Journal of Computational and Applied Mathematics 10 (1984) 179–193
North-Holland

A comparative study of ADI splitting methods for parabolic equations in two space dimensions

H.B. DE VRIES *
Mathematisch Centrum, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands

Received 28 June 1983

Abstract: The main purpose of the paper is a numerical comparison of three integration methods for semi-discrete parabolic partial differential equations in two space variables. Linear as well as nonlinear equations are considered. The integration methods are the well-known ADI method of Peaceman and Rachford, a global extrapolation scheme of the classical ADI method to order four and a fourth order, four-step ADI splitting method.

Keywords: Numerical analysis, parabolic partial differential equations, method of lines, ADI splitting methods, global extrapolation.

1. Introduction

Let the system of ordinary differential equations

$$\frac{dy}{dt} = f(t, y), \tag{1.1}$$

with prescribed values for y at $t = t_0$, originate from the semi-discretization on a uniform grid Ω_h (with mesh width h) of a *parabolic* two-dimensional partial differential equation. In a few recent papers [4,10] high order splitting methods are described for the numerical solution of (1.1) by using a splitting of the right-hand side function $f(t, y)$, e.g. $f(t, y) = f_1(t, y) + f_2(t, y)$ where the splitting functions f_1 and f_2 have 'simply structured' Jacobian matrices. More generally, one may use splitting functions $F(t, u, v)$ such that $F(t, y, y) = f(t, y)$ and $\partial F/\partial u$, $\partial F/\partial v$ are again 'simply structured'. Here, the numerical solution of parabolic partial differential equations with smooth initial data is considered.

The SC method analyzed and tested in [4] is a fourth order, four-step splitting method for semi-discrete parabolic equations. The method is a variant of the *method of successive corrections* described in [2]. In this method the fourth-order backward differentiation formula [5, p. 242] is chosen for the integration of (1.1). Then in each integration step a, usually nonlinear, system of equations is solved by a (nonlinear) ADI splitting method and this iteration process is accelerated by using *Chebyshev polynomials*. The parameters in the Chebyshev iteration process are chosen such that the *lower frequencies* in the initial error are strongly *damped*. Thus, if the problem is smooth so that no high frequencies are involved, a rather fast convergence is obtained to the solution of the system of equations originating from the fourth order backward differentiation formula. In Section 2 we briefly describe this method.

In [10] global extrapolation of the locally one-dimensional (LOD) method is advocated to increase the accuracy. This technique can be applied to any one-step splitting method for time-dependent, multi-space dimensional problems. Here, *global extrapolation* to order four of the classical ADI method of Peaceman and Rachford [6] is considered. Global extrapolation involves parallel integration with the same basic scheme on different time grids, but completely separated. Global extrapolation to order four requires twice as many operations per step as the basic scheme. By global extrapolation the accuracy is increased in a global way and by no means the stepwise stability of the solution process is influenced. In addition, global

* Present address: Fokker, B.V., Department RAMS, S018-29, 1117 ZJ Schiphol Oost, The Netherlands.

extrapolation is easy to implement. In Section 3 the classical ADI method and the global extrapolation scheme are briefly described.

Finally, in Section 4 the three integration methods are compared for a class of initial-boundary value problems. It is the purpose of this paper to give more insight into the use of an ADI splitting method for semi-discrete parabolic equations.

2. The SC method

In this section the SC method is briefly described. Details on the construction and analysis of this method are, as far as possible, omitted. The interested reader is referred to [4].

By applying the fourth-order backward differentiation formula [5, p. 242] to (1.1) we obtain at each integration step an implicit equation for the numerical solution y_{n+1} at t_{n+1}:

$$y_{n+1} - b_0\tau f(t_{n+1}, y_{n+1}) = \Sigma, \tag{2.1}$$

where $b_0 = \frac{12}{25}$, $\Sigma = \frac{1}{25}[48y_n - 36y_{n-1} + 16y_{n-2} - 3y_{n-3}]$ and τ is the integration step.

2.1. The iteration scheme

The systems of equations (2.1) are solved by the SC method, which is defined by

$$\tilde{y}^{(0)} = 4y_n - 6y_{n-1} + 4y_{n-2} - y_{n-3}, \tag{2.2a}$$

$$y^{(0)} = \Sigma + b_0\tau\tilde{F}(t_{n+1}, y^{(0)}, \tilde{y}^{(0)}), \tag{2.2b}$$

$$\begin{aligned} &y^{(j+1)} = (\mu_j - \lambda_j)y^{(j)} + (1-\mu_j)y^{(j-1)} + \lambda_j y^{**}, \quad j = 0, 1, \ldots, m-1, \\ &y_{n+1} = y^{(m)}, \end{aligned} \tag{2.2c}$$

where $\tilde{F}(t, u, v)$ denotes a Jacobi type splitting function such that $\tilde{F}(t, y, y) = f(t, y)$ and y^{**} is determined by the two equations

$$\begin{aligned} &\omega y^{*} + (1-\omega)y^{(j)} - b_0\tau F(t_{n+1}, y^{(j)}, y^{*}) = \Sigma, \\ &\omega y^{**} + (1-\omega)y^{*} - b_0\tau F(t_{n+1}, y^{**}, y^{*}) = \Sigma, \end{aligned} \tag{2.3}$$

with $F(t, u, v)$ an ADI splitting function [3] such that $F(t, y, y) = f(t, y)$.

2.2. The iteration parameters

Let T_j denote the Chebyshev polynomial of degree j, then the coefficients ω, μ_j and λ_j in the iteration scheme (2.2)–(2.3) are defined by

$$\begin{aligned} &\omega = \frac{\cosh\left(\frac{\operatorname{arccosh} 15}{m}\right) + 1}{\cosh\left(\frac{\operatorname{arccosh} 15}{m}\right) - \cos\frac{\pi}{2m}}, \\ &S^{*} = \frac{-2\omega(\omega-1) - \omega\left\{(\omega-1)\left(1+\cos\frac{\pi}{2m}\right)\left(3\omega - 2 - \omega\cos\frac{\pi}{2m}\right)\right\}^{1/2}}{-2 + \omega\left(1-\cos\frac{\pi}{2m}\right)}, \\ &a = \frac{(2\omega-1)(2S^{*}+1)}{(S^{*}+\omega)^2}, \qquad b = \frac{2\omega-1}{\omega}, \\ &\mu_0 = 1, \qquad w_0 = \frac{b+a}{b-a}, \qquad \mu_j = 2w_0\frac{T_j(w_0)}{T_{j+1}(w_0)}, \\ &\lambda_0 = \frac{2}{a+b}, \qquad \lambda_j = \frac{2\mu_j}{b+a}, \quad j = 1, 2, \ldots, m-1. \end{aligned} \tag{2.4}$$

2.3. The implicit relations

The predictor formula $y^{(0)}$ in the iteration scheme (2.2)–(2.3) is obtained by performing an adjusted Jacobi iteration (2.2b) on the third-order extrapolation formula (2.2a). The implicit equations in the Jacobi iteration are solved by performing one Newton iteration, where it is assumed that the matrix $\partial f/\partial y$ is evaluated in $(t_{n+1}, \tilde{y}^{(0)})$ and $\tilde{y}^{(0)}$ is the initial approximation in the Newton iteration. Then (2.2b) can be simplified into

$$y^{(0)} = (1+d)^{-1}\left[\Sigma + b_0\tau f\left(t_{n+1}, \tilde{y}^{(0)}\right) + d\tilde{y}^{(0)}\right], \tag{2.2b'}$$

where $d = \frac{15}{16}b_0\tau\sigma$ and σ is an estimate of the spectral radius of $\partial f/\partial y$. The estimate σ was either given in analytical form or computed by applying Gerschgorin's theorem to the matrix $\partial f/\partial y$.

In the numerical experiments the right-hand side of (1.1) can be linearly split into two terms, i.e., $f(t, y) = f_1(t, y) + f_2(t, y)$ where the splitting functions f_1 and f_2 correspond to one-space dimensional partial differential operators and have *tridiagonal Jacobian* matrices [3]. In this case the (nonlinear) ADI splitting function $F(t, u, v)$ is defined by

$$F(t, u, v) = f_1(t, u) + f_2(t, v). \tag{2.5}$$

The implicit equations (2.3) are solved by performing one Newton iteration, i.e.,

$$\begin{aligned} y^* &= y^{(j)} - \left[\omega I - b_0\tau F_v\right]^{-1}\left[y^{(j)} - b_0\tau f\left(t_{n+1}, y^{(j)}\right) - \Sigma\right], \\ y^{**} &= y^* - \left[\omega I - b_0\tau F_u\right]^{-1}\left[y^* - b_0\tau f(t_{n+1}, y^*) - \Sigma\right], \end{aligned} \tag{2.3'}$$

where F_v and F_u denote the tridiagonal Jacobian matrices evaluated in $(t_{n+1}, \tilde{y}^{(0)})$ of f_2 and f_1, respectively.

2.4. Stability

The SC method explicitly uses the information that (1.1) originates from a parabolic problem so that the eigenvalues of $\partial f/\partial y$ will be located in a long narrow strip along the *negative axis*. At the same time, this is also a restriction in the applicability of this method. The SC method is completely defined if we specify m. The resulting fourth-order four-step method is conditionally stable. To be more precise, the real *stability boundary* β of the SC method is of the form $\beta = cm^4$, where c is approximately equal to 4. In Table 2.1 the stability boundaries $\beta = \beta(m)$ of the SC method and the corresponding ω and S^*-values are listed for $m = 1$ until 6.

The SC method is stable for the S^*-values listed in Table 2.1 and for integration steps satisfying the condition

$$\tau \leqslant \frac{\beta(m)}{\sigma}. \tag{2.6}$$

In an actual application of the method we will choose for m the smallest integer such that (2.6) is satisfied when τ and σ are prescribed.

Table 2.1
The stability boundaries β of the SC method

	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$
$\omega \simeq$	1.07	1.8	3.2	5.18	7.75	10.88
$S^* \simeq$	0.48	4	18	54	129	264
$\beta \simeq$	20	101	385	1095	2549	5150

3. Global extrapolation of the classical ADI method of Peaceman and Rachford

In this section we shortly describe the global extrapolation of the classical ADI method of Peaceman and Rachford [6]. We assume that the right-hand side of (1.1), $f(t, y)$,can be written as

$$f(t, y) = f_1(t, y) + f_2(t, y), \tag{3.1}$$

where the splitting function f_i corresponds to a one-space dimensional partial differential operator and has a tridiagonal matrix J_i.

3.1. The ADI method

The following time integration formula,

$$y^* = y_n + \tfrac{1}{2}\tau f_1(t_n + \tfrac{1}{2}\tau, y^*) + \tfrac{1}{2}\tau f_2(t_n, y_n), \tag{3.2a}$$

$$y_{n+1} = 2y^* - y_n + \tfrac{1}{2}\tau f_2(t_n + \tau, y_{n+1}) - \tfrac{1}{2}\tau f_2(t_n, y_n), \tag{3.2b}$$

then defines the *second-order* ADI method of Peaceman and Rachford in the so-called Varga form [9]. The vectors y_n and y_{n+1} denote the numerical approximations to the exact solution $y(t)$ of (1.1) at the step points t_n and $t_{n+1} = t_n + \tau$, respectively.

3.2. Global extrapolation

The Peaceman–Rachford method (3.2) may be considered as a particular one-step integration method for the system of ordinary differential equations (1.1). Suppose that (3.2) is applied from $t_0 = 0$ up to $t_N = T$, using a time grid G_1 with stepsize τ and let f be M times differentiable with M sufficiently large. Then there exists an asymptotic expansion in the stepsize τ for the global error (see [8,10]). The existence of this asymptotic expansion for the global error forms the basis for global (Richardson) extrapolation of the ADI method (3.2).

Global extrapolation is easy to implement. It involves parallel integration with the basic scheme (3.2) on different time grids. Let us consider the coherent grids G_1, G_2 and G_3 depicted in Fig. 3.1. G_2 is obtained from G_1 by halving the stepsize τ, etc. Because of this coherence between the grids, the asymptotic expression of the global error holds for τ, $\frac{1}{2}\tau$ and $\frac{1}{3}\tau$, at all common gridpoints, i.e., on the whole of G_1. Let $y_{n,i}$ denote the approximation to $y(t_n)$ at the grid G_i. Then compute at all common points

$$y_n^{[4]} = \tfrac{27}{12}y_{n,3} - \tfrac{4}{3}y_{n,2} + \tfrac{1}{12}y_{n,1}, \tag{3.3}$$

and a *fourth-order* global extrapolation scheme of the ADI method is obtained. In the numerical experiments we apply (3.3) only in the endpoint $t = 1$. The integrations on the different grids are performed completely separated from each other. The results $y_{n,i}$ are only connected by the initial data $y_{0,i} = y_0$, for all i. This means that global extrapolation cannot interfere with the stability of the ADI method. Global extrapolation to order four requires twice as many operations per step as the basic scheme (3.2) on G_3.

It is well known that the classical ADI method will lose accuracy if the boundary conditions of the parabolic equation become time-dependent. The globally extrapolated results also suffer from this phenomenon.

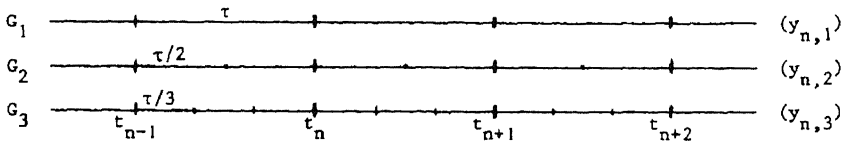

Fig. 3.1. Three coherent grids.

3.3. The implicit relations

The implicit equations (3.2a) and (3.2b) are solved by performing ν Newton iterations, i.e.,

$$x^{(0)} = y_n,$$

$$x^{(j+1)} = x^{(j)} - \left[I - \tfrac{1}{2}\tau J_1\right]^{-1}\left[x^{(j)} - y_n - \tfrac{1}{2}\tau f_1\left(t_n + \tfrac{1}{2}\tau, x^{(j)}\right) - \tfrac{1}{2}\tau f_2(t_n, y_n)\right], \quad j = 0, \ldots, \nu - 1, \tag{3.2a'}$$

$$y^* = x^{(\nu)}$$

and

$$v^{(0)} = y^*,$$

$$v^{(j+1)} = v^{(j)} - \left[I - \tfrac{1}{2}\tau J_2\right]^{-1}\left[v^{(j)} - 2y^* + y_n + \tfrac{1}{2}\tau f_2(t_n, y_n) - \tfrac{1}{2}\tau f_2\left(t_n + \tau, v^{(j)}\right)\right], \quad j = 0, \ldots, \nu - 1, \tag{3.2b'}$$

$$y_{n+1} = v^{(\nu)},$$

where J_1 and J_2 denote the *tridiagonal Jacobian* matrices evaluated in $(t_n + \frac{1}{2}\tau, y_n)$ and $(t_n + \tau, y_n)$, respectively. In case of linear problems we perform one iteration using the same Newton-type process. The ADI method of Peaceman and Rachford and the global extrapolation scheme will be denoted by PR(ν) and GEPR(ν), respectively, in the tables of results.

4. The set of test problems

In order to get insight into the behaviour of the various methods we applied them to a set of test equations. It is difficult to choose a representative set of test problems from the problem class under consideration. Here, a number of problems with a prescribed exact solution are constructed. Some of these problems served as a test example before [4,7,10]. The equations include difficulties like: arbitrary nonlinearities to test the stability of the methods, oscillating solutions and time-dependent boundary conditions.

4.1. The test examples

The equations are scalar equations and belong to the general class

$$u_t = G_1\left(t, x_1, x_2, u, u_{x_1}, u_{x_1x_1}\right) + G_2\left(t, x_1, x_2, u, u_{x_2}, u_{x_2x_2}\right), \tag{4.1}$$

denotes on $\{(t, x_1, x_2) | 0 \leqslant t \leqslant 1, (x_1, x_2) \in \Omega\}$, where Ω is given by

$$\Omega = \{(x_1, x_2) | 0 \leqslant x_1 \leqslant 1, 0 \leqslant x_2 \leqslant 1\}.$$

The initial conditions and the Dirichlet boundary conditions are obtained from the exact solutions. The space discretization of all equations is performed using *standard symmetric differences* on a uniform grid with grid size $h = \frac{1}{20}$, resulting in 361 internal grid points. The time integration aspects of the methods can be tested more or less separately from the effects of space discretization, because the equations are chosen such that discretization of the space variables on a uniform grid by standard finite differences does not give a space discretization error.

We now summarize the parabolic equations together with their exact solution:

I $$u_t = u_{x_1x_1} + u_{x_2x_2} + e^{-t}\left[\left(x_1^2 - x_1\right)\left(x_2^2 - x_2\right) + 2\left(x_1^2 - x_1\right) + 2\left(x_2^2 - x_2\right)\right],$$

$$u(t, x_1, x_2) = 1 - e^{-t}\left(x_1^2 - x_1\right)\left(x_2^2 - x_2\right).$$

II $$u_t = u_{x_1x_1} + u_{x_2x_2} - e^{-t}(x_1^2 + x_2^2 + 4),$$
$$u(t, x_1, x_2) = 1 + e^{-t}(x_1^2 + x_2^2).$$

III $$u_t = u_{x_1x_1} + u_{x_2x_2} - e^{-t}(x_1^3 + x_2^3 + 6x_1 + 6x_2),$$
$$u(t, x_1, x_2) = 1 + e^{-t}(x_1^3 + x_2^3).$$

IV $$u_t = u_{x_1x_1} + u_{x_2x_2} + 2t^2\left[(x_1^2 + x_2)\pi \cos 2\pi t - x_1 - \sin 2\pi t\right] + 2t\left[(x_1^2 + x_2)\sin 2\pi t + x_1x_2^2\right],$$
$$u(t, x_1, x_2) = 1 + t^2\left[(x_1^2 + x_2)\sin 2\pi t + x_1x_2^2\right].$$

V $$u_t = \frac{1}{1+t}(u_{x_1x_1} + u_{x_2x_2}) + (u_{x_1})^2 + (u_{x_2})^2 - e^{-t}\left[x_1^2 + x_2^2 + \frac{4}{1+t} + 4e^{-t}(x_1^2 + x_2^2)\right],$$
$$u(t, x_1, x_2) = 1 + e^{-t}(x_1^2 + x_2^2).$$

VI $$u_t = \frac{1}{1+t}(u_{x_1x_1} + u_{x_2x_2}) - \frac{u^2}{2(1+t)} + \frac{1}{2(1+t)}\left[1 + \frac{(x_1^2 - x_2^2)^2}{(1+t)^2}\right],$$
$$u(t, x_1, x_2) = 1 + \frac{x_1^2 - x_2^2}{1+t}.$$

VII $$u_t = \frac{x_1 + x_2}{2(1+t)}(u^3_{x_1x_1} + u^3_{x_2x_2}) - \frac{3(x_1 + x_2)^2}{4(1+t)}\sin^3 2\pi t + \pi(x_1 + x_2)\cos 2\pi t,$$
$$u(t, x_1, x_2) = \tfrac{1}{2}(x_1 + x_2)\sin 2\pi t.$$

VIII $$u_t = u(u_{x_1x_1} + u_{x_2x_2}) - 2t^2(x_1 + e^{-t})u + t(2-t)(x_1^2 + x_2)e^{-t} + 2tx_1x_2^2,$$
$$u(t, x_1, x_2) = 1 + t^2\left[(x_1^2 + x_2)e^{-t} + x_1x_2^2\right].$$

4.2. Strategy and results

The testing strategy is as simple as possible: all equations are integrated by the various methods using a sequence of constant stepsizes τ. In case of nonlinear problems the updating of the tridiagonal matrices is performed every integration step (see Sections 2 and 3). We thus do not use any strategy to estimate errors, to vary the stepsize and to control the updating of the tridiagonal matrices. The examples are such that an analytical expression for the Jacobian matrices was available.

In the SC method we need an estimate of the spectral radius of the matrix $\partial f/\partial y$, viz. σ (see Section 2). For the Examples V and VIII the estimate σ was computed by applying Gerschgorin's theorem to the matrix $\partial f/\partial y$ at each integration step. For the other examples the estimate σ was given in analytical form and the expression for σ is listed in the tables of results. The number of f-evaluations is minimized with respect to absolute stability requirements (see (2.6)). The estimate σ is constant for the linear Examples I, II, III and IV. For the Problems VI and VII, where we made a t-dependent estimate, m is minimized at each step.

The starting values needed by the SC method were obtained by computing them from the exact values prescribed at $t = -3\tau, -2\tau, -\tau, 0$.

The accuracy is measured by the number sd of correct significant digits defined by

$$\text{sd} = -\log_{10}|\text{maximum absolute error at } t = 1|. \tag{4.2}$$

The efficiency is measured by

fev = the total number of right-hand side evaluations ($f(t, y)$ in (1.1)),
Jev = the total number of Jacobian evaluations ($\partial f/\partial y$)

Table 4.0
The order of accuracy, the computational effort required per integration step τ and the number of arrays required for storage of the PR(ν), GEPR(ν) and SC method

	PR(ν)	GEPR(ν)	SC
Order of accuracy	2	4	4
Number of f-evaluations	$\frac{1}{2}+\nu$	$2\nu+1$	$2m+1$
Number of J-evaluations	1	2	1
Number of F–B substitutions	2ν	4ν	$2m$
Number of LU-decompositions	2	4	2
Number of storage arrays	11	13	14

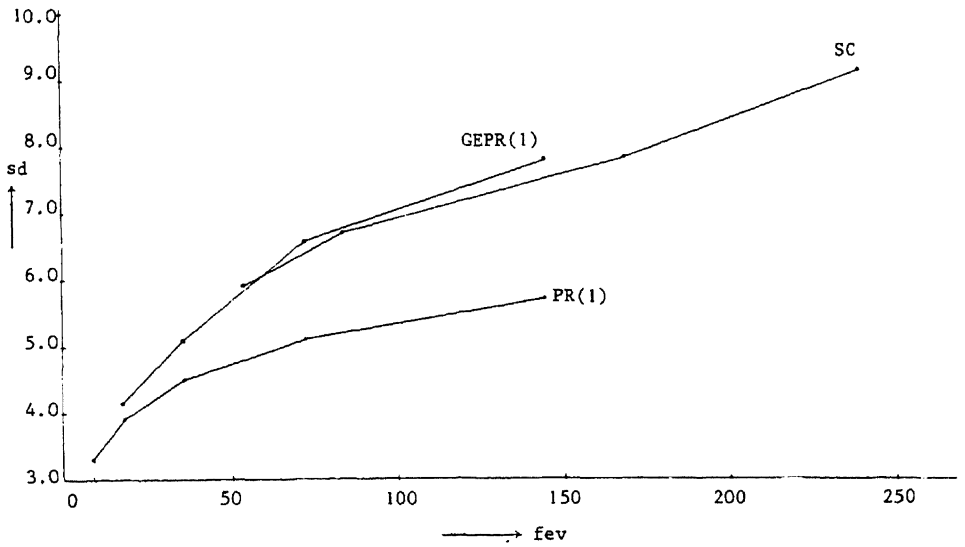

Fig. 4.1. Number of correct significant digits sd and number of f-evaluations fev for the linear Example I with constant boundary conditions and $h=\frac{1}{20}$.

Table 4.1.
Results for the linear Example I with constant boundary conditions and $h=\frac{1}{20}$ obtained by the PR(1), GEPR(1) and SC method. In the SC method $\sigma=8/h^2$

Method	τ	sd	fev	FBS
PR(1)	$\frac{1}{6}$	3.29	9	12
	$\frac{1}{12}$	3.92	18	24
	$\frac{1}{24}$	4.52	36	48
	$\frac{1}{48}$	5.12	72	96
	$\frac{1}{96}$	5.72	144	192
GEPR(1)	$\frac{1}{6}$	4.15	18	24
	$\frac{1}{12}$	5.12	36	48
	$\frac{1}{24}$	6.57	72	96
	$\frac{1}{48}$	7.83	144	192
SC	$\frac{1}{6}$	5.91	54	48
	$\frac{1}{12}$	6.72	84	72
	$\frac{1}{24}$	7.85	168	144
	$\frac{1}{48}$	9.16	240	192

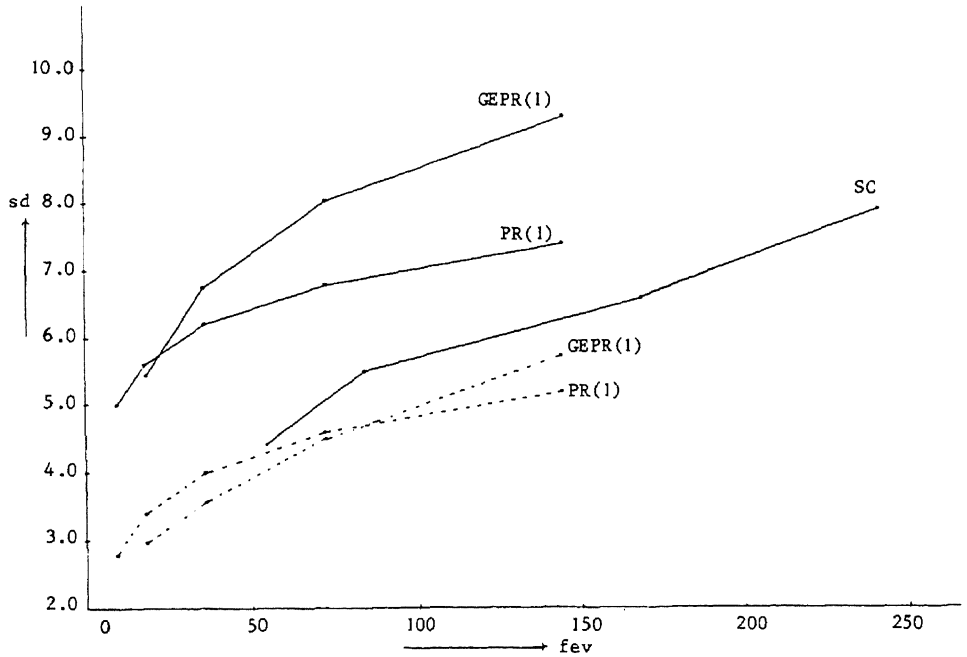

Fig. 4.2. Number of correct significant digits sd and number of f-evaluations fev for the linear Example II with $h = \frac{1}{20}$. The dotted lines refer to results obtained by PR(1) and GEPR(1), where the source term v was only included in the splitting function f_1.

and

FBS = the total number of forward–backward substitutions needed for the solution of tridiagonal systems.

In all methods the total number of LU-decompositions of the tridiagonal matrices is twice as many as the total number of Jacobian evaluations Jev. A conclusion based on the sd- and fev-values as to which method

Table 4.2
Results for the linear Example II with $h = \frac{1}{20}$ obtained by the PR(1), GEPR(1) and SC method. The numbers in the parentheses are the sd-values obtained by PR(1) and GEPR(1), where the source term v was only included in the splitting function f_1. In the SC method $\sigma = 8/h^2$

Method	τ	sd	fev	FBS
PR(1)	$\frac{1}{6}$	4.98(2.81)	9	12
	$\frac{1}{12}$	5.58(3.41)	18	24
	$\frac{1}{24}$	6.18(4.01)	36	48
	$\frac{1}{48}$	6.79(4.61)	72	96
	$\frac{1}{96}$	7.39(5.21)	144	192
GEPR(1)	$\frac{1}{6}$	5.45(2.95)	18	24
	$\frac{1}{12}$	6.74(3.56)	36	48
	$\frac{1}{24}$	8.03(4.50)	72	96
	$\frac{1}{48}$	9.34(5.76)	144	192
SC	$\frac{1}{6}$	4.39	54	48
	$\frac{1}{12}$	5.49	84	72
	$\frac{1}{24}$	6.60	168	144
	$\frac{1}{48}$	7.89	240	192

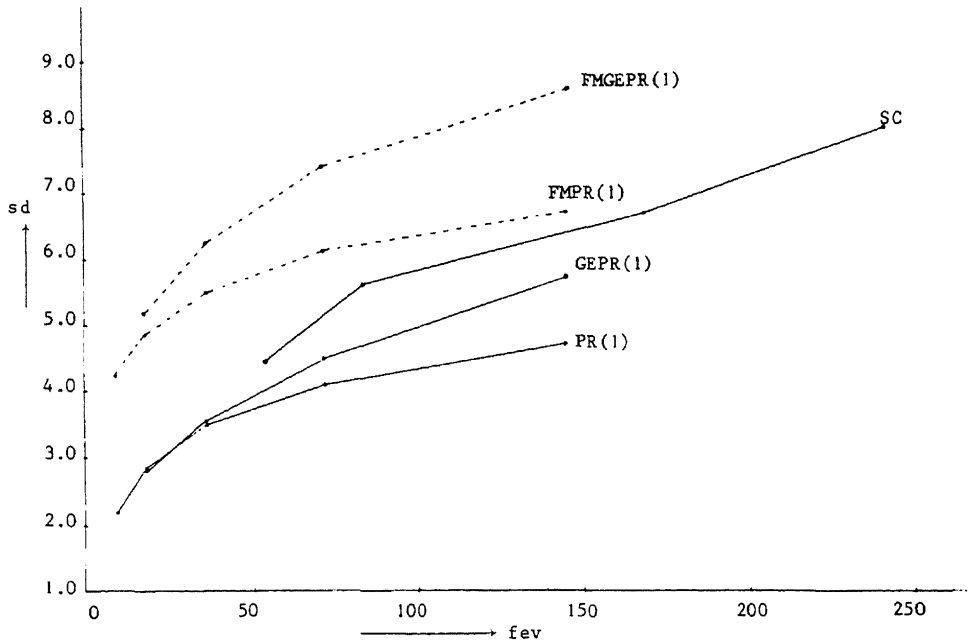

Fig. 4.3. Number of correct significant digits sd and number of f-evaluations fev for the linear Example III with $h = \frac{1}{20}$. The dotted lines refer to results obtained by PR(1) and GEPR(1) with the boundary-value correction (i.e., FMPR(1) and FMGEPR(1), respectively).

is the more efficient one is difficult, since one should also measure the additional computational effort required by the methods. Therefore, we list in the tables of results also the Jev- and FBS-values required by the various methods, so that the reader can judge the results himself. For linear problems the Jacobian matrices were determined once. In this case Jev is not listed in the tables of results. Other computations, such as the calculations of the extrapolation formula (2.2a) and Σ (2.1), the evaluation of the spectral radius of $\partial f/\partial y$, all initial work for estimating the iteration parameters and the Chebyshev iterations needed in the

Table 4.3
Results for the linear Example III with $h = \frac{1}{20}$ obtained by the PR(1), GEPR(1) and SC method. The numbers in the parentheses are the sd-values obtained by FMPR(1) and FMGEPR(1). In the SC method $\sigma = 8/h^2$

Method	τ	sd	fev	FBS
PR(1)	$\frac{1}{6}$	2.23(4.26)	9	12
	$\frac{1}{12}$	2.88(4.88)	18	24
	$\frac{1}{24}$	3.51(5.49)	36	48
	$\frac{1}{48}$	4.11(6.09)	72	96
	$\frac{1}{96}$	4.71(6.69)	144	192
GEPR(1)	$\frac{1}{6}$	2.83(5.18)	18	24
	$\frac{1}{12}$	3.57(6.26)	36	48
	$\frac{1}{24}$	4.50(7.39)	72	96
	$\frac{1}{48}$	5.76(8.61)	144	192
SC	$\frac{1}{6}$	4.44	54	48
	$\frac{1}{12}$	5.62	84	72
	$\frac{1}{24}$	6.72	168	144
	$\frac{1}{48}$	7.98	240	192

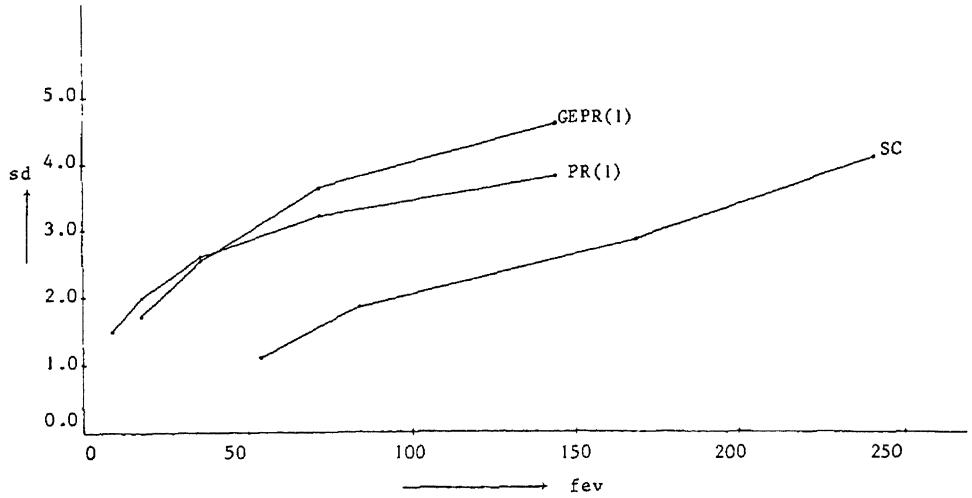

Fig. 4.4. Number of correct significant digits sd and number of f-evaluations fev for the linear Example IV with an oscillating solution and $h = \frac{1}{20}$.

SC method, are not taken into account in our efficiency measure. This slightly favours the SC method in our comparisons.

Table 4.0 summarizes for the various methods the order of accuracy, the computational effort required per integration step τ and the number of arrays of length corresponding to the number of grid points required for storage. Notice that in the global extrapolation scheme the integration step τ corresponds to the step on the finest time grid G_3 and the value of m in the SC method is not necessarily constant at each step (see Section 2.4).

It is well known that the ADI method of Peaceman and Rachford will lose accuracy if the boundary conditions become time-dependent [1,7]. In order to improve the accuracy Fairweather and Mitchell proposed a boundary-value correction technique (see [1,7]) for the ADI method. For the Examples III and V we have also used the Fairweather–Mitchell boundary-value correction in the basic PR(ν) scheme. This will be denoted by FMPR(ν) and FMGEPR(ν) in the tables of results and figures. The computational work of FMPR(ν) is hardly more than that of PR(ν).

Table 4.4
Results for the linear Example IV with $h = \frac{1}{20}$ obtained by the PR(1), GEPR(1) and SC method. In the SC method $\sigma = 8/h^2$

Method	τ	sd	fev	FBS
PR(1)	$\frac{1}{6}$	1.47	9	12
	$\frac{1}{12}$	1.99	18	24
	$\frac{1}{24}$	2.60	36	48
	$\frac{1}{48}$	3.20	72	96
	$\frac{1}{96}$	3.81	144	192
GEPR(1)	$\frac{1}{6}$	1.68	18	24
	$\frac{1}{12}$	2.55	36	48
	$\frac{1}{24}$	3.63	72	96
	$\frac{1}{48}$	4.57	144	192
SC	$\frac{1}{6}$	1.12	54	48
	$\frac{1}{12}$	1.86	84	72
	$\frac{1}{24}$	2.83	168	144
	$\frac{1}{48}$	4.09	240	192

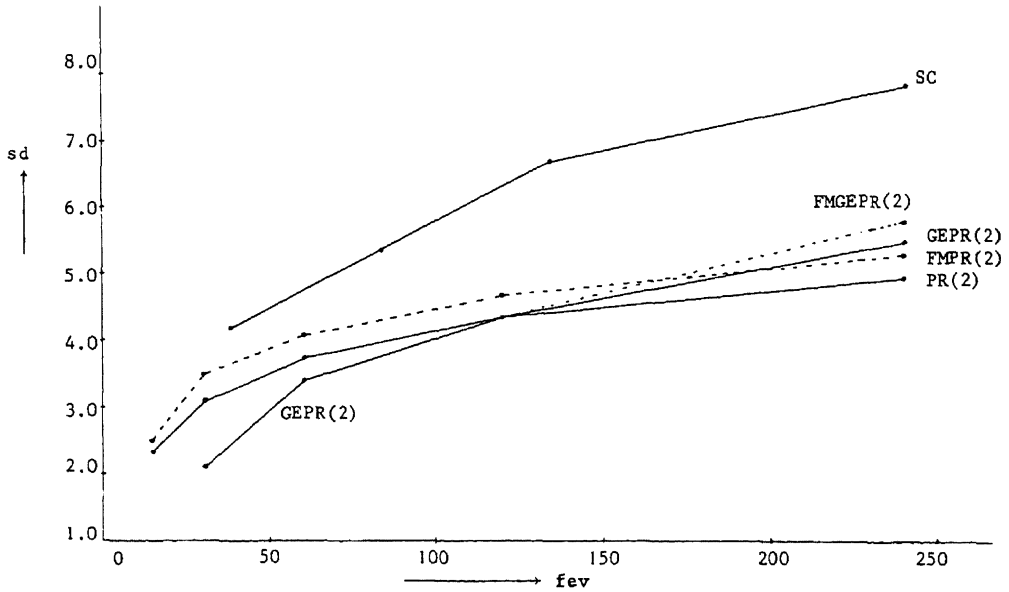

Fig. 4.5. Number of correct significant digits sd and number of f-evaluations fev for the nonlinear Example V with $h = \frac{1}{20}$. The dotted lines refer to results obtained by PR(2) and GEPR(2) with the boundary-value correction (i.e., FMPR(2) and FMGEPR(2), respectively).

In the examples a time-dependent source term $v(t, x_1, x_2, u)$ is present. A splitting of $v = \frac{1}{2}v + \frac{1}{2}v$ was used in all experiments, i.e., in the splitting functions f_i (see (2.5) and (3.1)) only a fraction of the source term (viz. $\frac{1}{2}v$) was included. For Example II we used also another splitting of v. In this splitting of v the entire source term was only included in f_1.

The results of the experiments are presented in Tables 4.1–4.8 and the corresponding figures. The τ-values correspond to the finest grid in the global extrapolation scheme.

For the linear Example I with constant boundary conditions the global extrapolation scheme is more or less comparable to the SC method. The basic PR(1) scheme is strongly sensitive to the splitting of the source term in the linear Example II. Using the most efficient splitting of v in the basic scheme the PR(1)

Table 4.5
Results for the nonlinear Example V with $h = \frac{1}{20}$ obtained by the PR(2), GEPR(2) and SC method. The numbers in the parentheses are the sd-values obtained by FMPR(2) and FMGEPR(2)

Method	τ	sd	fev	Jev	FBS
PR(2)	$\frac{1}{6}$	2.35(2.52)	15	6	24
	$\frac{1}{12}$	3.11(3.52)	30	12	48
	$\frac{1}{24}$	3.74(4.12)	60	24	96
	$\frac{1}{48}$	4.34(4.72)	120	48	192
	$\frac{1}{96}$	4.94(5.32)	240	96	384
GEPR(2)	$\frac{1}{6}$	2.1(2.13)	30	12	48
	$\frac{1}{12}$	3.42(3.41)	60	24	96
	$\frac{1}{24}$	4.32(4.34)	120	48	192
	$\frac{1}{48}$	5.52(5.81)	240	96	384
SC	$\frac{1}{6}$	4.19	38	6	32
	$\frac{1}{12}$	5.36	84	12	72
	$\frac{1}{24}$	6.69	134	24	110
	$\frac{1}{48}$	7.85	240	48	192

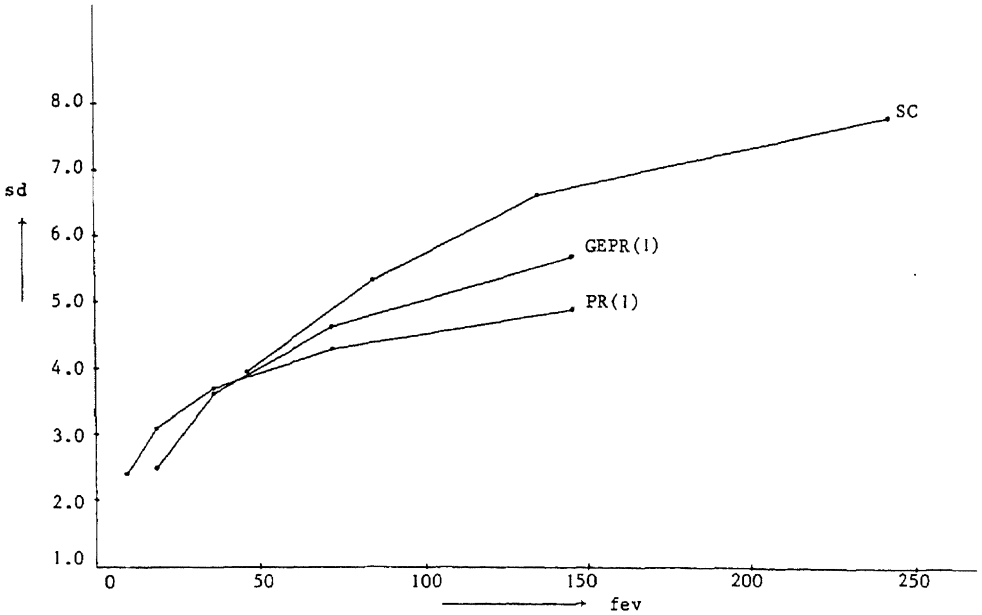

Fig. 4.6. Number of correct significant digits sd and number of f-evaluations fev for the mildly nonlinear Example VI with $h = \frac{1}{20}$.

and GEPR(1) method are superior to the SC method. For the linear Example III the global extrapolation scheme is only with the Fairweather–Mitchell boundary-value correction more efficient than the SC method. For the linear example IV the accuracy of all methods is low because of the oscillating solution. The SC method is less efficient than the PR(1) and GEPR(1) method.

The tables of results and figures of the nonlinear Examples V, VII and VIII illustrate the superiority of the SC method if high accuracy is desired. In the last two examples the global extrapolation scheme and Peaceman–Rachford scheme become unstable for larger stepsizes. For the mildly nonlinear Example VI the GEPR(1) scheme is slightly less efficient than the SC method.

Table 4.6
Results for the mildly nonlinear Example VI with $h = \frac{1}{20}$ obtained by the PR(1), GEPR(1) and SC method. In the SC method $\sigma = [8/h^2 + (t+2)/(t+1)]/(t+1)$.

Method	τ	sd	fev	Jev	FBS
PR(1)	$\frac{1}{6}$	2.41	9	6	12
	$\frac{1}{12}$	3.1	18	12	24
	$\frac{1}{24}$	3.7	36	24	48
	$\frac{1}{48}$	4.3	72	48	96
	$\frac{1}{96}$	4.9	144	96	192
GEPR(1)	$\frac{1}{6}$	2.46	18	12	24
	$\frac{1}{12}$	3.63	36	24	48
	$\frac{1}{24}$	4.63	72	48	96
	$\frac{1}{48}$	5.69	144	96	192
SC	$\frac{1}{6}$	3.96	46	6	40
	$\frac{1}{12}$	5.35	84	12	72
	$\frac{1}{24}$	6.63	134	24	110
	$\frac{1}{48}$	7.82	240	48	192

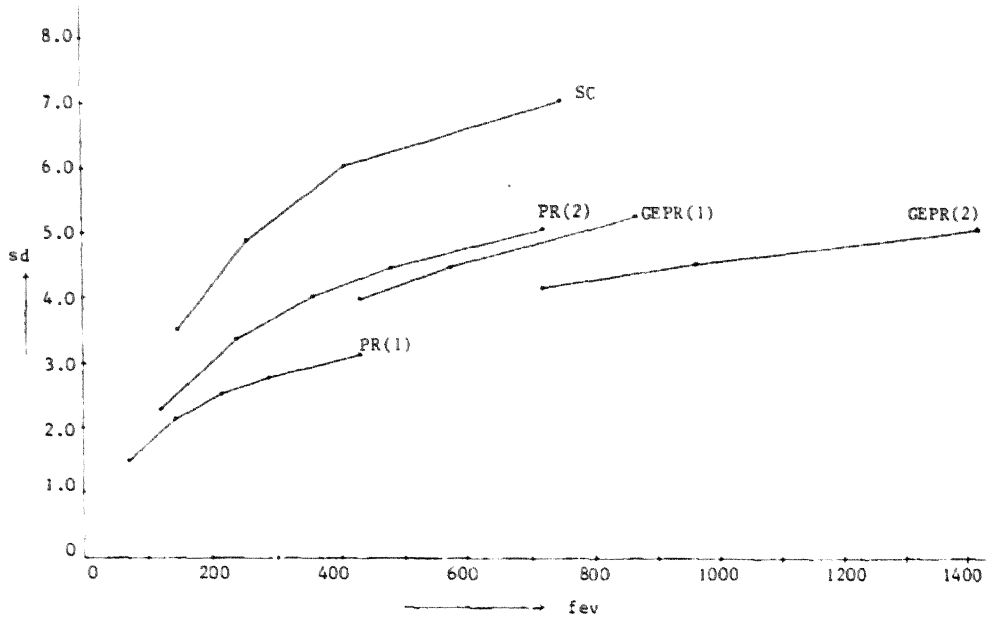

Fig. 4.7. Numbers of correct significant digits sd and number of f-evaluations fev for the strongly nonlinear Example VII with $h = \frac{1}{20}$.

5. Concluding remarks

From the tables of results and the figures we may draw the following conclusions:

(1) For the linear Example III and the nonlinear examples the SC method is superior to the Peaceman–Rachford method (PR(ν)) and the global extrapolation scheme (GEPR(ν)), whereas for the

Table 4.7
Results for the strongly nonlinear Example VII with $h = \frac{1}{20}$ obtained by the PR(ν), GEPR(ν) and SC method. The numbers in the parentheses are the results obtained by PR(2) and GEPR(2). In the SC method $\sigma = (24 \sin^2 2\pi t)/((1+t)h^2)$. An asterisk indicates unstable results

Method	τ	sd	fev	Jev	FBS
PR(1)	$\frac{1}{24}$	*	–	–	–
(PR(2))	$\frac{1}{48}$	1.51(2.32)	72(120)	48(48)	96(192)
	$\frac{1}{96}$	2.16(3.39)	144(240)	96(96)	192(384)
	$\frac{1}{144}$	2.53(4.03)	216(360)	144(144)	288(576)
	$\frac{1}{192}$	2.79(4.48)	288(480)	192(192)	384(768)
	$\frac{1}{288}$	3.15(5.09)	432(720)	288(288)	576(1152)
GEPR(1)	$\frac{1}{96}$	*	–	–	–
(GEPR(2))	$\frac{1}{144}$	3.97(4.21)	432(720)	288(288)	576(1152)
	$\frac{1}{192}$	4.53(4.54)	576(960)	384(384)	768(1536)
	$\frac{1}{288}$	5.3(5.08)	864(1440)	576(576)	1152(2304)
SC	$\frac{1}{24}$	3.54	150	24	126
	$\frac{1}{48}$	4.91	256	48	208
	$\frac{1}{96}$	6.03	412	96	316
	$\frac{1}{192}$	7.12	748	192	556

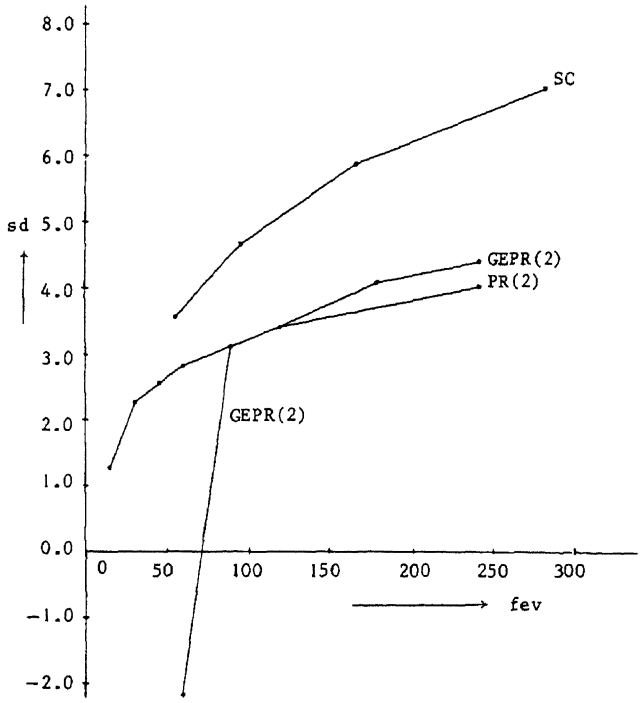

Fig. 4.8. Number of correct significant digits sd and number of f-evaluations fev for the nonlinear Example VIII with $h = \frac{1}{20}$.

linear Examples II and IV the global extrapolation scheme is the most efficient integrator. For the linear Example I the GEPR(1) method is competitive to the SC method.

(2) The results for the linear Example III illustrate that the inaccuracies caused by time-dependent boundary values can be removed by applying the Fairweather–Mitchell boundary-value correction. For

Table 4.8
Results for the nonlinear Example VIII with $h = \frac{1}{20}$ obtained by the PR(2), GEPR(2) and SC method.

Method	τ	sd	fev	Jev	FBS
PR(2)	$\frac{1}{6}$	1.28	15	6	24
	$\frac{1}{12}$	2.25	30	12	48
	$\frac{1}{18}$	2.56	45	18	72
	$\frac{1}{24}$	2.8	60	24	96
	$\frac{1}{36}$	3.13	90	36	144
	$\frac{1}{48}$	3.38	120	48	192
	$\frac{1}{96}$	3.97	240	96	384
GEPR(2)	$\frac{1}{12}$	−2.18	60	24	96
	$\frac{1}{18}$	3.15	90	36	144
	$\frac{1}{24}$	3.35	120	48	192
	$\frac{1}{36}$	4.07	180	72	288
	$\frac{1}{48}$	4.39	240	96	384
SC	$\frac{1}{6}$	3.55	56	6	50
	$\frac{1}{12}$	4.65	96	12	84
	$\frac{1}{24}$	5.86	168	24	144
	$\frac{1}{48}$	7.02	284	48	236

this problem the extrapolation scheme with the correction technique (FMGEPR(1)) is even more efficient than the SC method.

(3) For nonlinear problems the application of the boundary-value correction is less successful in the extrapolation scheme (see Table 4.5). Additional experiments have shown that solving the nonlinear equations more accurately (i.e., performing more Newton iterations) the effect of the Fairweather–Mitchell modification is more clearly noticeable in the extrapolation scheme. However, the SC method is still more efficient. For more general boundary conditions and regions in the (x_1, x_2)-space the Fairweather–Mitchell correction is of less practical value (see [7]).

(4) With the exception of the strongly nonlinear Example VII with the oscillating solution it pays to apply extrapolation of the PR(ν) scheme for small integration steps. For rather large integration steps the PR(ν) method is competitive or even more efficient.

(5) The SC method shows its fourth-order behaviour for realistic integration steps. The theoretical order of the GEPR(ν) scheme appears in the results for the nonlinear examples not so clearly as for the SC method. Additional experiments have shown that performing more Newton iterations in the basic PR(ν) scheme the order behaviour of GEPR(ν) stands out more clearly. However, for large integration steps the GEPR(ν) scheme becomes less efficient.

Summarizing, from the three methods considered, the SC method appears to be the most efficient and robust one for the numerical solution of nonlinear parabolic equations in two space dimensions if high accuracies are desired. The ADI method of Peaceman and Rachford is particularly suited if one is satisfied with low accuracy results. For linear or mildly nonlinear problems the global extrapolation scheme is a useful alternative. In addition, the global extrapolation scheme is easier to implement than the SC method.

It should be noted that the SC method is slightly favoured by using four exact starting values and the smoothed extrapolation formula (2.2a–2.2b) as initial approximation in the Chebyshev iteration. By choosing better initial approximations in the Newton processes a more robust global extrapolation scheme can be constructed for nonlinear problems. The numerical solutions on the finest time grid can be used to construct (e.g., interpolation techniques) initial approximations in the Newton processes on the two other time grids. However, a price has to be paid for the easy applicability of the algorithm and a few additional experiments have shown that the gain in efficiency is not surprising. Further, the storage requirements of the SC method and the global extrapolation scheme are more or less comparable (see Table 4.0).

Acknowledgment

The author would like to thank Dr. J.G. Verwer and Prof. dr. P.J. van der Houwen for their constructive remarks and careful reading of the manuscript.

References

[1] G. Fairweather and A.R. Mitchell, A new computational procedure for ADI methods, *SIAM J. Numer. Anal.* **4** (1967) 163–170.
[2] P.J. van der Houwen, Multistep splitting methods of high order for initial value problems, *SIAM J. Numer. Anal.* **17** (1979) 291–309.
[3] P.J. van der Houwen and J.G. Verwer, One-step splitting methods for semi-discrete parabolic equations, *Computing* **22** (1979) 291–309.
[4] P.J. van der Houwen and H.B. de Vries, A fourth order ADI method for semidiscrete parabolic equations, *J. Comput. Appl. Math.* **9** (1983) 41–63.
[5] J.D. Lambert, *Computational Methods in Ordinary Differential Equations* (Wiley, London, 1973).
[6] D.W. Peaceman and H.H. Rachford, Jr., The numerical solution of parabolic and elliptic differential equations, *J. Soc. Ind. Appl. Math.* **3** (1955) 28–41.
[7] B.P. Sommeijer, P.J. van der Houwen and J.G. Verwer, On the treatment of time-dependent boundary conditions in splitting methods for parabolic differential equations, *Internat. J. Numer. Meths. Engrg.* **17** (1981) 335–346.
[8] H.J. Stetter, *Analysis of Discretization Methods for Ordinary Differential Equations* (Springer, Berlin, 1973).
[9] Varga, R.S., *Matrix Iterative Analysis* (Prentice-Hall, Englewood Cliffs, NJ, 1962).
[10] J.G. Verwer and H.B. de Vries, Global extrapolation of a first order splitting method, Rept. NW 150/83, Mathematisch Centrum, Amsterdam, 1983.