Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational and Applied Mathematics 169 (2004) 171–182
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

The various order explicit multistep exponential fitting for systems of ordinary differential equations

Chen Tang, Haiqing Yan*, Hao Zhang, WenRun Li

Department of Applied Physics, University of Tianjin, Tianjin 300072, PR China

Received 7 April 2003; received in revised form 4 November 2003

Abstract

We develop various order-explicit multistep schemes of exponential fitting for systems of ordinary differential equations. The schemes may be applied to systems whose linear part is nondiagonal. Various precisions can be achieved by employing the various order schemes. Comparisons of accuracy and efficiency of numerical solutions are mentioned in the examples. The numerical results show that the schemes are highly accurate and computationally efficient.

Keywords: Explicit multistep method; Exponential fitting; Higher-order calculation

1. Introduction

In the last decade, a lot of research has been performed in the area of numerical solutions of initial-value problems related to systems of first-order ordinary differential equations. The numerical method treated in this paper is the so-called "exponential time differencing" (ETD) scheme [2,6], or shortly "exponential fitting"(EF) [8]. Explicit exponential fitting with second- and fourth-order accuracy was constructed by employing Hermite interpolants and was used in delayed recruitment/renewal equation [9]. Both explicit ETD schemes of arbitrary order and Runge–Kutta ETD schemes were derived for stiff systems [3]. Runge–Kutta EF scheme with fifth-order accuracy was developed [1]. Although explicit ETD schemes of arbitrary order have been proposed [3], the implementation of the higher-order ETD schemes does not previously seem to have been carried out. To develop EF methods further, in this paper we derive new variable order-explicit multistep EF schemes, which are more extensive and practical than those methods previously given. We also carry out higher-order

* Corresponding author. Tel.: +86-22-87894268; fax: +86-22-27890681.
E-mail addresses: tangchen1963@yahoo.com.cn (C. Tang), haiqingyan@yahoo.com.cn (H. Yan).

0377-0427/$ - see front matter
doi:10.1016/j.cam.2003.12.015

explicit multistep EF schemes. We then demonstrate the accuracy and efficacy of the new schemes via application to a variety of test cases and comparison with other analytical and numerical results. The numerical results show that the schemes are highly accurate and computationally efficient.

2. The explicit multistep EF schemes

The explicit ETD schemes of arbitrary order have been derived elsewhere [3], but here we give a more extensive method, which can be applied to systems whose linear part is nondiagonal. The nth-order system of first-order differential equations for initial value problems are in the form

$$\begin{cases} \dot{y}_1 = f_1(t, y_1, \ldots, y_n), \\ \vdots \\ \dot{y}_j = f_j(t, y_1, \ldots, y_n), \\ \vdots \\ \dot{y}_n = f_n(t, y_1, \ldots, y_n). \end{cases} \tag{1}$$

We introduce the vectors

$$\mathbf{y} = (y_1, y_2, \ldots, y_n)^{\mathrm{T}}, \quad \mathbf{f} = (f_1, f_2, \ldots, f_n)^{\mathrm{T}}.$$

Eq. (1) is expressed as

$$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, t). \tag{2}$$

Introducing a constant matrix $\mathbf{H}$, we rearrange Eq. (2) as

$$\dot{\mathbf{y}} - \mathbf{H}\mathbf{y} = \mathbf{F}, \tag{3}$$

where

$$\mathbf{F} = \mathbf{f}(t, \mathbf{y}) - \mathbf{H}\mathbf{y} \tag{4}$$

and the constant matrix $\mathbf{H}$ should satisfy $\det(\mathbf{H}) \neq 0$.

We multiply (3) through by an integrating factor $\exp(-\mathbf{H}t)$, then integrate from t_k to t_{k+1} to give

$$\mathbf{y}_{k+1} = \exp(\mathbf{H}h) \cdot \mathbf{y}_k + \int_{t_k}^{t_{k+1}} \exp \mathbf{H}(t_{k+1} - t) \cdot \mathbf{F}(t)\, \mathrm{d}t. \tag{5}$$

Let $t_{k+1} = t_k + h$, Eq. (5) is expressed as

$$\mathbf{y}_{k+1} = \exp(\mathbf{H}h) \cdot \mathbf{y}_k + \int_{t_k}^{t_k+h} \exp \mathbf{H}(t_k + h - t) \cdot \mathbf{F}(t)\, \mathrm{d}t. \tag{6}$$

Let $t = t_k + \tau$, we have

$$\mathbf{y}_{k+1} = \exp(\mathbf{H}h) \cdot \mathbf{y}_k + \int_0^h \exp \mathbf{H}(h - \tau) \cdot \mathbf{F}(t_k + \tau)\, \mathrm{d}\tau. \tag{7}$$

On the right-hand side of (7) there appears the unknown solution $\mathbf{F}(t_k+\tau)$. But since the approximations $\mathbf{y}_{k-i},\dots,\mathbf{y}_{k-1},\mathbf{y}_k$ at time points $t_{k-i},\dots,t_{k-1},t_k$ are known, the values $\mathbf{F}_{k-i},\dots,\mathbf{F}_{k-1},\mathbf{F}_k$ at time points $t_{k-i},\dots,t_{k-1},t_k$ are also available by formula (4). It is natural to replace the function $\mathbf{F}(t_k+\tau)$ in (7) by interpolating polynomial $\mathbf{N}(t_k+\tau)$ through the points $(t_k,\mathbf{F}_k),(t_{k-1},\mathbf{F}_{k-1}),\dots,(t_{k-i},\mathbf{F}_{k-i})$. Although any form of the interpolating polynomial can be used for the derivation, it is most convenient to use the Newton backward-difference formula. The Newton interpolating polynomial formed through $(t_k,\mathbf{F}_k),(t_{k-1},\mathbf{F}_{k-1}),\dots,(t_{k-i},\mathbf{F}_{k-i})$ can be expressed as

$$\mathbf{N}(t_k+\tau)=\sum_{s=0}^{j}(-1)^s\sum_{j=s}^{i}(-1)^j\begin{pmatrix}\frac{-\tau}{h}\\ j\end{pmatrix}\begin{pmatrix}j\\ s\end{pmatrix}\mathbf{F}_{k-s}, \tag{8}$$

where

$$\begin{pmatrix}\frac{-\tau}{h}\\ j\end{pmatrix}=\frac{\left(\frac{-\tau}{h}\right)\left(\frac{-\tau}{h}-1\right)\cdots\left(\frac{-\tau}{h}-j+1\right)}{j!},\quad\text{and}\quad\begin{pmatrix}\frac{-\tau}{h}\\ 0\end{pmatrix}=1. \tag{9}$$

Inserting (8) into (7), we obtain the $(i+1)$-step explicit EF scheme

$$\begin{aligned}\mathbf{y}_{k+1}&=\exp(\mathbf{H}h)\cdot\mathbf{y}_k+h\left[(-1)^0\sum_{j=0}^{i}\mathbf{g}_j\begin{pmatrix}j\\ 0\end{pmatrix}\right]\mathbf{F}_k+h\left[(-1)^1\sum_{j=1}^{i}\mathbf{g}_j\begin{pmatrix}j\\ 1\end{pmatrix}\right]\mathbf{F}_{k-1}\\&\quad+\cdots+h\left[(-1)^s\sum_{j=s}^{i}\mathbf{g}_j\begin{pmatrix}j\\ s\end{pmatrix}\right]\mathbf{F}_{k-s}+\cdots+h(-1)^i\mathbf{g}_i\mathbf{F}_{k-i}.\end{aligned} \tag{10}$$

Considering Eq. (9), the $(i+1)$-step explicit EF scheme with $(i+2)$-order is consequently expressed as

$$\begin{aligned}\mathbf{y}_{k+1}&=\exp(\mathbf{H}h)\cdot\mathbf{y}_k+h(\mathbf{g}_0+\mathbf{g}_1+\cdots+\mathbf{g}_i)\mathbf{F}_k-h(\mathbf{g}_1+2\mathbf{g}_2+\cdots+i\mathbf{g}_i)\mathbf{F}_{k-1}\\&\quad+\cdots+(-1)^s h\left(\mathbf{g}_s\begin{pmatrix}s\\ s\end{pmatrix}+\mathbf{g}_{s+1}\begin{pmatrix}s+1\\ s\end{pmatrix}\right.\\&\quad\left.+\cdots+\mathbf{g}_i\begin{pmatrix}i\\ s\end{pmatrix}\right)\mathbf{F}_{k-s}+\cdots+h(-1)^i\mathbf{g}_i\mathbf{F}_{k-i},\end{aligned} \tag{11}$$

where

$$\mathbf{g}_j=\int_0^1\exp\mathbf{H}h(1-\tau)\cdot(-1)^j\begin{pmatrix}-\tau\\ j\end{pmatrix}\,\mathrm{d}\tau. \tag{12}$$

We now demonstrate the calculation of the coefficients $\mathbf{g}_j$.

Let $A = \mathbf{H}^{-1}/h,\ \mathbf{T} = \exp(\mathbf{H}h)$

$$\begin{aligned}
\mathbf{g}_j &= \int_0^1 \exp \mathbf{H}h(1-\tau)\cdot(-1)^j \begin{pmatrix} -\tau \\ j \end{pmatrix} \mathrm{d}\tau \\
&= (-1)^j \left[(-\mathbf{A})\int_0^1 [\exp \mathbf{H}h(1-\tau)]' \begin{pmatrix} -\tau \\ j \end{pmatrix} \mathrm{d}\tau\right] \\
&= (-1)^j \left\{(-\mathbf{A})\left[\mathbf{I}\cdot\begin{pmatrix} -\tau \\ j \end{pmatrix} - \mathbf{T}\cdot\begin{pmatrix} -\tau \\ j \end{pmatrix} - \int_0^1 \exp \mathbf{H}h(1-\tau)\begin{pmatrix} -\tau \\ j \end{pmatrix}' \mathrm{d}\tau\right]\right\} \\
&\ \vdots \\
&= (-1)^j \sum_{s=0}^{j} (-\mathbf{A}^{s+1}) \left[\mathbf{I}\cdot\begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Bigg|_{\tau=1} - \mathbf{T}\cdot\begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Bigg|_{\tau=0}\right],
\end{aligned} \tag{13}$$

where $\begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Big|_{\tau=1}$ denotes the sth-order derivative of the function $(-\tau)(-\tau-1)\ldots(-\tau-j+1)/j!$ at the point $\tau=1$, and $\begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Big|_{\tau=0}$ is the sth-order derivative of the function $(-\tau)(-\tau-1)\ldots(-\tau-j+1)/j!$ at the point $\tau=0$. When the integral calculation of $\mathbf{g}_j$ is transformed to the sum calculation of $\mathbf{g}_j$, Eq. (11) can be used to implement the arbitrary order calculation.

Since

$$\begin{pmatrix} -\tau \\ 0 \end{pmatrix}^{(0)}\Bigg|_{\tau=1} = \begin{pmatrix} -\tau \\ 0 \end{pmatrix}^{(0)}\Bigg|_{\tau=0} = 1. \tag{14}$$

Differentiating the expression $\begin{pmatrix} -\tau \\ j \end{pmatrix}$ gives

$$\begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Bigg|_{\tau=1} = \frac{s!}{j!} B_j^{j-s}, \quad \begin{pmatrix} -\tau \\ j \end{pmatrix}^{(s)}\Bigg|_{\tau=0} = \frac{s!}{j!} B_{j-1}^{j-s}, \tag{15}$$

where the symbol B_j^{j-s} denotes the constant

$$B_j^{j-s} = \sum_{m=1}^{C_j^{j-s}} p_m \quad \text{and} \quad B_j^0 = 1. \tag{16}$$

The symbol p_m is the product of all elements in mth-combination. For example,

$$C_3^2 = 3, \quad B_3^2 = 3\times 2 + 3\times 1 + 2\times 1 = 11,$$

$$\begin{aligned}
C_5^3 = 10, \quad B_5^3 &= 5\times 4\times 3 + 5\times 4\times 2 + 5\times 3\times 2 + 4\times 3\times 2 + 5\times 4\times 1 + 5\times 3\times 1 \\
&\quad + 4\times 3\times 1 + 5\times 2\times 1 + 4\times 2\times 1 + 3\times 2\times 1 = 225.
\end{aligned}$$

Hence, from Eq. (13),

$$\mathbf{g}_0 = -\frac{\mathbf{H}^{-1}}{h}(\mathbf{I} - \mathbf{T}), \tag{17}$$

$$\mathbf{g}_j = -\frac{\mathbf{H}^{-1}}{h}\mathbf{I} - \sum_{s=1}^{j}\left(\frac{\mathbf{H}^{-1}}{h}\right)^{s+1}\left[\frac{s!}{j!}(B_j^{j-s}\mathbf{I} - B_{j-1}^{j-s}\mathbf{T})\right], \quad j = 1, \ldots, i. \tag{18}$$

Here the arbitrary-precision arithmetic is used to calculate $\mathbf{g}_j$. The local truncation error for the $(i+1)$-step explicit EF scheme is $O(h^{i+2})$.

For example, to derive the seven-step explicit EF scheme, consider Eq. (11) with $i = 6$,

The seven-step explicit scheme EF7 is, consequently,

$$\begin{aligned}\mathbf{y}_{k+1} = {} & \exp(\mathbf{H}h)\cdot\mathbf{y}_k + h(\mathbf{g}_0 + \mathbf{g}_1 + \mathbf{g}_2 + \mathbf{g}_3 + \mathbf{g}_4 + \mathbf{g}_5 + \mathbf{g}_6)\mathbf{F}_k \\ & - h(\mathbf{g}_1 + 2\mathbf{g}_2 + 3\mathbf{g}_3 + 4\mathbf{g}_4 + 5\mathbf{g}_5 + 6\mathbf{g}_6)\mathbf{F}_{k-1} \\ & + h(\mathbf{g}_2 + 3\mathbf{g}_3 + 6\mathbf{g}_4 + 10\mathbf{g}_5 + 15\mathbf{g}_6)\mathbf{F}_{k-2} \\ & - h(\mathbf{g}_3 + 4\mathbf{g}_4 + 10\mathbf{g}_5 + 20\mathbf{g}_6)\mathbf{F}_{k-3} \\ & + h(\mathbf{g}_4 + 5\mathbf{g}_5 + 15\mathbf{g}_6)\mathbf{F}_{k-4} - h(\mathbf{g}_5 + 6\mathbf{g}_6)\mathbf{F}_{k-5} + h\mathbf{g}_6\mathbf{F}_{k-6},\end{aligned} \tag{19}$$

where the coefficients $\mathbf{g}_0, \mathbf{g}_1, \ldots, \mathbf{g}_6$ can be easily evaluated by Eqs. (17) and (18).

In the foregoing, we have derived the explicit EF schemes of arbitrary order in a matrix form. It is inconvenient to implement the matrix calculation if the number of Eq. (1) n is very large. In that case, Eq. (1) is transformed equally to the following form:

$$\begin{cases} \dot{y}_1 - c_1 y_1 = F_1(t, y_1, \ldots, y_n), \\ \vdots \\ \dot{y}_m - c_m y_m = F_m(t, y_1, \ldots, y_n), \\ \vdots \\ \dot{y}_n - c_n y_n = F_n(t, y_1, \ldots, y_n), \end{cases} \tag{20}$$

where $F_m(t, y_1, \ldots, y_n) = f_m(t, y_1, \ldots, y_n) - c_m y_m$, $m = 1, 2, \ldots, n$

The $(i+1)$-step explicit EF scheme is expressed as

$$\begin{aligned} y_{m,k+1} = {} & \exp(c_m h)\cdot y_{m,k} + h(g_{m,0} + g_{m,1} + \cdots + g_{m,i})F_{m,k} - h(g_{m,1} + 2g_{m,2} + \cdots + ig_{m,i})F_{m,k-1} \\ & + \cdots + (-1)^s h\left(g_{m,s}\binom{s}{s} + g_{m,s+1}\binom{s+1}{s} + \cdots + g_{m,i}\binom{i}{s}\right)F_{m,k-s} \\ & + \cdots + h(-1)^i g_{m,i}F_{m,k-i}, \end{aligned} \tag{21}$$

where

$$g_{m,0} = -\frac{c_m^{-1}}{h}(1 - \exp(c_m h)),$$

$$g_{m,j} = -\frac{c_m^{-1}}{h} - \sum_{s=1}^{j}\left(\frac{c_m^{-1}}{h}\right)^{s+1}\left[\frac{s!}{j!}(B_j^{j-s} - B_{j-1}^{j-s}\exp(c_m h))\right], \quad j = 1,\dots,i, \ \ m = 1,\dots,n.$$

For example, the ray equation is

$$\begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}t} = p_1, \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}t} = p_2, \\ \dfrac{\mathrm{d}p_1}{\mathrm{d}t} = -c^{-3}\dfrac{\partial c}{\partial x_1}, \\ \dfrac{\mathrm{d}p_2}{\mathrm{d}t} = -c^{-3}\dfrac{\partial c}{\partial x_2}, \end{cases} \tag{22}$$

where $c(x_1, x_2) = 1 - 0.2\sin(3\pi x_1)\sin(0.5\pi x_2)$.

If the matrix form is adopted, Eq. (22) should be transformed equally to the following form:

$$\begin{bmatrix} \dfrac{\mathrm{d}x_1}{\mathrm{d}t} \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}t} \\ \dfrac{\mathrm{d}p_1}{\mathrm{d}t} \\ \dfrac{\mathrm{d}p_2}{\mathrm{d}t} \end{bmatrix} - \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -x_1 - c^{-3}\dfrac{\partial c}{\partial x_1} \\ -x_2 - c^{-3}\dfrac{\partial c}{\partial x_2} \end{bmatrix}. \tag{23}$$

If not, Eq. (22) is transformed equally to

$$\begin{cases} \dfrac{\mathrm{d}x_1}{\mathrm{d}t} - x_1 = p_1 - x_1, \\ \dfrac{\mathrm{d}x_2}{\mathrm{d}t} - x_2 = p_2 - x_2, \\ \dfrac{\mathrm{d}p_1}{\mathrm{d}t} - p_1 = -c^{-3}\dfrac{\partial c}{\partial x_1} - p_1, \\ \dfrac{\mathrm{d}p_2}{\mathrm{d}t} - p_2 = -c^{-3}\dfrac{\partial c}{\partial x_2} - p_2. \end{cases} \tag{24}$$

Table 1
The results for Eq. (25) by various methods

Method	NFCN	ERR
Four-step EF	20	4.7664e−010
Adams–Bashforth–Moulton	48	1.8886e−006
ode113	98	1.8669e−009
ode45	230	8.5109e−010

3. Examples and numerical results

In this section we test the described various order explicit multistep EF schemes on a set of examples. A multistep method normally needs the starting values. The starting values in the present multistep method are evaluated by the explicit one-step EF scheme. The following cases have been considered.

Example 1.

$$y'' - y = t. \tag{25}$$

Initial conditions are $y(0) = 1,\ y'(0) = 1$.

Introducing $\dot{y}_1 = y_2$, Eq. (25) is expressed as

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0 \\ t \end{bmatrix}. \tag{26}$$

We choose the four-step explicit EF scheme with the fixed stepsize h=0.1, and compare with Adams–Bashforth–Moulton and the solvers of Matlab. The following solvers are used for comparison: ode45 [4] and ode113 [10]. ode45 is based on an explicit Runge–Kutta (4,5) formula. ode113 is a variable order Adams–Bashforth–Moulton solver. We list the results in Table 1.

We use the following abbreviations:

NFCN: number of function evaluations

- ERR: the discrete rms error, is defined by

$$\mathrm{ERR} = \frac{\left[\sum_{l=1}^{n} (y_\mathrm{a} - y_\mathrm{e})_l^2\right]^{1/2}}{n^{1/2}},$$

y_a is a numerical solution, y_e is an exact solution.

Example 2. The second example is the motion of a spring with a cubic nonlinearity

$$y'' + y - \varepsilon y^3 = 0, \quad y(0) = 1,\ y'(0) = 0. \tag{27}$$

This system is conservative, so the solutions satisfy the energy integral

$$E = \frac{1}{2}(y^2 + y'^2) - \frac{\varepsilon}{4} y^4 \equiv \frac{1}{2} - \frac{\varepsilon}{4}. \tag{28}$$

For $\varepsilon = 10^{-4}$ the total energy E is 0.499975, which we use to assess the accuracy of the present method. The period of the oscillation is about 2π and we integrated to 100π. In Table 2, we list

Table 2
The total energy at 100π with the various step schemes

Stepsize	0.5	0.4	0.3	0.2	0.1
Third-step (EF3)	0.500390	0.500210	0.500083	0.500009	0.499979
Fourth-step (EF4)	0.500166	0.500039	0.499993	0.499977	0.499975
Fifth-step (EF4)	0.499932	0.499958	0.499970	0.499974	0.499975
Sixth-step (EF5)	0.499922	0.499961	0.499974	0.499975	0.499975
Seventh-step (EF6)	0.499966	0.499976	0.499974	0.499975	0.499975

Table 3
ERR for $t \in [0, 100\pi]$

Method	NFCN	ERR
Eighth-step EF	3142	2.5841e−008
Seventh-step EF	3142	4.9542e−008
Sixth-step EF	3142	6.7664e−008
Fifth-step EF	3142	7.5058e−007
Fourth-step EF	3142	1.5163e−006
ode113	6256	3.2641e−006
ode113	7068	7.2633e−007
ode113	7824	7.0196e−008
ode45	55202	9.7695e−008

the total energy at 100π with the various step-explicit EF schemes. To get an impression of the performance of the present method we give the discrete rms error of total energy with our method and the solvers of Matlab in Table 3.

Example 3. The third example is the ray Eq. (22).

The Hamilton values of this system satisfy the following equation:

$$H(\mathbf{x}, \mathbf{p}) = 0.5^* \{ p_1^2 + p_2^2 - 1/c^2 \}. \tag{29}$$

When initial conditions are $x_1(0) = 0.5$, $x_2(0) = 0$, $p_1(0) = \cos(1.2)$, $p_2(0) = -\sin(1.2)$, $H(x_1, x_2, p_1, p_1) \equiv 0$. We choose fixed stepsize $h = 0.0125$, and calculate the Hamilton values for $t \in [0, 2000]$ with the various order-explicit schemes. To make it clear, we only show the Hamilton values for $t \in [1900, 2000]$ in Fig. 1.

In Table 4, we list the Hamilton values at $t = 100$ with the various step-explicit EF schemes.

Example 4. Burgers equation is a useful model for many physically interesting problems, particularly those of a fluid-flow nature. Burgers equation behaves as an elliptic, parabolic or hyperbolic partial differential equation. Therefore, Burgers equation has been used widely as a model equation for testing and comparing computational techniques [7].

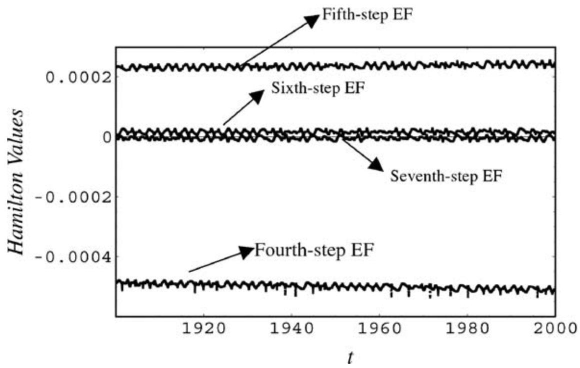

Fig. 1. The Hamilton values versus time.

Table 4
The Hamilton values H for Eq. (29) at $t = 100$

Method	H
Four-step EF	−6.52379e−005
Five-step EF	3.78435e−005
Six-step EF	4.22052e−006
Seven-step EF	−1.32372e−006
Eight-step EF	−8.08752e−007

The first problem:

We consider the following Burgers equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \varepsilon\frac{\partial^2 u}{\partial x^2}, \quad 0 < x < 1, \ t > 0, \tag{30}$$

Whose analytical solution is known (see [5]), the initial and boundary conditions are taken from analytical solution. We put $x_i = i\Delta x$ $(i = 1, 2, \ldots, m-1)$ and define $u_i(t) = u(x_i, t)$

Discretizing the derivatives with respect to the space variables in (30), we obtain

$$\frac{\mathrm{d}u_i(t)}{\mathrm{d}t} - \left(-\frac{2\varepsilon}{(\Delta x)^2}\right)u_i(t) = \frac{\varepsilon}{(\Delta x)^2}[u_{i+1}(t) + u_{i-1}(t)] - u_i(t)\,\frac{u_{i+1}(t) - u_{i-1}(t)}{2(\Delta x)}. \tag{31}$$

We choose grid size $\Delta x = 0.1$ for $\varepsilon = 0.1$, and $\Delta x = 0.01$ for $\varepsilon = 0.003$, and Eq. (30) is respectively transformed into a system of 9 ODEs and 99 ODEs. Let $\lambda = h/(\Delta x)^2$, where λ is the Courant number. Here we choose four-step explicit EF scheme and compare the results with [5,11], see Tables 5 and 6.

The second problem:

We consider the following Burgers equation [7]

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \varepsilon\frac{\partial^2 u}{\partial x^2}. \tag{32}$$

Table 5
The absolute error of numerical solution for Eq. (30), $t = 1.0$, $\varepsilon = 0.1$, $m = 10$

x_i	Present method $\lambda = 2.5$	(D) AGE [5] $\lambda = 1.0$	PR AGE-IMP [5] $\lambda = 1.0$	DR AGE-CN [5] $\lambda = 1.0$	PARALLEL [11] $\lambda = 2.5$
0.1	1.59e−004	1.83e−004	2.9e−002	1.60e−004	4.23e−005
0.2	2.82e−004	3.26e−004	6.1e−002	2.66e−004	2.57e−004
0.3	2.68e−004	3.88e−004	9.4e−002	2.59e−004	6.15e−004
0.4	1.27e−004	2.93e−004	1.236e−001	7.17e−005	1.69e−004
0.5	3.17e−004	9.60e−005	1.4498e−001	3.3e−004	2.64e−003
0.6	7.83e−004	6.94e−004	1.542e−001	8.94e−004	3.78e−003
0.7	1.40e−003	1.19e−003	1.4769e−001	1.45e−003	4.44e−003
0.8	1.60e−003	1.51e−003	1.2229e−001	1.71e−003	4.46e−003
0.9	1.30e−003	1.22e−003	7.485e−002	1.34e−003	3.09e−003

Table 6
The numerical solutions for Eq. (30), $t = 0.5$, $\varepsilon = 0.003$, $m = 100$

x_i	PARALLEL [11] $\lambda = 10$	Present method $\lambda = 20$	(D) AGE [5] $\lambda = 1$	PR AGE-IMP [5] $\lambda = 1$	DR AGE-CN [5] $\lambda = 1$	Exact solution
0.1	1.0000	1.000000	1.0000	0.999999	1.000000	1.000000
0.2	1.0000	1.000000	1.0000	0.999999	0.999999	1.000000
0.3	1.0000	1.000000	1.0000	0.999995	0.999999	1.000000
0.4	1.0000	1.000000	1.0000	0.992646	0.999999	1.000000
0.5	1.0000	0.999996	1.0000	0.620453	1.000001	0.999986
0.6	0.9336	0.951840	0.9552	0.360375	0.953063	0.941313
0.7	0.1154	0.114505	0.1145	0.109650	0.114373	0.113837
0.8	0.1000	0.100027	0.1000	0.100049	0.100026	0.100018
0.9	0.1000	0.100000	0.1000	0.100000	0.100000	0.100000

The solution will be sought in the region $-1 \leqslant x \leqslant 1$ for $t \geqslant 0$. Initial and boundary conditions are taken to be

$$u_0(x) = u(x,0) = \begin{cases} 1 & \text{if } -1 \leqslant x \leqslant 0, \\ 0 & \text{if } 0 < x \leqslant 1, \end{cases}$$

$$u(-1,t) = 1, \quad u(1,t) = 0.$$

Here we solve Eq. (32) for $\varepsilon = 0.01$ at $t = 0.92$. We choose five-step explicit EF scheme, grid size $\Delta x = 0.025$ and time step $h = 0.01$, while the time step $h = 0.001$ in [7]. We list the results in Table 7, and compare them with [7] and the exact solution.

Table 7
The numerical solutions of Eq. (32), $t = 0.92$, $\varepsilon = 0.01$

x_i	Present method	Exact solution	Traditional Galerkin [7]	Ref. [7]
−0.9	1.0000	1.0000	0.9956	1.0000
−0.8	1.0000	1.0000	1.0456	1.0000
−0.7	1.0000	1.0000	1.0672	1.0000
−0.6	1.0000	1.0000	1.0402	1.0000
−0.5	1.0000	1.0000	0.9831	1.0000
−0.4	1.0000	1.0000	0.9303	1.0000
−0.3	1.0000	1.0000	0.9128	1.0000
−0.2	1.0000	1.0000	0.9444	1.0000
−0.1	1.0000	1.0000	1.0159	1.0000
0	1.0000	1.0000	1.0963	1.0000
0.1	1.0000	1.0000	1.1411	1.0000
0.2	1.0000	1.0000	1.1057	1.0000
0.3	0.9999	0.9998	0.9613	0.9995
0.4	1.0043	0.9714	0.7099	0.9723
0.5	0.1428	0.1861	0.3933	0.2034
0.6	0.0017	0.0015	0.0905	0.0004
0.7	0.0000	0.0000	−0.1017	0.0000
0.8	0.0000	0.0000	−0.1154	0.0000
0.9	0.0000	0.0000	0.0091	0.0000

4. Conclusion

We have developed the various order-explicit multistep EF schemes and carried out higher-order schemes. The present schemes can be applied to systems whose linear part is nondiagonal. We have demonstrated the accuracy and efficacy of the new schemes via application to a variety of test cases.

Future areas of further development include the implicit multistep EF schemes and predictor–corrector EF schemes. This work is currently underway.

Acknowledgements

This work was supported by State Key Laboratory of Vibration, Shock and Noise, Shanghai Jiaotong University (Grant No: VSN-2003-03) and 985 Education Development Plan of Tianjin University.

References

[1] G.V. Berghe, L.G. Ixaru, H.D. Meyer, Frequency determination and step-length control for exponentially-fitted Runge–Kutta methods, J. Comput. Appl. Math. 132 (2001) 95–105.
[2] G. Beylkin, J.M. Keiser, L. Vozovoiy, A new class of time discretization schemes for the solution of nonlinear PDEs, J. Comput. Phys. 147 (1998) 362–387.
[3] S.M. Cox, P.C. Matthews, Exponential time differencing for stiff systems, J. Comput. Phys. 176 (2002) 430–455.

[4] J.R. Dormand, P.J. Prince, A family of embedded Runge–Kutta formulae, J. Comput. Appl. Math. 6 (1980) 19–26.
[5] D.J. Evans, M.S. Sahimi, The numerical solution of Burgers equations by the alternating group explicit (AGE) method, Intern. J. Comput. Math. 29 (1989) 39–64.
[6] R. Holland, Finite-difference time-domain (FDTD) analysis of magnetic diffusion, IEEE Trans. Electromagn. Compat. 36 (1994) 32–39.
[7] E.B. Lin, X. Zhou, Connection coefficients on an interval and wavelet solutions of Burgers equation, J. Comput. Appl. Math. 135 (2001) 63–78.
[8] B.J. McCartin, R.J. Mitchell, A numerical model of chaotic respiration, Proceedings of the Fourth Conference on Mathematical Modeling in the Undergraduate Curriculum, University of Wisconsin, La Crosse, 2000.
[9] B.J. McCartin, Exponential fitting of the delayed recruitment/renewal equation, J. Comput. Appl. Math. 136 (2001) 343–356.
[10] L.F. Shampine, M.K. Gordon, Computer Solution of Ordinary Differential Equations: The Initial Value Problem, W. H. Freeman, San Francisco, 1975.
[11] W.Q. Wang, A parallel computing method for Burgers equation, Chinese J. Computat. Phys. 18 (2001) 385–389 (in Chinese).