



ELSEVIER

Applied Numerical Mathematics 31 (1999) 227–238



APPLIED
NUMERICAL
MATHEMATICS

www.elsevier.nl/locate/apnum

Runge–Kutta methods for linear ordinary differential equations

D.W. Zingg*, T.T. Chisholm

Institute for Aerospace Studies, University of Toronto, 4925 Dufferin Street, Downsview, Ontario, Canada M3H 5T6

Abstract

Three new Runge–Kutta methods are presented for numerical integration of systems of linear inhomogeneous ordinary differential equations (ODEs) with constant coefficients. Such ODEs arise in the numerical solution of partial differential equations governing linear wave phenomena. The restriction to linear ODEs with constant coefficients reduces the number of conditions which the coefficients of the Runge–Kutta method must satisfy. This freedom is used to develop methods which are more efficient than conventional Runge–Kutta methods. A fourth-order method is presented which uses only two memory locations per dependent variable, while the classical fourth-order Runge–Kutta method uses three. This method is an excellent choice for simulations of linear wave phenomena if memory is a primary concern. In addition, fifth- and sixth-order methods are presented which require five and six stages, respectively, one fewer than their conventional counterparts, and are therefore more efficient. These methods are an excellent option for use with high-order spatial discretizations. © 1999 Elsevier Science B.V. and IMACS. All rights reserved.

Keywords: Runge–Kutta methods; Linear differential equations; Ordinary differential equations

1. Introduction

We consider the numerical integration of large linear inhomogeneous systems of ordinary differential equations in the form

$$\frac{d\mathbf{u}}{dt} = \mathbf{A}\mathbf{u} - \mathbf{g}(t), \quad (1)$$

where \mathbf{A} is an M by M matrix whose elements depend on neither \mathbf{u} nor t , and \mathbf{u} and $\mathbf{g}(t)$ are vectors of length M . Such essentially autonomous systems arise in the numerical solution of partial differential equations (PDEs) governing linear wave phenomena after application of a spatial discretization such as a finite-difference, finite-volume, or finite-element method. Examples of such PDEs are the linearized Euler equations governing acoustic waves and the Maxwell equations governing electromagnetic waves. The elements of \mathbf{A} depend on the PDE and the spatial discretization. The inhomogeneous term $\mathbf{g}(t)$ is

* Corresponding author. E-mail: dwz@oddjob.utias.utoronto.ca.

associated with either a source term or the boundary conditions. In the context of wave propagation, the system of ODEs is often mildly stiff with the eigenvalues of \mathbf{A} typically lying near the imaginary axis.

The system of ODEs arising from the application of a spatial discretization to a system of PDEs can be very large, especially in three-dimensional simulations. Consequently, the constraints on the methods used for integrating these systems are somewhat different from those which have driven much of the development of numerical methods for initial value problems. Due to their high accuracy and modest memory requirements, explicit Runge–Kutta methods have become popular for simulations of wave phenomena [7–9,18,20]. The classical fourth-order Runge–Kutta method requires three memory locations per dependent variable [1,6], but low-storage methods requiring only two memory locations per dependent variable can be derived [4,16,17]. This property is easily achieved by a third-order Runge–Kutta method [17], but an additional stage is required for a fourth-order method [4]. Since the primary cost of the integration is in the evaluation of the derivative function, and each stage requires a function evaluation, the additional stage represents a significant increase in expense. For the same reason, error checking is generally not performed when solving very large systems of ODEs arising from the discretization of PDEs.

There have been several attempts to develop efficient methods for integrating linear systems of ODEs [5,12–14]. The basic premise of these methods is that the major cost in evaluating the derivative function is in forming the matrix \mathbf{A} and the vector $\mathbf{g}(t)$. In the application considered here, the simulation of linear wave phenomena, the matrix \mathbf{A} is never explicitly formed or stored. Hence the methods previously proposed for linear systems are not appropriate for this application.

It is well known that a Runge–Kutta method with p stages has an order of accuracy not exceeding p [2, 3]. For $p \leq 4$, methods of order p can be derived with p stages. However, fifth- and sixth-order methods require at least six and seven stages, respectively. Nine stages are required for seventh-order accuracy and eleven for eighth-order accuracy [2]. Since the cost for our application is roughly proportional to the number of stages, this represents a significant limitation of higher-order Runge–Kutta methods.

Several authors have considered various approximations to reduce the number of stages and the storage requirements of high-order Runge–Kutta methods. Shanks [15] was able to develop schemes with a reduced number of stages by requiring only that the accuracy conditions be approximately satisfied. Zingg et al. [19,20] propose methods with low storage requirements which are of high order for linear homogeneous ODEs but second-order otherwise. A similar idea was proposed previously by Lorenz [11].

In this paper, we develop Runge–Kutta methods specifically for linear ODEs with constant coefficients. By removing the constraints imposed by nonlinearity in the derivative function, high-order Runge–Kutta methods can be derived which are more efficient in some respect than the classical methods. In the next section, we present a fourth-order method which requires less memory than the classical fourth-order Runge–Kutta method. We then present fifth- and sixth-order methods requiring fewer derivative function evaluations per time step than fifth- and sixth-order Runge–Kutta methods applicable to nonlinear problems.

2. General form of an explicit Runge–Kutta method

Without loss of generality, we consider the following scalar ODE:

$$\frac{du}{dt} = f(t, u). \quad (2)$$

A general p -stage explicit Runge–Kutta method can be written as

$$\begin{aligned} k_1 &= f(t_n, u_n), \\ k_i &= f\left(t_n + c_i h, u_n + h \sum_{j=1}^{i-1} a_{ij} k_j\right), \quad i = 2, \dots, p, \\ u_{n+1} &= u_n + h \sum_{i=1}^p b_i k_i, \end{aligned} \quad (3)$$

where $h = \Delta t$ is the time step, $t_n = nh$, and u_n is an approximation to $u(t_n)$.

3. Low-storage fourth-order method

We consider first the case $p = 4$. With the constraints

$$\begin{aligned} c_2 &= a_{21}, \\ c_3 &= a_{31} + a_{32}, \\ c_4 &= a_{41} + a_{42} + a_{43} \end{aligned} \quad (4)$$

there remain ten parameters. For fourth-order accuracy, there are eight conditions which must be satisfied. Four of these arise even for linear homogeneous constant-coefficient ODEs. A further three conditions must be met if the ODEs are inhomogeneous. The final condition is associated with non-constant coefficients or nonlinearity. Therefore, fourth-order Runge–Kutta methods are a two-parameter family of which the classical method is a particular choice.

If we restrict our attention to linear constant-coefficient ODEs, the number of conditions is reduced to seven. They are

$$\begin{aligned} \sum_{i=1}^4 b_i &= 1, \\ \sum_{i=2}^4 c_i b_i &= \frac{1}{2}, \\ c_2 a_{32} b_3 + b_4 (c_2 a_{42} + c_3 a_{43}) &= \frac{1}{6}, \\ c_2 a_{32} a_{43} b_4 &= \frac{1}{24}, \\ \sum_{i=2}^4 b_i c_i^2 &= \frac{1}{3}, \\ \sum_{i=2}^4 b_i c_i^3 &= \frac{1}{4}, \\ b_3 c_2^2 a_{32} + b_4 (c_2^2 a_{42} + c_3^2 a_{43}) &= \frac{1}{12}. \end{aligned} \quad (5)$$

These conditions can be derived in the same manner as the classical conditions. The eighth condition is eliminated by exploiting the fact that, for linear ODEs,

$$\frac{\partial^2 f}{\partial u^2} = \frac{\partial^2 f}{\partial u \partial t} = 0.$$

The reduction in the number of conditions to be satisfied does not permit us to reduce the number of stages. However, we can obtain reduced storage requirements.

Following the approach of Wray [17], the requirement that only two memory locations be used imposes the following three constraints:

$$\begin{aligned} b_1 &= a_{41} = a_{31}, \\ b_2 &= a_{42}. \end{aligned} \tag{6}$$

With these constraints, only two memory locations are required for both the dependent variable and the value of the time derivative. Hence the method requires minimal storage even when compact or spectral methods are used for the spatial discretization. With the memory locations denoted A and B , the method proceeds as follows:

1. Initially, u_n is stored in A , and B is empty.
2. The term $k_1 = f(t_n, u_n)$ is evaluated and stored in B .
3. The quantity $u_n + ha_{31}k_1$, is calculated and stored in A .
4. The quantity $u_n + ha_{21}k_1$ is calculated and stored in B .
5. The term $k_2 = f(t_n + c_2h, u_n + ha_{21}k_1)$ is evaluated and stored in B .
6. The contents of the two memory locations are linearly combined to form $u_n + h(a_{31}k_1 + a_{32}k_2)$, which is stored in B .
7. With $a_{41} = a_{31}$, another linear combination gives $u_n + h(a_{41}k_1 + a_{42}k_2)$, which is stored in A .
8. The term $k_3 = f[t_n + c_3h, u_n + h(a_{31}k_1 + a_{32}k_2)]$ is evaluated and stored in B .
9. The contents of the two memory locations are linearly combined to form $u_n + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)$, which is stored in B .
10. With $b_1 = a_{41}$ and $b_2 = a_{42}$, another linear combination gives $u_n + h(b_1k_1 + b_2k_2 + b_3k_3)$, which is stored in A .
11. The term $k_4 = f[t_n + c_4h, u_n + h(a_{41}k_1 + a_{42}k_2 + a_{43}k_3)]$ is evaluated and stored in B .
12. The contents of the two memory locations are linearly combined to form u_{n+1} .

With the additional constraints imposed by the low-storage requirement, we are left with seven parameters to satisfy the seven conditions given in Eq. (5). Although this system may possess more than one solution, the only solution we have found is

$$\begin{aligned} a_{21} &= c_2 = 0.69631521002413, & c_3 &= 0.29441651741, \\ c_4 &= 0.82502163765, & b_1 &= a_{41} = a_{31} = 0.07801567728325, \\ a_{32} &= 0.21640084013679, & b_2 &= a_{42} = 0.04708870117112, \\ a_{43} &= 0.69991725920066, & b_3 &= 0.47982272993855, \\ b_4 &= 0.39507289160708. \end{aligned}$$

4. Five-stage fifth-order method

For the case $p = 5$, we have, in addition to the constraints given in Eq. (4), the following condition:

$$c_5 = a_{51} + a_{52} + a_{53} + a_{54}. \tag{7}$$

Consequently, adding the fifth stage has produced five additional parameters for a total of fifteen. The coefficients must satisfy the following eleven conditions in order to produce fifth-order accuracy for linear constant-coefficient ODEs:

$$\begin{aligned} \sum_{i=1}^5 b_i &= 1, \\ \sum_{i=2}^5 c_i b_i &= \frac{1}{2}, \\ c_2 a_{32} b_3 + b_4 (c_2 a_{42} + c_3 a_{43}) + b_5 (c_2 a_{52} + c_3 a_{53} + c_4 a_{54}) &= \frac{1}{6}, \\ c_2 a_{32} a_{43} b_4 + b_5 [c_2 (a_{42} a_{54} + a_{32} a_{53}) + c_3 a_{43} a_{54}] &= \frac{1}{24}, \\ c_2 a_{32} a_{43} a_{54} b_5 &= \frac{1}{120}, \\ \sum_{i=2}^5 b_i c_i^2 &= \frac{1}{3}, \\ \sum_{i=2}^5 b_i c_i^3 &= \frac{1}{4}, \\ b_3 c_2^2 a_{32} + b_4 (c_2^2 a_{42} + c_3^2 a_{43}) + b_5 (c_2^2 a_{52} + c_3^2 a_{53} + c_4^2 a_{54}) &= \frac{1}{12}, \\ \sum_{i=2}^5 b_i c_i^4 &= \frac{1}{5}, \\ (b_4 a_{42} + b_3 a_{32} + b_5 a_{52}) c_2^3 + (b_5 a_{53} + b_4 a_{43}) c_3^3 + b_5 a_{54} c_4^3 &= \frac{1}{20}, \\ b_5 [a_{54} (a_{42} c_2^2 + a_{43} c_3^2) + a_{53} a_{32} c_2^2] + b_4 a_{43} a_{32} c_2^2 &= \frac{1}{60}. \end{aligned} \tag{8}$$

Thus a four-parameter family of solutions is obtained. Several different criteria can be applied in order to choose a method from this family. The following values have been found by minimizing the L_2 norm of a vector containing the coefficients of the method:

$$\begin{aligned} a_{21} = c_2 = 0.21, & \quad c_3 = 0.43, & \quad c_4 = 0.68, & \quad c_5 = 0.85, \\ a_{32} = 0.47418546365915, & & a_{42} = 0.13437223603429, & \\ a_{52} = 0.26302355344001, & & b_2 = 0.16574368303091, & \\ a_{43} = 0.57068167533284, & & a_{53} = 0.10434139625551, & \\ b_3 = 0.41041645692809, & & a_{54} = 0.39377303853165, & \\ b_4 = -0.04092124960122, & & b_5 = 0.37240141154501, & \\ b_1 = 0.09235969809721 & & & \end{aligned}$$

with

$$a_{31} = c_3 - a_{32}, \quad a_{41} = c_4 - a_{42} - a_{43}, \quad a_{51} = c_5 - a_{52} - a_{53} - a_{54}.$$

5. Six-stage sixth-order method

With $p = 6$, the following condition must be satisfied in addition to the constraints given in Eqs. (4) and (7):

$$c_6 = a_{61} + a_{62} + a_{63} + a_{64} + a_{65}. \quad (9)$$

Therefore, there remain twenty-one free coefficients. The requirement of sixth-order accuracy for linear constant-coefficient ODEs produces the following sixteen conditions:

$$\begin{aligned} \sum_{i=1}^6 b_i &= 1, & \sum_{i=2}^6 c_i b_i &= \frac{1}{2}, \\ c_2 a_{32} b_3 + b_4 (c_2 a_{42} + c_3 a_{43}) + b_5 (c_2 a_{52} + c_3 a_{53} + c_4 a_{54}) + b_6 (c_2 a_{62} + c_3 a_{63} + c_4 a_{64} + c_5 a_{65}) &= \frac{1}{6}, \\ c_2 a_{32} a_{43} b_4 + b_5 [c_2 (a_{42} a_{54} + a_{32} a_{53}) + c_3 a_{43} a_{54}] \\ &+ b_6 [a_{65} (a_{54} c_4 + a_{53} c_3 + a_{52} c_2) + a_{64} (a_{43} c_3 + a_{42} c_2) + a_{63} a_{32} c_2] = \frac{1}{24}, \\ b_6 (a_{65} [a_{54} (a_{43} c_3 + a_{42} c_2) + a_{53} a_{32} c_2] + a_{64} a_{43} a_{32} c_2) + b_5 a_{54} a_{43} a_{32} c_2 &= \frac{1}{120}, \\ c_2 a_{32} a_{43} a_{54} a_{65} b_6 &= \frac{1}{720}, \\ \sum_{i=2}^6 b_i c_i^2 &= \frac{1}{3}, & \sum_{i=2}^6 b_i c_i^3 &= \frac{1}{4}, \\ b_3 c_2^2 a_{32} + b_4 (c_2^2 a_{42} + c_3^2 a_{43}) + b_5 (c_2^2 a_{52} + c_3^2 a_{53} + c_4^2 a_{54}) \\ &+ b_6 (a_{65} c_5^2 + a_{64} c_4^2 + a_{63} c_3^2 + a_{62} c_2^2) = \frac{1}{12}, \quad (10) \\ \sum_{i=2}^6 b_i c_i^4 &= \frac{1}{5}, \\ (b_4 a_{42} + b_3 a_{32} + b_5 a_{52} + b_6 a_{62}) c_2^3 + (b_5 a_{53} + b_4 a_{43} + b_6 a_{63}) c_3^3 \\ &+ (b_5 a_{54} + b_6 a_{64}) c_4^3 + b_6 a_{65} c_5^3 = \frac{1}{20}, \\ b_6 [a_{65} (a_{52} c_2^2 + a_{53} c_3^2 + a_{54} c_4^2) + a_{64} (a_{42} c_2^2 + a_{43} c_3^2) + a_{63} a_{32} c_2^2] \\ &+ b_5 [a_{54} (a_{42} c_2^2 + a_{43} c_3^2) + a_{53} a_{32} c_2^2] + b_4 a_{43} a_{32} c_2^2 = \frac{1}{60}, \\ \sum_{i=2}^6 b_i c_i^5 &= \frac{1}{6}, \\ b_6 (a_{62} c_2^4 + a_{63} c_3^4 + a_{64} c_4^4 + a_{65} c_5^4) + b_5 (a_{52} c_2^4 + a_{53} c_3^4 + a_{54} c_4^4) + b_4 (a_{42} c_2^4 + a_{43} c_3^4) + b_3 a_{32} c_2^4 &= \frac{1}{30}, \\ c_2^3 [b_6 (a_{65} a_{52} + a_{64} a_{42} + a_{63} a_{32}) + b_5 (a_{54} a_{42} + a_{53} a_{32}) + b_4 a_{43} a_{32}] \\ &+ c_3^3 [b_6 (a_{65} a_{53} + a_{64} a_{43}) + b_5 a_{54} a_{43}] + c_4^3 b_6 a_{65} a_{54} = \frac{1}{120}, \\ b_6 [a_{65} a_{54} (a_{43} c_3^2 + a_{42} c_2^2) + a_{64} a_{43} a_{32} c_2^2] + b_5 a_{54} a_{43} a_{32} c_2^2 &= \frac{1}{360}. \end{aligned}$$

Using the same criterion as for the fifth-order method, the following coefficients have been chosen from the five-parameter family of solutions to the above conditions:

$$\begin{aligned}
 a_{21} = c_2 = 0.15, & \quad c_3 = 0.36, & \quad c_4 = 0.57, & \quad c_5 = 0.75, & \quad c_6 = 0.90, \\
 a_{32} = 0.45818181818182, & \quad a_{42} = 0.09769454545455, \\
 a_{52} = 0.10861879806510, & \quad a_{62} = 0.20874226393025, \\
 b_2 = 0.24971305394585, & \quad a_{43} = 0.48766666666667, \\
 a_{53} = 0.04655817933320, & \quad a_{63} = 0.12686271445897, \\
 b_3 = 0.11278150363005, & \quad a_{54} = 0.44703799502007, \\
 a_{64} = 0.02734417934727, & \quad b_4 = 0.35718962665957, \\
 a_{65} = 0.37591957583530, & \quad b_5 = -0.00478351095633, \\
 b_6 = 0.24659027402511, & \quad b_1 = 0.03850905269576
 \end{aligned}$$

with

$$\begin{aligned}
 a_{31} = c_3 - a_{32}, & \quad a_{41} = c_4 - a_{42} - a_{43}, \\
 a_{51} = c_5 - a_{52} - a_{53} - a_{54}, & \quad a_{61} = c_6 - a_{62} - a_{63} - a_{64} - a_{65}.
 \end{aligned}$$

6. Stability contours

The stability contours of the three new methods are shown in Fig. 1. Satisfaction of the first four conditions in Eq. (5) ensures that the new fourth-order method has the same stability contour as the classical fourth-order Runge–Kutta method. Similarly, the stability contours of the five-stage fifth-order

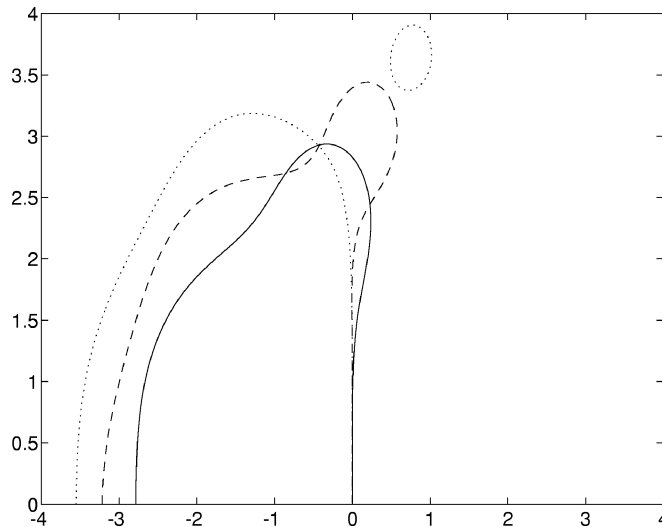


Fig. 1. Stability contours for the fourth-order (—), fifth-order (---), and sixth-order (· · ·) methods.

method and the six-stage sixth-order method are uniquely defined and do not depend on which members of the respective families are selected.

Although the stable regions of the fifth- and sixth-order methods are somewhat larger than that of the fourth-order method, the increase is not sufficient to compensate for the cost of the additional stages. Therefore, the fourth-order method is a better choice if the time step is limited by stability considerations. It should be emphasized that the present fourth-order method has exactly the same properties with respect to stiff ODEs as the classical fourth-order Runge–Kutta method.

The stable regions of the fifth- and sixth-order methods do not include the imaginary axis. Systems with pure imaginary eigenvalues are obtained when central differencing is applied to the spatial derivatives in partial differential equations governing wave propagation phenomena with no physical dissipation, in the absence of boundary conditions. However, Zingg et al. [20] have demonstrated that by adding a small amount of numerical dissipation to the spatial discretization, stable schemes can be obtained using such methods. The amount of dissipation required is sufficiently low that the overall accuracy of the scheme is not compromised. The stability contour of the method successfully used in [9] for simulations of the propagation and scattering of electromagnetic waves is identical to that of the present sixth-order method.

7. Fourier error analysis

Using Fourier analysis we can determine the errors produced by an integration method when applied to a linear homogeneous ODE. Since our interest is in wave propagation, we consider a scalar ODE of the form

$$\frac{du}{dt} = i\omega u, \quad (11)$$

where ω is a real constant. The Runge–Kutta methods developed here produce a solution in the form

$$u_n = \sigma^n u_0, \quad (12)$$

where

$$\sigma = \sum_{k=0}^p \frac{1}{k!} (i\omega)^k \quad (13)$$

and p is the number of stages. The local amplitude and phase errors are determined from σ as follows:

$$\text{er}_a = |\sigma| - 1, \quad (14)$$

$$\text{er}_p = -\frac{\tan^{-1}(\sigma_i/\sigma_r)}{\omega h} + 1, \quad (15)$$

where σ_r and σ_i denote the real and imaginary parts of σ .

Figs. 2 and 3 show the local amplitude and phase errors produced by the three new methods. In order to account for the number of stages, the errors are plotted versus $\omega h/p$. Hence the errors shown are for approximately equal computational effort. Since the time step is thus proportional to p , the amplitude error shown is $|\sigma|^{1/p} - 1$. The figures show that each increase in the order of the method produces an increase in accuracy even though the extra work has been accounted for. Hence the fifth- and sixth-order methods can be more efficient than the fourth-order method if a sufficiently accurate spatial discretization is used.

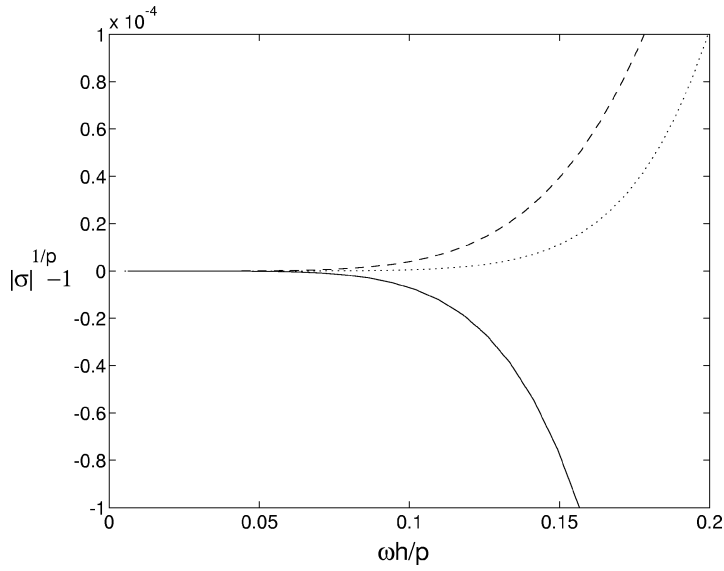


Fig. 2. Amplitude error produced by the fourth-order (—), fifth-order (---), and sixth-order (· · ·) methods.

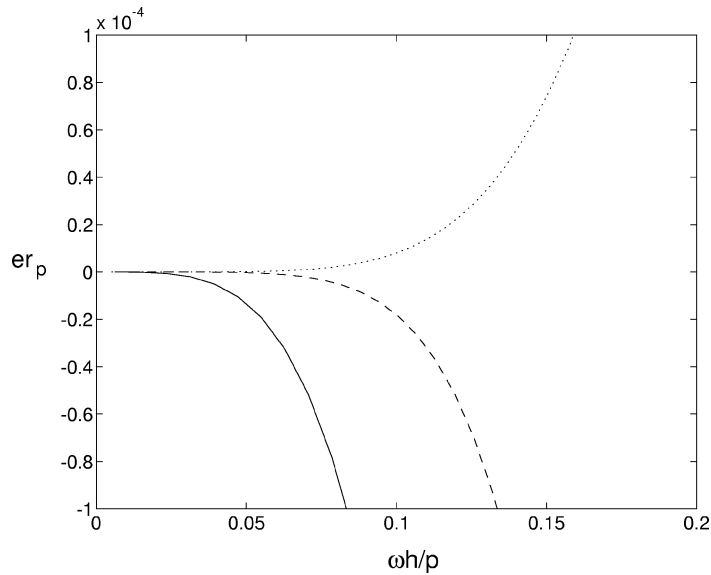


Fig. 3. Phase error produced by the fourth-order (—), fifth-order (---), and sixth-order (· · ·) methods.

8. Application to the numerical solution of a PDE

We now apply the new methods to the numerical solution of the linear convection equation given by

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \tag{16}$$

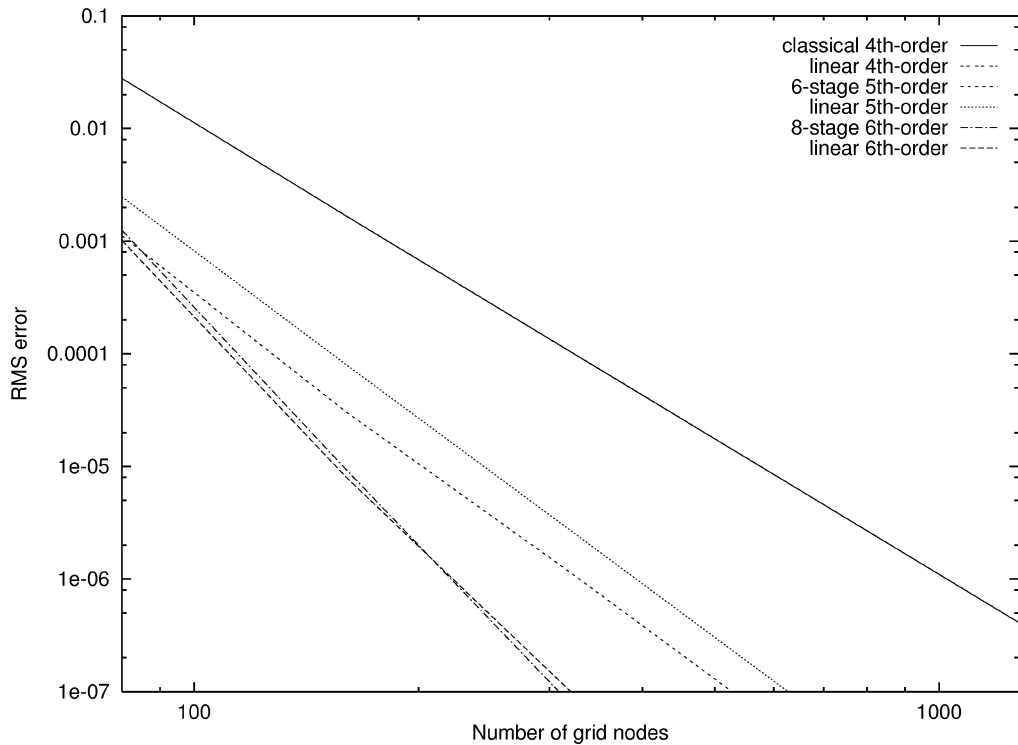


Fig. 4. Root-mean-square error as a function of the number of grid nodes for the numerical solution of the linear convection equation.

where $u = u(x, t)$ and $0 \leq x \leq 1$. This equation is commonly used as a model equation in the study of numerical algorithms for the solution of hyperbolic systems of PDEs. The boundary condition is $u(0, t) = \sin \omega t$ with $\omega = 16\pi$. We run until a periodic steady state is obtained. The spatial operator is seventh-order, with a nine-point stencil and suitable boundary operators. The error is displayed as a function of the number of grid nodes in Fig. 4. The ratio $\Delta t / \Delta x$ is unity in all cases. The three methods presented here are designated “linear”. In addition, the classical fourth-order Runge–Kutta method is shown, along with a six-stage fifth-order method [3, p. 202] and an eight-stage sixth-order method [10, p. 122]. The error produced by the new low-storage fourth-order method is indistinguishable from that produced by the classical fourth-order method. The linear fifth-order method produces somewhat more error than the six-stage fifth-order method, but is much more accurate than the fourth-order methods. The two sixth-order methods produce comparable errors, with the linear method requiring two fewer stages. Overall, these results confirm the validity and usefulness of the new methods for use in the numerical solution of linear systems of hyperbolic PDEs, such as the time-domain Maxwell equations.

9. Conclusions

Three new Runge–Kutta methods have been presented for the integration of linear systems of ODEs with constant coefficients. If the time step size is limited by stability, then the new fourth-order method

is the most suitable of the new methods. This method requires less memory than the classical fourth-order Runge–Kutta method and less computational effort than the low-storage method proposed in [4]. In any application to linear ODEs where the classical fourth-order Runge–Kutta method is currently the method of choice, the new fourth-order method provides the same stability characteristics and equivalent accuracy with reduced memory requirements. If the time step is limited by accuracy, and memory is a secondary concern, then the new fifth- and sixth-order methods present an efficient new alternative. Since the expense of the methods is roughly proportional to the number of stages for the problems of interest here, the new fifth- and sixth-order methods are significantly more efficient than their counterparts for nonlinear ODEs. The sixth-order method is a particularly good choice for use with high-order spatial discretizations in the numerical solution of partial differential equations governing linear wave phenomena.

References

- [1] E.K. Blum, A modification of the Runge–Kutta fourth-order method, *Math. Comp.* 16 (1962) 176–187.
- [2] J.C. Butcher, The non-existence of ten stage eighth order explicit Runge–Kutta methods, *BIT* 25 (1985) 521–540.
- [3] J.C. Butcher, *The Numerical Analysis of Ordinary Differential Equations* (Wiley, New York, 1987).
- [4] M.H. Carpenter and C.A. Kennedy, Fourth-order $2N$ -storage Runge–Kutta schemes, NASA TM 109112 (June 1994).
- [5] W.H. Enright, The efficient solution of linear constant-coefficient systems of differential equations, *Simulation* 30 (1978) 129–133.
- [6] S. Gill, A process for the step-by-step integration of differential equations in an automatic digital computing machine, *Proc. Cambridge Philos. Soc.* 47 (1951) 96–108.
- [7] Z. Haras and S. Ta’Asan, Finite-difference schemes for long-time integration, *J. Comput. Phys.* 114 (1994) 265–279.
- [8] F.Q. Hu, M.Y. Hussaini and J.L. Mantney, Low-dissipation and low-dispersion Runge–Kutta schemes for computational acoustics, *J. Comput. Phys.* 124 (1996) 177–191.
- [9] H. Jurgens and D.W. Zingg, Implementation of a high-accuracy finite-difference scheme for linear wave phenomena, *Houston J. Math.* (July 1996).
- [10] J.D. Lambert, *Computational Methods in Ordinary Differential Equations* (Wiley, New York, 1973).
- [11] E.N. Lorenz, An N -cycle time-differencing scheme for stepwise numerical integration, *Monthly Weather Rev.* 99 (1971) 644–648.
- [12] L. Ranzani, Optimizing the numerical integration of initial value problems in shooting methods for linear boundary value problems, *SIAM J. Sci. Comput.* 14 (1993) 860–871.
- [13] L.F. Shampine, Cheaper integration of linear systems, *Simulation* 20 (1973) 17.
- [14] L.F. Shampine, Solution of structured non-stiff ODEs, *J. Comput. Appl. Math.* 15 (1986) 293–300.
- [15] E.B. Shanks, Higher-order approximations of Runge–Kutta type, NASA TN D-2920 (September 1965).
- [16] J.H. Williamson, Low-storage Runge–Kutta schemes, *J. Comput. Phys.* 35 (1980) 48–56.
- [17] A.A. Wray, Minimal storage time advancement schemes for spectral methods, Manuscript, NASA Ames Research Center (unpublished).
- [18] D.W. Zingg, A review of high-order and optimized finite-difference methods for simulating linear wave phenomena, AIAA Paper 97-2088, in: *Proceedings of the 13th AIAA Computational Fluid Dynamics Conference*, American Institute for Aeronautics and Astronautics (June 1997).

- [19] D.W. Zingg and H. Lomax, Some aspects of high-order numerical solutions of the linear convection equation with forced boundary conditions, AIAA Paper 93-3381, in: *Proceedings of the 11th AIAA Computational Fluid Dynamics Conference*, American Institute for Aeronautics and Astronautics (June 1993).
- [20] D.W. Zingg, H. Lomax and H.M. Jurgens, High-accuracy finite-difference schemes for linear wave propagation, *SIAM J. Sci. Comput.* 17 (1996) 328–346.