respectively. A distinctive feature of these new schemes is the exact evaluation of the contribution of the linear part. That is, if the nonlinear part is zero, then the scheme reduces to the evaluation of the exponential function of the operator (or matrix) $\mathcal{L}$ representing the linear part. We show that such schemes have very good stability properties and, in fact, describe explicit schemes with stability regions similar to those of typical implicit schemes used in, for example, fluid dynamics applications.

Computing and applying the exponential or other functions of operators in the usual manner typically requires evaluating dense matrices and, for that reason, is highly inefficient. An exception is the case where there is a (fast) transform that diagonalizes the operator. For example, if $\mathcal{L}$ is a convolution (or a circulant) matrix which is diagonalized by the Fourier transform (FT), then computing functions of operators can be accomplished by a fast algorithm, e.g. the FFT. It is clear that in this case the need for FT for the diagonalization prevents one from extending this approach to the case of variable coefficients.

We note that the problem of computing the exponential of large matrices has been of interest in numerical analysis (see, e.g. [11] and references therein) due to the ubiquitous nature of this operator in physics and mathematics. It turns out [5] that the wavelet transform produces sparse representations (up to a finite but arbitrary accuracy) for a wide class of operators. This fact may be used for computing functions of operators (see [5, 6]), in particular, of elliptic operators with variable coefficients. In the wavelet system of coordinates computing the exponential of such operators always results in sparse matrices and, therefore, using the exponential of operators for numerical purposes is an efficient option [7].

In this paper we further develop the approach of [7], concentrating on issues of stability of time-discretization schemes with *exact* treatment of the *linear part* (ELP) schemes. We study the stability of these schemes using an approach developed in [14] and show that ELP schemes have distinctly different stability properties as compared with known implicit–explicit schemes.

In particular, we are interested in applications of ELP schemes to advection–diffusion equations. Among equations for which ELP schemes appear to be very natural are, for example, the Navier–Stokes equations which may be written in the form (1.1) (see Appendix A). The stability properties of time-discretization schemes for advection–diffusion equations are controlled by the linear term and, therefore, these equations require an implicit treatment to avoid choosing an unreasonably small time step. As we show in this paper, using an explicit ELP scheme, it is possible to achieve stability usually associated with implicit predictor–corrector schemes. Even if an implicit ELP scheme is used, as is done in [7], an approximation is used only for the nonlinear term, giving one a chance to clearly distinguish numerical errors due to that term. Moreover, in the usual implicit schemes for advection–diffusion equations the corrector part of the scheme requires iterations that involve either both linear and nonlinear terms or only the linear term (see [14, A consistent adaptive approach may be used to perform all computations in the wavelet domain [7]. We note, however, that applying the exponential function of operators in the wavelet domain may also be combined with the evaluation of the nonlinear part in the physical domain. This may be the simplest way to implement ELP schemes, short of implementing adaptive schemes of [7]. ELP schemes may also be used in combination with pseudo-spectral methods in space. For example, using the pseudo-spectral Fourier method for solving equations with constant coefficients, differential operators can be applied in the Fourier domain and the nonlinear part in the physical domain.

In Section 2 we introduce new multistep methods where the linear part is treated exactly and the nonlinear part is evaluated either implicitly or explicitly. We introduce an integral representation for the time evolution problem and then discretize the integral equation in time. Finally, we derive expressions for the operator-valued coefficients representing contributions from different time levels. Then, in Section 3, we describe algorithms for the evaluation of the operator-valued coefficients and in Section 4 we study stability regions of new schemes. We describe a method for linear stability analysis following [14] and construct stability regions for the new class of schemes in order to analyze their properties.

2. SCHEMES WITH EXACT TREATMENT OF THE LINEAR TERM

2.1. *Integral Formulation*

We are interested in the solution of nonlinear evolution equations of the form

$$u_t = \mathcal{L}u + \mathcal{N}(u) \quad \text{in } \Omega \in \mathbb{R}^d, \tag{2.1}$$

where $u = u(x, t), x \in \mathbb{R}^d, d = 1, 2, 3$, and $t \in [0, T]$. We also supply the initial conditions,

$$u(x, 0) = u_0(x) \quad \text{in } \Omega, \tag{2.2}$$

and the linear boundary conditions

$$\mathcal{B}u(x, t) = 0 \quad \text{on } \partial\Omega \in \mathbb{R}^{d-1}, t \in [0, T]. \tag{2.3}$$

form of (2.4),

$$u(x,t) = \sum_{i=1} c_i e^{(t-\kappa_i)\mathcal{L}} u(x,\kappa_i) + \int_0^t e^{(t-\tau)\mathcal{L}} \mathcal{N}(u(x,\tau))\,d\tau, \qquad \sum_{i=1} c_i = 1, \tag{2.5}$$

can be considered, where $u(x,\kappa_i)$, $i, 1\ldots,n$, are assumed to be known (we use (2.5) with $n = 1$).

The operator $\mathcal{L}$ and the exponential operator $\mathrm{e}^{\mathcal{L}t}$ in (2.4) incorporate the boundary conditions. For example, writing $u(x,t) = \mathrm{e}^{\mathcal{L}t}u_0(x)$ implies that the function $u(x,t)$ solves $u_t = \mathcal{L}u$ with the initial condition $u(x,0) = u_0(x)$ and the boundary condition $\mathcal{B}u(x,t) = 0$ for $x \in \partial\Omega$.

The integral equation (2.4) is easy to use for numerical purposes if, for example, $\mathcal{L}$ is an operator with constant coefficients and u is a periodic function. In this case $\mathcal{L}$ can be represented by a diagonal matrix in the Fourier basis. For instance, if $\mathcal{L}$ is the Laplacian and $u(x) = e^{ikx}$, then $\Delta u = -k^2 u$ and, in such a case, the exponential operator $e^{(t-\tau)\mathcal{L}}$ simply reduces to multiplication by $e^{-k^2(t-\tau)}$. However, for a general linear operator $\mathcal{L}$ with variable coefficients, the exponential operators appearing in this equation are represented by dense matrices. As far as we know, this is the main reason for the limited use of (2.4) as a starting point of numerical discretization.

We observe that the situation is different for the exponential operators on a wide class of linear operators in a wavelet system of coordinates. The sparsity of the exponential operators was utilized in [7] for constructing a numerical algorithm for the solution of PDEs of the form (2.1). In this paper we develop this approach further in order to construct a collection of high order discretizations of (2.4) with good stability properties.

2.2. *A Procedure for Time Discretization*

In order to simplify the notation in our derivation, we replace a linear operator $\mathcal{L}$ by a scalar q since the coefficients of the scheme are analytic (operator) functions of $\mathcal{L}$. Since all such functions commute with each other, it is sufficient to consider a scalar in deriving the coefficients of the numerical scheme.

Thus, instead of (2.1) and (2.4), it is sufficient to examine

$$u_t = qu + \mathcal{N}(u) \tag{2.6}$$

and

$$u(x,t) = e^{q(t-\kappa)}u(x,\kappa) + \int_0^t e^{q(t-\tau)}\mathcal{N}(u(x,\tau))\,d\tau, \tag{2.7}$$

where $0 \le \kappa \le t$ and $u(x,\kappa)$ is given.

Let us consider the function $u(x,t)$ at the discrete moments of time $t_n = t_0 + n\Delta t$, where Δt is the time step so that $u_n \equiv u(x,t_n)$ and $N_n \equiv \mathcal{N}(u(x,t_n))$. Discretizing (2.7) yields

$$u_{n+1} = e^{ql\Delta t}u_{n+1-l} + \Delta t\left(\gamma N_{n+1} + \sum_{m=0}^{M-1}\beta_m N_{n-m}\right), \tag{2.8}$$

where $M + 1$ is the number of time levels involved in the discretization and $l \le M$. The expression in parenthesis in (2.8) may be viewed as the numerical quadrature for the integral in (2.7). The coefficients γ and β_m are the functions of $q\Delta t$ (to simplify the notation, we suppress the dependence of γ and β_m on l).

We observe that the algorithm is explicit if $\gamma = 0$ and it is implicit otherwise. Typically, for a given M, the order of accuracy is M for an explicit scheme and $M + 1$ for an implicit scheme, due to one more degree of freedom, γ (see later for a more detailed discussion). We refer to this family of schemes as exact linear part (ELP) schemes.

Remark 1. Using (2.8) to discretize partial differential equations (or, in general, using semigroup approach as in (2.4)) may be viewed as a way to “reduce” partial differential to ordinary differential equations. For example, one can apply the first Dahlquist criterion for (2.8) (see, e.g. [12]) to determine the maximal order for the implicit schemes. The reason this can be done is that the operator coefficients in (2.8) commute with each other and, thus, polynomials with such coefficients can be manipulated in the usual manner.

Remark 2. In the particular case where $l = 2;\ \gamma = 0$, and $M = 1$, Eq. (2.8) turns into the explicit scheme known as the “slave-frog” scheme,

$$u_{n+1} = e^{2q\Delta t}u_{n-1} + \Delta t\beta_0 N_n; \quad \beta_0 = \frac{e^{2q\Delta t} - 1}{q\Delta t}. \tag{2.9}$$

This scheme has been used in computational fluid dynamics (see, e.g. [9]). We do not know other examples of temporal schemes related to the family (2.8). As we will see below, the scheme in (2.9) does not have good stability properties (see Section 4.2).

Let us first consider the case $l = 1$ (so that $\tau = t_n$ in Eq. (2.7) is the nearest time level t_{n+1}),

$$u_{n+1} = e^{q\Delta t}u_n + \Delta t\left(\gamma N_{n+1} + \sum_{m=0}^{M-1}\beta_m N_{n-m}\right). \tag{2.10}$$

Our task now is to find the coefficients γ and β_m of scheme (2.10) in terms of $q\Delta t$. In [7] these coefficients are derived so that (i) the expression in parentheses is an $M + 1$ order quadrature approximation to the integral in (2.7) and (ii) the quadrature uses the fewest number of nonzero coefficients, β_m. In this paper we adopt a different approach and obtain expressions for the coefficients γ and β_m in (2.10) by using the differential equation (2.6) and by expanding the terms u_{n+1} and $e^{q\Delta t}u_n$ into the Taylor series. The resulting expressions differ from those in [7] by higher order terms in $q\Delta t$ which are beyond the order of the approximation (see discussion below).

We start by expanding u_{n+1} in the Taylor series at the time level t_n,

$$u_{n+1} = \sum^{\infty}$$

etc., and, in general,

$$u^{(k)} = q^k u + \sum_{j=0}^{k-1} N^{(j)} q^{k-1-j}. \tag{2.13}$$

Substituting (2.12) into (2.11), we have

$$u_{n+1} = \sum_{k=0}^{\infty} q^k \frac{(\Delta t)^k}{k!} u_n + \sum_{k=1}^{\infty} \sum_{j=0}^{k-1} N^{(j)} q^{k-1-j} (\Delta t)$$

TABLE I

Coefficients of Implicit ELP Schemes for $l = 1$, where $Q_k = Q_k(q \cdot t)$

M	γ	β_0	β_1	β_2	Order
1	Q_2	$Q_1 - Q_2$	0	0	2
2	$Q_2/2 + Q_3$	$Q_1 - 2Q_3$	$Q_3 - Q_2/2$	0	3
3	$Q_2/3 + Q_3 + Q_4$	$Q_1 + Q_2/2 - 2Q_3 - 3Q_4$	$-Q_2 + Q_3 + 3Q_4$	$Q_2/6 - Q_4$	4

and we have

$$\begin{aligned} \gamma + \sum_{m=0}^{M-1} \beta_m &= Q_1 \\ \frac{1}{k!}\left(\gamma + (-1)^k \sum_{m=1}^{M-1} m^k \beta_m\right) &= Q_{k+1}, \quad k = 1, \ldots, M. \end{aligned} \tag{2.20}$$

In the explicit case $\gamma = 0$ and we obtain

$$\begin{aligned} \sum_{m=0}^{M-1} \beta_m &= Q_1, \\ \frac{(-1)^k}{k!} \sum_{m=1}^{M-1} m^k \beta_m &= Q_{k+1}, \quad k = 1, \ldots, M - 1. \end{aligned} \tag{2.21}$$

For

TABLE III

Coefficients of Implicit ELP Schemes for *l* = 2, where Q_k^2 = Q_k(2*q* ¢ *t*)

M	°	*fl*$_0$	*fl*$_1$	*fl*$_2$	Order
1	Q_2^2 ¡ Q_1^2	2Q_1^2 ¡ Q_2^2	0	0	2
2	Q_3^2 ¡ Q_2^2=2	2.Q_2^2 ¡ Q_3^2/	Q_1^2 ¡ 3Q_2^2=2 C Q_3^2	0	3
3	¡ Q^2				

As we have shown in Section 2, the coefficients of ELP schemes are written in terms of operators $Q_k = Q_k(\mathcal{L}\Delta t)$, where

$$Q_j(\mathcal{L}\Delta t) = \frac{e^{\mathcal{L}\Delta t} - E_j(\mathcal{L}\Delta t)}{(\mathcal{L}\Delta t)^j}, \tag{3.1}$$

and

$$E_j(\mathcal{L}\Delta t) = \sum_{k=0}^{j-1} \frac{(\mathcal{L}\Delta t)^k}{k!}, \tag{3.2}$$

for $j = 0, 1, \ldots$. We have

$$\begin{aligned} Q_0(\mathcal{L}\Delta t) &= e^{\mathcal{L}\Delta t}, \\ Q_1(\mathcal{L}\Delta t) &= (e^{\mathcal{L}\Delta t} - \mathcal{I})(\mathcal{L}\Delta t)^{-1}, \\ Q_2(\mathcal{L}\Delta t) &= (e^{\mathcal{L}\Delta t} - \mathcal{I} - \mathcal{L}\Delta t)(\mathcal{L}\Delta t)^{-2}, \\ &\cdots \end{aligned} \tag{3.3}$$

We will now describe a method that permits us to compute operators $Q_0, Q_1, Q_2, \ldots$ without computing $(\mathcal{L}\Delta t)^{-1}$. The problem in using the Taylor expansion directly is that it will result in a loss of accuracy due to possibly large singular values of $\mathcal{L}\Delta t$. To avoid this problem in computing the exponential Q_0, we use the scaling and squaring method. This method results in a fast algorithm if the evaluation is performed in a wavelet basis with a sufficient number of vanishing moments (for a chosen accuracy) [6].

The scaling and squaring method for the exponential is based on the identity

$$Q_0(2x) = Q_0^2(x). \tag{3.4}$$

First we compute $Q_0(\mathcal{L}\Delta t 2^{-l})$ for some l chosen so that the largest singular value of $\mathcal{L}\Delta t 2^{-l}$ is less than one. This computation is performed using the Taylor expansion. Using (3.4), the resulting matrix is then squared l times to obtain the final answer.

A similar algorithm may be used for computing $Q_j(\mathcal{L}\Delta t)$, $j = 1, 2, \ldots,$ for any finite j. Let us illustrate this approach by considering $j = 1, 2, \ldots, 6$.

It is not difficult to verify that

$$\begin{aligned} Q_1(2x) &= \frac{1}{2}(Q_0(x)Q_1(x) + Q_1(x)), \\ Q_2(2x) &= \frac{1}{4}(Q_1(x)Q_1(x) + 2Q_2(x)), \\ Q_3(2x) &= \frac{1}{8}(Q_1 \end{aligned}$$

The identities in (3.5) are obtained by considering the difference between $2^j Q_j(2x)$ and $Q_{j/2}(x)Q_{j/2}(x)$ if j is even, and $Q_{(j-1)/2}(x)Q_{(j+1)/2}(x)$ if j is odd. This difference is then expressed using a linear combination of $Q_j(x), Q_{j-1}$

We start by providing several examples of stability diagrams for discretization methods discussed in [14] in order to establish a baseline for comparison with the stability properties of ELP schemes.

Let us first provide an interpretation for parameters q and ν for a model advection–diffusion equation,

$$u_t = \alpha u_{xx} + U u_x, \tag{4.4}$$

where U is a constant convective velocity and $\alpha > 0$ is a damping constant. Considering periodic boundary conditions, we obtain for the Fourier transform $\hat{u}$ of the function u,

$$\hat{u}_t = -\alpha k^2 \hat{u} + ikU\hat{u}. \tag{4.5}$$

The equation (4.5) has the same structure as Eq. (4.3) with $q = \alpha k^2 > 0$ and $\nu = ikU$. For nonlinear equations under consideration the splitting of the right-hand side of (4.5) into two terms corresponds to the splitting into linear (diffusion) and nonlinear (advection) parts, respectively.

We now turn to an example from [14] which deals with a mixed implicit–explicit method, where the linear term is approximated using the implicit third-order Adams–Moulton scheme and the nonlinear term is approximated using the explicit third-order Adams–Bashforth scheme as

$$u_n = u_{n-1} - q\Delta t \sum_{k=0}^{2} \alpha_k u_{n-k} + \nu \Delta t \sum_{k=1}^{3} \beta_k u_{n-k}, \tag{4.6}$$

where $\alpha_0 = 5/12, \alpha_1 = 8/12, \alpha_2 = -1/12$, and $\beta_1 = 23/12, \beta_2 = -16/12, \beta_3 = 5/12$. The corresponding stability diagram is plotted in Fig. 1. We observe that this method is conditionally stable with a region of stability that separates from the imaginary (vertical) axis for $q\Delta t \geq 6$ (that is, for high wavenumbers, since $q \propto k^2$). In this figure, as well as other stability diagrams below, the label of the curve (the boundary of the stability region), $q\Delta t$, can be identified by finding the point of intersection of the corresponding curve with the horizontal axis in the right half-plane.

In our second example, borrowed from the same paper [14], we reproduce the stability diagram for a third-order stiffly stable scheme. According to [10], a method is stiffly stable if it is consistent in a neighborhood of the origin and absolutely stable away from the origin in the left imaginary plane. Stiffly stable multistep methods are implicit and their coefficients are available up to eleventh order (see [13]).

In [14] a mixed explicit–implicit stiffly stable scheme has been introduced,

$$\sum_{k=0}^{3} \alpha_k u_{n-k} = -q\Delta t u_n + \nu \Delta t \sum_{k=1}^{3} \beta_k u_{n-k}, \tag{4.7}$$

with α-coefficients being those of the standard third-order stiffly stable scheme from [10], $\alpha_0 = 11/6; \alpha_1 = -3; \alpha_2 = 3/2$, and $\alpha_3 = 1/3$. The coefficients for the explicit part are $\beta_1 = 3; \beta_2 = -$

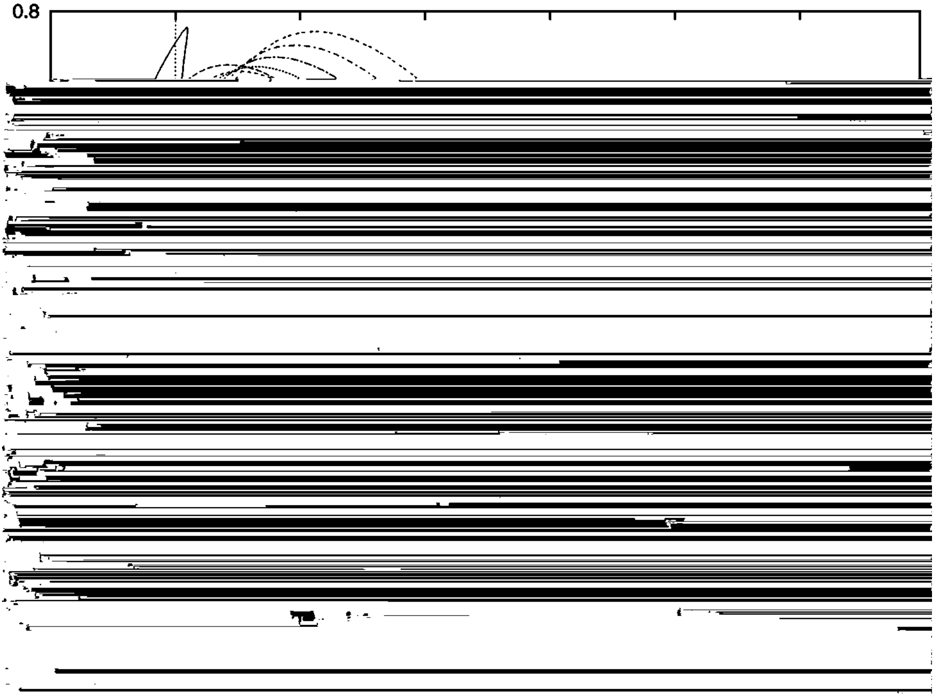


FIG. 1. Stability diagram of the third-order Adams–Moulton/Adams–Bashforth scheme. The curves correspond to different values of the parameter $q\Delta t$ ($q\Delta t = 0$ corresponds to the explicit scheme). This plot reproduces Fig. 6 in [14].

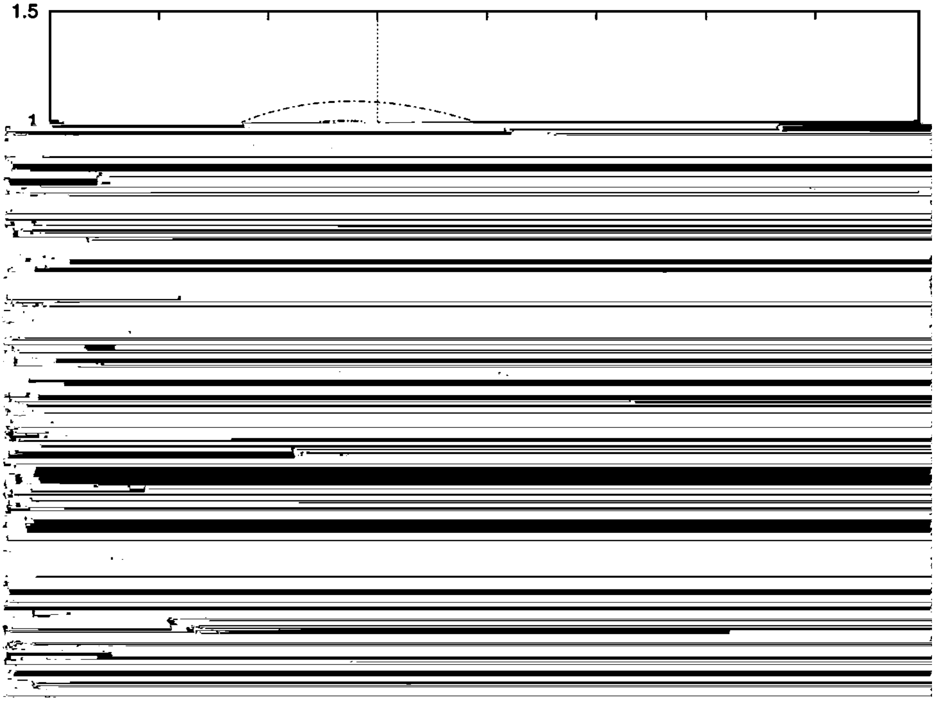
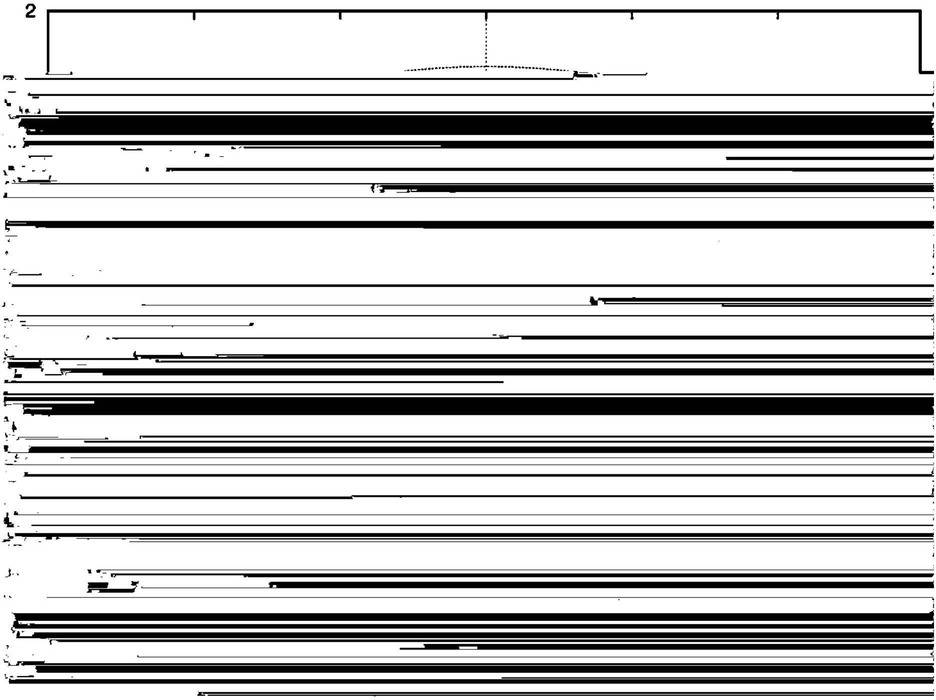


FIG. 2. Stability diagram of the mixed implicit–explicit stiffly stable scheme of third order ($q\Delta t = 0$ corresponds to the explicit scheme). This plot reproduces Fig. 7 in [14].


2

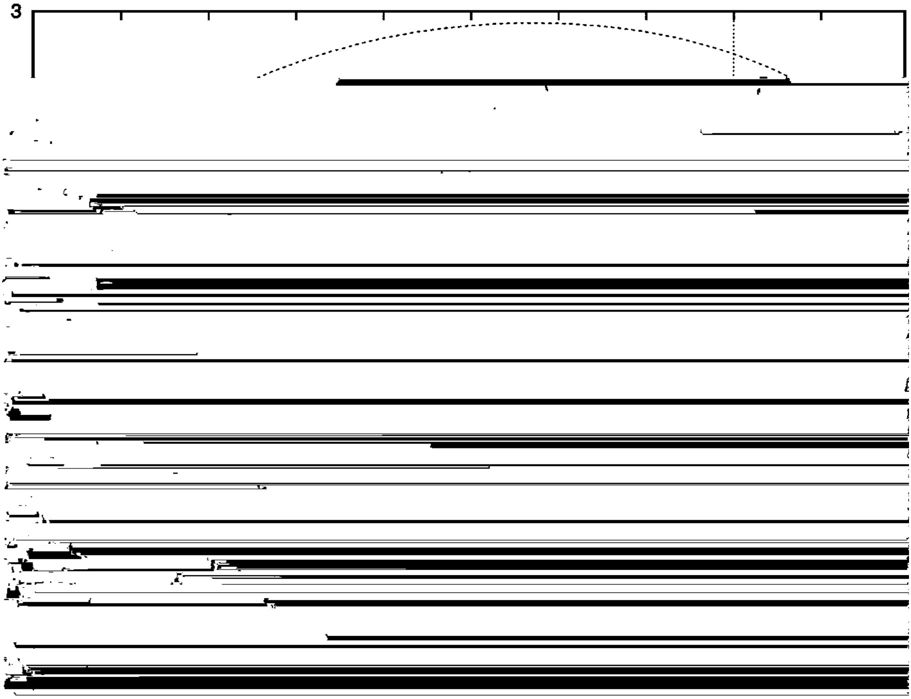


FIG. 4. Stability diagram for the implicit third-order scheme with $l = 2; M = 2$ (see Table III).

We provide in Appendix C an explanation for such behavior of the stability region of the $l = 2$ schemes in the limit $q \to 0$. We now turn to a class of schemes with better stability properties.

Implicit ELP schemes $(l = 1)$.

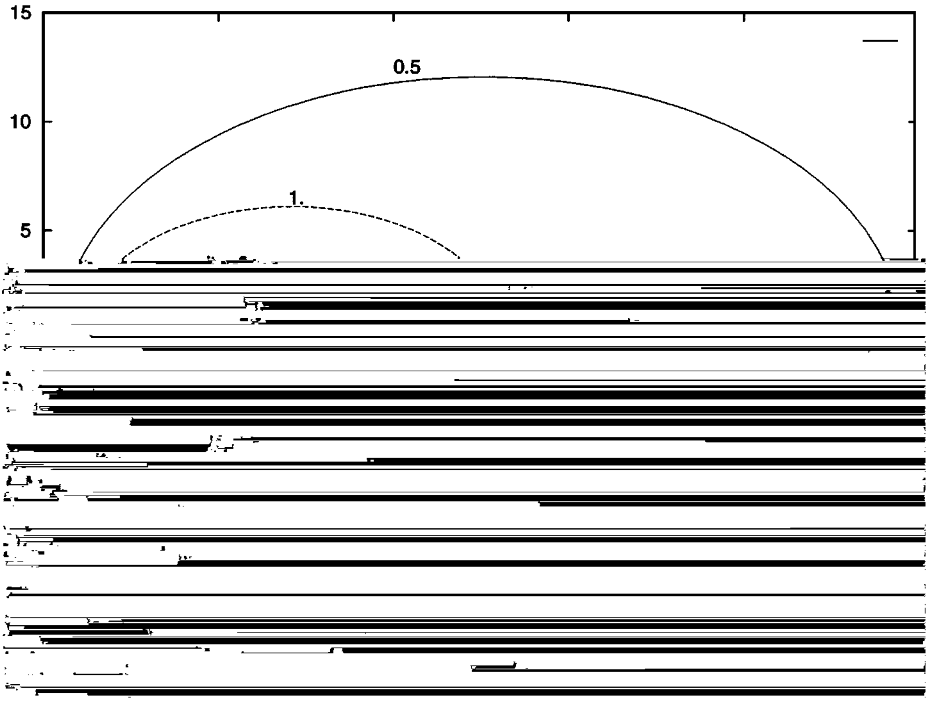


FIG. 5. Stability diagram for the second-order implicit scheme ($M = 1; l = 1$) using bare coefficients $\gamma = Q_2$ and $\beta_0 = Q_1 - Q_2$ from Table I. (Here the interior of the curves are the regions of

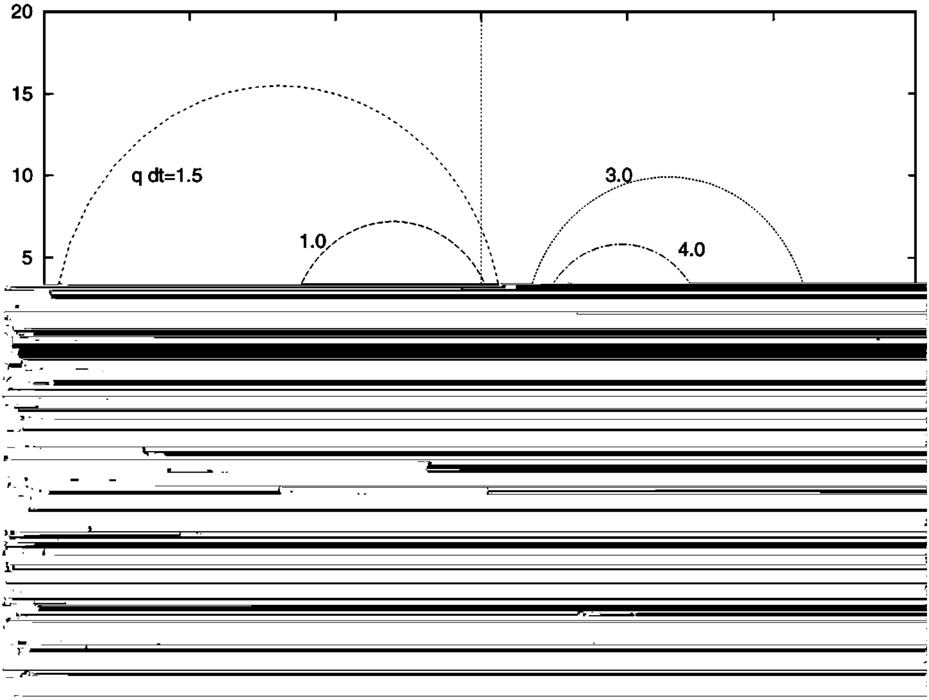


FIG. 7. Stability diagram for the third-order implicit scheme with $l = 1; M = 2$ and coefficients from Table I.

of the fourth-order implicit scheme is shown in Fig. 8. It is still larger than that of the third-order stiffly stable scheme. This scheme becomes super-stable for $q\Delta t > 4.26$ (not shown in Fig. 8).

Explicit ELP schemes $(l = 1)$. Finally, we consider the stability of explicit ELP schemes. Let us illustrate the stability regions of the second $M = 2$ and third $M = 3$ order schemes in Figs. 9 and 10. We note that the stability regions of these schemes for $q\Delta t > 0$ are more reminiscent of those for implicit schemes (compare, for example, Fig. 10 with Fig. 2). In the limit $q\Delta t \to 0$ the stability regions tend to those of the second- and third-order Adams–Bashforth schemes. This is expected since in (2.8) the exponential function $e^{q\Delta t} \to 1$ as $q\Delta t \to 0$, and the bare coefficients β_k in (2.10) coincide with those of the corresponding order Adams–Bashforth scheme.

For useful schemes it is important that the stability regions obtained via the stability analysis of [14] grow as $q\Delta t$

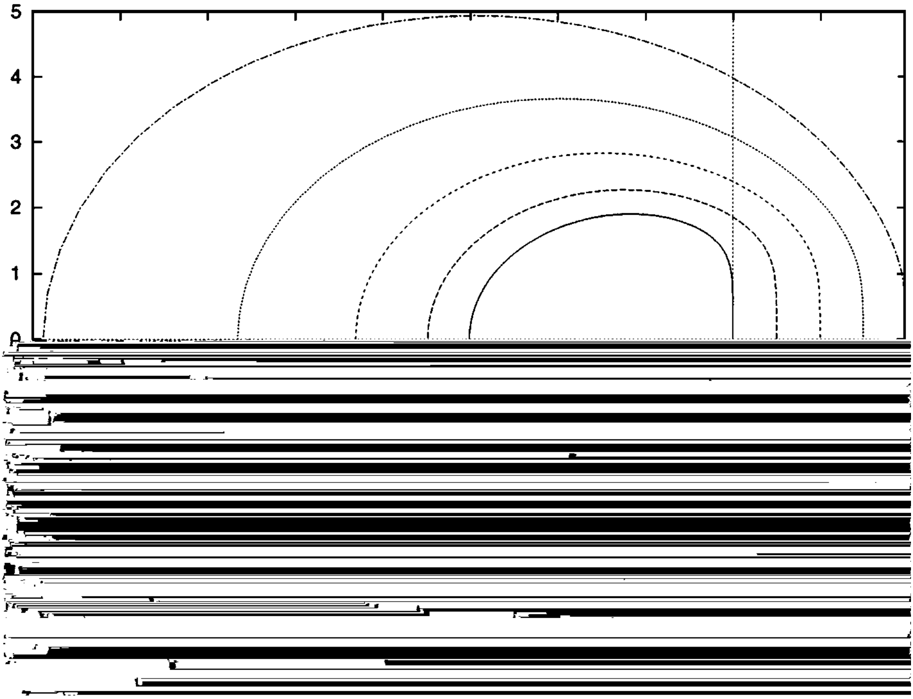


FIG. 8. Stability diagram for the fourth-order implicit scheme with $l = 1; M = 3$ and coefficients from Table I.

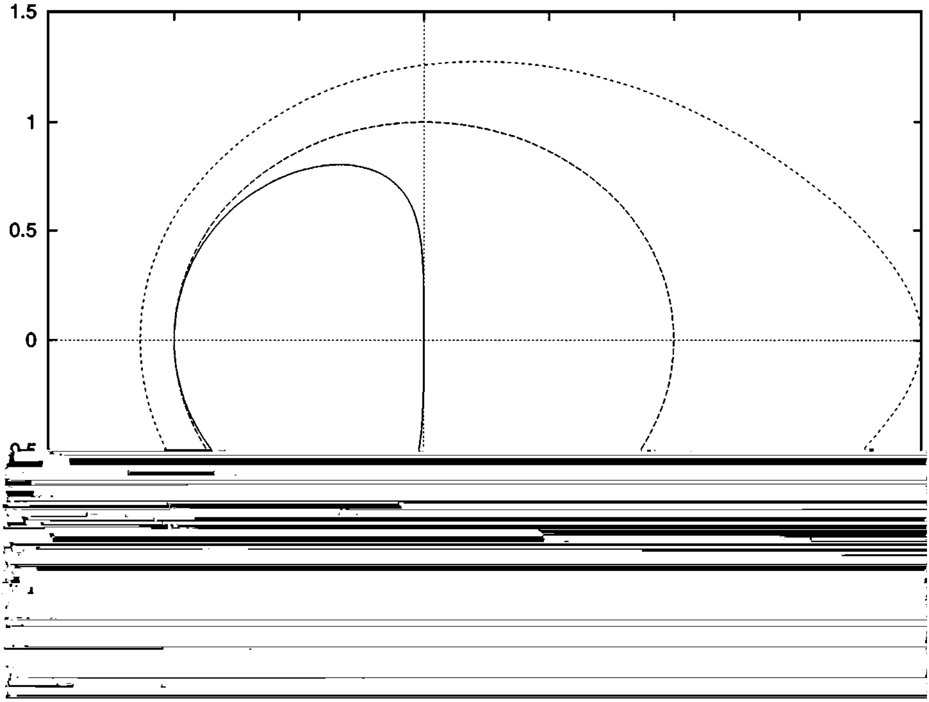


FIG. 9. Stability diagram for the second-order explicit scheme with $M = 2; l = 1$ and coefficients from Table II. The curve at $q\Delta t = 0$ corresponds to the boundary of the stability region of the second-order Adams–Bashforth method.

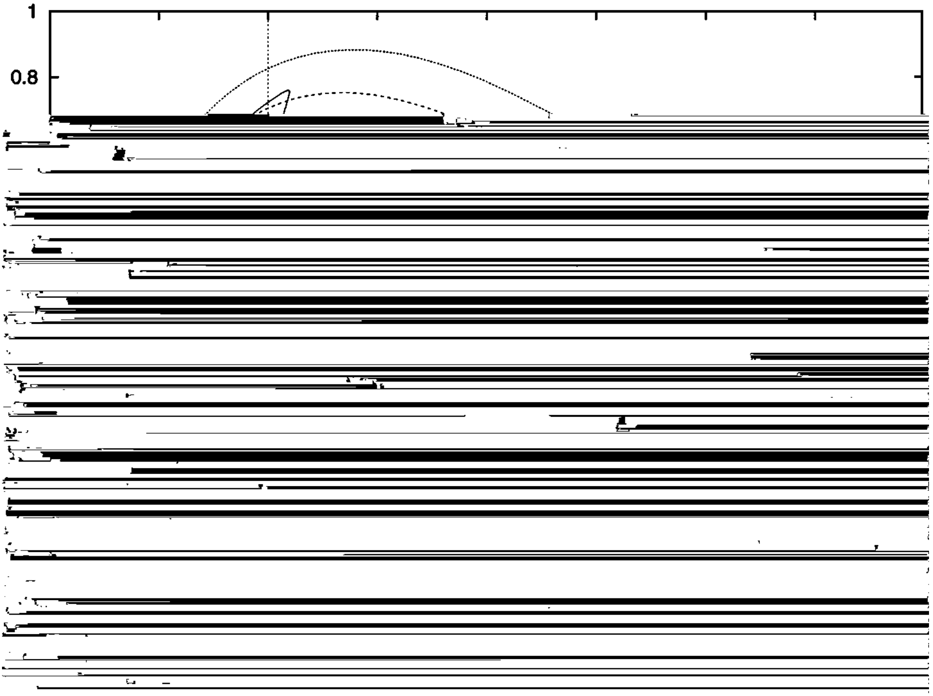


FIG. 10. Stability diagram for the third-order explicit scheme with $M = 3, l = 1$ and coefficients from Table II. The curve at $q\Delta t =$

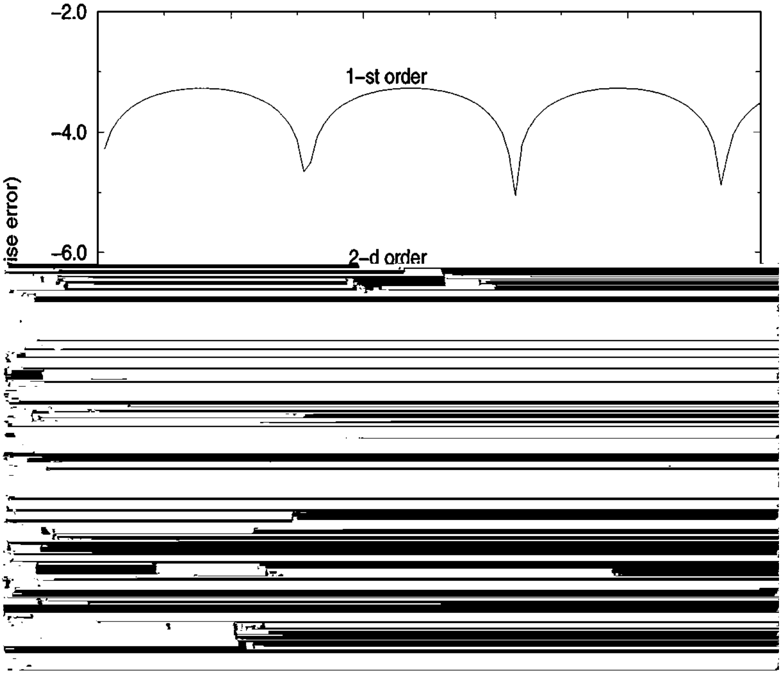


FIG. 11. Error as a function of time for Example 1 for first-, second-, and third-order explicit ELP schemes.

EXAMPLE 2. Again using explicit ELP schemes in time and multiwavelets in space, we consider Burgers' equation

$$u_t = \nu u_{xx} - u u_x, \quad x \in [0, 1]. \tag{4.10}$$

A reference solution with periodic boundary conditions may be written as

$$u^{(ref)} = -2\nu \frac{\phi_x(x - ct, t + \tau)}{\phi(x - ct, t + \tau)}, \quad \tau > 0 \tag{4.11}$$

where

$$\phi(x, t) = \sum_{n=-\infty}^{\infty} e^{-(x-n)^2/4\nu t}. \tag{4.12}$$

In Fig. 12 we display the numerical solution for $c = 4$, $\nu = 0.1/\pi$, and $\tau = 1/(2\pi)$ (these are parameters of the standard test case) so that the profile moves at speed $c = 4$. The pointwise numerical error for the solution at $t = 1/16$ is plotted in Fig. 13. The maximum numerical error is given in Table V for the explicit first-, second-, and third-order ELP

TABLE V
Maximum Error of the Solution of the Periodic Burgers' Equation at the Time $t = 1/16$, $\nu = 0.1/\pi$, and $c = 0, 4$

c\order	First	Second	Third
0	9.6×10^{-4}	3.6×10^{-7}	5.9×10^{-10}
4	3.0×10^{-2}	4.2×10^{-4}	6.5×10^{-6}

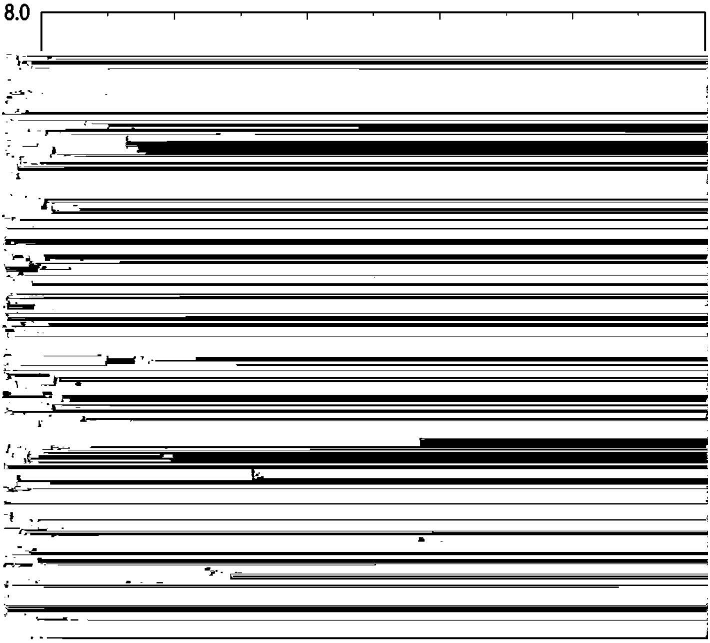


FIG. 12. Solution of the periodic Burgers' equation at *t* D 0 and *t* D 1=16; ″ D 0:1=...; *c* D 4.

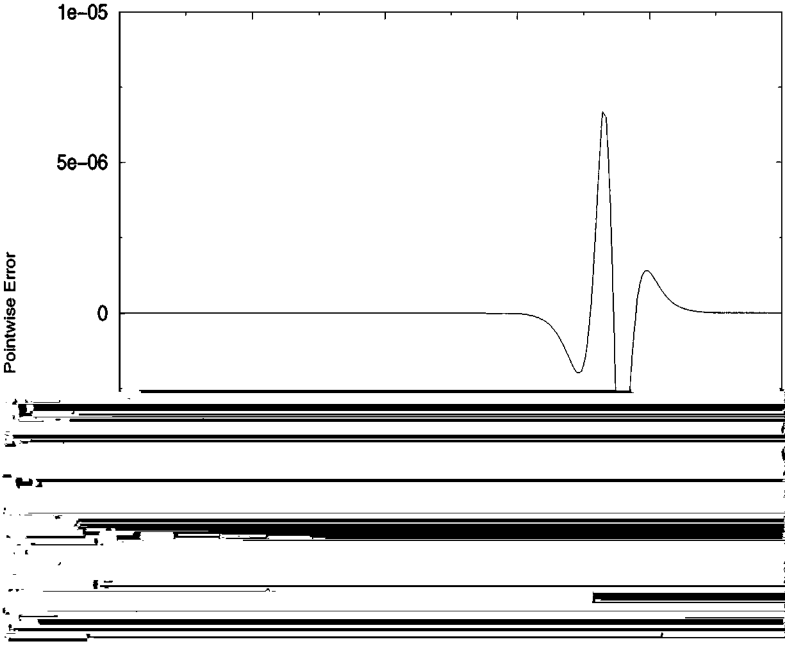


FIG. 13. The pointwise error for the solution of periodic Burgers' equation at *t* D 1=16.

schemes for $\Delta t = 10^{-4}, t = 1/16$ and $c = 0, c = 4$. Again, the spatial resolution was chosen to make temporal errors dominant.

5. CONCLUSIONS

We have demonstrated excellent stability properties of both explicit and implicit ELP schemes. The efficiency of using ELP schemes for nonlinear partial differential equations depends on the sparsity of the exponential of the linear part of the operator (obtained by

In order to show that **P**(**u**) satisfies (A.2), let us compose the divergence operator and (A.6), so that we need to verify

$$\begin{aligned} &\partial_1 R_1^2 + \partial_2 R_2 R_1 + \partial_3 R_3 R_1 = \partial_1 \\ &\partial_1 R_1 R_2 + \partial_2 R_2^2 + \partial_3 R_3 R_2 = \partial_2 \\ &\partial_1 R_1 R_3 + \partial_2 R_2 R_3 + \partial_3 R_3^2 = \partial_1. \end{aligned} \tag{A.7}$$

It is easy to check relations (A.7) by examining symbols of these operators in the Fourier domain; for example, the first equation in (A.7) reduces to

$$i\xi_1 \frac{\xi_1^2}{|\xi|^2} + i\xi_2 \frac{\xi_2 \xi_1}{|\xi|^2} + i\xi_3 \frac{\xi_3 \xi_1}{|\xi|^2} = i\xi_1. \tag{A.8}$$

Applying the divergence operator to (A.1), we obtain

$$-\Delta p = \sum_{k,l=1}^{3} \partial_k \partial_l u_k u_l$$

Finally, in rewriting (A.12) as

$$\mathbf{u}_t = \mathcal{L}\mathbf{u} + \mathcal{N}(\mathbf{u}), \tag{A.13}$$

we incorporate the boundary conditions into the operator $\mathcal{L}$. For example, $\mathbf{u} = \mathcal{L}^{-1}\mathbf{v}$ means that u solves $\mathcal{L}\mathbf{u} = \mathbf{v}$ with the boundary conditions $\mathcal{B}u = 0$. Similarly, $u(x,t) = e^{\mathcal{L}t}u_0(x)$ means that u solves $u_t = \mathcal{L}u$, $u(x,0) = u_0(x)$, and $\mathcal{B}u(x,t) = 0$.

APPENDIX B

For the convenience of the reader, we describe a technique for the computation of a marginal stability curve (or the boundary of a stability region). We start with the test problem

$$u_t = \lambda u, \tag{B.1}$$

where λ is a complex parameter. Applying a time-discretization method to this equation, we obtain a homogeneous linear difference equation,

$$\sum^{M}$$

APPENDIX C

In this section we show that the family of schemes with $l = 2$ has a weak instability in the limit $q \to 0$. As an example let us analyze the third-order implicit scheme with the stability region illustrated in Fig. 4. The coefficients of this scheme are given in Table III with $M = 2$. In the limit $q \to 0$, the values of $Q_k^2 = Q_k(2q\Delta t)$ are

$$Q_1^2 = 2; \quad Q_2^2 = 2; \quad Q_3^2 = \frac{4}{3}; \quad Q_4^2 = \frac{2}{3}. \tag{C.1}$$

For the corresponding values of the coefficients we have $\gamma = 1/3$; $\beta_0 = 4/3$; $\beta_1 = 1/3$.

The linear homogeneous difference equation in this case is

$$u_{n+1} = u$$

11. M. Hochbruck and C. Lubich, On Krylov subspace approximations to the matrix exponential operator, *SIAM J. Numer. Anal.* **34**(5), 1911 (1997).
12. A. Iserles, *A First Course in the Numerical Analysis of Differential Equations* (Cambridge Univ. Press, Cambridge, 1996).
13. M. K. Jain and V. K. Srivastava, *High Order Stiffly Stable Methods for Ordinary Differential Equations*, Technical Report 394, University of Illinois, Urbana, II, 1970.
14. G. E. Karniadakis, M. Israeli, and S. A. Orszag, High order splitting methods for the incompressible Navier–Stokes equations, *J. Comput. Phys.* **97**, 414 (1991).