. Savastano and G. C. Vansteenkiste (eds), *Simulation of Systems '79*, North-Holland, New York, 1979, pp. 437–445.
11. L. F. Richardson, *Trans. R. Soc. Lond. Ser. A*, **210,** 307–357 (1910).