



ELSEVIER

Applied Numerical Mathematics 24 (1997) 203–218



APPLIED  
NUMERICAL  
MATHEMATICS

## Euler–Chebyshev methods for integro–differential equations

P.J. van der Houwen\*, B.P. Sommeijer

*CWI, P.O. Box 94079, 1090 GB Amsterdam, Netherlands*

---

### Abstract

We construct and analyse explicit methods for solving initial value problems for systems of differential equations with expensive right-hand side functions whose Jacobian has its stiff eigenvalues along the negative axis. Such equations arise after spatial discretization of parabolic integro–differential equations of Volterra or Fredholm type with nonstiff integral parts. The methods to be developed in this paper may be interpreted as stabilized forward Euler methods. They require only one right-hand side evaluation per step and the construction of a stabilizing matrix. This matrix should be tuned to the class of problems to be integrated. In the case of parabolic integro–differential equations, the stabilizing matrix will be based on Chebyshev polynomials and will be constructed by means of recursions satisfied by these polynomials. This construction is related to the construction of the intermediate stages in the so-called Runge–Kutta–Chebyshev methods for ordinary differential equations. In analogy with these methods, we shall call the stabilized Euler methods, Euler–Chebyshev methods. They are second-order accurate, and although they are explicit, their stepsize restriction is not prescribed by the stiff eigenvalues. For integro–differential equations in which the parabolic part consists of a one-dimensional diffusion term, we describe an efficient implementation of the stabilizing matrix, which is based on factorization properties of Chebyshev polynomials. © 1997 Published by Elsevier Science B.V.

*Keywords:* Numerical analysis; Initial-value problems; Extended real stability boundaries

---

### 1. Introduction

We consider the numerical solution of initial value problems (IVPs) for systems of differential equations with relatively expensive right-hand side functions. In particular, we shall study IVPs of the form

$$\frac{d\mathbf{y}(t)}{dt} = D(t)\mathbf{y}(t) + \mathbf{v}(t), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \quad \mathbf{y}, \mathbf{v} \in \mathbb{R}^r, \quad (1.1)$$

where  $D(t)$  is an  $r$ -by- $r$  matrix whose eigenvalues are assumed negative and  $\mathbf{v}(t)$  is an expensive function. Examples of such problems can be found in the class of semi-discrete parabolic integro–differential equations of Volterra or Fredholm type.

---

\* Corresponding author. E-mail: p.j.van.der.houwen@cwi.nl.

### 1.1. Volterra integro–differential equations

Consider the parabolic Volterra integro–differential equation

$$\frac{\partial u(t, \mathbf{x})}{\partial t} = Lu(t, \mathbf{x}) + g(t, \mathbf{x}) + \int_{t_0}^t k(t, \tau, \mathbf{x}, u(t, \mathbf{x})u(\tau, \mathbf{x})) \, d\tau, \quad \mathbf{x} \in \Omega, \quad (1.2)$$

where  $L$  is an elliptic operator and  $\Omega$  is a domain in  $\mathbb{R}^d$ . We replace  $\Omega$  by a set of  $r$  points  $\mathbf{x}_i$  and  $u(t, \mathbf{x})$  by the set of  $r$  values  $u(t, \mathbf{x}_i)$ . Let  $\mathbf{y}(t) = (y_i(t))$  where  $y_i(t)$  represents an approximation to  $u(t, \mathbf{x}_i)$ . Then, the initial-boundary value problem for (1.2) can be discretized into an initial value problem (IVP) for the  $r$ -dimensional system of Volterra integro–differential equations

$$\frac{dy_i(t)}{dt} = \mathbf{d}_i(t)^T \mathbf{y}(t) + g_i(t) + \int_{t_0}^t k(t, \tau, \mathbf{x}_i, y_i(t), y_i(\tau)) \, d\tau, \quad i = 1, \dots, r, \quad (1.2')$$

where  $g_i(t) := g(t, \mathbf{x}_i)$  and  $\mathbf{d}_i(t)$  is an  $r$ -dimensional vector representing the discretization of  $L$  at  $\mathbf{x}_i$ . Next, we define

$$z_i(t, s) := \int_{t_0}^s k(t, \tau, \mathbf{x}_i, y_i(t), y_i(\tau)) \, d\tau, \quad i = 1, \dots, r, \quad (1.3)$$

and we replace (1.2') by the equivalent system

$$\begin{aligned} \frac{dy_i(t)}{dt} &= \mathbf{d}_i(t)^T \mathbf{y}(t) + g_i(t) + z_i(t, t), & y_i(t_0) &= u(t_0, \mathbf{x}_i), \\ \frac{\partial z_i(t, s)}{\partial s} &= k(t, s, \mathbf{x}_i, y_i(t), y_i(s)), & z_i(t, t_0) &= 0, \end{aligned} \quad i = 1, \dots, r. \quad (1.4)$$

This problem can be cast into the form (1.1) with  $\mathbf{y}_0 = (u(t_0, \mathbf{x}_i))$ ,  $D$  the  $r$ -by- $r$  matrix the row vectors of which are given by  $\mathbf{d}_i$ , and with

$$\mathbf{v}(t) = \mathbf{g}(t) + \mathbf{z}(t, t), \quad \frac{\partial \mathbf{z}(t, s)}{\partial s} = \mathbf{K}(t, s, \mathbf{y}(t), \mathbf{y}(s)), \quad \mathbf{z}(t, t_0) = \mathbf{0}, \quad \mathbf{y}, \mathbf{z} \in \mathbb{R}^r. \quad (1.5)$$

Here,  $\mathbf{g}(t) := (g_i(t))$ ,  $\mathbf{z}(t, s) := (z_i(t, s))$  and  $\mathbf{K}(t, s, \mathbf{y}, \mathbf{w}) := (k(t, s, \mathbf{x}_i, y_i, w_i))$ . This Volterra integro–differential equation problem is an example of the form (1.1) where the right-hand side is rather expensive, because each evaluation of  $\mathbf{v}(t)$  requires the integration of the initial value problem in (1.5) from  $s = t_0$  until  $s = t$ . Furthermore, since  $L$  is elliptic, the spectrum of its discretization  $D$  is expected to be negative.

**Example 1.1.** The mathematical model for the evolution of a community of species (or population) that is allowed to diffuse spatially is described by (cf. [4, p. 6 and p. 183])

$$\begin{aligned} \frac{\partial N(t, x)}{\partial t} &= \frac{\partial^2 N(t, x)}{\partial x^2} + g(t, x) + N(t, x) \left( 1 - \int_{t_0}^t N(s, x) K(t-s) \, ds \right), \\ K(t) &:= \frac{1}{T^2} t \exp\left(\frac{-t}{T}\right), \end{aligned} \quad (1.6)$$

where  $N$  is the size of the population,  $T$  is the point where the so-called “strong” generic delay kernel  $K(t)$  assumes its maximum, and  $g(t, x)$  represents external influences. Note that the “birth term”  $N(t, x)$  is considered as part of the function  $\mathbf{v}(t)$  occurring in (1.1).

### 1.2. Fredholm integro–differential equations

A second class of problems consists of partial integro–differential equations of Fredholm type

$$\frac{\partial u(t, \mathbf{x})}{\partial t} = Lu(t, \mathbf{x}) + g(t, \mathbf{x}) + \int_{\Omega} k(t, \mathbf{x}, \xi, u(t, \xi)) \, d\xi, \quad \mathbf{x} \in \Omega, \tag{1.7}$$

where again  $L$  is an elliptic operator and  $\Omega$  is a domain in  $\mathbb{R}^d$ . Replacing  $\Omega$  by a set of  $r$  points  $\mathbf{x}_i$  and  $u(t, \mathbf{x})$  by the set of  $r$  values  $u(t, \mathbf{x}_i)$ , and defining  $\mathbf{y}(t) = (y_i(t))$  where  $y_i(t)$  represents an approximation to  $u(t, \mathbf{x}_i)$ , this equation can be discretized resulting in the  $r$ -dimensional system

$$\frac{dy_i(t)}{dt} = \mathbf{d}_i(t)^T \mathbf{y}(t) + g_i(t) + \sum_{j=1}^r w_j k(t, \mathbf{x}_i, \mathbf{x}_j, y_j(t)), \quad i = 1, \dots, r, \tag{1.7'}$$

where  $w_j$  is the quadrature weight associated with  $\mathbf{x}_j$ . This system is of the form (1.1) with

$$\mathbf{v}(t) = \mathbf{g}(t) + \mathbf{K}(t, \mathbf{y}(t)) := \left( g_i(t) + \sum_{j=1}^r w_j k(t, \mathbf{x}_i, \mathbf{x}_j, y_j(t)) \right). \tag{1.8}$$

Again, we see that the function  $\mathbf{v}(t)$  is usually quite expensive.

**Example 1.2.** The behaviour in time of the temperature distribution above the earth can be modeled by an equation of the form (1.7), that is by (cf. [14])

$$\frac{\partial u(t, x)}{\partial t} = \frac{\partial^2 u(t, x)}{\partial x^2} - \int_0^1 \frac{u^4(t, \xi)}{(1 + |x - \xi|)^2} \, d\xi, \quad 0 \leq x \leq 1. \tag{1.9}$$

In this paper, we are interested in explicit numerical integration methods requiring only a few right-hand side evaluations. Sections 2–4 will describe such methods in the case of parabolic integro–differential equations of Volterra type. In Section 5, we show how a similar approach can be used for solving parabolic integro–differential equations of Fredholm type.

## 2. Stabilized forward Euler methods

A family of integration methods for (1.1), requiring only *one* right-hand side evaluation, is given by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + hS(hD_{n+1/2})(D_{n+1/2}\mathbf{y}_n + \mathbf{v}(t_n + \frac{1}{2}h)), \quad D_{n+1/2} := D(t_n + \frac{1}{2}h), \tag{2.1}$$

where  $S(x)$  is a polynomial or rational function. In fact, if  $\mathbf{v}(t)$  is a given function, then  $S(x)$  can be expressed in terms of the *stability function* of (2.1). Here, the stability function  $R(x)$  of a method is

defined by the relation  $\mathbf{y}_{n+1} = R(h\lambda)\mathbf{y}_n$  which is obtained if the method is applied to the test equation  $\mathbf{y}' = \lambda\mathbf{y}$ , where  $\lambda$  is a scalar parameter. If  $|R(h\lambda)| \leq 1$ , then the numerical solution values  $\{\mathbf{y}_n\}$  remain bounded as  $n \rightarrow \infty$ . It is easily seen that application of (2.1) to the above test equation yields the relation  $\mathbf{y}_{n+1} = (I + h\lambda S(h\lambda))\mathbf{y}_n$ , so that  $R$  and  $S$  are related according to  $R(x) = 1 + xS(x)$ . Hence,  $S$  should be such that  $|R|$  is bounded by 1 on a large, negative interval on the  $x$ -axis. If we can determine  $S$  in such a way, then  $S(hD_{n+1/2})$  can be interpreted as a stabilizing matrix. The use of stabilizing or smoothing matrices for relaxing the stability (or convergence) conditions in numerical methods for partial differential equations has been proposed in many papers. Usually, smoothing matrices are used for *residue smoothing* in iterative solvers (see, e.g., [6,7,9,10,13]), or for *right-hand side smoothing* (e.g., in [16–18]). The method (2.1) may be considered as a forward Euler method in which the right-hand side of the IVP is stabilized (or smoothed) by the matrix  $S$ . Therefore, we shall call (2.1) a *stabilized forward Euler method*.

Most methods employing right-hand side smoothing are only first-order accurate. An advantage of the stabilized Euler method (2.1) is that it can easily be made second-order accurate. To see this, we substitute the exact solution  $\mathbf{y}$  into (2.1) and we expand at the point  $t_{n+1/2}$  to find the residual term

$$\begin{aligned} h\mathbf{y}' + \frac{1}{24}h^3\mathbf{y}''' - h(S(0)I + hS'(0)D + \frac{1}{2}h^2S''(0)D^2)(D(\mathbf{y} - \frac{1}{2}h\mathbf{y}' + \frac{1}{8}h^2\mathbf{y}'') + \mathbf{v}) + O(h^4) \\ = h\mathbf{y}' + \frac{1}{24}h^3\mathbf{y}''' - h(S(0)I + hS'(0)D + \frac{1}{2}h^2S''(0)D^2)(\mathbf{y}' - \frac{1}{2}hD\mathbf{y}' + \frac{1}{8}h^2D\mathbf{y}'') + O(h^4) \\ = h(1 - S(0))\mathbf{y}' + h^2D(\frac{1}{2}S(0) - S'(0))\mathbf{y}' \\ + \frac{1}{24}h^3(\mathbf{y}''' - 3S(0)D\mathbf{y}'' + 12(S'(0) - S''(0))D^2\mathbf{y}') + O(h^4), \end{aligned} \quad (2.2)$$

where  $\mathbf{y}$  and its derivatives are all evaluated at  $t_{n+1/2}$ . Hence, the stabilized forward Euler method is second-order accurate if  $S(0) = 1 + O(h^2)$  and  $S'(0) = 1/2 + O(h)$ . It is not possible to achieve third-order accuracy by a special choice of  $S$ . In the case where  $\mathbf{v}(t)$  originates from a Volterra integral term as in (1.5), we assume that the ODE solver used for integrating (1.5) is at least second-order accurate, to achieve second-order overall accuracy [2, p. 151]. Note that the conditions on  $S$  imply that the stability function is second-order consistent, i.e.,  $R(0) = 1$ ,  $R'(0) = 1 + O(h^2)$ ,  $R''(0) = 1 + O(h)$ .

**Remark 2.1.** Instead of (1.1), we may also consider the more general IVP

$$\frac{d\mathbf{y}(t)}{dt} = \mathbf{f}(t, \mathbf{y}(t)), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \quad \mathbf{y} \in \mathbb{R}^r, \quad (1.1')$$

where  $\mathbf{f}(t, \mathbf{y})$  is an arbitrary (expensive) right-hand side function. The stabilized forward Euler method (2.1) takes the form

$$\mathbf{y}_{n+1} = \mathbf{y}_n + hS(hJ)\mathbf{f}(t_n + \frac{1}{2}h, \mathbf{y}_n), \quad (2.1')$$

where  $J$  is an approximation to the Jacobian  $J_n := \partial\mathbf{f}(t_n, \mathbf{y}_n)/\partial\mathbf{y}$ . It can be shown that it is second-order accurate if  $S(0) = 1$ ,  $S'(0) = 1/2$  and  $J = J_n + O(h)$ .

Next, we investigate the stability of (2.1) in the case where  $\mathbf{v}(t)$  is defined by (1.5). Let us apply (2.1) to the familiar integro-differential test equation of Brunner and Lambert [3]

$$\frac{dy(t)}{dt} = \xi y(t) + \eta \int_{t_0}^t y(\tau) d\tau, \quad y(t_0) = y_0, \quad (2.3)$$

equivalently, to

$$\frac{dy(t)}{dt} = \xi y(t) + z(t), \quad y(t_0) = y_0; \quad \frac{dz(t)}{dt} = \eta y(t), \quad z(t_0) = 0. \tag{2.4}$$

where,  $\xi$  and  $\eta$  represent eigenvalues of  $D_{n+1/2}$  and  $(\partial/\partial \mathbf{y} + \partial/\partial \mathbf{w})\mathbf{K}(t, s, \mathbf{y}, \mathbf{w})$ , respectively. It will be assumed that both  $\xi$  and  $\eta$  are negative. The test equation (2.4) results into the relation

$$y_{n+1} = y_n + hS(h\xi)(\xi y_n + z_{n+1/2}). \tag{2.5}$$

A relation for  $z_{n+1/2}$  can be obtained by specifying the integration method for (1.5). We separately discuss the explicit midpoint rule and the two-step backward differentiation formula.

*1. Explicit midpoint rule*

Suppose that the differential equation in (1.5) is integrated by the explicit midpoint rule

$$z_{\nu+1/2} = z_{\nu-1/2} + h\mathbf{K}(t_{n+1/2}, t_\nu, \mathbf{y}_{n+1/2}, \mathbf{y}_\nu), \quad \nu = 1, \dots, n.$$

where,  $\mathbf{y}_{n+1/2}$  should be approximated by means of the step point values  $\mathbf{y}_\nu$ . To avoid the solution of implicit relations, we use extrapolation (rather than interpolation), to obtain the modified midpoint value

$$z_{\nu+1/2} = z_{\nu-1/2} + h\mathbf{K}(t_{n+1/2}, t_\nu, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1}), \mathbf{y}_\nu), \quad \nu = 1, \dots, n. \tag{2.6}$$

For the test equation (2.4) we are led to the recursion

$$\begin{aligned} y_{n+1} - hS(h\xi)z_{n+1/2} &= (I + h\xi S(h\xi))y_n, \\ z_{n+1/2} &= h\eta y_n + z_{n-1/2} \end{aligned}$$

we remark that the same recursion results if  $\mathbf{y}_{n+1/2}$  would have been approximated by interpolation of the values  $\mathbf{y}_n$  and  $\mathbf{y}_{n+1}$ . It is easily verified that the characteristic equation is given by

$$\zeta^2 - [R + Q]\zeta + R = 0, \quad R := 1 + h\xi S(h\xi), \quad Q := 1 + h^2\eta S(h\xi).$$

Since  $R$  and  $Q$  are real, it follows that the eigenvalues  $\zeta$  are within the unit circle if  $R < 1$ ,  $Q < 1$  and  $2R + Q > -1$ . In the  $(h\xi, h^2\eta)$ -plane, this stability region is given by

$$-\beta_{\text{real}} < h\xi < 0, \quad 2h\xi \frac{1 + R(h\xi)}{1 - R(h\xi)} < h^2\eta < 0, \tag{2.7}$$

where  $\beta_{\text{real}}$  is the real stability boundary associated with the stability function  $R$ . Evidently, the function should be bounded away from  $-1$ , otherwise negative values of  $\eta$  are not allowed.

*2. Backward differentiation formula*

Next we show that integrating (1.5) by a backward differentiation formula (BDF) allows that the stability function  $R$  assumes values close to  $-1$ . Applying the two-step formula

$$z_{\nu+1/2} = \frac{4}{3}z_{\nu-1/2} - \frac{1}{3}z_{\nu-3/2} + \frac{2}{3}h\mathbf{K}(t_{n+1/2}, t_{\nu+1/2}, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1}), \frac{1}{2}(\mathbf{y}_\nu + \mathbf{y}_{\nu+1})), \tag{2.8}$$

for  $\nu = 2, \dots, n$ , and proceeding as in the preceding subsection, we derive for the test equation

$$\begin{aligned} y_{n+1} - hS(h\xi)z_{n+1/2} &= (I + h\xi S(h\xi))y_n, \\ -h\eta y_{n+1} + 3z_{n+1/2} &= h\eta y_n + 4z_{n-1/2} - z_{n-3/2}, \end{aligned}$$

with characteristic equation

$$\begin{aligned} [4 - Q]\zeta^3 - [3 + 3R + Q]\zeta^2 + [1 + 4R]\zeta - R &= 0, \\ R &:= 1 + h\xi S(h\xi), \quad Q := 1 + h^2\eta S(h\xi). \end{aligned} \quad (2.9)$$

We derive from the Hurwitz criterion the following stability region in the  $(R, Q)$ -plane:

$$\begin{aligned} R > -1, \quad Q < 1, \quad R + Q < 2, \quad 2R + Q < 7, \\ 2R^2 + 5RQ + Q^2 - 13R - 7Q > -12. \end{aligned} \quad (2.10)$$

Note that this region contains the strip  $-1 < R < 1$  and  $Q < 1$ , so that nonzero values of  $\eta$  are allowed when  $R$  is close to  $-1$ . However, it should be remarked that the method (2.1) becomes implicit if  $\nu = n$  in (2.8). In fact, on substitution of  $z(t_n + \frac{1}{2}h) = z_{n+1/2}$  into (2.1), we obtain for  $\mathbf{y}_{n+1}$

$$\begin{aligned} \mathbf{y}_{n+1} &= \frac{2}{3}h^2S(hD_{n+1/2})\mathbf{K}(t_{n+1/2}, t_{n+1/2}, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1}), \frac{1}{2}(\mathbf{y}_n + \mathbf{y}_{n+1})) + \mathbf{b}_n, \\ \mathbf{b}_n &:= \mathbf{y}_n + \frac{1}{3}hS(hD_{n+1/2})(3D_{n+1/2}\mathbf{y}_n + 4z_{n-1/2} - z_{n-3/2}). \end{aligned} \quad (2.11)$$

We solve (2.11) by fixed point iteration, i.e.,

$$\mathbf{y}_{n+1}^{(j)} = \frac{2}{3}h^2S(hD_{n+1/2})\mathbf{K}(t_{n+1/2}, t_{n+1/2}, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1}), \frac{1}{2}(\mathbf{y}_n + \mathbf{y}_{n+1}^{(j-1)})) + \mathbf{b}_n. \quad (2.12)$$

The iteration error  $\mathbf{e}_{n+1}^{(j)} := \mathbf{y}_{n+1} - \mathbf{y}_{n+1}^{(j)}$  satisfies, in a first approximation, the relation

$$\mathbf{e}_{n+1}^{(j)} = \frac{1}{3}h^2S(hD_{n+1/2})K'\mathbf{e}_{n+1}^{(j-1)}, \quad K' := \frac{\partial}{\partial \mathbf{w}}\mathbf{K}(t_{n+1/2}, t_{n+1/2}, \mathbf{y}_n, \mathbf{y}_n).$$

For the test equation (2.4), this recursion reduces to

$$\mathbf{e}_{n+1}^{(j)} = \frac{1}{3}h^2\eta S(h\xi)\mathbf{e}_{n+1}^{(j-1)} = \frac{1}{3}(Q - 1)\mathbf{e}_{n+1}^{(j-1)}.$$

Hence, we should add to (2.10) the convergence condition  $-2 < Q < 4$ . Note that *nonstiff* error components (i.e.,  $|h\xi|$  small) are amplified by a factor  $h^2|\eta|/3$  and *stiff* error components (i.e.,  $|h\xi|$  large) by a factor less than  $2h^2|\eta|/(3|h\xi|)$ . Thus, for moderate values of the eigenvalues  $\eta$  we may expect fast convergence of the iteration process (2.12). In the  $(R, Q)$ -plane, the stability-convergence region always contains the region defined by  $-1 < R < 1$  and  $-2 < Q < 1$ . In the  $(h\xi, h^2\eta)$ -plane this stability-convergence region is determined by

$$-\beta_{\text{real}} < h\xi < 0, \quad \frac{3h\xi}{1 - R(h\xi)} < h^2\eta < 0, \quad (2.13)$$

showing that stability functions assuming values close to  $-1$  can be used.

**Remark 2.2.** In (2.8) the values of  $z(t, t)$  are computed halfway the step points  $t_n$ . Alternatively, we may compute them at the step points to obtain the integration method

$$\begin{aligned} z_{\nu+1} &= \frac{4}{3}z_{\nu} - \frac{1}{3}z_{\nu-1} + \frac{2}{3}hK(t_{n+1/2}, t_{\nu+1}, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1}), \mathbf{y}_{\nu+1}), \quad \nu = 2, \dots, n, \\ z_{n+1/2} &= \frac{1}{2}(z_n + z_{n+1}). \end{aligned} \quad (2.8')$$

It turns out that this approach leads to the same stability and convergence conditions. To see this we apply (2.8') to the test equation:

$$\begin{aligned} 2y_{n+1} - hS(h\xi)z_{n+1} &= 2(1 + h\xi S(h\xi))y_n + hS(h\xi)z_n, \\ -2h\eta y_{n+1} + 3z_{n+1} &= 4z_n - z_{n-1}. \end{aligned}$$

This recursion also possesses the characteristic equation (2.9). Hence, the stability region is again determined by (2.10). The equation for  $y_{n+1}$  becomes

$$\begin{aligned} y_{n+1} &= \frac{1}{3}h^2S(hD_{n+1/2})K(t_{n+1/2}, t_{n+1}, \frac{1}{2}(3y_n - y_{n-1}), y_{n+1}) + b_n, \\ b_n &:= y_n + \frac{1}{6}hS(hD_{n+1/2})(6D_{n+1/2}y_n + 7z_n - z_{n-1}). \end{aligned}$$

Solving this equation by fixed point iteration yields the same convergence condition as for (2.12).

**Remark 2.3.** We can remove the implicitness from (2.11) by replacing in (2.8) the last argument of  $K$  by the extrapolated value  $\frac{1}{2}(3y_n - y_{n-1})$ . However, then the stability polynomial  $R$  is again required to be bounded away from  $-1$ . This can be deduced from the characteristic equation

$$3\zeta^3 - [3R + 3Q + 1]\zeta^2 + [4R + Q]\zeta - R = 0$$

which leads to the stability region

$$\begin{aligned} 2R + Q > -1, \quad Q < 1, \quad 2R - Q < 5, \quad R + Q < 2, \\ 2R^2 + 3RQ - 11R - 3Q > -9. \end{aligned} \tag{2.14}$$

The first inequality requires  $\eta$  to be nonnegative if  $R$  approximates  $-1$ .

### 3. Explicit Euler–Chebyshev methods

We first consider the case where the stability function  $R$  is a polynomial. The stability polynomials satisfying  $|R| \leq 1$  on large negative intervals  $[-\beta_{\text{real}}, 0]$  proposed in the literature involve the Chebyshev polynomials

$$T_m(x) := \cos(m \arccos(x)). \tag{3.1}$$

We shall consider the following cases of second-order consistent stability polynomials:

$$R(x) = \frac{1}{3m^2} \left( 2m^2 + 1 + (m^2 - 1)T_m \left( 1 + \frac{3x}{m^2 - 1} \right) \right), \tag{3.2a}$$

$$\beta_{\text{real}} = \frac{2}{3}(m^2 - 1) \approx \frac{2}{3}m^2,$$

$$R(x) = \frac{1}{2-x} \left( 2 - xT_m(\cos(\pi/m) + \frac{1}{2}x(1 - \cos(\pi/m))) \right), \tag{3.2b}$$

$$\beta_{\text{real}} = \frac{2}{(\tan(\pi/2m))^2} \approx \frac{4}{5}m^2.$$

These polynomials have been discovered in [1] and [12], respectively. Note that the *effective* stability boundary  $m^{-1}\beta_{\text{real}}$  increases (almost) linearly with  $m$ , so that the  $h\xi$ -condition in (2.7) and (2.13)

allows increasingly larger *effective* stepsizes (scaled with respect to the amount of work per step). We shall use the stability polynomials (3.2) in the method (2.1) by defining the stabilizing matrix

$$S(hD_{n+1/2}) = \varepsilon(W - I)^{-1}(T_m(W) - I), \quad (3.3)$$

where  $\{\varepsilon, W\}$  is respectively given by

$$\{\varepsilon, W\} := \left\{ \frac{1}{m^2}, I + \frac{3hD_{n+1/2}}{m^2 - 1} \right\}, \quad (3.4a)$$

$$\{\varepsilon, W\} := \left\{ \frac{1}{2}(1 - \cos(\pi/m)), \cos(\pi/m)I + \frac{1}{2}(1 + \cos(\pi/m))hD_{n+1/2} \right\}. \quad (3.4b)$$

The methods  $\{(2.1), (3.3)\}$  may be considered as a matrix version of the Runge–Kutta–Chebyshev methods designed in [11] (see also [8]). Stability plots corresponding to (3.2a) and (3.2b) can be computed by means of (2.7) and (2.13), and are given in Section 3.2.

### 3.1. Computation of $S(hD_{n+1/2})$

The computation of the stabilizing matrix  $S(hD_{n+1/2})$  can efficiently be done by using the following recursive relations satisfied by the Chebyshev polynomials:

$$T_{j+1}(w) = 2wT_j(w) - T_{j-1}(w), \quad T_{2j}(w) = 2(T_j(w))^2 - 1. \quad (3.5)$$

From these relations, we can derive the following lemma (the proof of part (a) is straightforward, that of part (b) can be found in [10]):

**Lemma 3.1.** *Let  $W - I$  be nonsingular, and define  $S_m := (W - I)^{-1}(T_m(W) - I)$ .*

(a) *For all  $m \geq 1$ ,  $S_m$  can be generated by the recursion*

$$S_1 = I, \quad S_2 = 2(W + I), \quad S_j = 2WS_{j-1} - S_{j-2} + 2I, \quad j = 3, \dots, m. \quad (3.6)$$

(b) *If  $m = 2^q$ , then*

$$S_m = F_q \cdot F_{q-1} \cdot \dots \cdot F_1, \quad F_1 = 2(W + I), \quad F_j = (2I - F_{j-1})^2, \quad j = 2, \dots, q. \quad (3.7)$$

By virtue of this lemma, the stabilizing matrix  $S(hD_{n+1/2})$  occurring in (2.1) can be computed by defining  $\{\varepsilon, W\}$  according to (3.4), by performing either the recursion (3.6) or (3.7) to obtain  $S_m$ , and by setting  $S(hD_{n+1/2}) = \varepsilon S_m$ . Part (a) of this lemma presents the conventional way of computing the matrix  $S_m$  and requires  $m - 2$  matrix–matrix multiplications. However, if we allow  $m$  to be a power of 2, then we can use part (b) of the lemma, requiring only  $2q - 2$  matrix–matrix multiplications. Note that if the matrix  $D$  in (1.1) does not depend on  $t$ , then the stabilizing matrix  $S$  is independent of  $n$ , so that it needs only to be recomputed if  $h$  changes.

It should be remarked that instead of computing the matrix  $S_m$ , we may also recursively compute for any given vector  $\mathbf{a}$ , the “stabilized” vector  $\mathbf{a}_m = S_m \mathbf{a}$ . This can be achieved by the recursion

$$\mathbf{a}_1 = \mathbf{a}, \quad \mathbf{a}_2 = 2(W + I)\mathbf{a}, \quad \mathbf{a}_j = 2W\mathbf{a}_{j-1} - \mathbf{a}_{j-2} + 2\mathbf{a}, \quad j = 3, \dots, m. \quad (3.6')$$

This approach is usually more storage economic than using (3.6) or (3.7). Unfortunately, it seems not possible to convert the matrix recursion (3.7) into a vector recursion.



### 3.2. Stability

For the cases where the quadrature term is computed by means of the explicit midpoint rule and the backward differentiation formulas (2.6) and (2.8), the stability(-convergence) region in the quarter plane  $\{h\xi \leq 0, h^2\eta \leq 0\}$  is determined by the conditions (2.7) and (2.13). Figs. 1–3 present plots for  $m = 4$  and  $m = 16$ . In these plots,  $h\xi$  is on the horizontal axis and the shaded region corresponds to the stability region.

The derivation of explicit stability conditions from (2.7) and (2.13) is discussed in the following subsections.

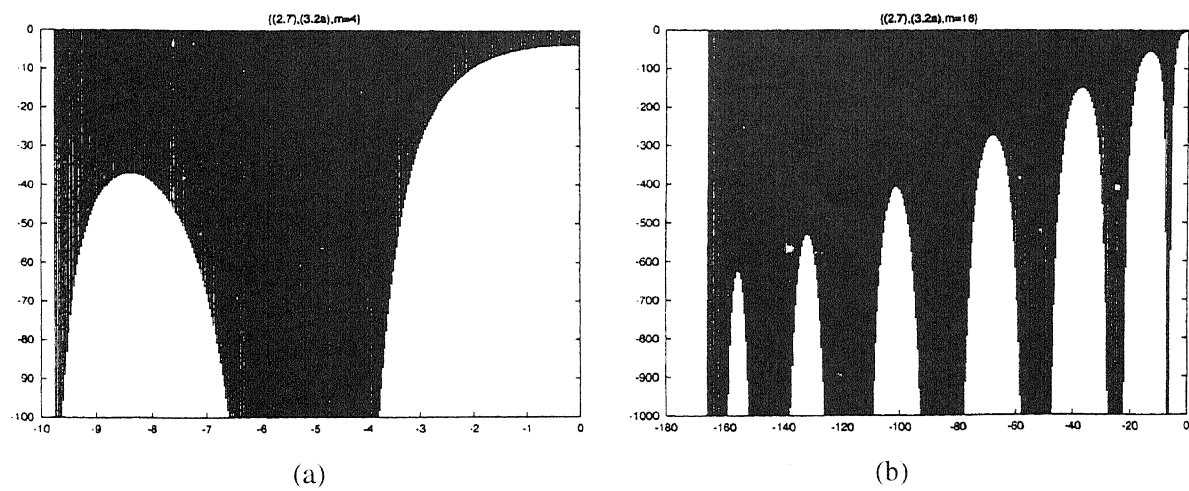


Fig. 1. (a) The case  $\{(2.7), (3.2a), m = 4\}$ . (b) The case  $\{(2.7), (3.2a), m = 16\}$ .

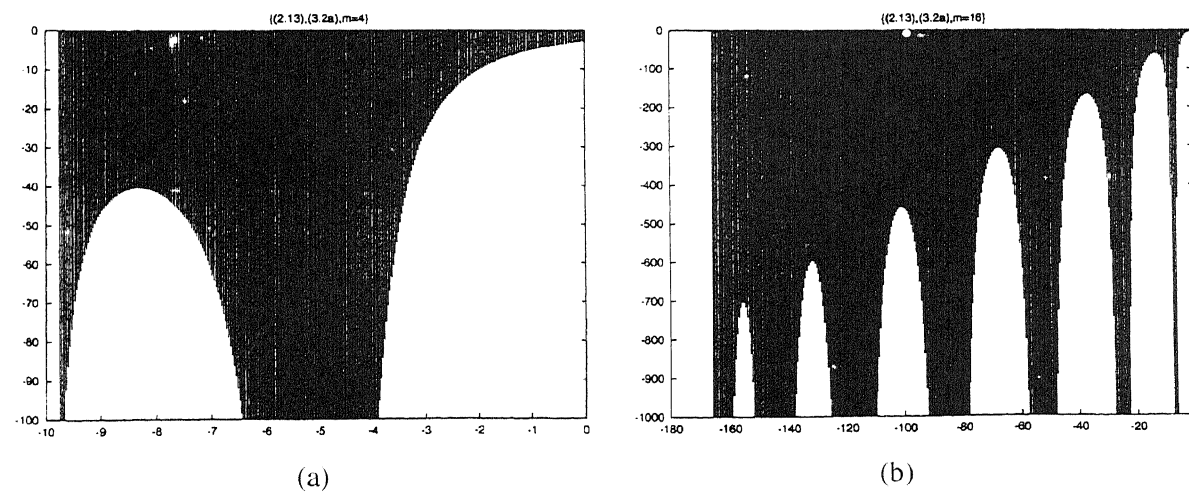


Fig. 2. (a) The case  $\{(2.13), (3.2a), m = 4\}$ . (b) The case  $\{(2.13), (3.2a), m = 16\}$ .

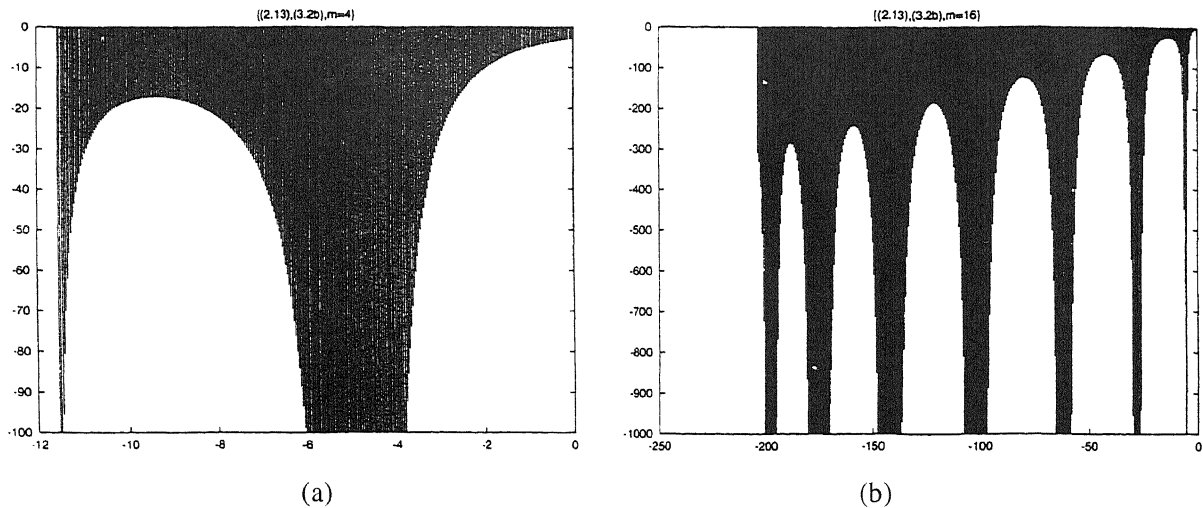


Fig. 3. (a) The case  $\{(2.13), (3.2b), m = 4\}$ . (b) The case  $\{(2.13), (3.2b), m = 16\}$ .

### 3.2.1. Explicit midpoint rule

In the case (3.2a), we have  $\frac{1}{3} \leq R \leq 1$  in the interval  $-\frac{2}{3}m^2 \leq h\xi \leq 0$ . Hence, in all points where  $R$  approximates the value  $\frac{1}{3}$ , we have to satisfy the condition  $4h\xi < h^2\eta < 0$ . Furthermore, for  $h\xi = 0$ , we have  $-4 < h^2\eta < 0$ . Thus, we may conclude from (2.7) that the stability region contains a polygonal region with  $(-\frac{2}{3}m^2, 0)$ ,  $(0, 0)$ ,  $(0, -4)$ ,  $(-1, -4)$  and  $(-\frac{2}{3}m^2, -\frac{8}{3}m^2)$  as its corner points. Suppose that  $m$  is prescribed and that  $h$  is chosen as large as allowed by the condition  $-\beta_{\text{real}} < h\xi < 0$  with  $\beta_{\text{real}} \approx \frac{2}{3}m^2$  (cf. (3.2)), i.e.,

$$h = h_{\text{max}} := \frac{\beta_{\text{real}}}{\rho(D)}, \quad (3.8)$$

where  $\rho(D)$  denotes the spectral radius of  $D$ . Furthermore, let  $\mu(D)$  denote the minimal magnitude of the eigenvalues  $\xi$  of  $D$ . Then, it is easily verified that for  $h\mu(D) \leq 1$ , we should satisfy  $h^2\eta > -4$ , and for  $h\mu(D) \geq 1$ , we have to satisfy  $h^2\eta > -4h\mu(D)$ . On substitution of (3.8), we are led to an upper bound for  $|\eta|$  as listed in Table 1.

If (3.2b) is used, we have  $(2 + h\xi)(2 - h\xi)^{-1} \leq R \leq 1$  in the interval  $-\frac{4}{5}m^2 \leq h\xi \leq 0$ , so that the minimal values of  $R$  lie on the curve  $(2 + h\xi)(2 - h\xi)^{-1}$ . Upon substitution into (2.7), we find that the stability region contains the rectangle  $\{-\frac{4}{5}m^2 \leq h\xi \leq 0, -4 < h^2\eta < 0\}$ . Again choosing  $h$  according to (3.8), we obtain the upper bound for  $|\eta|$  as listed in Table 1.

### 3.2.2. Backward differentiation formula

Using the stability polynomial (3.2a), the condition (2.13) yields a domain that contains a polygone with corner points  $(-\frac{2}{3}m^2, 0)$ ,  $(0, 0)$ ,  $(0, -3)$ ,  $(-\frac{2}{3}, -3)$ , and  $(-\frac{2}{3}m^2, -3m^2)$ . Proceeding as in the previous subsection, we find the upper bounds for  $|\eta|$  as given in Table 1.

If the stability polynomial (3.2b) is used, we obtain a polygone with corner points  $(-\frac{4}{5}m^2, 0)$ ,  $(0, 0)$ ,  $(0, -3)$ ,  $(-2, -3)$  and  $(-\frac{4}{5}m^2, -\frac{6}{5}m^2)$ . For  $h\mu(D) \leq 2$ , we require  $h^2\eta > -3$ , whereas

Table 1  
Maximal stable values of  $|\eta|$  for explicit Euler–Chebyshev methods with  $h = h_{\max} = \beta_{\text{real}}/\rho(D)$

$R(x)$	Volterra term by midpoint rule (cf. (2.7))		Volterra term by BDF (cf. (2.13))	
(3.2a)	$\frac{4}{h_{\max}^2}$	if $h_{\max} \leq \frac{1}{\mu(D)}$	$\frac{3}{h_{\max}^2}$	if $h_{\max} \leq \frac{2}{3\mu(D)}$
	$\frac{4\mu(D)}{h_{\max}}$	if $h_{\max} \geq \frac{1}{\mu(D)}$	$\frac{9\mu(D)}{2h_{\max}}$	if $h_{\max} \geq \frac{2}{3\mu(D)}$
(3.2b)	$\frac{4}{h_{\max}^2}$		$\frac{3}{h_{\max}^2}$	if $h_{\max} \leq \frac{2}{\mu(D)}$
			$\frac{3\mu(D)}{2h_{\max}}$	if $h_{\max} \geq \frac{2}{\mu(D)}$

$h^2\eta > -\frac{3}{2}h\mu(D)$  has to be satisfied if  $h\mu(D) \geq 2$ . The resulting upper bounds for  $|\eta|$  are listed in Table 1.

### 3.3. Application to model problems

Consider the model problem where  $L$  is the  $d$ -dimensional Laplace operator and let  $D_{n+1/2}$  be the standard symmetric  $(2d + 1)$ -point discretization on a uniform grid with mesh size  $\Delta$  and with Dirichlet boundary conditions. This discretization will be denoted by  $D$ . For example, if  $d = 1$ , then

$$D = \frac{1}{\Delta^2} \begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \end{pmatrix}. \tag{3.9}$$

By Gerschgorin’s theorem we find that  $\rho(D) = 4d\Delta^{-2}$ , so that by choosing  $h$  according to (3.8)

$$h = \frac{c}{4d}m^2\Delta^2, \tag{3.8'}$$

where  $c$  is a constant given by  $2/3$  and  $4/5$  for the polynomials (3.2a) and (3.2b), respectively. Notice that for a given spatial mesh size  $\Delta$ , the right-hand side can be made as large as we want by choosing  $m$  sufficiently large.

In [10] it was pointed out that in the above model situation, the factor matrices  $F_j$  in (3.7) turn out to be rather sparse, so that (3.7) can be used without extreme storage requirements.

Let  $D_{n+1/2}$  be given by (3.9), let  $m = 2^q$ , and let  $R(x)$  be defined by (3.2a). From (3.4a) it follows that  $W = I + \frac{1}{2}\Delta^2 D_{n+1/2}$ , so that (3.7) yields

$$F_1 = \begin{pmatrix} 2 & 1 & & & \\ 1 & 2 & 1 & & \\ & 1 & 2 & 1 & \\ & & \ddots & \ddots & \ddots \end{pmatrix}, \quad F_2 = \begin{pmatrix} 1 & 0 & 1 & & & \\ 0 & 2 & 0 & 1 & & \\ 1 & 0 & 2 & 0 & 1 & \\ & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}, \quad \dots \tag{3.10}$$



4.1. Direct linear solvers

If a direct linear solver is used, we may adopt (4.1) as the stability function of the method. For the explicit midpoint rule, we find on substitution of (4.1) into (2.7) that in the quarter plane  $\{h\xi \leq 0, h^2\eta \leq 0\}$  the stability region is given by an infinite strip along the negative  $h\xi$ -axis defined by  $-4 < h^2\eta < 0$ . A comparison with the stability region associated with the explicit Euler-Chebyshev methods generated by the stability polynomials (3.2) and observing that the stepsizes of these methods can be chosen as large as we want by choosing  $m$  sufficiently large reveals that the explicit Euler-Chebyshev methods allow values of  $-\eta$  that are equal to or larger than those allowed by the method generated by (4.1).

Similarly, we find on substitution of (4.1) into (2.13) that for the backward differentiation formula, the stability region is given by the infinite wedge  $-3 + \frac{3}{2}h\xi < h^2\eta < 0$ . This leads to the same conclusion as for the explicit midpoint rule.

4.2. Chebyshev semi-iteration process

Let  $Ax = g$  be the system to be solved, where  $A$  has its eigenvalues in the positive interval  $[p, q]$ . Then the Chebyshev semi-iteration process is defined by (see, e.g., [5])

$$\begin{aligned} x_1 &= x_0 + b_0(Ax_0 - g), & x_{j+1} &= a_j x_j + (1 - a_j)x_{j-1} + b_j(Ax_j - g), & j \geq 1, \\ b_0 &= \frac{w_1}{w_0}, & a_j &= 2w_0 \frac{T_j(w_0)}{T_{j+1}(w_0)}, & b_j &= 2w_1 \frac{T_j(w_0)}{T_{j+1}(w_0)}, & j \geq 1, \\ w_0 &:= -\frac{p+q}{p-q}, & w_1 &:= \frac{2}{p-q}, \end{aligned} \tag{4.4}$$

where  $x_0$  is the initial approximation and where  $T_j$  is again the Chebyshev polynomial of the first kind. Let us apply (4.4) to (4.3) and suppose that  $m$  iterations are performed with  $y_n$  as the initial approximation to  $y_{n+1}$ . Then, it can be verified that the resulting approximation to  $y_{n+1}$  is of the form (2.1) with

$$\begin{aligned} S(hD_{n+1/2}) &:= w_1(W - w_0I)^{-1} \left( I - \frac{T_m(W)}{T_m(w_0)} \right), \\ W &:= w_0I + w_1 \left( I - \frac{1}{2}hD_{n+1/2} \right), \\ w_0 &:= \frac{4 + h(\rho(D) + \mu(D))}{h(\rho(D) - \mu(D))}, & w_1 &:= -\frac{4}{h(\rho(D) - \mu(D))}, \end{aligned} \tag{4.5}$$

where  $\rho(D)$  and  $\mu(D)$  are defined as in the preceding section. Thus, after  $m$  iterations, the stability polynomial  $R(x) = 1 + xS(x)$  is given by

$$R(x) = \frac{1}{2-x} \left( 2 + x - 2x \frac{T_m(w_0 + w_1 - w_1x/2)}{T_m(w_0)} \right). \tag{4.6}$$

Since  $R(0) = 1$ ,  $R'(0) = 1 + O(h^m)$  and  $R''(0) = 1 + O(h^{m-1})$ , the method  $\{(2.1), (4.5)\}$  is second-order accurate within two iterations ( $m \geq 2$ ). Furthermore, it follows from (4.6) that in the interval  $-h\rho(D) \leq x \leq -h\mu(D)$ ,  $R(x)$  is bounded by the curves  $(2 + x \pm 2x/T_m(w_0))/(2 - x)$ . By

Table 2  
Effective stability boundaries for the method {(2.1), (4.5)}

$\alpha$	$m = 1$	$m = 2$	$m = 4$	$m = 6$	$m = 8$	$m = 10$	$m = 12$	$m = 15$	$m = 20$	$m = 25$
0	4.00	1.86	1.00	0.73	0.59	0.51	0.45	0.39	0.33	0.29
0.010	4.04	1.90	1.05	0.80	0.68	0.61	0.58	0.56	0.58	0.69
0.025	4.10	1.97	1.14	0.92	0.84	0.84	0.89	1.09	2.00	5.12
0.050	4.21	2.09	1.32	1.19	1.29	1.60	2.25	4.64	22.78	139.39

requiring that these curves are between  $-1$  and  $+1$ , we obtain the condition  $-2T_m(w_0) < x < 0$ . Thus, we have stability if  $h$  satisfies the inequality

$$h\rho(D) < 2T_m\left(\frac{4 + h(\rho(D) + \mu(D))}{h(\rho(D) - \mu(D))}\right). \quad (4.7)$$

The corresponding upper bound for  $h\rho(D)$  can be interpreted as the real stability boundary  $\beta_{\text{real}}$  of the method {(2.1), (4.5)}. Suppose that  $\mu(D) = \alpha\rho(D)$ . Then  $\beta_{\text{real}}$  equals the largest root of the equation

$$\beta = 2T_m\left(\frac{4 + (1 + \alpha)\beta}{(1 - \alpha)\beta}\right). \quad (4.8)$$

In Table 2 the values of  $m^{-2}\beta_{\text{real}}$  are listed for a few values of  $\alpha$  and  $m$ . These figures show that for  $\alpha = 0$ , the values  $m^{-2}\beta_{\text{real}}$  decrease with  $m$ , but for all nonzero values of  $\alpha$  in this table,  $m^{-2}\beta_{\text{real}}$  starts to increase for sufficiently large values of  $m$ .

## 5. Euler–Chebyshev methods for Fredholm integro–differential equations

The Euler–Chebyshev methods developed in the preceding sections can directly be applied to parabolic integro–differential equations of Fredholm type. However, the stability properties do change. Let us consider the stability test equation

$$\frac{dy(t)}{dt} = \xi y(t) + v(t), \quad v(t) = \eta y(t), \quad y(t_0) = y_0, \quad (5.1)$$

where  $\xi$  and  $\eta$  represent eigenvalues of the matrices  $D_{n+1/2}$  and  $\partial\mathbf{K}/\partial\mathbf{y} := (w_j \partial k(t, \mathbf{x}_i, \mathbf{x}_j, y_j) / \partial y_j)$ , respectively. It will be assumed that  $\xi$  is negative and  $\eta$  is complex.

Applying (2.1) to (5.1) leads to the recursion (2.5) with  $z_{n+1/2}$  replaced by  $v_{n+1/2}$ . We shall consider the stability of this recursion in the case where  $v(t_n + \frac{1}{2}h)$  is computed by extrapolation of  $\mathbf{y}$ -values. Interpolation would be much more stable, but leads to implicit relations, the solution of which is rather costly, because of the expensive function  $\mathbf{K}$  (cf. (1.8)). This extrapolation procedure is similar to the approach of Verwer et al. in their IMEX schemes [15].

From (1.8) we see that  $v(t_n + \frac{1}{2}h)$  can be approximated by

$$\mathbf{v}_{n+1/2} = \mathbf{K}\left(t_n + \frac{1}{2}h, \frac{1}{2}(3\mathbf{y}_n - \mathbf{y}_{n-1})\right). \quad (5.2)$$

Table 3  
Numerical results obtained by Euler–Chebyshev methods

	$h = \frac{1}{5}$	$h = \frac{1}{10}$	$h = \frac{1}{20}$	$h = \frac{1}{40}$	$h = \frac{1}{80}$	$h = \frac{1}{160}$	$h = \frac{1}{320}$	$h = \frac{1}{640}$
Accuracy	$10^{-1.7}$	$10^{-2.5}$	$10^{-3.2}$	$10^{-3.8}$	$10^{-4.3}$	$10^{-4.6}$	$10^{-4.7}$	$10^{-4.7}$
(3.2a): $m =$	89	64	45	32	23	16	12	8
(3.2b): $m =$	80	57	40	29	20	15	11	8

Hence, for the test equation (5.1), we have  $v_{n+1/2} = \frac{1}{2}\eta(3y_n - y_{n-1})$ , so that we obtain the recursion

$$y_{n+1} = \left(1 + h\xi S(h\xi) + \frac{3}{2}h\eta S(h\xi)\right)y_n - \frac{1}{2}h\eta S(h\xi)y_{n-1},$$

with characteristic equation

$$\zeta^2 - \left(R + \frac{3}{2}\eta\xi^{-1}(R - 1)\right)\zeta + \frac{1}{2}\eta\xi^{-1}(R - 1) = 0, \quad R := 1 + h\xi S(h\xi). \tag{5.3}$$

The same stability functions  $R$  as used in the Volterra case can be employed, that is,  $R$  is defined by either (3.2a), or (3.2b), or (4.1) or (4.6). For stability of the resulting method, Eq. (5.3) should have its zeros within the unit circle. As for the Volterra case, we obtain the  $h\xi$ -condition  $-\beta_{\text{real}} < h\xi < 0$ . The condition involving  $\eta$  becomes a condition on  $h\eta$  rather than on  $h^2\eta$ . Moreover, where  $h\eta$  was assumed to be real in the Volterra case (corresponding to models of the form (1.2) with kernels  $k$  not depending on  $x$ , see, e.g., Example 1.1), we now have to assume that  $\eta$  is complex-valued because the kernel  $k$  in (1.8) is expected to depend on  $x$  (see Example 1.2).

### 6. Numerical experiments

We illustrate the performance of the explicit Euler–Chebyshev method by integrating the population dynamics problem (1.6) over the domain  $\{0 \leq x \leq 1, 0 \leq t \leq 2\}$ . We set  $T = 1$  and imposed homogeneous Dirichlet conditions at  $x = 0$  and  $x = 1$ . In order to check the accuracy of the method, we chose the function  $g$  such that the exact solution is given by  $N(t, x) = \exp(-t)\sin(\pi x)$ . We applied the Euler–Chebyshev method (2.1) generated by the stability polynomials (3.2a) and (3.2b) using the explicit midpoint rule (2.6) and the BDF (2.8) for computing the integral term. In these experiments, the stabilizing matrix  $S(hD_{n+1/2})$  was generated by the vector recursion (3.6’).

In order to compare the four methods, we applied these methods with the same stepsize  $h$  and we chose  $m$  as small as allowed by the stability condition  $-\beta_{\text{real}} < h\xi$ . It turned out that for a given integration step  $h$ , the four methods produced almost the same accuracies. In Table 3, the maximal absolute errors produced at the end point, together with the corresponding values of  $m$ , are listed for  $\Delta = 1/80$  and a sequence of integration steps. These figures show the second-order accuracy of the Euler–Chebyshev methods for relatively large time steps  $h$ . If  $h$  decreases below  $1/80$ , the spatial discretization error becomes dominant and the overall accuracy is largely determined by the spatial error.

In an actual implementation, it is recommended to choose  $q$  (rather than  $h$ ), put  $m = 2^q$ , define  $h$  according to (3.8’), and generate  $S(hD_{n+1/2})$  by means of the factor matrices  $F_j$  given by (3.10) and (3.11). For example, the results of Table 3 obtained by the generating polynomial (3.2a) with stepsizes  $h = 1/10, 1/40, 1/160$  and  $1/640$  require respectively  $q = 6, 5, 4$  and  $3$ , that is, the construction of

the stabilizing matrix  $S(hD_{n+1/2})$  requires respectively 6, 5, 4 and 3 matrix–vector multiplications by matrices that are essentially tridiagonal. Furthermore, since the quadrature rule used for evaluating the Volterra term does not affect the overall accuracy, it is recommended to use the explicit midpoint rule because of its easy implementation.

## References

- [1] M. Bakker, Analytic aspects of a minimax problem, Report TN 62, Mathematisch Centrum, Amsterdam (1971) (in Dutch).
- [2] H. Brunner and P.J. van der Houwen, *The Numerical Solution of Volterra Equations*, CWI Monographs 3 (North-Holland, Amsterdam, 1986).
- [3] H. Brunner and J.D. Lambert, Stability of numerical methods for Volterra integro–differential equations, *Computing* 12 (1974) 75–89.
- [4] J.M. Cushing, *Integro–Differential Equations and Delay Models in Population Dynamics*, Lecture Notes in Biomathematics 20 (Springer, Berlin, 1977).
- [5] G.H. Golub and C.F. van Loan, *Matrix Computations* (John Hopkins University Press, London, 2nd. ed., 1989).
- [6] A. Jameson, The evolution of computational methods in aerodynamics, *J. Appl. Mech.* 50 (1983) 1052–1076.
- [7] A. Lerat, A class of implicit difference schemes for hyperbolic conservation laws, *C. R. Acad. Sci. Paris Sér. I Math.* 288 (1979) 1033–1036 (in French).
- [8] B.P. Sommeijer and P.J. van der Houwen, On the economization of stabilized Runge–Kutta methods with applications to parabolic initial value problems, *Z. Angew. Math. Mech.* 61 (1981) 105–114.
- [9] E. Turkel, Acceleration to a steady state for the Euler equations, in: *Numerical Methods for the Euler Equations of Fluid Dynamics* (SIAM, Philadelphia, PA, 1985) 218–311.
- [10] P.J. van der Houwen, C. Boon and F.W. Wubs, Analysis of smoothing matrices for the preconditioning of elliptic difference equations, *Z. Angew. Math. Mech.* 68 (1988) 3–10.
- [11] P.J. van der Houwen and B.P. Sommeijer, On the internal stability of explicit,  $m$ -stage Runge–Kutta methods for large  $m$ -values, *Z. Angew. Math. Mech.* 60 (1980) 479–485.
- [12] P.J. van der Houwen and B.P. Sommeijer, A special class of multistep Runge–Kutta methods with extended real stability interval, *IMA J. Numer. Anal.* 2 (1982) 183–209.
- [13] P.J. van der Houwen and B.P. Sommeijer, Improving the stability of predictor–corrector methods by residue smoothing, *IMA J. Numer. Anal.* 10 (1990) 361–378.
- [14] A.S. Vasudeva Murthy and J.G. Verwer, Solving parabolic integro–differential equations by an explicit integration method, *J. Comput. Appl. Math.* 39 (1992) 121–132.
- [15] J.G. Verwer, J.G. Blom and W.H. Hundsdorfer, An implicit–explicit approach for atmospheric transport–chemistry problems, *Appl. Numer. Math.* 20 (1996) 191–209.
- [16] J.C. Wilson, Stability of Richtmyer-type difference schemes in any finite number of space variables and their comparison with multistep Strang schemes, *J. Inst. Math. Appl.* 10 (1972) 238–257.
- [17] F.W. Wubs, Stabilization of explicit methods for hyperbolic partial differential equations, *Internat. J. Numer. Methods Fluids* 6 (1986) 641–657.
- [18] F.W. Wubs, Numerical solution of the shallow-water equations, Thesis, University of Amsterdam (1987).