INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 19, 1039–1047 (1994)

FULLY DISCRETE ARBITRARY-ORDER SCHEMES FOR A MODEL PARABOLIC EQUATION

J. SHI

Department of Aerospace Science, College of Aeronautics, Cranfield Institute of Technology, Cranfield, Bedfordshire MK43 0AL, U.K.

SUMMARY

A fully discrete methodology is investigated from which two-level, explicit, arbitrary-order, conservative numerical schemes for a model parabolic equation can be derived. To illustrate this, fully discrete three-, five-, seven- and nine-point conservative numerical schemes are presented, revealing that a higher-order scheme has a better stability condition. A method from which high-order numerical schemes for a scalar advection–diffusion equation can be developed is discussed. This method is based on high-order schemes of both the advection and diffusion equations.

KEY WORDS Fully discrete High-order Conservative

1. INTRODUCTION

An important research subject in computational fluid dynamics (CFD) concerns the development of high-order numerical schemes for direct simulation of viscous flows. Since second-order schemes in practice are inadequate for many applications, great efforts have been made towards developing high-order-accurate schemes in the past. There are many application areas for which such research is of vital importance. One example of considerable interest is the solution of problems which require long-time evolution. For this kind of problem low-order methods will produce unacceptable dispersive and diffusive errors in a very short time. Another example concerns direct numerical simulation of turbulent flows. Low-order methods contain large numerical diffusion and dispersion and are thus totally inaccurate for simulating turbulent structures, since large amounts of turbulent fluctuations associated with different scales of eddies are artificially dissipated. In large computational problems low-order methods would require vast amounts of computer memory (possibly not available with current computers) in order to attain a satisfactory degree of accuracy. A high-order method would attain the same accuracy with coarser meshes requiring less sophisticated hardware and making it possible to actually run the problem.

Essentially there are two different techniques which can be used to construct high-order numerical schemes: semidiscrete and fully discrete methods. In the semidiscrete method (for definition see Reference 1) one divides the discretization process into two separate stages. In the first stage one discretizes in space only, leaving the problem continuous in time; in the second stage one has sets of ordinary differential equations (ODEs) in time which can be discretized appropriately. Often this technique is called the *method of lines*.

Today most high-order numerical schemes rely on the semidiscrete approach. One typical

CCC 0271–2091/94/231039–19

Received February 1993
Revised June 1994

example is the conventional implicit multistep scheme[2] which is widely applied in CFD. However, because of the severe restrictions on high time levels, the highest-order method in practice is second-order, i.e. the two-step method.

One approach to dealing with viscous flows is to use the convection–diffusion operator-splitting method,[3] which splits the Navier–Stokes equations into two parts, an inviscid part (Euler equations) and a viscous part (Stokes equations), and then solves the two parts sequentially by applying an optimum numerical scheme to each part. This approach avoids the homogeneous treatment of both advection and diffusion in the Navier–Stokes equations and reduces the task to developing high-order schemes for each individual part, which is much easier to deal with technically than the full advection–diffusion equations.

The model equation for studying the Stokes equations is a scalar parabolic equation. In this paper a fully discrete technique which has the important property of combining time and space discretization of a model parabolic equation in a single stage is investigated. By applying this technique, two-level, explicit, arbitrary-order numerical schemes for a scalar parabolic equation can be constructed. The resulting schemes are expressed in a conservative form, i.e. in terms of a diffusion numerical flux function, which can be readily combined with an advection numerical flux to form a high-order advection–diffusion scheme if necessary.

The paper is organized as follows. Section 2 establishes a formula from which two-level, explicit, fully discrete, arbitrary-order numerical schemes can be derived. Section 3 applies the technique to construct some high-order, fully discrete, conservative numerical schemes and gives the stability condition for the schemes. Section 4 presents a way to construct high-order schemes for the advection–diffusion equation. Section 5 contains conclusions.

2. FULLY DISCRETIZING A MODEL PARABOLIC EQUATION

We consider the initial value problem (IVP) for a one-dimensional linear scalar model parabolic partial differential equation (PDE), namely the scalar diffusion equation in a conservative form

$$\begin{aligned} &u_t - (\nu u_x)_x = 0, \quad -\infty < x < \infty, \quad t > 0, \\ &u(x, 0) = u_0(x). \end{aligned} \tag{1}$$

Here $u(x, t)$ is the unknown function and ν is a diffusion coefficient.

We discretize the computational half-plane by choosing a uniform mesh with a mesh width $h = \Delta x$ and a time step $k = \Delta t$ and define the computational grid $x_j = jh$, $t_n = nk$. We use U_j^n to denote the computed approximation to the exact solution $u(x_j, t_n)$ of equation (1).

Theorem 1

The fully discrete formula from which a two-level, fully discrete, explicit, mth-order-accurate, finite difference method can be derived for the model parabolic equation $u_t - \nu u_{xx} = 0$ is defined as

$$U_j^{n+1} = \sum_{\alpha=0}^{p} B_{k_\alpha} U_{j+k_\alpha}^n, \tag{2}$$

where α is the grid point number, $p = 2m + 1$ is the number of grid points used, m is the order of accuracy in time and B_{k_α} are constant coefficients determined by

$$B_{k_\alpha=0} = 1 - \sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha}, \quad \begin{bmatrix} B_{k_1} \\ B_{k_2} \\ \vdots \\ \vdots \\ \vdots \\ B_{k_{2m}} \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & \cdots & k_{2m} \\ k_1^2 & k_2^2 & \cdots & k_{2m}^2 \\ \vdots & \vdots & \vdots & \vdots \\ k_1^n & k_2^n & \cdots & k_{2m}^n \\ \vdots & \vdots & \vdots & \vdots \\ k_1^{2m} & k_2^{2m} & \cdots & k_{2m}^{2m} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 2d \\ \vdots \\ y \\ \vdots \\ [(2m)!/m!]d^m \end{bmatrix} \quad (k_\alpha \neq 0), \tag{3}$$

where

$$y = \begin{cases} 0 & \text{if } n \text{ is an odd number,} \\ [n!/(n/2)!]d^{n/2} & \text{if } n \text{ is an even number,} \end{cases}$$

with $d = vk/h^2$ a diffusion number.

Proof. In order to prove the theorem, we first analyse the local truncation error of equation (1) by Taylor series expansion of both sides of the equation. This can be written as

$$E(x, t) = u(x, t) + \sum_{n=1}^{m} \frac{(\Delta t)^n}{n!} u_{t^n} + O[(\Delta t)^{m+1}] - \sum_{\alpha=1}^{p} B_{k_\alpha}\left(u(x, t) + \sum_{n=1}^{m} \frac{(k_\alpha \Delta x)^n}{n!} u_{x^n}\right) + O[(\Delta x)^{m+1}], \tag{4}$$

where m is the order of accuracy of the scheme ($1 \leqslant m < \infty$), $u_{t^n} = \partial^n u/\partial t^n$ and $u_{x^n} = \partial^n u/\partial x^n$. From equation (1) it is easy to obtain

$$u_{t^n} = v^n u_{x^{2n}}. \tag{5}$$

Substitution of equation (5) into equation (4) gives

$$E(x, t) = \left(1 - \sum_{\alpha=1}^{p} B_{k_\alpha}\right) u(x, t) + \sum_{n=1}^{m} \left(\frac{(\Delta t)^n}{n!} v^n u_{x^{2n}} - \sum_{\alpha=1}^{p} B_{k_\alpha} \frac{(k_\alpha \Delta x)^n}{n!} u_{x^n}\right) + O[(\Delta t)^{m+1}, (\Delta x)^{2m+1}]. \tag{6}$$

Note here that the order of the truncation error in equation (6) is $m + 1$ in time and $2m + 1$ in space, because $\Delta t \sim \Delta x^2$. Obviously the relationship between m and p is

$$p = 2m + 1, \tag{7}$$

In order to achieve mth-order accuracy in time, it is sufficient to require that

$$1 - \sum_{\alpha=1}^{p} B_{k_\alpha} = 0, \tag{8}$$

$$\frac{(\Delta t)^n}{n!} v^n u_{x^{2n}} - \sum_{\alpha=1}^{p} B_{k_\alpha} \frac{(k_\alpha \Delta x)^n}{n!} u_{x^n} = 0 \quad (n = 1, 2, 3, \ldots, m). \tag{9}$$

Equation (9) can be rewritten as

$$\frac{[d(\Delta x)^2]^n}{n!} u_{x^{2n}} - \sum_{\alpha=1}^{p} \frac{(\Delta x)^n}{n!} B_{k_\alpha} k_\alpha^n u_{x^n} = 0. \tag{10}$$

Incorporating the left-hand side of (10) in terms of n and reorganizing it, equations (8) and (10) become

$$
\begin{aligned}
&B_0 = 1 - \sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha}, \\
&\sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha} k_\alpha^n = 0 \quad (n = 1, 3, \ldots, 2m-1), \\
&\sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha} k_\alpha^n = \frac{n!}{(n/2)!} d^{n/2} \quad (n = 2, 4, \ldots, 2m).
\end{aligned} \tag{11}
$$

Equation (11) can be transformed into the alternative forms

$$
\begin{aligned}
&B_{k_\alpha=0} = 1 - \sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha}, \\
&k_1 B_{k_1} + k_2 B_{k_2} + \cdots + k_{2m} B_{k_{2m}} = 0, \\
&k_1^2 B_{k_1} + k_2^2 B_{k_2} + \cdots + k_{2m}^2 B_{k_{2m}} = 2d, \\
&\quad \vdots \\
&k_1^n B_{k_1} + k_2^n B_{k_2} + \cdots + k_{2m}^n B_{k_{2m}} = \begin{cases} 0 & \text{if } n \text{ is an odd number,} \\ [n!/(n/2)!] d^{n/2} & \text{if } n \text{ is an even number,} \end{cases} \\
&\quad \vdots \\
&k_1^{2m} B_{k_1} + k_2^{2m} B_{k_2} + \cdots + k_{2m}^{2m} B_{k_{2m}} = \frac{(2m)!}{m!} d^m \quad (k_\alpha \neq 0)
\end{aligned} \tag{12}
$$

or

$$
B_{k_\alpha=0} = 1 - \sum_{\alpha=1, k_\alpha \neq 0}^{2m} B_{k_\alpha}, \quad
\begin{bmatrix} B_{k_1} \\ B_{k_2} \\ \vdots \\ \vdots \\ \vdots \\ B_{k_{2m}} \end{bmatrix}
=
\begin{bmatrix} k_1 & k_2 & \cdots & k_{2m} \\ k_1^2 & k_2^2 & \cdots & k_{2m}^2 \\ \vdots & \vdots & \vdots & \vdots \\ k_1^n & k_2^n & \cdots & k_{2m}^n \\ \vdots & \vdots & \vdots & \vdots \\ k_1^{2m} & k_2^{2m} & \cdots & k_{2m}^{2m} \end{bmatrix}^{-1}
\begin{bmatrix} 0 \\ 2d \\ \vdots \\ y \\ \vdots \\ [(2m)!/m!] d^m \end{bmatrix}
\quad (k_\alpha \neq 0), \tag{13}
$$

where

$$
y = \begin{cases} 0 & \text{if } n \text{ is an odd number,} \\ [n!/(n/2)!] d^{n/2} & \text{if } n \text{ is an even number,} \end{cases}
$$

which is formula (3) and establishes the theorem.

3. CONSERVATIVE HIGH-ORDER NUMERICAL SCHEMES

In this section we use some examples to demonstrate how to apply the method presented previously to derive high-order, conservative numerical schemes

$$
U_j^{n+1} = U_j^n - \frac{k}{h} [T(U^n; j) - T(U^n; j-1)], \tag{14}
$$

where $T(U^n; j)$ is a diffusion numerical flux which satisfies the consistency condition

$$T(\bar{u}, \bar{u}, \ldots, \bar{u}) = f(\bar{u}) = 0, \tag{15}$$

where $\bar{u}$ is constant.

3.1. *Fully discrete three-point schemes*

From equation (7) we know that the three-point schemes are first-order in time and second-order in space, i.e. order (1, 2).

3.1.1. Upwind-biased scheme. Let us denote the three-point centred scheme as $U_j^{n+1} = f(U_{j-2}^n, U_{j-1}^n, U_j^n)$. Here $k_1 = -2$, $k_2 = -1$ and $k_3 = 0$ in equation (12), which gives

$$-2B_{-2} - B_{-1} = 0, \qquad 4B_{-2} + B_{-1} = 2d, \qquad B_0 = 1 - B_{-2} - B_{-1},$$

i.e.

$$B_0 = 1 + d, \qquad B_{-1} = -2d, \qquad B_{-2} = d. \tag{16}$$

Therefore the numerical scheme is

$$U_j^{n+1} = (1 + d)U_j^n - 2dU_{j-1}^n + dU_{j-2}^n, \tag{17}$$

Applying the stability analysis method introduced in Reference 4, the amplification factor λ of the scheme is

$$\lambda = 1 + 4d. \tag{18}$$

For stability one requires $|\lambda| \leqslant 1$, which is satisfied if

$$-\tfrac{1}{2} \leqslant d \leqslant 0. \tag{19}$$

However, it is physically meaningless for the diffusion number d to be negative. Obviously the one-side upwind scheme (17) is physically not right. Our numerical analysis of other upwind-biased schemes either gave the same conclusion or showed severe stability restraints. Therefore from now on we will not discuss the upwind-biased schemes.

3.1.2. Centred scheme. We denote the scheme by $U_j^{n+1} = f(U_{j-1}^n, U_j^n, U_{j+1}^n)$. Here $k_1 = -1$, $k_2 = 0$ and $k_3 = 1$ in equation (12), which gives

$$-B_{-1} + B_1 = 0, \qquad B_{-1} + B_1 = 2d, \qquad B_0 = 1 - B_{-1} - B_1$$

or

$$B_0 = 1 - 2d, \qquad B_{-1} = d, \qquad B_1 = d. \tag{20}$$

Note here that $B_{-1} = B_1$. Actually this is a general law of parabolic numerical schemes which states that the mirror points of numerical schemes, such as points $j - 1$ and $j + 1$ in this scheme, have identical coefficients.

Therefore the centred numerical scheme is

$$U_j^{n+1} = (1 - 2d)U_j^n + d(U_{j-1}^n + U_{j+1}^n). \tag{21}$$

Applying the method introduced in Reference 5, the numerical flux of the scheme is

$$T(U^n; j) = F_j^d - F_{j+1}^d, \tag{22}$$

where $F_j^d = \beta U_j^n$ is a local diffusion flux and β, which has the dimension of velocity, is called the 'diffusive velocity' and is defined by

$$\beta = \nu/h. \tag{23}$$

The amplification factor of the scheme is

$$\lambda = 1 - 4d. \tag{24}$$

For stability one requires $|\lambda| \leqslant 1$, which is satisfied if

$$0 \leqslant d \leqslant \tfrac{1}{2}. \tag{25}$$

3.2. *Fully discrete five-point centred scheme*

The five-point schemes are second-order in time and fourth-order in space, i.e. order (2, 4).

Let us consider the scheme which is denoted as $U_j^{n+1} = f(U_{j-2}^n, U_{j-1}^n, U_j^n, U_{j+1}^n, U_{j+2}^n)$. Here $k_1 = -2$, $k_2 = -1$, $k_3 = 0$, $k_4 = -1$ and $k_5 = -2$ in equation (12), which gives

$$B_0 = 1 - B_{-2} - B_{-1} - B_1 - B_2, \qquad -2B_{-2} + B_{-1} + B_1 + 2B_2 = 0,$$
$$4B_{-2} + B_{-1} + B_1 + 4B_2 = 2d,$$
$$-8B_{-2} + B_{-1} + B_1 + 8B_2 = 0, \qquad 16B_{-2} + B_{-1} + B_1 + 16B_2 = 12d^2$$

or

$$B_0 = 1 + 3d^2 - \tfrac{5}{2}d, \qquad B_{-1} = B_1 = \tfrac{4}{3}d - 2d^2, \qquad B_{-2} = B_2 = \tfrac{1}{2}d^2 - \tfrac{1}{12}d. \tag{26}$$

Therefore the scheme is

$$U_j^{n+1} = (1 + 3d^2 - \tfrac{5}{2}d)U_j^n + (\tfrac{4}{3}d - 2d^2)(U_{j-1}^n + U_{j+1}^n) + (\tfrac{1}{2}d^2 - \tfrac{1}{12}d)(U_{j-2}^n + U_{j+2}^n). \tag{27}$$

The numerical flux of the scheme is

$$T(U^n; j) = (\tfrac{1}{2}d - \tfrac{1}{12})F_{j-1}^d + (\tfrac{5}{4} - \tfrac{3}{2}d)F_j^d + (\tfrac{3}{2}d - \tfrac{5}{4})F_{j+1}^d + (\tfrac{1}{12} - \tfrac{1}{2}d)F_{j+2}^d. \tag{28}$$

The amplification factor of the scheme is

$$\lambda = 1 - 2(\tfrac{8}{3}d - 4d^2). \tag{29}$$

Therefore the stability condition for $|\lambda| \leqslant 1$ is

$$0 \leqslant d \leqslant \tfrac{2}{3}. \tag{30}$$

Note that compared with the three-point centred scheme which has a stability condition $0 \leqslant d \leqslant \frac{1}{2}$, the five-point centred scheme has a better stability condition.

3.3. *Fully discrete seven-point centred scheme*

The accuracy with seven-point schemes is third-order in time and sixth-order in space i.e. order (3, 6).

By repeating the same procedure as before, the seven-point centred scheme is

$$\begin{aligned} U_j^{n+1} &= (1 - \tfrac{10}{3}d^3 + \tfrac{14}{3}d^2 - \tfrac{49}{18}d)U_j^n + (\tfrac{15}{6}d^3 - \tfrac{13}{4}d^2 + \tfrac{3}{2}d)(U_{j-1}^n + U_{j+1}^n) \\ &\quad + (d^2 - d^3 - \tfrac{3}{20}d)(U_{j-2}^n + U_{j+2}^n) + (\tfrac{1}{6}d^3 - \tfrac{1}{12}d^2 + \tfrac{1}{90}d)(U_{j-3}^n + U_{j+3}^n). \end{aligned} \tag{31}$$

The numerical flux of the scheme is

$$\begin{aligned} T(U^n; j) &= (\tfrac{1}{6}d^2 - \tfrac{1}{12}d + \tfrac{1}{90})F_{j-2}^d + (\tfrac{11}{12}d - \tfrac{5}{6}d^2 - \tfrac{5}{36})F_{j-1}^d + (\tfrac{5}{3}d^2 - \tfrac{7}{3}d + \tfrac{49}{36})F_j^d \\ &\quad + (\tfrac{7}{3}d - \tfrac{5}{3}d^2 - \tfrac{49}{36})F_{j+1}^d + (\tfrac{5}{6}d^2 - \tfrac{11}{12}d + \tfrac{5}{36})F_{j+2}^d + (\tfrac{1}{12}d - \tfrac{1}{6}d^2 - \tfrac{1}{90})F_{j+3}^d. \end{aligned} \tag{32}$$

The amplification factor of this scheme is

$$\lambda = 1 - \tfrac{32}{3}d^3 + \tfrac{40}{3}d^2 - \tfrac{272}{45}d. \tag{33}$$

For stability one requires $|\lambda| \leqslant 1$, which is satisfied if

$$0 \leqslant d \leqslant 0{\cdot}85. \tag{34}$$

Note that the stability condition of the scheme is again improved compared with the five-point centred scheme.

3.4. *Fully discrete nine-point central scheme*

The accuracy of this scheme is fourth-order in time and eighth-order in space i.e. order (4, 8)

$$\begin{aligned} U_j^{n+1} &= (1 - 2{\cdot}847222d + 5{\cdot}6875008d^2 - 6{\cdot}24999984d^3 + 2{\cdot}9166648d^4)U_j^n \\ &\quad + (1{\cdot}6d - 4{\cdot}066666656d^2 + 4{\cdot}83333324d^3 - 2{\cdot}333331184d^4)(U_{j-1}^n + U_{j+1}^n) \\ &\quad + (1{\cdot}408333332d^2 - 0{\cdot}2d - 2{\cdot}16666667d^3 + 1{\cdot}16666667d^4)(U_{j-2}^n + U_{j+2}^n) \\ &\quad + (0{\cdot}025396824d - 0{\cdot}2d^2 + 0{\cdot}5d^3 - 0{\cdot}33333216d^4)(U_{j-3}^n + U_{j+3}^n) \\ &\quad + (0{\cdot}01453324d^2 - 0{\cdot}00178572d - 0{\cdot}04166664d^3 + 0{\cdot}04166568d^4)(U_{j-4}^n + U_{j+4}^n). \end{aligned} \tag{35}$$

The numerical flux of the scheme is

$$\begin{aligned} T(U^n; j) &= (0{\cdot}01453324d - 0{\cdot}00178572 - 0{\cdot}04166664d^2 + 0{\cdot}04166568d^3)F_{j-3}^d \\ &\quad + (0{\cdot}023611104 - 0{\cdot}18546676d + 0{\cdot}45833336d^2 - 0{\cdot}29166648d^3)F_{j-2}^d \\ &\quad + (1{\cdot}222866572d - 0{\cdot}176388896 - 1{\cdot}70833331d^2 + 0{\cdot}87500019d^3)F_{j-1}^d \\ &\quad + (1{\cdot}423611104 - 2{\cdot}843800084d + 3{\cdot}12499993d^2 - 1{\cdot}45833165d^3)F_j^d \\ &\quad - (1{\cdot}423611104 - 2{\cdot}843800084d + 3{\cdot}12499993d^2 - 1{\cdot}45833165d^3)F_{j+1}^d \\ &\quad - (1{\cdot}222866572d - 0{\cdot}176388896 - 1{\cdot}70833331d^2 + 0{\cdot}87500019d^3)F_{j+2}^d \\ &\quad - (0{\cdot}023611104 - 0{\cdot}18546676d + 0{\cdot}45833336d^2 - 0{\cdot}29166648d^3)F_{j+3}^d \\ &\quad - (0{\cdot}01453324d - 0{\cdot}00178572 - 0{\cdot}04166664d^2 + 0{\cdot}04166568d^3)F_{j+4}^d. \end{aligned} \tag{36}$$

The amplification factor of the scheme is

$$\lambda = 1 - 2(3{\cdot}250794d - 8{\cdot}533333d^2 + 10{\cdot}66666692d^3 - 5{\cdot}33333304d^4). \tag{37}$$

The stability condition of the scheme is

$$0 \leqslant d \leqslant 1. \tag{38}$$

4. HIGH-ORDER SCHEMES FOR SCALAR ADVECTION–DIFFUSION EQUATION

In Section 3 we presented some conservative high-order schemes for the model parabolic equation. In this section we discuss a way to construct conservative high-order schemes for a scalar advection–diffusion equation

$$u_t + (au - \nu u_x)_x = 0, \tag{39}$$

where $au - \nu u_x$ is the physical advection–diffusion flux which is a combination of the advection flux and the diffusion flux.

The numerical flux $H(U^n; j)$ which simulates the physical advection–diffusion flux can be easily defined by combining a numerical advection flux $F(U^n; j)$ and a diffusion flux $T(U^n; j)$, i.e.

$$H(U^n; j) = F(U^n; j) + T(U^n; j). \tag{40}$$

Therefore the numerical scheme can be written as

$$\begin{aligned} U_j^{n+1} &= U_j^n - \frac{k}{h}[H(U^n; \ ) - H(U^n; j-1)] \\ &= U_j^n - \frac{k}{h}[F(U^n; j) - F(U^n; j-1) + T(U^n; j) - T(U^n; j-1)]. \end{aligned} \tag{41}$$

Taking the three-point centred advection numerical flux of the Lax–Wendroff scheme,

$$F(U^n; j) = \tfrac{1}{2}(1 + c)F_j^a + \tfrac{1}{2}(1 - c)F_{j+1}^a, \tag{42}$$

where c is the Courant number, $F_j^a = Re\, F_j^d = aU_j^n$ is the local advection flux and $Re = c/d = ah/\nu$ is the local cell Reynolds number, and the five-point centred diffusion numerical flux (28), for example, we have the following advection–diffusion numerical flux:

$$\begin{aligned} H(U^n; j) = {} & \left(\frac{1}{2} + \frac{c}{2} + \frac{5}{4Re} - \frac{3c}{2Re^2}\right)F_j^a + \left(\frac{c}{2Re} - \frac{1}{12}\right)\frac{1}{Re}F_{j-1}^a \\ & + \left(\frac{1}{2} - \frac{c}{2} + \frac{3c}{2Re^2} - \frac{5}{3Re}\right)F_{j+1}^a + \left(\frac{1}{12} - \frac{1}{2Re}\right)\frac{1}{Re}F_{j+2}^a. \end{aligned} \tag{43}$$

This is a five-point, second-order (in time and space) scheme. By following this example, any high-order numerical scheme can be constructed for the scalar advection–diffusion equation.

5. CONCLUSIONS

An approach for constructing two-level, explicit, fully discrete, arbitrary-order, conservative numerical methods for a one-dimensional scalar parabolic equation has been presented. To illustrate the technique, fully discrete three-, five-, seven- and nine-point conservative numerical schemes are given. One remarkable property of the parabolic numerical schemes is that a higher-order scheme has a better stability condition.

Based on high-order schemes of both the advection and diffusion equations, a way to construct high-order schemes for a scalar advection–diffusion equation is discussed. For systems of advection–diffusion equations, by using the operator-splitting method, high-order numerical schemes for both hyperbolic and parabolic equations can be explicitly applied.

REFERENCES

1. R. J. Leveque, *Numerical Methods for Conservation Laws*, Basel, Boston, Berlin: Birkhauser, 1990.
2. R. M. Beam and R. F. Warming, 'Alternating direction implicit method for parabolic equations with a mixed derivative', *SIAM J. Sci. Stat. Comput.*, **1**, 131–59 (1980).
3. P. E. Brown, 'Numerical solution of the 2D unsteady Navier–Stokes equations using viscous–convective operator splitting', *M.Sc. Thesis*, Cranfield Institute of Technology, 1990.
4. J. Shi, 'A linear stability analysis method for numerical schemes', *COA Rep. 9310*, Cranfield Institute of Technology, 1993.
5. J. Shi and E. F. Toro, 'Fully discrete arbitrary-order schemes for a scalar hyperbolic conservation law', *COA Rep. 9307*, Cranfield Institute of Technology, 1993.