JOURNAL OF COMPUTATIONAL PHYSICS 28, 135–166 (1978)

The Operator Compact Implicit Method for Parabolic Equations*

MELVYN CIMENT,† STEPHEN H. LEVENTHAL,†† AND BERNARD C. WEINBERG§

Mathematics and Engineering Analysis Branch, Naval Surface Weapons Center, White Oak Laboratory, Silver Spring, Maryland 20910

Received May 17, 1977; revised December 7, 1977

This paper attempts to trace out the broad characteristics of a class of higher order finite difference schemes which are applicable to the solution of parabolic partial differential equations associated with viscous fluid flow problems. The basic method developed here uses the approach of the *compact implicit* techniques applied to the full spatial operator. The resulting spatial approximation, referred to here as the *operator compact implicit* method can be implemented with a variety of temporal integration schemes. In particular, a simple factorization technique is employed to resolve higher space dimension problems in terms of simple tridiagonal systems. The operator compact implicit method is compared to standard techniques and to some of the newer compact implicit methods. Stability characteristics, computational efficiency and the results of numerical experiments are discussed.

1. INTRODUCTION

The current engineering requirements for providing computational fluid dynamics codes for realistic viscous flow problems have provided the impetus for the development and implementation of higher order finite difference techniques [8, 1, 24]. It has been repeatedly demonstrated on model problems, that even the simplest types of higher order methods should provide tremendous practical advantages in terms of diminishing the required number of points (storage) and also the overall computing time for a desired resolution.

The present effort was undertaken to confront the full range of associated computational problems that would be involved in practical viscous flow field calculations. Our goal was to try to develop a cohesive set of higher order approximation tools which would help to indicate what methods ultimately might be best employed to form the basis of a major new code.

It appeared to several people almost simultaneously (sparked by a suggestion of Kreiss [17]) that from among the various techniques available a fruitful class of methods might emerge from the so-called compact implicit techniques [3], [8], [25]. Although there appear to be a variety of forms and implementations, the approaches do share some broad characteristics. The higher order is usually sought for the spatial part of the differential operator. The method developed is generally required to—

* This research has been supported jointly by the NSWC IR fund, NAVAIR, and NAVSEA.

† Present Address: Applied Mathematics Division, National Bureau of Standards, Washington, D.C. 20234.

†† Present Address: Simulation Research Section, Gulf Science and Technology Co., P.O. Drawer 2038, Pittsburgh, Pa. 15230.

§ Present Address: Scientific Research Associates, P.O. Box 498, Glastonbury, Conn. 06033.

0021-9991/78/0282-0135$02.00/0

1. reduce to tridiagonal form for fourth order accuracy;

2. allow for nonuniform spatial grids (usually at the expense of one order of accuracy);

3. allow for flexibility in choosing the time step.

In the various methods developed so far all these conditions have been met for simple model problems. However, further important concerns still remain.

As pointed out by [3, 4, 2] the usual compact implicit techniques, because of their implicit complexity, are not generally applicable in a direct manner to problems with varying order derivative terms unless a vector unknown of the derivative values is considered. Indeed, adopting the factorization technique suggested in [3] for a wave equation problem to a model parabolic problem resulted in numerical instabilities. To circumvent such problems, we advocate the use of a more general spatial approximation method, an operator compact implicit method suggested by Swartz [26]. Essentially, the same basic ideas are involved and instead of setting up spatial approximations for *individual* derivative terms one now poses the difference approximation in terms of the spatial *operator*. This spatial approximation has been previously derived in [20]; however, the basic derivation and implementation there proceeds along lines different from those taken here.

Another serious concern that one has relates to the stability characteristics of the overall method. If the spatial operator is associated with implicit temporal schemes, as might be expected, a variety of unconditionally stable schemes result for the linear model. However, the cell Reynolds number spatial stability characteristics are now somewhat more difficult to elucidate for spatially implicit methods. Our analysis in section IV is incomplete since it only applies for homogeneous equations with constant coefficients. However, our analysis and experiments indicate that for the operator compact implicit (OCI) approximation, there is a wider range of admissible cell Reynolds number than for the usual compact implicit methods for general homogeneous problems.

In our numerical studies of nonlinear models we have chosen to use two different approaches. As a benchmark, we have taken the basic Crank-Nicolson routine which requires linearization or iteration. Our second approach adapts a Lees type method [13] which does not require temporal iterations for a nonlinear problem. This latter simple scheme has proven to be very effective in numerical experiments. What emerges from our investigation is that a promising class of methods can be developed around the operator compact implicit method. In the future we hope to resolve questions concerning the treatment of mixed spatial derivative terms and to more fully resolve the limitations associated with cell Reynolds number effects.

2. Basic Difference Equations

The classical finite difference approach for solving two-point boundary value problems of the form

$$L(u) = a(x)\, u_{xx} + b(x)\, u_x = f,\ x \in [0, 1] \tag{2.1}$$

with $u(0)$, $u(1)$ given is to separately substitute standard approximations for the first and second derivatives in (2.1) and then solve the resulting system of equations. Accordingly, the centered second order approximation for these terms is

$$\frac{\delta_x U_j}{2h} \equiv \frac{U_{j+1} - U_{j-1}}{2h} = (u_x)_j + O(h^2), \tag{2.2}$$

$$\frac{\delta_x^2 U_j}{h^2} \equiv \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} = (u_{xx})_j + O(h^2), \tag{2.3}$$

where $x_j = jh, j = 0, 1,..., J$ and $U_j \sim u(x_j)$ and $h = 1/J$ is the mesh size.

The resulting system of equations that is derived upon substitution of (2.2), (2.3) into (2.1) is tridiagonal, and hence easily solved. For the case of Dirichlet data, there is no need to create fictitious points (i.e., to extrapolate information) in order to implement the scheme. However, if higher order accuracy is desired, the classical approach is to enlarge the basic mesh star, i.e. use more points in the discretization. Again, for the centered type of approximation fourth order accuracy is achieved by the equations

$$\left[I - \frac{1}{6}\delta_x^2\right]\frac{\delta_x U_j}{2h} = \frac{U_{j-2} - 8U_{j-1} + 8U_{j+1} - U_{j+2}}{12h}$$

$$= (u_x)_j + O(h^4), \tag{2.4}$$

$$\left[I - \frac{1}{12}\delta_x^2\right]\frac{\delta_x^2 U_j}{h^2} = \frac{-U_{j-2} + 16U_{j-1} - 30U_j + 16U_{j+1} - U_{j+2}}{12h^2}$$

$$= (u_{xx})_j + O(h^4). \tag{2.5}$$

By substituting (2.4), (2.5) into (2.1) a pentadiagonal system of linear equations is obtained, and it is necessary to use fictitious points near both boundaries.

A different fourth order approximation can be obtained by following a suggestion of Kreiss [17]. The resulting representation is of an implicit nature in that there are relationships among the function and its derivative at each of three adjacent mesh points. Because the method achieves the highest order accuracy possible on the smallest star it has been called the *compact implicit* method. For the derivatives considered above, following our notation, one obtains

$$\left[I + \frac{\delta_x^2}{6}\right]^{-1}\frac{\delta_x}{2h}U_j = (u_x)_j + O(h^4)$$

or

$$\frac{\delta_x}{2h}U_j = \frac{U_{j+1} - U_{j-1}}{2h} = \left[I + \frac{\delta_x^2}{6}\right](u_x)_j + O(h^4)$$

$$= \frac{(u_x)_{j+1} + 4(u_x)_j + (u_x)_{j-1}}{6} + O(h^4) \tag{2.6b}$$

and

$$\left[I + \frac{\delta_x^2}{12}\right]^{-1} \frac{\delta_x^2}{h^2} U_j = (u_{xx})_j + O(h^4) \tag{2.7a}$$

or

$$\frac{\delta_x^2}{h^2} U_j = \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} = \left[I + \frac{\delta_x^2}{12}\right](u_{xx})_j + O(h^4)$$

$$= \frac{(u_{xx})_{j+1} + 10(u_{xx})_j + (u_{xx})_{j-1}}{12} + O(h^4). \tag{2.7b}$$

Equations (2.6) and (2.7) are derivable by either a Taylor series analysis or a Hermite polynomial interpolation or by thinking of (2.4) and (2.5) as Neumann series representations (up to fourth order) of (2.6) and (2.7), respectively. These formulas had been described in the earlier work of Collatz [6]. As a reference for these formulas in the case of an uneven grid, see [1].

By substituting (2.6) and (2.7) into (2.1) it becomes apparent that in general it is not possible to directly obtain a tractable system of equations in terms of U_j alone. Indeed, to solve the resulting system one can define new variables $F_j \sim (u_x)_j$ and $S_j \sim (u_{xx})_j$ and develop the following 3×3 block tridiagonal system of equations approximating (2.1):

$$\text{(a)} \quad \frac{U_{j+1} - U_{j-1}}{2h} - \frac{F_{j+1} + 4F_j + F_{j-1}}{6} = 0,$$

$$\text{(b)} \quad \frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} - \frac{S_{j+1} + 10S_j + S_{j-1}}{12} = 0, \tag{2.8}$$

$$\text{(c)} \quad b_j F_j + a_j S_j = f_j,$$

where $b_j = b(x_j)$ and $a_j = a(x_j)$ and the above equations hold for $j = 1, 2, \ldots, J-1$. Alternatively, omitting S_j and using only U_j, F_j, a 2×2 block tridiagonal system results from using (2.8a) with

$$\frac{U_{j+1} - 2U_j + U_{j-1}}{h^2} + \frac{1}{12}\left(\frac{b_{j+1}}{a_{j+1}} F_{j+1} + \frac{10b_j}{a_j} F_j + \frac{b_{j-1}}{a_{j-1}} F_{j-1}\right)$$

$$= \frac{1}{12}\left(\frac{f_{j+1}}{a_{j+1}} + \frac{10f_j}{a_j} + \frac{f_{j-1}}{a_{j-1}}\right). \tag{2.9}$$

Equations (2.8) and (2.9) require more work to solve them than the second order method, but generally the higher order accuracy permits one to solve with considerably fewer points to achieve a comparable accuracy. Moreover, for Dirichlet data, no fictitious points are needed. Boundary values ($j = 0, J$) are required for F_j in (2.9) and for F_j and S_j in (2.8). These are obtained by either Hamming type formula [21], or a Pade type formula [9].

These spatial approximation methods have been used by Hirsh [8], Rubin [24, 25],

and Adam [1] with a resulting block tridiagonal system of equations. Our goal was to achieve scalar tridiagonal systems. However it becomes apparent that the use of compact implicit schemes with mixed order derivatives will not result in such simple systems. (See [5], [8], [1] for details.) However, by using a different approach for the spatial operator these goals for parabolic problems are still attainable. Namely, we abandon our attempts to represent the separate derivative terms in the spatial operator and adopt an approach which looks for a relationship on three adjacent points between $L(u)$ and the function u. The resulting fourth order accurate relationship may be derived by a Taylor series development and can be represented in the following equations (see Appendix A for details).

$$q_j^+(L(U))_{j+1} + q_j^0(L(U))_j + q_j^-(L(U))_{j-1} = \frac{r_j^+U_{j+1} + r_j^0U_j + r_j^-U_{j-1}}{h^2} \tag{2.10a}$$

or

$$\frac{Q^{-1}R}{h^2}\, u(x_j) = L(u)_j + O(h^4), \tag{2.10b}$$

where the operators Q and R are each tridiagonal displacement operators, namely,

$$QU_j = q_j^+U_{j+1} + q_j^0U_j + q_j^-U_{j-1}, \tag{2.11a}$$

$$RU_j = r_j^+U_{j+1} + r_j^0U_j + r_j^-U_{j-1}, \tag{2.11b}$$

and where (for simplicity we omit the j index from Q and R)

$$\begin{aligned}
q_j^+ &= 6a_ja_{j-1} + h(5a_{j-1}b_j - 2a_jb_{j-1}) - h^2b_jb_{j-1}, \\
q_j^0 &= 4[15a_{j+1}a_{j-1} - 4h(a_{j+1}b_{j-1} - b_{j+1}a_{j-1}) - h^2b_{j+1}b_{j-1}], \\
q_j^- &= 6a_ja_{j+1} - h(5a_{j+1}b_j - 2a_jb_{j+1}) - h^2b_jb_{j+1}, \\
r_j^+ &= \tfrac{1}{2}[q_j^+(2a_{j+1} + 3hb_{j+1}) + q_j^0(2a_j + hb_j) + q_j^-(2a_{j-1} - hb_{j-1})], \\
r_j^- &= \tfrac{1}{2}[q_j^+(2a_{j+1} + hb_{j+1}) + q_j^0(2a_j - hb_j) + q_j^-(2a_{j-1} - 3hb_{j-1})], \\
r_j^0 &= -(r_j^+ + r_j^-).
\end{aligned} \tag{2.12}$$

These relationships were first presented by Swartz [26]. Equation (2.10a) retains the scalar tridiagonal feature of a second order method while not requiring additional fictitious points at the boundary. Note, in the case where either $a(x)$ or $b(x)$ is identically zero, with the other coefficient identically a constant, the usual compact implicit schemes (either (2.6) or (2.7)) will result. Because of these characteristics we have adopted the terminology of referring to (2.10) as the *operator compact implicit* (OCI) method. Note, a formula of structure similar to (2.10) — (2.12) is presented in Appendix A for the case of an uneven grid. In that case the method is third order accurate.

At least symbolically, we refer to the inverse of Q. The determination of when Q can be inverted is in general a difficult problem. In the case of constant coefficients ($a(x) \equiv$

$a = \text{const}$, $b(x) \equiv b = \text{const}$) the invertibility of Q on l_2 can be fully analyzed by Fourier analysis [26]. Defining $R_c = hb/a$ as the cell Reynolds number, then Q^{-1} exists for (see Appendix C for finite dimensional proof)

$$R_c \leqslant (12)^{1/2} \doteq 3.464. \tag{2.13}$$

The invertibility of Q on a finite dimensional space for variable coefficients is harder to specify in general.

As indicated in Appendix A a standard Taylor series analysis of (2.10a) (i.e. setting as many lower order derivative terms to zero in the truncation error) results in (2.11), (2.12). This Swartz [26] approximation will be referred to as the *standard* OCI scheme. However, it is possible to generate multi-parameter families of fourth order OCI schemes by allowing various lower order derivative terms to appear. In a future paper a generalized Taylor series development will be presented to provide a variety of OCI schemes. In this paper we are exclusively interested in the full implementation of the *standard* OCI scheme. The investigations here will provide guidelines for our future paper on how to best select OCI schemes for wider applicablility and improved robustness.

The above standard OCI scheme can be extracted from the works of [20], [11] where approximation of a parabolic operator was considered. Our approach focuses attention on the more general combinations of time dependent methods with OCI schemes that are possible. It should also be observed that similar compact implicit spatial approximations have been developed under various names, in particular Collatz had sometime ago advocated such approaches which he refers to as "mehrstellen" methods [6]. More recently Collatz describes a format for even more general operator implicit methods. See *Topics in Numerical Analysis, Proceedings of Royal Irish Academy Conference on Numerical Analysis*, (J. J. H. Miller, Ed.), Academic Press, New York, 1972. Several earlier efforts for constant coefficient problems can be found in [7], [16].

3. The Operator Compact Implicit Method

In this section we consider time integration methods which can be used with the OCI spatial approximation for parabolic problems. The method is first developed for a one dimensional problem and then by use of a factorization technique multi-dimensional problems are reduced down to a sequence of one dimensional type problems.

The methods presented here are unconditionally stable. However, as with most other methods for this problem there is a cell Reynolds number condition (2.13). The discussion of stability will be reserved for section IV.

3.1. *One-Dimensional Problems*

Consider the equation

$$u_t = a(x, t)\, u_{xx} + b(x, t)\, u_x = L(u). \tag{3.1}$$

Let n indicate the time dependence in the difference approximation to u at the nth time level.

The first time discretization method considered here is Crank-Nicolson.

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{(Q^{n+1})^{-1} R^{n+1} U_j^{n+1} + (Q^n)^{-1} R^n U_j^n}{2h^2}, \tag{3.2}$$

which requires that one solve

$$[I - \lambda(Q^{n+1})^{-1} R^{n+1}] U_j^{n+1} = [I + \lambda(Q^n)^{-1} R^n] U_j^n \equiv G_j^n, \tag{3.3}$$

where $\lambda = \Delta t/2h^2$. (Note well, for simplicity in the presentation of the equations we will be redefining λ from time to time.) Denote the right-hand side of (3.3) by G_j^n; then

$$[Q^{n+1} - \lambda R^{n+1}] U_j^{n+1} = Q^{n+1} G_j^n. \tag{3.4}$$

Note the following facts about (3.4).

(1) The matrix represented by $Q^{n+1} - \lambda R^{n+1}$ is tridiagonal, thus very easily solved.

(2) No fictitious points, or extra boundary conditions are needed after initialization.

(3) The righthand side G_j^n may be computed by the simple recurrence relation

$$G_j^n = 2U_j^n - G_j^{n-1}. \tag{3.5}$$

(4) The method is second order accurate in time, fourth order accurate in space, and *unconditionally* stable (see Section 4).

(5) For constant coefficients the matrix $(Q - \lambda R)$ invertible for $\lambda \geqslant 0$ when $R_c \leqslant (12)^{1/2}$. (See Appendix C.)

The second method to be considered is adapted from a Lees type scheme [13]. The Lees method combined with an operator compact implicit spatial differencing suggests the following method,

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} = \frac{(Q^n)^{-1} R^n (U_j^{n+1} + U_j^n + U_j^{n-1})}{3h^2}, \tag{3.6}$$

which requires the solution of

$$[I - \lambda(Q^n)^{-1} R^n] U_j^{n+1} = \lambda(Q^n)^{-1} R^n U_j^n + [I + \lambda(Q^n)^{-1} R^n] U_j^{n-1}, \tag{3.7}$$

where now $\lambda = 2\Delta t/3h^2$. Multiply (3.7) by Q^n to obtain

$$[Q^n - \lambda R^n] U_j^{n+1} = \lambda R^n [U_j^n + U_j^{n-1}] + Q^n U_j^{n-1}. \tag{3.8}$$

As pointed out by the reviewer, a matrix multiplication can be saved by grouping (3.8) alternatively as

$$[Q^n - \lambda R^n][U_j^{n+1} - U_j^{n-1}] = \lambda R^n[U_j{}^n + 2U_j^{n-1}]. \tag{3.8'}$$

Note the following facts about (3.8), (3.8').

(1) The matrix to be solved is tridiagonal.

(2) No fictitious points or extra boundary conditions are needed.

(3) The righthand side is easily computed.

(4) The method is second order accurate in time, fourth order accurate in space, and *unconditionally* stable (see Section 4).

(5) It is necessary to generate U_j^1 by some other method to begin the computation.

(6) No iteration is necessary for a nonlinear problem.

3.2. *Two-Dimensional Problems*

We now turn to the consideration of the two dimensional parabolic problem

$$u_t = L_x(u) + L_y(u) \equiv L(u), \tag{3.9}$$

where

$$L_x(u) = au_{xx} + bu_x\,, \tag{3.10a}$$

$$L_y(u) = cu_{yy} + du_y\,. \tag{3.10b}$$

As pointed out in [5], our factorization technique can not be properly adapted with the usual compact implicit method to spatial operators with different order terms. Thus, the discussion here is restricted to the implementation of the OCI method.

For simplicity (3.10) is solved on a rectangular region given by

$$\{(x_j\,, y_k)\colon x_j = jh_x; j = 0, 1,..., J, y_k = kh_y; k = 0, 1,..., K\},$$

where boundary data is prescribed for all t for $j = 0, J$ and for $k = 0, K$, and initial data is prescribed for $t = 0$. As in [3] it is possible to directly extend the method developed here to rectangular-like L-shaped domains. We shall denote the OCI approximations to the operators in (3.10a), (3.10b) by equations (2.10) – (2.12) with subscripts x and y, respectively.

The methods to be presented are of the ADI (Alternating Direction Implicit) variety and their derivations are similar to those developed in [4] for the treatment of the wave equation.

Crank-Nicolson Time Discretization

As before, the first method to be examined uses a Crank-Nicolson time discretization

$$\frac{U_{j,k}^{n+1} - U_{j,k}^{n}}{\Delta t} = \frac{(Q_x^{n+1})^{-1} R_x^{n+1} U_{j,k}^{n+1} + (Q_x^{n})^{-1} R_x^{n} U_{j,k}^{n}}{2h_x^2} + \frac{(Q_y^{n+1})^{-1} R_y^{n+1} U_{j,k}^{n+1} + (Q_y^{n})^{-1} R_y^{n} U_{j,k}^{n}}{2h_y^2}. \tag{3.11}$$

As in the one dimensional case where each of the derivatives was represented separately, there is no way to "unravel" the different inverse operators in (3.11) except by adding to (3.11) the by now familiar second order perturbation cross term

$$-\frac{\Delta t^2}{4} \frac{\delta_t^+}{\Delta t} \left[(Q_x^{n})^{-1} \frac{R_x^{n}}{h_x^2}\right]\left[(Q_y^{n})^{-1} \frac{R_y^{n}}{h_y^2}\right] U_{j,k}^{n}, \tag{3.12}$$

where δ_t^+ is the forward difference operator. The resulting equations are easily seen to assume the factored form

$$[I - \lambda_x (Q_x^{n+1})^{-1} R_x^{n+1}][I - \lambda_y (Q_y^{n+1})^{-1} R_y^{n+1}] U_{j,k}^{n+1} = [I + \lambda_x (Q_x^{n})^{-1} R_x^{n}][I + \lambda_y (Q_y^{n})^{-1} R_y^{n}] U_{j,k}^{n}, \tag{3.13}$$

where $\lambda_x = \Delta t/2h_x^2$ and $\lambda_y = \Delta t/2h_y^2$.

By introducing an intermediate variable, (3.14) splits into two tridiagonal systems

$$[I - \lambda_x (Q_x^{n+1})^{-1} R_x^{n+1}] Z_{j,k}^{n+1} = G_{j,k}^{n}, \tag{3.14a}$$

$$[I - \lambda_y (Q_y^{n+1})^{-1} R_y^{n+1}] U_{j,k}^{n+1} = Z_{j,k}^{n+1}, \tag{3.14b}$$

where

$$G_{j,k}^{n} = [I + \lambda_x (Q_x^{n})^{-1} R_x^{n}][I + \lambda_y (Q_y^{n})^{-1} R_y^{n}] U_{j,k}^{n}. \tag{3.15}$$

$G_{j,k}^{n}$ is easily computed using previous values by the relationship

$$G_{j,k}^{n} = 2(U_{j,k}^{n} - Z_{j,k}^{n} + \lambda_x (Q_x^{n})^{-1} R_x^{n} U_{j,k}^{n}) + G_{j,k}^{n-1}. \tag{3.16}$$

In order to solve (3.14a), boundary conditions for $Z_{j,k}^{n+1}$ on the $x = $ const. boundaries are needed. Likewise, in order to solve (3.14b) boundary conditions for $Z_{j,k}^{n+1}$ on the $y = $ const. boundaries are needed. These intermediate boundary conditions are obtained in the following manner:

(1) Use one sided differences to compute $Z_{j,k}^{n+1}$ at the four corner points. Here, the fact that $Z_{j,k}^{n+1}$ is a fourth order approximation to $u_{j,k}^{n+1} - (\Delta t/2)(cu_{yy} + du_y)_{j,k}^{n+1}$ is used.

(2) On the x = const. boundaries, (3.14b) is employed to solve for $Z_{j,k}^{n+1}$:

$$Q_y^{n+1} Z_{j,k}^{n+1} = [Q_y^{n+1} - \lambda_y R_y^{n+1}]\, U_{j,k}^{n+1}.$$

(3) Now that the x = const. boundary data for $Z_{j,k}^{n+1}$ have been obtained, one can proceed with the x sweeps of the ADI scheme using (3.14a). Included in these sweeps are the y = const. boundaries. Thus, the $Z_{j,k}^{n+1}$ boundary values necessary for the y sweeps in (3.14b) are now fully available.

Lees Time Discretization

Finally, a method which is a generalization of the one dimensional OCI-Lees scheme is examined. Approximate (3.10) by

$$\frac{U_{j,k}^{n+1} - U_{j,k}^{n-1}}{2\Delta t} = \left[(Q_x{}^n)^{-1} \frac{R_x{}^n}{h_x{}^2} + (Q_y{}^n)^{-1} \frac{R_y{}^n}{h_y{}^2}\right] \frac{(U_{j,k}^{n+1} + U_{j,k}^n + U_{j,k}^{n-1})}{3}. \quad (3.17)$$

Again, in order to obtain a factored tridiagonal method one adds the second order perturbation term

$$-\frac{4}{9}\Delta t^2 \left[(Q_x{}^n)^{-1} \frac{R_x{}^n}{h_x{}^2}\right]\left[(Q_y{}^n)^{-1} \frac{R_y{}^n}{h_y{}^2}\right] \frac{\delta_t}{\Delta t}\, U_{j,k}^n$$

to obtain

$$\begin{aligned}
&[I - \lambda_x(Q_x{}^n)^{-1} R_x{}^n][I - \lambda_y(Q_y{}^n)^{-1} R_y{}^n]\, U_{j,k}^{n+1} \\
&\quad = [\lambda_x(Q_x{}^n)^{-1} R_x{}^n + \lambda_y(Q_y{}^n)^{-1} R_y{}^n]\, U_{j,k}^n \\
&\qquad + [I + \lambda_x(Q_x{}^n)^{-1} R_x{}^n][I + \lambda_y(Q_y{}^n)^{-1} R_y{}^n]\, U_{j,k}^{n-1}. \qquad (3.18)
\end{aligned}$$

Or alternatively, as pointed out by the reviewer, a more efficient form results from solving the left had side for $(U_{j,k}^{n+1} - U_{j,k}^{n-1})$ namely,

$$\begin{aligned}
&[I - \lambda_x(Q_x{}^n)^{-1} R_x{}^n][I - \lambda_y(Q_y{}^n)^{-1} R_y{}^n][U_{j,k}^{n+1} - U_{j,k}^{n-1}] \\
&\quad = [\lambda_x(Q_x{}^n)^{-1} R_x{}^n + \lambda_y(Q_y{}^n)^{-1} R_y{}^n][U_{j,k}^n + 2U_{j,k}^{n-1}] \qquad (3.18')
\end{aligned}$$

Denote the righthand side of (3.18) by $G_{j,k}^n$, introduce an intermediate value $Z_{j,k}^{n+1}$, and apply our usual splitting to obtain

$$[Q_x{}^n - \lambda_x R_x{}^n]\, Z_{j,k}^{n+1} = Q_x{}^n G_{j,k}^n, \qquad (3.19a)$$

$$[Q_y{}^n - \lambda_y R_y{}^n]\, U_{j,k}^{n+1} = Q_y{}^n Z_{j,k}^{n+1}. \qquad (3.19b)$$

Note the following:

(1) There does not appear to be any simple algorithm for computing the righthand side. However, upon multiplying $G_{j,k}^n$ by $Q_x{}^n$ (as in (3.19a)) it is clear that only a backsolve of the tridiagonal matrix $Q_y{}^n$ for different righthand sides is required.

(2) The intermediate boundary condition for $Z_{j,k}^{n+1}$ is obtained in the same manner as in the Crank-Nicolson case once the $Z_{j,k}^{n+1}$ at the four corner points are computed.

(3) As in the one-dimensional problem an extra plane of information must be generated to begin the computation and no iteration is necessary for nonlinear problems.

4. Stability Considerations

In this section we discuss two stability characteristics which enter into the evaluation of the usefulness of difference schemes for parabolic equations. At the threshold one must consider the Lax-Richtmyer stability of the evolutionary operator [22]. More recently, it has come to be appreciated that the stability characteristics associated with the spatial operator should be examined [23], [9], [12]. The ability of a spatial difference scheme to resolve the spatial variation in a region of sharp gradients (boundary layer) often gives rise to a so called cell Reynolds number condition. Here we examine these stability questions for the compact implicit schemes previously discussed.

4.1. *Temporal Stability Analysis*

For the case of constant coefficients one can analyze the L_2 stability of the difference scheme of interest by Fourier analysis [22]. Here the discussion is limited to OCI schemes. Substituting $U_j{}^n = \rho_{\text{CN}}^n e^{ij\theta}$ into (3.2) yields

$$\rho_{\text{CN}} = \left(\frac{2 + \lambda I(\theta)}{2 - \lambda I(\theta)}\right), \tag{4.1a}$$

where

$$I(\theta) = 3a \frac{24(\cos\theta - 1) + iR_c(12 - R_c{}^2)\sin\theta}{30 - 2R_c{}^2 + (6 - R_c{}^2)\cos\theta + i\,3R_c\sin\theta}. \tag{4.1b}$$

The term $I(\theta)$ is associated with the Fourier transform of the spatial operator alone [26]. For stability it is required that $|\rho_{\text{CN}}^2| \leqslant 1$. Imposing this condition directly on (4.1) yields, Re $I(\theta) \leqslant 0$ as a necessary and sufficient condition for stability. This latter condition requires that

$$24(\cos\theta - 1)[30 - 2R_c{}^2 + (6 - R_c{}^2)\cos\theta] + (12 - R_c{}^2)\,3R_c{}^2\sin^2\theta \leqslant 0.$$

Collecting terms and factoring out a $(\cos\theta - 1)$ term yields

$$(\cos\theta - 1)[720 - 84R_c{}^2 + 3R_c{}^4 + \cos\theta(144 - 60R_c{}^2 + 3R_c{}^4)] \leqslant 0 \tag{4.2}$$

Regrouping, and noting from (2.13) that the region of interest is $R_c^2 \leqslant 12$, yields

$$(\cos\theta - 1)[12(12 - R_c^2) + 12(12 - R_c^2\cos\theta) + 288 + (144 - 72R_c^2 + 3R_c^4) \times (\cos\theta + 1)] \leqslant 0. \tag{4.3}$$

To see that this inequality is always satisfied for $R_c^2 \leqslant 12$, note that the term in the left parentheses is $\leqslant 0$ and the term in the bracket is the sum of four terms, the first three of which are clearly nonnegative. The last term in the bracket takes on a negative minimum at $R_c^2 = 12$ and even when $\cos\theta = 1$ this minimum is just the negative of the third term. This establishes that $|\rho_{CN}| \leqslant 1$ for $R_c^2 \leqslant 12$, and the unconditional temporal stability of OCI-CN. The two space dimension case follows directly.

To see that OCI-Lees is similarly stable, substitute $U_j^n = \rho_L^n e^{ij\theta}$ into (3.6) to obtain a quadratic for ρ_L,

$$\rho_L^2 + \tfrac{1}{2}(K + 1)\rho_L + K = 0 \tag{4.4}$$

where $K = \rho_{CN}$ as in (4.1) above (with λ replaced by $\frac{2}{3}\lambda$). Since the OCI-CN method is unconditionally stable, clearly in the range of $R_c^2 \leqslant 12$, $|K| \leqslant 1$. The stability of the OCI-Lees method is now contained in the statement of the following lemma

LEMMA. *For the roots* ρ_L *of* (4.4)

$$|\rho_L| \leqslant 1 \qquad \text{iff } |K| \leqslant 1.$$

Proof. First we prove the lemma for the case of equality in both inequalities. Say $K = e^{i\phi}$ then solve for ρ_L directly as $\rho_L = \bar{\rho}e^{i\phi/2}$ where $\bar{\rho} = e^{\pm i\psi}$, $\cos(\phi/2) = -2\cos\psi$. Clearly such ψ exists and thus $|\rho_L| = |\bar{\rho}| = 1$. On the other side, if $|\rho_L| = 1$, say $\rho_L = e^{i\phi/2}$, then soving for K yields

$$K = -\frac{1 + 2e^{i\phi/2}}{1 + 2e^{-i\phi/2}}.$$

Thus $|K| = 1$. This completes the proof that $|\rho_L| = 1$ iff $|K| = 1$. To show that $|\rho_L| < 1$ iff $|K| < 1$ examine the variation of the roots ρ_L with respect to the unit circle as K varies from 0 to $+\infty$. At $K = 0$, $\rho_L = 0, -\frac{1}{2}$, both roots are inside the unit circle. Now by a connectivity argument, and the fact that the ρ_L roots depend continuously on the coefficient K [18], varying K such that $|K| < 1$ then the corresponding roots ρ_L must remain strictly inside the unit circle. Indeed, if some root "touched" the unit circle, i.e., $|\rho_L| = 1$, then by our proof above $|K| = 1$. This argument demonstrates that for $|K| < 1$, $|\rho_L| < 1$. Conversely at $K = +\infty$ both ρ_L roots are outside the unit circle thus, again by a connectivity argument the ρ_L must remain outside the unit circle for all K such that $|K| > 1$.

Finally the stability of the two dimensional Lees-OCI method is established using the above Lemma. Substituting $U_{j,k}^n = \rho^n e^{i(j\theta + k\phi)}$ into (3.19) one obtains

$$\rho^2 + \tfrac{1}{2}(1 - \alpha\beta)\rho - \alpha\beta = 0, \tag{4.5}$$

where $\alpha = (1 + \lambda_x l(\theta))/(1 - \lambda_x l(\theta))$, $\beta = (1 + \lambda_y l(\phi))/(1 - \lambda_y l(\phi))$, and where $l(\theta)$ and $l(\phi)$ are defined by (4.1b) for x and y, respectively. Noting that α, β are each separately in the form of a ρ_{CN} as found above, one concludes from the above lemma, that in the range $|R_c^x| \leqslant (12)^{1/2}$, $|R_c^y| \leqslant (12)^{1/2}$ (i.e., where the cell Reynolds number invertibility condition is satisfied for each spatial operator) $|\alpha| \leqslant 1$, $|\beta| \leqslant 1$. Now identifying $K = -\alpha\beta$ in (4.5) clearly our above Lemma implies $|\rho| \leqslant 1$.

4.2. *Spatial Stability*

Experience with computations involving diffusion convection equations has long shown that nonphysical oscillations will appear in the computed solution when the spatial mesh size is not sufficiently small [23], [9], [12]. Here we use the standard linear analysis to attempt to predict some of the cell Reynolds number limitations associated with the methods discussed in this paper. Through-out this subsection, for discussion purposes, we will consider the following model "boundary layer" problem

$$\begin{gathered} au_{xx} - bu_x = 0, \qquad a, b \text{ positive constants} \\ u(0) = 0, \qquad u(1) = 1 \end{gathered} \tag{4.6}$$

where in general b/a is large. Note, the solution of (4.6) is

$$u(x_j) = c_1 + c_2 e^{bx/a} = c_1 + c_2 e^{R_c j}, \qquad x_j = j\,\Delta x. \tag{4.7}$$

Operator Compact Implicit

The spatial stability analysis for this method is quite straightforward and provides a practical guide for the range of usefulness of the scheme. Assuming $Q^{-1}RU_j = 0$ is applied to (4.6) then one is to consider the three point homogeneous difference equation

$$RU_j = 0. \tag{4.8}$$

Substituting a solution of the form $U_j = \mu^j$ into (4.8) (using (2.11b)) leads to the general difference solution

$$U_j = c_1 + c_2\mu^j; \qquad \mu = \frac{24 + R_c(12 - R_c^2)}{24 - R_c(12 - R_c^2)}. \tag{4.9}$$

Three cases are possible for general R_c :

1. $R_c < (12)^{1/2}$, $\mu > 1$. The difference solution is monotone increasing, concave up, and properly approximates the true solution

2. $(12)^{1/2} < R_c < 4.207607$ (R_c value where numerator of μ vanishes), $0 < \mu < 1$. The difference solution is monotone increasing but concave down and completely wrong.

3. $R_c > 4.207607$, $-1 < \mu < 0$. The difference solution is oscillatory.

In summary, the spatial modal analysis, of essentially the operator R indicates that the cell Reynolds number R_c should be restricted to the exact same condition used for the invertibility of Q, i.e. $R_c \leqslant (12)^{1/2}$. This represents no additional limitation on how one would prudently employ the OCI method.

Compact Implicit-Block Methods

To check the spatial stability of any of the block tridiagonal compact implicit methods it is sufficient to consider any one of them since each method (either the 2×2, or the 3×3) has the same set of characteristic roots. Thus, the fundamental modes of the system can be obtained by taking a solution of (2.8a, c) for (4.6) in the form

$$\begin{pmatrix} U_j \\ F_j \end{pmatrix} = \mu^j \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \qquad j = 0, 1, \ldots, J. \tag{4.10}$$

A nontrivial solution results if the determinantal equation

$$(\mu - 1)[(4 - R_c)\,\mu^3 + (12 - 11R_c)\,\mu^2 - (12 + 11R_c)\,\mu - (4 + R_c)] = 0 \tag{4.11}$$

holds. A study of (4.11) will at least provide an indication of what types of nonphysical results are possible. However, there are four roots (and corresponding arbitrary constants) to contend with now. A proper analysis involves consideration of the particular schemes used to approximate the required derivatives at the boundaries. Here we present a qualitative analysis of the possible numerical solutions of (4.6) along with some illustrative computational experiments.

For our model example (4.6) one would like to obtain a U_j which is monotone, or at least, does not have large oscillatory modes which are dominant. Generally, this is accomplished by restricting R_c so that if $\mathrm{Re}\,\mu < 0$ then $|\mu| \leqslant 1$. However, a simple inspection of the bracketed cubic in (4.11) at $\mu = -1, 0, 1$ reveals that such a condition can not be found, since there are (for $R_c < 4$) always three real roots of (4.11), μ_+, μ_-, μ_0 such that

$$\mu_+ > 1, \mu_- < -1, -1 < \mu_0 < 0.$$

Thus the block tridiagonal schemes for (4.6) do not satisfy what has been generally considered a reasonable stability requirement. Yet the schemes are useful in practice; see Section 5. The reason why the oscillatory modes do not even appear in some calculations, let alone dominate them, is tied to a consideration of the way the coefficients are determined by the boundary conditions.

A series of numerical experiments was made for (4.6) and qualitatively we can conclude the following. In the range of R_c values ($0 < R_c < 4/(15)^{1/2} = 1.0328$) where $\mu_+ \leqslant |\mu_-|$ no dominant oscillations occur. While in the range $4/(15)^{1/2} \leqslant R_c \leqslant 2.14383$ corresponding to the μ_+ range $|\mu| \leqslant \mu_+ \leqslant e^{R_c}$ the negative oscillations tend to affect more of the region. For $\mu_+ > e^{R_c}$ the oscillations are apparent in most of the region. Typical results are presented for $R_c = 1.0, 1.5, 2.0, 2.4$ in Tables 4.1, 4.2.

TABLE 4.1

Compact Implicit (2 × 2) Block Tridiagonal Solution of (4.6)

	$R_e = 1.0, b/a = 30$			$R_e = 1.5, b/a = 45$	
j	U_j	$u(x_j)$	j	U_j	$u(x_j)$
1	0.	0.	1	0.	0.
2	.16513E−12	.16079E−12	2	.26899E−13	.99664E−19
3	.61378E−12	.59786E−12	3	−.50788E−13	.54633E−18
4	.18330E−11	.17860E−11	4	.14292E−12	.25481E−17
5	.51414E−11	.50155E−11	5	−.33484E−12	.11520E−16
6	.14134E−10	.13794E−10	6	.84259E−12	.51727E−16
7	.38533E−10	.37658E−10	7	−.20584E−11	.23192E−15
8	.10486E−09	.10253E−09	8	.50913E−11	.10395E−14
9	.28480E−09	.27885E−09	9	−.12520E−10	.46589E−14
10	.77395E−09	.75816E−09	10	.30904E−10	.20880E−13
12	.57087E−08	.56027E−08	12	.18793E−09	.41938E−12
13	.15495E−07	.15230E−07	13	−.45992E−09	.18795E−11
14	.42104E−07	.41399E−07	14	.11474E−08	.84235E−11
15	.11428E−06	.11254E−06	15	−.27645E−08	.37751E−10
16	.31053E−06	.30590E−06	16	.70919E−08	.16919E−09
17	.84279E−06	.83153E−06	17	−.16225E−07	.75826E−09
18	.22903E−05	.22603E−05	18	.45561E−07	.33983E−08
19	.62155E−05	.61442E−05	19	−.87359E−07	.15230E−07
20	.16892E−04	.16702E−04	20	.32645E−06	.68256E−07
21	.45839E−04	.45400E−04	21	−.30805E−06	.30590E−06
22	.12458E−03	.12341E−03	22	.29748E−05	.13710E−05
23	.33806E−03	.33546E−03	23	.25611E−05	.61442E−05
24	.91885E−03	.91188E−03	24	.37845E−04	.27536E−04
25	.24931E−02	.24788E−02	25	.10386E−03	.12341E−03
26	.67769E−02	.67379E−02	26	.62403E−03	.55308E−03
27	.18386E−01	.18316E−01	27	.23905E−02	.24788E−02
28	.49982E−01	.49787E−01	28	.11645E−01	.11109E−01
29	.13560E+00	.13534E+00	29	.49587E−01	.49787E−01
30	.36864E+00	.36788E+00	30	.22727E+00	.22313E+00
31	.10000E+01	.10000E+01	31	.10000E+01	.10000E+01

TABLE 4.2

Compact Implicit (2 × 2) Block Tridiagonal Solution of (4.6)

	$R_e = 2.0, b/a = 60$			$R_e = 2.4, b/a = 72$	
j	U_j	$u(x_j)$	j	U_j	$u(x_j)$
1	0.	0.	1	0.	0.
2	.40527E−11	.55946E−25	2	.10396E−09	.53927E−30
3	−.60778E−11	.46933E−24	3	−.13186E−09	.64837E−29
4	.16219E−10	.35239E−23	4	.34476E−09	.72010E−28
5	−.32360E−10	.26094E−22	5	−.60936E−09	.79432E−27
6	.73391E−10	.19287E−21	6	.12990E−08	.87565E−26
7	−.15680E−09	.14252E−20	7	−.25178E−08	.96525E−25
8	.34427E−09	.10531E−19	8	.51159E−08	.10640E−23
9	−.74643E−09	.77811E−19	9	−.10152E−07	.11729E−22
10	.16277E−08	.574956−18	10	.20383E−07	.12929E−21
11	−.35401E−08	.42484E−17	11	−.40686E−07	.14252E−20
12	.77089E−08	.31391E−16	12	.81453E−07	.15710E−19
13	−.16777E−07	.23195E−15	13	−.16283E−06	.17317E−18
14	.36522E−07	.17139E−14	14	.32573E−06	.19089E−17
15	−.79496E−07	.12664E−13	15	−.65138E−06	.21042E−16
16	.17304E−06	.93576E−13	16	.13028E−05	.23195E−15
17	−.37667E−06	.69144E−12	17	−.26056E−05	.25569E−14
18	.81991E−06	.51091E−11	18	.52113E−05	.28185E−13
19	−.17847E−05	.37751E−10	19	−.10423E−04	.31068E−12
20	.38851E−05	.27895E−09	20	.20845E−04	.34247E−11
21	−.84541E−05	.20612E−08	21	−.41690E−04	.37751E−10
22	.18423E−04	.15230E−07	22	.83381E−04	.41614E−09
23	−.39949E−04	.11254E−06	23	−.16676E−03	.45872E−08
24	.88083E−04	.83153E−06	24	.33357E−03	.50565E−07
25	−.18344E−03	.61442E−05	25	−.66651E−03	.55739E−06
26	.46035E−03	.45400E−04	26	.13401E−02	.61442E−05
27	−.55256E−03	.33546E−03	27	−.26014E−02	.67729E−04
28	.45128E−02	.24788E−02	28	.60827E−02	.74659E−03
29	.14551E−01	.18316E−01	29	−.23290E−02	.82297E−02
30	.14782E+00	.13534E+00	30	.11462E+00	.90718E−01
31	.10000E+01	.10000E+01	31	.10000E+01	.10000E+01

The case $R_c = 2.4$ is particularly interesting because here $\mu_- = -2$ and the μ_- term is the dominant term in the solution in the interior part of region as is apparent by observing that the ratio of successive terms is -2.

Since the circumstances where these spatial oscillations will *dominate* (they are *always* present for constant coefficients) is not easily anticipated, one should be aware of this potential problem for the block compact implicit methods.

5. Numerical Experiments

5.1. *Introduction*

In this section results of numerical experiments conducted with the various schemes that were discussed in Sections 2 and 3 are presented. These calculations were performed in order to determine the viability of the OCI method for solving parabolic problems, to understand its characteristics and limitations, and to compare its performance with classical second order techniques as well as to other fourth order approaches.

One of our major concerns is the efficiency of the various schemes, i.e. computation time required to obtain a given accuracy. Obviously this is machine as well as programmer dependent. In order not to bias any of the techniques care was taken to program the algorithms in an efficient and consistent manner. The computing times that are given include time for: matrix setups, inversions, boundary condition evaluations and (for nonlinear problems) iteration procedures. All results were computed on the NSWC/WOL *CDC* 6500 computer.

The operation count estimates (multiplications and divisions) for the block tridiagonal inversion algorithm is given in [10] as

$$\text{ops} = (3n - 2)(m^3 + m^2) \tag{5.1}$$

where m is the order of the block and n is the number of equations. This estimate assumes full blocks. However, if the specific values of the elements of the blocks are taken into account, e.g., zeros and ones, the actual operation count can be greatly reduced. Such modified algorithms were used to obtain the reported results.

A comparison of operation counts for the various inversion procedures (assuming full blocks) and the modified algorithms are presented in Table 5.1. Also included there are the matrix setup operation counts. Note that for the block methods the inversion of the matrix is the dominant factor in the running time, while for the OCI technique the matrix setup accounts for most of the time,

For completeness the operation count estimates for the explicit pentadiagonal method are also included (although no calculations were performed with it). The low operation count is offset by the need for extrapolation formulas near both boundaries. The unfavorable spatial stability characteristics of the block methods (four roots) are also possessed by this method.

TABLE 5.1

Matrix Setup and Inversion Operations[a] Uniform Mesh

Matrix	Inversion		Setup	Total setup + actual inversion
	Estimated[b]	Actual		
Scalar tridiagonal (OCI–CN)	$5N - 4$	$5N - 4$	$22N - 22$	$27N - 26$
2×2 Block tridiagonal (C–N)	$36N - 24$	$27N - 60$	$8N + 16$	$35N - 44$
3×3 Block tridiagonal (CN)	$108N - 72$	$49N - 62$	$4N + 24$	$53N - 38$
Scalar pentadiagonal	$11N - 16$		$10N$[c]	$21N-16$

[a] Here it is assumed that multiplications and divisions are equivalent. However, on certain machines this may not be true; e.g., on the CDC 6600 a division is comparable to six multiplications. The operation counts would have to be changed accordingly for the methods.

[b] Reference [10].

[c] Does not include operation counts for extrapolation formulas for points adjacent to the boundaries.

5.2. *Linear Parabolic Equation*

The first numerical experiment involved the solution of a one dimensional linear parabolic partial differential equation with variable coefficients

$$u_t = a(x, t)\, u_{xx} + b(x, t)\, u_x\,; \qquad t \geqslant 0; 0 \leqslant x \leqslant 1, \tag{5.2a}$$

where

$$a(x, t) = \frac{1}{2}\frac{(x + 1)}{(t + 2)^2}\,; \qquad b(x, t) = \frac{1}{2}\frac{(x + 1)}{(t + 2)}\,,$$

with the exact solution

$$u(x, t) = u_e(x, t) = \exp[(x + 1)(t + 2)]. \tag{5.2b}$$

Initial and boundary conditions are given by

$$\begin{aligned} u(x, 0) &= u_e(x, 0), \\ u(0, t) &= u_e(0, t); \qquad u(1, t) = u_e(1, t). \end{aligned} \tag{5.2c}$$

This example was constructed in order to test the stability and convergence properties of the methods under consideration for a variable coefficient problem. Results are shown in Table 5.2 and Fig. 1. All the methods tested were stable and show the predicted convergence rates. Crank–Nicolson temporal integration was used for all the schemes.

TABLE 5.2

Linear Variable Coefficient Parabolic Equation

$$u_t = a(x,t)u_{xx} + b(x,t)u_x$$
$$u = \exp\{(x+1)(t+2)\}$$

Method	N	Time steps ($\Delta t = 0.0001$)	L_2 Error	L_2 Rate	Computing time[a] (sec)
Second order Crank–Nicolson	100	2000	$0.20 * 10^{-04}$		35.5
	160	2000	$0.79 * 10^{-05}$	1.98	55.4
	200	2000	$0.51 * 10^{-05}$	1.96	68.8
	400	2000	$0.13 * 10^{-05}$	1.97	135.5
3×3 block Crank–Nicolson	5	2000	$0.15 * 10^{-04}$		12.2
	10	2000	$0.90 * 10^{-06}$	4.06	17.7
	20	2000	$0.51 * 10^{-07}$	4.14	30.4
	40	2000	$0.20 * 10^{-08}$	4.67	58.5
2×2 block Crank–Nicolson	5	2000	$0.83 * 10^{-05}$		7.4
	10	2000	$6.70 * 10^{-06}$	3.58	11.9
	20	2000	$0.48 * 10^{-07}$	3.87	20.8
	40	2000	$0.20 * 10^{-08}$	4.59	40.2
Operator compact implicit Crank–Nicolson	5	2000	$0.24 * 10^{-04}$		5.4
	10	2000	$0.15 * 10^{-05}$	4.00	8.8
	20	2000	$0.94 * 10^{-07}$	4.00	15.6
	40	2000	$0.41 * 10^{-08}$	4.52	30.0

[a] Computation times are for a CDC 6500.

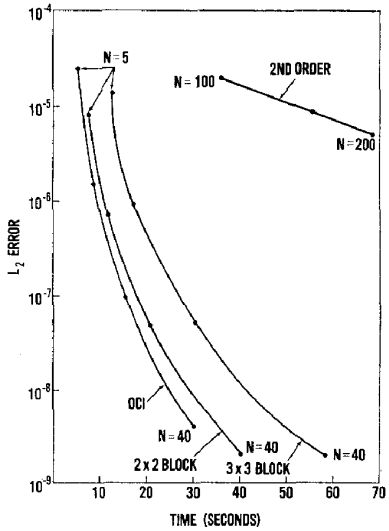

FIG. 1. Linear variable coefficient equation. L_2 Error vs running time.

Of basic interest is the savings that can be obtained in storage and computational time. As noted in Table 5.2 and Fig. 1 the OCI technique compares favorably with the other methods tested. This is not wholly unexpected, since the block methods require additional work to compute the first and/or second derivatives.

It is also important to note the differences in the computed L_2 errors of the fourth order methods. These result from several factors among which are the local truncation error and boundary conditions. The spatial truncation errors, which are dominant for the case considered, are given below.

Compact Implicit-(Block Methods)

First derivative:

$$E_f = \frac{-h^4}{180} u^{(5)} + O(h^6).$$

Second derivative:

$$E_s = \frac{-h^4}{240} u^{(6)} + O(h^6).$$

Thus for Eq. (2.1) with a and b constant the local spatial truncation error at point j would be

$$E = -h^4 \left(\frac{a}{240} u_j^{(6)} + \frac{b}{180} u_j^{(5)}\right). \tag{5.3}$$

OCI

Specializing Eq. (A16) for constant coefficients yields

$$E = -h^4 \left(\frac{a}{200} u_j^{(6)} + \frac{3b}{200} u_j^{(5)}\right). \tag{5.4}$$

In achieving a scalar tridiagonal system, the OCI technique leads to an unsymmetric difference formula and thus has a larger local truncation error than the block methods that were derived from symmetric formulations. Were it not for the different boundary conditions, Pade relations for the 3×3 block method and a Hamming type formula for the 2×2 block method (see [5] for details), both block techniques would give identical errors.

5.2.1. *General boundary conditions.* The OCI method can also be applied to problems with more general boundary conditions of the form

$$Au_x + Bu = g. \tag{5.5}$$

A linear fourth order accurate expression is sought relating u_x at the boundary with

the coefficients in (5.6) can be evaluated. These coefficients and the truncation error are given in Appendix B. As an example, Eq. (52a) was solved with the boundary conditions:

$$x = 0, \qquad u + u_x = u_e(0) + (u_e)_x(0) = (t + 3)\exp[t + 2],$$
$$x = 1, \qquad u = u_e(1) = \exp[2(t + 2)].$$

TABLE 5.3

Linear Variable Coefficient Parabolic Equation

$u_t = a(x, t)u_{xx} + b(x, t)u_x$
$u = u_e = \exp(x + 1)(t + 2)$
$u(0) + u_x(0) = (t + 1)\exp[t + 2]$, $u(1) = u_e(1)$
OCI — 2000 Time steps

N	L_2 Error	L_2 Rate	Computing time[a] (sec)
5	$0.222 * 10^{-02}$		6.57
		4.796	
10	$0.122 * 10^{-03}$		9.58
		4.049	
20	$0.737 * 10^{-05}$		16.71
		4.063	
40	$0.441 * 10^{-06}$		30.42

[a] Computation times for a CDC 6500.

Table 5.3 shows the L_2 errors and L_2 rates of convergence for different mesh widths. Comparisons with the results in Table 5.2 indicate that for general boundary conditions the L_2 error is larger and the computation time is increased.

5.3. *Burgers Equation*

In order to test the various methods for a nonlinear problem that is indicative of viscous flows the one dimensional Burgers equation was investigated. Consider

$$u_t = -(u - \alpha)\, u_x + \nu u_{xx}\,. \tag{5.7}$$

With the exact steady state solution given by

$$u_e(x) = \alpha\{1 - \tanh(\alpha x/2\nu)\}. \tag{5.8}$$

Near $x = 0$, $u(x)$ exhibits large gradients, and as $\nu \to 0$, a steep shock wave forms. The ability to resolve this flow field would demonstrate the viability of the various methods.

Solutions were obtained in the domain $-5 \leqslant x \leqslant 5$ with $\alpha = \frac{1}{2}$ and for various

values of ν, and with the exact values of $u(x)$ specified at the boundaries. The initial conditions employed for all cases are

$$u(x, 0) = \begin{cases} 1, & -5 < x < 0, \\ 0.5, & x = 0, \\ 0, & 0 < x < 5. \end{cases}$$

Results of computations with the OCI (Crank–Nicolson and Lees) methods and the second order Crank-Nicolson finite difference scheme are presented in Tables 5.4-5.7 and Fig. 2.

Since Eq. (5.5) is nonlinear, a linearization such as proposed by McDonald and Briley [14] or iteration is necessary for the Crank-Nicolson temporal discretization. Here, we adapt the OCI method with successive approximation for the nonlinear term, uu_x, i.e.,

$$U_j^{n+1}(U_x)_j^{n+1} = U^*{}_j(U_x)_j^{n+1}, \tag{5.9}$$

where U_j^* is the latest iterant value. This procedure converges linearly.

The second order finite difference scheme uses a different type of linearization, viz.

$$(U_j - \alpha)^{n+1/2}\,(U_x)_j^{n+1/2} = \frac{(U_j^{n+1/2} - \alpha)}{2}\left\{\frac{U_{j+1}^{n+1} - U_{j-1}^{n+1}}{2\Delta x} + \frac{U_{j+1}^{n} - U_{j-1}^{n}}{2\Delta x}\right\}, \tag{5.10a}$$

where $U_j^{n+1/2}$ is replaced by

$$(U^*{}_j + U_j{}^n)/2, \tag{5.10b}$$

TABLE 5.4

Steady State Solution of Burgers Equation:
Second Order Crank–Nicolson

ν	N	$h = \Delta x$	$\nu\Delta t/h^2$	Max error	L_2 Error	L_2 Rate
0.500	50	0.20	6.25	$0.633 * 10^{-3}$	$0.125 * 10^{-2}$	
	100	0.10	25.00	$0.158 * 10^{-3}$	$0.311 * 10^{-3}$	2.007
	200	0.05	100.00	$0.395 * 10^{-4}$	$0.778 * 10^{-4}$	1.999
0.250	50	0.20	3.125	$0.303 * 10^{-2}$	$0.442 * 10^{-2}$	
	100	0.10	12.50	$0.747 * 10^{-3}$	$0.109 * 10^{-2}$	2.020
	200	0.05	50.00	$0.186 * 10^{-3}$	$0.273 * 10^{-3}$	1.997
0.125	50	0.20	1.5625	$0.128 * 10^{-1}$	$0.131 * 10^{-1}$	
	100	0.10	6.25	$0.303 * 10^{-2}$	$0.314 * 10^{-2}$	2.061
	200	0.05	25.00	$0.749 * 10^{-3}$	$0.775 * 10^{-3}$	2.019
0.062	50	0.20	0.775	$0.694 * 10^{-1}$	$0.473 * 10^{-1}$	
	100	0.10	3.100	$0.130 * 10^{-1}$	$0.940 * 10^{-2}$	2.331
	200	0.05	12.40	$0.308 * 10^{-2}$	$0.224 * 10^{-2}$	2.069
0.031	100	0.10	1.55	$0.694 * 10^{-1}$	$0.334 * 10^{-1}$	
	200	0.05	6.20	$0.130 * 10^{-2}$	$0.665 * 10^{-2}$	2.328

TABLE 5.5

Steady State Solution of Burgers Equation:
OCI Crank–Nicolson and Lees

ν	N	$h = \Delta x$	$\nu\Delta t/h^2$	Max error	L_2 Error	L_2 Rate
0.500	10	1.00	0.25	$0.132 * 10^{-2}$	$0.231 * 10^{-2}$	
	20	0.50	2.00	$0.796 * 10^{-4}$	$0.137 * 10^{-3}$	4.076
	50	0.20	6.25	$0.205 * 10^{-5}$	$0.348 * 10^{-5}$	4.028
	100	0.10	25.00	$0.128 * 10^{-6}$	$0.216 * 10^{-6}$	3.985
0.250	10	1.00	0.125	$0.189 * 10^{-1}$	$0.267 * 10^{-1}$	
	20	0.50	0.500	$0.126 * 10^{-2}$	$0.153 * 10^{-2}$	4.125
	50	0.20	3.125	$0.312 * 10^{-4}$	$0.370 * 10^{-4}$	4.062
	100	0.10	12.500	$0.194 * 10^{-5}$	$0.230 * 10^{-5}$	4.008
0.125	20	0.5	0.250	$0.187 * 10^{-1}$	$0.188 * 10^{-1}$	
	50	0.20	1.563	$0.466 * 10^{-3}$	$0.431 * 10^{-3}$	4.120
	100	0.10	6.250	$0.312 * 10^{-4}$	$0.261 * 10^{-4}$	4.046
0.062	50	0.20	0.388	$0.868 * 10^{-2}$	$0.554 * 10^{-2}$	
	100	0.10	3.100	$0.484 * 10^{-3}$	$0.313 * 10^{-3}$	4.145
0.031	60	0.167	0.558	$0.598 * 10^{-1}$	$0.346 * 10^{-1}$	
	100	0.10	1.550	$0.868 * 10^{-2}$	$0.392 * 10^{-2}$	4.263

TABLE 5.6

Steady State Solution of Burgers Equation;
Comparison of U Profiles
($\nu = 0.500$)

		OCI–CN			
X	Exact U	$N = 10$	$N = 20$	$N = 100$	Second order CN $N = 200$
−5.00	0.993307	0.993307	0.993307	0.993307	0.993307
−4.00	0.982014	0.982042	0.982015	0.982014	0.982021
−3.00	0.952574	0.952845	0.952589	0.952574	0.952595
−2.00	0.880797	0.881716	0.880850	0.880797	0.880833
−1.00	0.731059	0.732380	0.731138	0.731059	0.731094
−0.40	0.598688			0.598688	0.598705
−0.20	0.549834			0.549834	0.549842
−0.00	0.500000	0.500000	0.500000	0.500000	0.500000

TABLE 5.7

Steady State Solution of Burgers Equation
Comparison of U Profiles
$\nu = 0.031$

		OCI–CN		
X	Exact U	$N = 60$	$N = 100$	Second order CN $N = 200$
−1.200	1.000000		1.000000	1.000000
−1.167	1.000000	0.999997		
−1.000	1.000000	0.999990	1.000000	1.000000
−0.833	0.999999	0.999963		
−0.800	0.999998		0.999995	1.000000
−0.667	0.999979	0.999894		
−0.600	0.999937		0.999903	1.000000
−0.500	0.999086	0.999651		
−0.400	0.998425		0.998091	0.999981
−0.333	0.995397	0.998843		
−0.200	0.961794		0.962779	0.994937
−0.167	0.936325	0.996115		
0.000	0.500000	0.500000	0.500000	0.500000
0.167	0.063675	0.003885		
0.200	0.038206		0.037221	0.005062
0.333	0.004603	0.001157		
0.400	0.001575		0.001909	0.000019
0.500	0.000314	0.000349		
0.600	0.000063		0.000097	0.000000
0.667	0.000021	0.000106		
0.800	0.000002		0.000005	0.000000
0.833	0.000001	0.000032		
1.000	0.000000	0.000010	0.000000	0.000000
1.167	0.000000	0.000003		
1.200	0.000000		0.000000	0.000000

U^*_j being the last iterate. This form of iteration has super-linear convergence properties [19].

Both methods assume an initial guess for $U_j^{n+1} = U_j^*$ which is used to solve the resultant tridiagonal system of equations. Iteration is employed until the difference between successive iterants is less than some preset tolerance. The steady state is assumed when differences in solution values at two time steps is less than some predetermined value.

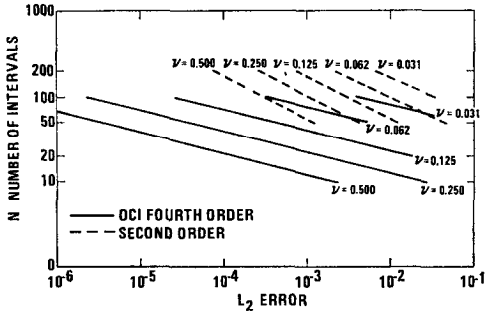

FIG. 2. Steady state Burgers equation. L_2 Error vs number of intervals.

In contrast to the above procedure, the OCI–Lees discretization does not require iteration and generally approached the steady state in about the same number of time steps as the OCI–CN method.

Figure 2 presents a graph of the computed L_2 error versus the number of intervals, for the fourth order and second order schemes. The storage savings possible with the OCI method are readily evident from the figure. Tables 5.6 and 5.7 compare solution values obtained from the fourth order and second order methods with the exact value, for two cases, $\nu = 0.5$ and $\nu = 0.031$.

Although the cell Reynolds number analysis for the OCI method given in Section 4 was derived for a linear spatial operator, this theory can be useful in predicting the behavior for nonlinear time dependent problems. For the Burgers equation it was found that physical solutions were obtained for a steady state only when $| R_c |_{\max} < 2.55$, where

$$| R_c |_{\max} = \frac{(u - \alpha)\, h}{\nu} = \frac{h}{2\nu}.$$

However, a careful inspection of the numerical results indicates that for $| R_c |_{\max} > 2.55$, in computing the transient solution, values are encountered which yield cell Reynolds numbers exceeding $(12)^{1/2}$, and physical steady state solutions cannot be obtained. These results suggest that when the homogeneous case maintains, one should monitor the evolution of the local cell Reynolds number and consider modifying the spatial mesh when necessary.

The results of the computations presented above suggest that the OCI method can be adapted to handle nonlinearities with very little additional effort and can resolve regions with sharp gradients.

5.4. *Two Dimensional Problems*

The OCI method was tested for a two dimensional parabolic equation

$$u_t = a(x, y, t)\, u_{xx} + b(x, y, t)\, u_x + c(x, y, t)\, u_{yy} + d(x, y, t)\, u_y,$$

whose coefficients were constructed in order to obtain the solution

$$u(x, y, t) = \exp\{(x + 1)(y + 1)(t + 1)\}.$$

Neither efficiency studies nor comparisons with other methods were made. The aim here was mainly to check the order of accuracy and the viability of the ADI formulation. Table 5.8 demonstrates that the splitting technique given in Section 3 yields fourth order accuracy. Ciment and Leventhal [3] have demonstrated that for hyperbolic equations this type of ADI scheme retains fourth order accuracy on other than rectangular domains, e.g., L shaped domains. Similar results are expected for parabolic equations.

TABLE 5.8

Two Dimensional Parabolic Equation: OCI–Crank–Nicolson
$u_t = a(x, y, t)u_{xx} + b(x, y, t)u_x + c(x, y, t)u_{yy} + d(x, y, t)u_y$
$u(x, y, t) = \exp\{(x + 1)(y + 1)(t + 1)\}$
Domain is square $\Omega = [\frac{1}{2} \leqslant x, y \leqslant 1]$, $\Delta x = \Delta y = h$

Time steps	h	Δt	L_2 Error	L_2 Rate	Max relative error	Max relative rate
5	0.1	0.1	3.235 − 03		1.544 − 04	
				4.414		3.905
20	0.05	0.025	1.517 − 04		1.031 − 05	
				4.955		3.977
80	0.025	0.00625	4.890 − 06		6.549 − 07	
10	0.1	0.1	3.903 − 02		3.909 − 04	
				4.364		3.933
40	0.05	0.025	1.896 − 03		2.559 − 05	
				4.909		3.982
160	0.025	0.00625	6.311 − 05		1.619 − 06	

APPENDIX A

The Swartz operator compact implicit formulas are derived here for uniform and nonuniform grids, with their associated truncation errors.

Given the spatial operator

$$L(u) = a(x)u_{xx} + b(x)u_x\,, \tag{A1}$$

a linear relationship between u and Lu at x_j is sought in the form

$$r^- u_- + r^0 u_0 + r^+ u_+ = q^- L(u)_- + q^0 L(u)_0 + q^+ L(u)_+ \tag{A2}$$

where as shorthand notation the subscripts $-$, 0, $+$ are used for $j - 1$, j, and $j + 1$, respectively, and the j dependence of the coefficients is not indicated, see (2.12).

The function values u_- and u_+ and the spatial operators $L(u)_-$ and $L(u)_+$ can be obtained through Taylor's series expansion about the point j.

$$u_+ = u_0 + h_+u_0^{(1)} + \frac{h_+^2}{2!}u_0^{(2)} + \frac{h_+^3}{3!}u_0^{(3)} + \frac{h_+^4}{4!}u_0^{(4)} + \frac{h_+^5}{5!}u_0^{(5)} + \frac{h_+^6}{6!}u_0^{(6)} \cdots, \quad \text{(A3a)}$$

$$u_- = u_0 - h_-u_0^{(1)} + \frac{h_-^2}{2!}u_0^{(2)} - \frac{h_-^3}{3!}u_0^{(3)} + \frac{h_-^4}{4!}u_0^{(4)} - \frac{h_-^5}{5!}u_0^{(5)} + \frac{h_-^6}{6!}u_0^{(6)} \cdots, \quad \text{(A3b)}$$

$$\begin{aligned} L(u)_- &= a_-u_-^{(2)} + b_-u_-^{(1)} \\ &= b_-u_0^{(1)} + (a_- - h_-b_-)\,u_0^{(2)} \\ &\quad - h_-\left(a_- - \frac{h_-}{2}b_-\right)u_0^{(3)} + \frac{h_-^2}{2!}\left(a_- - \frac{h_-}{3}b_-\right)u_0^{(4)} \\ &\quad - \frac{h_-^3}{3!}\left(a_- - \frac{h_-}{4}b_-\right)u_0^{(5)} + \frac{h^4}{4!}\left(a_- - \frac{h_-}{5}b_-\right)u_0^{(6)} \cdots, \end{aligned} \quad \text{(A3c)}$$

$$\begin{aligned} L(u)_+ &= a_+u_+^{(2)} + b_+u_+^{(1)} \\ &= b_+u_0^{(1)} + (a_+ + h_+b_+)\,u_0^{(2)} \\ &\quad + h_+\left(a_+ + \frac{h_+}{2}b_+\right)u_0^{(3)} + \frac{h_+^2}{2!}\left(a_+ + \frac{h_+}{3}b_+\right)u_0^{(4)} \\ &\quad + \frac{h_+^3}{3!}\left(a_+ + \frac{h_+}{4}b_+\right)u_0^{(5)} + \frac{h_+^4}{4!}\left(a_+ + \frac{h_+}{5}b_+\right)u_0^{(6)} \cdots, \end{aligned} \quad \text{(A3d)}$$

where superscripts in parenthesis indicate derivatives and $h_+ = x_{j+1} - x_j$ and $h_- = x_j - x_{j-1}$. Multiplying (A3a) — (A3d) by α, β, γ, δ, respectively, and collecting terms the following relation is obtained.

$$\alpha u_+ + \beta u_- + \gamma L(u)_- + \delta L(u)_+ = (\alpha + \beta)\,u_0 + \bar{B}u_0^{(1)} + \bar{A}u_0^{(2)} + \bar{C}u_0^{(3)} + \bar{D}u_0^{(4)} + \text{Truncation Error} \quad \text{(A4)}$$

or

$$\alpha u_+ - (\alpha + \beta)\,u_0 + \beta u_- = -\gamma L(u)_- - \delta L(u)_+ + \bar{A}u_0^{(2)} + \bar{B}u_0^{(1)} + \bar{C}u_0^{(3)} + \bar{D}u_0^{(4)} + \text{Truncation Error,} \quad \text{(A5)}$$

where, in order to obtain (A2) directly from (A5)

$$\begin{aligned} \bar{B} &\equiv \alpha h_+ - \beta h_- + \gamma b_- + \delta b_+ = b_0\,, \\ \bar{A} &\equiv \frac{\alpha h_+^2}{2} + \frac{\beta h_-^2}{2} + \gamma(a_- - h_-b_-) + \delta(a_+ + h_+b_+) = a_0\,, \\ \bar{C} &\equiv \frac{\alpha h_+^3}{3!} - \frac{\beta h_-^3}{3!} - \gamma h_-\left(a_- - \frac{h_-}{2}b_-\right) + \delta h_+\left(a_+ + \frac{h_+}{2}b_+\right) = 0, \\ \bar{D} &\equiv \frac{\alpha h_+^4}{4!} + \frac{\beta h_-^4}{4!} + \frac{\gamma h_-^2}{2}\left(a_- - \frac{h_-}{3}b_-\right) + \frac{\delta h_+^2}{2}\left(a_+ + \frac{h_+}{3}b_+\right) = 0. \end{aligned} \quad \text{(A6)}$$

Define

$$\hat{\alpha} = \alpha\mathscr{D}, \qquad \hat{\beta} = \beta\mathscr{D}, \qquad \hat{\gamma} = \gamma\mathscr{D}, \qquad \hat{\delta} = \delta\mathscr{D}, \tag{A7}$$

where $\mathscr{D}$ is the determinant of (A6),

$$\begin{aligned}\mathscr{D} = \{&12a_-a_+(h_+^3 + 4h_+^2h_- + 4h_+h_-^2 + h_-^3) - 2a_+b_-h_-(3h_+^3 + 7h_+^2h_- \\ &+ 5h_+h_-^2 + h_-^3) + 2a_-b_+h_+(h_+^3 + 5h_+^2h_- + 7h_+h_-^2 + 3h_-^3) \\ &- h_+h_-b_+b_-(h_+ + h_-)^3\}.\end{aligned}$$

Then the variables $\hat{\alpha}$, $\hat{\beta}$, $\hat{\gamma}$ and $\hat{\delta}$ are given by

$$\begin{aligned}\hat{\gamma} = \{&12h_+a_+a_0(h_+^2 - h_+h_- - h_-^2) \\ &+ 2a_+b_0h_+^2h_-(3h_+ + 2h_-) + 2a_0b_+h_+^2(h_+^2 - h_+h_- - 2h_-^2) \\ &+ h_+^3h_-b_0b_+(h_+ + h_-)\},\end{aligned} \tag{A9}$$

$$\begin{aligned}\hat{\delta} = \{&12a_0a_-h_-(h_-^2 - h_+h_- - h_+^2) + 2a_0b_-h_-^2(2h_+^2 + h_+h_- - h_-^2) \\ &- 2a_-b_0h_+h_-^2(2h_+ + 3h_-) + b_0b_-h_+h_-^3(h_+ + h_-)\},\end{aligned} \tag{A10}$$

$$\hat{\beta}h_-(h_+ + h_-) = \mathscr{D}[2a_0 - h_+b_0] - \hat{\delta}[2a_+ + h_+b_+] - \hat{\gamma}[2a_- - b_-(2h_- + h_+)], \tag{A11}$$

$$\hat{\alpha}h_+(h_+ + h_-) = \mathscr{D}[2a_0 + h_-b_0] - \hat{\delta}[2a_+ + b_+(2h_+ + h_-)] - \hat{\gamma}[2a_- - h_-b_-]. \tag{A12}$$

Multiplying thru by $\mathscr{D}$, the q's and r's become

$$\begin{aligned} q^- &= \hat{\gamma}, \qquad q^+ = \hat{\delta}, \qquad q^0 = -\mathscr{D}, \\ r^- &= -\hat{\beta}, \qquad r^0 = (\hat{\alpha} + \hat{\beta}), \qquad r^+ = -\hat{\alpha}, \end{aligned} \tag{A13}$$

such that the operators Q and R are given in the form

$$\begin{aligned} Q &= \hat{\delta}S_+ - \mathscr{D}I + \hat{\gamma}S_-, \\ R &= -\hat{\alpha}S_+ + (\hat{\alpha} + \hat{\beta})\, I - \hat{\beta}S_-, \end{aligned} \tag{A14}$$

where S is the shift operator.

Using the relations (A7) — (A12) the truncation error given by

$$\begin{aligned} E_T = &\frac{1}{\mathscr{D}}\left\{\frac{\hat{\alpha}h_+^5}{5!} - \frac{\hat{\beta}h_-^5}{5!} + \frac{\hat{\delta}h_+^3}{3!}\left(a_+ + \frac{h_+}{4}b_+\right) - \frac{\hat{\gamma}h_-^3}{3!}\left(a_- - \frac{h_-}{4}b_-\right)\right\} u_0^{(5)} \\ &+ \frac{1}{\mathscr{D}}\left\{\frac{\hat{\alpha}h_+^6}{6!} + \frac{\hat{\beta}h_-^6}{6!} + \frac{\hat{\delta}h_+^4}{4!}\left(a_+ + \frac{h_+}{5}b_+\right) + \frac{\hat{\gamma}h_-^4}{4!}\left(a_- - \frac{h_-}{5}b_-\right)\right\} u_0^{(6)} \end{aligned} \tag{A15}$$

is seen to be third order accurate for small h.

For a uniform mesh, $h_+ = h_- = h$, the truncation error reduces to

$$E_T = \frac{h^4}{1800 a_+ a_-}\{-[9a_0 a_- a_+]\, u_0^{(6)} + [4a_0 a_- b_+ - 35 a_- a_+ b_0 + 4a_0 a_+ b_-]\, u_0^{(5)}\}, \quad \text{(A16)}$$

which is fourth order accurate.

Note that in Eq. (2.12), common factors in the q's and r's have been canceled (involving constant h), so that (2.12) differs from (A13) by a multiplicative constant, $2h^3$.

In a future paper it will be shown how a family of OCI schemes can be obtained by expanding α, β, γ and δ in an asymptotic series in powers of h and retaining lower order derivative terms in the truncation error while still achieving fourth order accuracy.

Appendix B

The coefficients of (5.6), namely,

$$F_0 \equiv (u_x)_0 = H_0 U_0 + H_1 U_1 + H_2 U_2 + G_0 f_0 + G_1 f_1 + G_2 f_2\,, \quad \text{(B1)}$$

are derived.

Consider the compact implicit formulas

$$F_0 + 4F_1 + F_2 = \frac{3}{h}(U_2 - U_0), \quad \text{(B2a)}$$

$$S_0 + 10S_1 + S_2 = \frac{12}{h^2}(U_0 - 2U_1 + U_2), \quad \text{(B2b)}$$

$$S_0 + 4S_1 + S_2 = \frac{3}{h}(F_2 - F_0), \quad \text{(B2c)}$$

and the differential equation at points $j = 0, 1, 2$ expressed as

$$a_j S_j + b_j F_j = f_j\,, \qquad j = 0, 1, 2 \qquad (a_j b_j \neq 0). \quad \text{(B3)}$$

Equations (B2)-(B3) form a system of six equations in nine unknowns, and thus F_0 can be determined as a function of u_0, u_1, u_2, f_0, f_1, and f_2.

The coefficients in (B1) are listed below.

$$H_0 = \left\{-\left(\frac{6}{h}\right)\left(\frac{b_1}{a_1}\right)\left(3\,\frac{b_2}{a_2} + \frac{15}{h}\right) + \frac{48}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)\right\}\Big/C,$$

$$H_1 = -\frac{96}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)\Big/C,$$

$$H_2 = \left\{\left(\frac{6}{h}\right)\left(\frac{b_1}{a_1}\right)\left(3\,\frac{b_2}{a_2} + \frac{15}{h}\right) + \frac{48}{h^2}\left(\frac{b_2}{a_2} - \frac{b_1}{a_1} + \frac{3}{h}\right)\right\}\Big/C,$$

$$G_0 = \frac{-2}{a_0}\left(\frac{3b_1}{a_1} + \frac{6}{h}\right)/C,$$

$$G_1 = \frac{-8}{a_1}\left(3\,\frac{b_2}{a_2} + \frac{15}{h}\right)/C,$$

$$G_2 = \frac{-2}{a_2}\left(\frac{3b_1}{a_1} + \frac{6}{h}\right)/C,$$

$$C = 2\left\{3\,\frac{b_1}{a_1}\left(\frac{b_2}{a_2} - \frac{b_0}{a_0}\right) + \frac{6}{h}\left(5\,\frac{b_1}{a_1} - \frac{b_2}{a_2} - \frac{b_0}{a_0}\right)\right\}. \tag{B4}$$

The truncation error is given by

$$E_{\text{TRUNC}} = \left\{-\left(4\,\frac{b_3}{a_3} - 10\,\frac{b_2}{a_2}\right)E_S + \left[4\,\frac{b_3}{a_3}\left(\frac{b_3}{a_3} - \frac{b_2}{a_2} + \frac{3}{h}\right)\right.\right.$$
$$\left.\left. - \left(\frac{b_3}{a_3} + \frac{3}{h}\right)\left(4\,\frac{b_3}{a_3} - 10\,\frac{b_2}{a_2}\right)\right]E_F + 4\left(\frac{b_3}{a_3} - \frac{b_2}{a_2} + \frac{3}{h}\right)E_T\right\},$$

where

$$E_S = 5\,\frac{h^4}{30}\,u^{(6)}, \qquad E_F = \frac{h^4}{30}\,u^{(5)}, \qquad E_T = \frac{h^4}{20}\,u^{(6)}.$$

Equation (B5) can be specialized for constant coefficients

$$E_{\text{TRUNC}} = \frac{h^5}{(b/a)\,180}\left[\left(5\,\frac{b}{a} + \frac{3}{h}\right)u^{(6)} + \left(\frac{b}{a} + \frac{5}{h}\right)u^{(5)}\right]. \tag{B6}$$

In the case of time dependent problems modifications to (B1) are necessary. Consider the one dimensional parabolic equation

$$u_t = L(u) \equiv aS + bF = f. \tag{B7}$$

The first derivative at an end point at time level $(n + 1)$ in the form of

$$F_0^{n+1} = H_0^{n+1}U_0^{n+1} + H_1^{n+1}U_1^{n+1} + H_2^{n+1}U_2^{n+1}$$
$$+ G_0^{n+1}f_0^{n+1} + G_1^{n+1}f_1^{n+1} + G_2^{n+1}f_2^{n+1} \tag{B8}$$

is sought.

Again, as before, use the compact implicit formulas (B2), but with u, F, and S evaluated at time level $(n + 1)$. The differential equation (B7), however, is discretized temporally by a Crank–Nicolson scheme to yield

$$\frac{U_j^{n+1} - U_j{}^n}{\Delta t} = [a_j^{n+1}S_j^{n+1} + b_j^{n+1}F_j^{n+1}]/2 + [a_j{}^nS_j{}^n + b_j{}^nF_j{}^n]/2 \equiv \frac{f_j^{n+1} + f_j^n}{2}. \tag{B9}$$

Thus f appearing in (B8) is the spatial operator evaluated at $(n+1)$ and is given by

$$f_j^{n+1} \equiv a_j^{n+1} S_j^{n+1} + b_j^{n+1} + \frac{2}{\Delta t} U_j^{n+1} - \left(f_j^n + \frac{2}{\Delta t} U_j^n\right). \tag{B10}$$

Hence, substituting (B10) into (B8), the desired relationship is obtained:

$$\begin{aligned} F_0^{n+1} = &\left[H_0^{n+1} + \frac{2}{\Delta t} G_0^{n+1}\right] U_0^{n+1} + \left[H_1^{n+1} + \frac{2}{\Delta t} G_1^{n+1}\right] U_1^{n+1} \\ &+ \left[H_2^{n+1} + \frac{2}{\Delta t} G_1^{n+1}\right] U_2^{n+1} + G_0^{n+1}\left[f_0^n + \frac{2}{\Delta t} U_0^n\right] \\ &+ G_1^{n+1}\left[f_1^n + \frac{2}{\Delta t} U_1^n\right] + G_2^{n+1}\left[f_2^n + \frac{2}{\Delta t} U_2^n\right]. \end{aligned} \tag{B11}$$

The local spatial truncation error remains unchanged.

Appendix C

Here we prove the invertibility of $S \equiv (Q - \lambda R)$, $\lambda \geqslant 0$ for the case of constant coefficients. Let $s^{+,0,-} = q^{+,0,-} - \lambda r^{+,0,-}$. Let $d_J = \det S$ ($J \times J$ matrix). Expanding by minors, since S is tridiagonal $d_j = s^0 d_{j-1} - s^+ s^- d_{j-2}$, $j = 2, 3, \cdots J$. Where $d_1 = s^0$, $d_0 \equiv 1$. Observe from (2.12) that for $R_c \leqslant 12^{1/2}$, $s^0 > 0$. Thus if $s^+ s^- \leqslant 0$ the d_j are a strictly (nondecreasing) positive sequence. The only case that needs further consideration is when $s^+ s^- > 0$. In this case sign $s^+ =$ sign s^- which implies that (by direct substitution from (2.12))

$$\begin{aligned} |s^+| + |s^-| = |s^+ + s^-| &= |(12 - R_c^2) - R_c^2 - 144\lambda| \\ &\leqslant 12 + 144\lambda \leqslant s^0. \end{aligned}$$

In this case S is an irreducibly diagonally dominant matrix and hence invertible [18]. The authors thank Charles R. Johnson for the suggestion to expand out the above determinant and to consider the sign pattern.

References

1. Y. Adam, "A Hermitian Finite Difference Method for the Solution of Parabolic Equations," Comp. & Maths. with Appls., Vol. 1, pp. 393–406, Pergamon, New York, 1975.
2. W. R. Briley and H. McDonald, "Solution of the Multidimensional Compressible Navier–Stokes Equations by a Generalized Implicit Method," United Technologies R75-911363-15, Jan. 1976.
3. M. Ciment and S. H. Leventhal, *Math. Comp.* **29** (1975), 985.
4. M. Ciment and S. H. Leventhal, *Math. Comp.* **32** (1978), 143.
5. M. Ciment, S. Leventhal, and B. Weinberg, "The Operator Compact Implicit Method for Parabolic Equations," NSWC/WOL TR 77-29, April 1977.

6. L. COLLATZ, "The Numerical Treatment of Differential Equations," Springer-Verlag, Berlin, 1960.
7. J. DOUGLAS, *J. Math. Phys.* **35** (1956), 145.
8. R. S. HIRSH, *J. Computational Phys.* **19** (1975), 90.
9. R. S. HIRSH AND D. H. RUDY, *J. Computational Phys.* **25** (1974), 304.
10. E. ISAACSON AND H. B. KELLER, "Analysis of Numerical Methods," Wiley, New York, 1966.
11. E. KRAUSE, E. H. HIRSCHEL, AND W. KORDULLA, *Computers and Fluids* **4** (1976), 77.
12. D. C. L. LAM AND R. B. SIMPSON, *J. Computational Phys.* **22** (1976), 486.
13. M. LEES, *Math. Comp.* **20** (1966), 516.
14. H. MACDONALD AND W. R. BRILEY, *J. Computational Phys.* **19** (1975), 150.
15. A. R. MITCHELL, "Computational Methods in Partial Differential Equations," Wiley, London, 1969.
16. A. R. MITCHELL AND G. FAIRWEATHER, *Numer. Math.* **6** (1964), 285.
17. S. A. ORSZAG AND M. ISRAELI, Numerical simulation of viscous incompressible flows, *in* "Annual Review of Fluid Mechanics," Vol. 6, 1974.
18. J. M. ORTEGA, "Numerical Analysis," Academic Press, New York, 1972.
19. J. M. ORTEGA AND W. RHEINBOLDT, "Iterative Solution of Nonlinear Equations in Several Variables," Academic Press, New York, 1970.
20. N. PETERS, Boundary layer calculations by a Hermitian finite difference method, *in* "Proceedings of the Fourth International Conference on Numerical Methods in Fluid Mechanics, Boulder, Colorado," Springer–Verlag, Berlin, 1974.
21. A. RALSTON, "A First Course in Numerical Analysis," McGraw–Hill, New York, 1965.
22. R. D. RICHTMYER AND K. W. MORTON, "Difference Methods for Initial-Value Problems," 2nd ed., Interscience, New York, 1967.
23. P. J. ROACHE, "Computational Fluid Dynamics," Hermosa, Albuquerque, N.M., 1972.
24. S. G. RUBIN AND P. K. KHOSLA, *J. Computational Phys.* **24** (1977), 217.
25. S. G. RUBIN AND R. A. GRAVES, JR., "A Cubic Spline Approximation for Problems in Fluid Dynamics," Old Dominion University, TR 74-T1, Norfolk, Va., June 1974.
26. B. K. SWARTZ, The construction of finite difference analogs of some finite element schemes, *in* "Mathematical Aspects of Finite Elements in Partial Differential Equations" (C. DeBoor, Ed.), pp. 279–312, Academic Press, New York, 1974.