Boundary Value Problems
a SpringerOpen Journal

RESEARCH Open Access

On the solution of high order stable time integration methods

Owe Axelsson[1,2], Radim Blaheta[2], Stanislav Sysala[2] and Bashir Ahmad[1*]

*Correspondence:
bashirahmad_qau@yahoo.com
[1]King Abdulaziz University, Jeddah, Saudi Arabia
Full list of author information is available at the end of the article

Abstract

Evolution equations arise in many important practical problems. They are frequently stiff, *i.e.* involves fast, mostly exponentially, decreasing and/or oscillating components. To handle such problems, one must use proper forms of implicit numerical time-integration methods. In this paper, we consider two methods of high order of accuracy, one for parabolic problems and the other for hyperbolic type of problems. For parabolic problems, it is shown how the solution rapidly approaches the stationary solution. It is also shown how the arising quadratic polynomial algebraic systems can be solved efficiently by iteration and use of a proper preconditioner.

1 Introduction

Evolution equations arise in many important practical problems, such as for parabolic and hyperbolic partial differential equations. After application of a semi-discrete Galerkin finite element or a finite difference approximation method, a system of ordinary differential equations,

$$M\frac{d\mathbf{u}}{dt} + A\mathbf{u}(t) = \mathbf{f}(t), \quad t > 0, \mathbf{u}(0) = \mathbf{u}_0,$$

arises. Here, $\mathbf{u}, \mathbf{f} \in \Re^n$, M is a mass matrix and M, A are $n \times n$ matrices. For a finite difference approximation, $M = I$, the identity matrix.

In the above applications, the order n of the system can be very large. Under reasonable assumptions of the given source function $\mathbf{f}$, the system is stable, *i.e.* its solution is bounded for all $t > 0$ and converges to a fixed stationary solution as $t \to 0$, independent of the initial value $\mathbf{u}_0$. This holds if A is a normal matrix, that is, has a complete eigenvector space, and has eigenvalues with positive real parts. This condition holds for parabolic problems, where the eigenvalues of A are real and positive. In more involved problems, the matrix A may have complex eigenvalues with arbitrary large imaginary parts.

Clearly, not all numerical time-integration methods preserve the above stability properties. Unless the time-step is sufficiently small, explicit time-integration methods do not converge and/or give unphysical oscillations in the numerical solution. Even with sufficiently small time-steps, algebraic errors may increase unboundedly due to the large number of time-steps. The simplest example where the stability holds is the Euler implicit method,

$$\tilde{\mathbf{u}}(t+\tau) + \tau A\tilde{\mathbf{u}}(t+\tau) = \tilde{\mathbf{u}}(t) + \tau\mathbf{f}(t+\tau), \quad t = \tau, 2\tau, \ldots, \tilde{\mathbf{u}}(0) = \tilde{\mathbf{u}}_0,$$

where $\tau > 0$ is the time-step. Here, the eigenvalues of the inverse of the resulting matrix in the corresponding system,

$$(I + \tau A)\tilde{\mathbf{u}}(t + \tau) = \tilde{\mathbf{u}}(t) + \tau \mathbf{f}(t + \tau)$$

equal $(1 + \tau\lambda)^{-1}$ and satisfy the stability condition,

$$\left|\mu(\lambda)\right| = \left|(1 + \tau\lambda)^{-1}\right| < 1, \quad \lambda \in \sigma(A).$$

Here, $\sigma(A)$ denotes the set of eigenvalues of A. To more quickly damp out initial transients in the solution, which arises for instance due to that the initial value may not satisfy boundary conditions given in the parabolic problem, one should preferably have eigenvalues of the inverse of the discrete matrix B, that satisfies $|\mu(\lambda)| \to 0$ for eigenvalues $\lambda \to \infty$. This holds for the implicit Euler method, where

$$B = I + \tau A \quad \text{and} \quad \mu(\lambda) = (1 + \tau\lambda)^{-1}.$$

This method is only first-order accurate, *i.e.* its global time discretization error is $O(\tau)$. Therefore, to get a sufficiently small discretization error, one must choose very small time-steps, which means that the method becomes computationally expensive and also causes a stronger increase of round-off errors. However, there exists stable time-integration methods of arbitrary high order. They are of implicit Runge-Kutta quadrature type (see *e.g.* [1–5]), and belong to the class of A-stable methods, *i.e.* the eigenvalues $\mu(B^{-1})$ of the corresponding matrix B where $B\tilde{\mathbf{u}}(t + \tau) = \tilde{\mathbf{u}}(t) + \tau\tilde{\mathbf{f}}(t)$, and $\tilde{\mathbf{f}}(t)$ is a linear function of $\mathbf{f}(t)$ at the quadrature points in the interval $[t, t + \tau]$, satisfy $|\mu(B^{-1})| < 1$ for all normal matrices $M^{-1}A$ with $\Re e(\lambda) > 0$. The highest order achieved, $O(\tau^{2m})$ occurs for Gauss quadrature where m equals to the number of quadrature points within each time interval.

To satisfy the second, desirable condition,

$$\lim_{\lambda\to\infty}\left|\mu(\lambda)\right| \to 0,$$

one can use a special subclass of such methods, based on Radau quadrature; see, *e.g.* [1, 5]. The discretization error is here only one order less, $O(\tau^{2m-1})$. For linear problems, all such stable methods lead to rational polynomial approximation matrices B, and hence the need to solve quadratic polynomial equations. For stable methods, it turns out that the roots of these polynomials are complex.

In Section 2, a preconditioning method is described that is very efficient when solving such systems, without the need to factorize the quadratic polynomials in first order factors, thereby avoiding the need to use complex arithmetics. Section 3 discusses the special case where $m = 2$. It shows also how the general case, where $m > 2$, can be handled.

Section 4 deals with the use of implicit Runge-Kutta methods of Gauss quadrature type for solving hyperbolic systems of Hamiltonian type.

Section 5 presents a method to derive time discretization errors.

In Section 6, some illustrating numerical tests are shown. The paper ends with concluding remarks.

2 Preconditioners for quadratic matrix polynomials

From the introduction, it follows that it is of importance to use an efficient solution method for quadratic matrix polynomials and not factorize them in first order factors when this results in complex valued factors. For a method to solve complex valued systems in real arithmetics, see, *e.g.* [6]. Here, we use a particular method that is suitable for the arising quadratic matrix polynomials.

Consider then the matrix polynomial,

$$B = M + aA + b^2 AM^{-1}A. \tag{1}$$

We assume that M is spd and that $|a| < 2b$, which latter implies that the first order factors of B are complex. Systems with B will be solved by iteration. As a preconditioner, we use the matrix

$$C_\alpha = (M + \alpha A)M^{-1}(M + \alpha A),$$

where $\alpha > 0$ is a parameter. We assume that A is a normal matrix, that is, has a full eigenvector space and further that the symmetric part, $A + A^T$ of A is spd. To estimate the eigenvalues of $C_\alpha^{-1}B$, we write

$$(C_\alpha \mathbf{x}, \mathbf{x}) - (B\mathbf{x}, \mathbf{x}) = (2\alpha - a)(A\mathbf{x}, \mathbf{x}) + \left(\alpha^2 - b^2\right)\left(AM^{-1}A\mathbf{x}, \mathbf{x}\right).$$

After a two-sided multiplication with $M^{-1/2}$, we get

$$(\tilde{C}_\alpha \tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - (\tilde{B}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) = (2\alpha - a)(\tilde{A}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) + \left(\alpha^2 - b^2\right)\left(\tilde{A}^2\tilde{\mathbf{x}}, \tilde{\mathbf{x}}\right), \tag{2}$$

where $\tilde{C}_\alpha = M^{-1/2}C_\alpha M^{-1/2} = (I + \alpha\tilde{A})^2$, *etc.* and $\tilde{\mathbf{x}} = M^{1/2}\mathbf{x}$. Note that, by similarity, $C_\alpha^{-1}B$ and $\tilde{C}_\alpha^{-1}\tilde{B}$ have the same eigenvalues.

We are interested in cases where $\tilde{A}$ may have large eigenvalues. (In our application, $\tilde{A}$ involves a time-step factor τ, but since we use higher order time-discretization methods, τ will not be very small and cannot damp out the inverse to some power of the space-discretization parameter h that also occurs in $\tilde{A}$.) Therefore, we choose $\alpha = b$. Note that this implies that $2\alpha - a > 0$.

The resulting relation (2) can now be written

$$(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - \left(\tilde{C}_\alpha^{-1}\tilde{B}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}\right) = (2\alpha - a)\left(\tilde{C}_\alpha^{-1}\tilde{A}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}\right), \tag{3}$$

where

$$\left(\tilde{C}_\alpha^{-1}\tilde{A}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}\right) = \left((I + \alpha\tilde{A})^{-2}\tilde{A}\tilde{\mathbf{x}}, \tilde{\mathbf{x}}\right).$$

Since $2\alpha - a > 0$, the real part of the eigenvalues of $\tilde{C}_\alpha^{-1}\tilde{B}$ are bounded above by 1. To find estimates of the eigenvalues $\lambda(\mu)$ of $\tilde{C}_\alpha^{-1}\tilde{B}$, let $(\mu, \mathbf{z})$ be eigensolutions of $\tilde{A}$, *i.e.* let

$$\tilde{A}\mathbf{z} = \mu\mathbf{z}, \quad |\mathbf{z}| = 1.$$

It follows from (3) that for $\tilde{\mathbf{x}} = \mathbf{z}$,

$$\lambda(\mu) = \left(\tilde{C}_\alpha^{-1}\tilde{B}\mathbf{z}, \mathbf{z}\right) = 1 - \left(1 - \frac{a}{2\alpha}\right)\frac{2\alpha\mu}{1 + 2\alpha\mu + (\alpha\mu)^2}$$
$$= 1 - \left(1 - \frac{a}{2\alpha}\right)\frac{1}{1 + \frac{1}{2}(\alpha\mu + \frac{1}{\alpha\mu})}.$$

We write $\alpha\mu = \mu_0 e^{i\varphi}$ so $\frac{1}{2}(\alpha\mu + \frac{1}{\alpha\mu}) = \frac{1}{2}(\mu_0 + \frac{1}{\mu_0})\cos(\varphi) + \frac{i}{2}(\mu_0 - \frac{1}{\mu_0})\sin(\varphi)$, where i is the imaginary unit. Note that $\mu_0 > 0$ so $\frac{1}{2}(\mu_0 + \frac{1}{\mu_0}) \geq 1$. Since, by assumption, the real part of μ is positive, it holds $|\varphi| \leq \varphi_0 < \pi/2$. A computation shows that the values of the factor $\frac{1}{1+\frac{1}{2}(\alpha\mu+\frac{1}{\alpha\mu})}$ are located in a disc in the complex plane with center at $\delta/2$ and radius $\delta/2$, where $\delta = 1/(1 + \cos\varphi_0)$.

Hence, $\lambda(\mu)$ is located in a disc with center at $1 - \frac{1}{2}(1 - \frac{a}{2\alpha})\delta$ and radius $\frac{1}{2}(1 - \frac{a}{2\alpha})\delta$.

For $\varphi_0 = 0$, *i.e.* for real eigenvalues of $\tilde{A}$, then $\delta = 1/2$ and $1 \geq \lambda(\mu) \geq \frac{3}{4} + \frac{1}{8}\frac{a}{\alpha}$.

3 A stiffly stable time integration method

Consider a system of ordinary differential equations,

$$M\frac{d\mathbf{x}}{dt} + \sigma(t)\big(A\mathbf{x}(t) - \mathbf{f}(t)\big) = 0, \quad t > 0, \mathbf{x}(0) = \mathbf{x}_0, \tag{4}$$

where $\mathbf{x}, \mathbf{f} \in \Re^n$, $\sigma(t) \geq \sigma_0 > 0$, M, A are $n \times n$ matrices, where M is assumed to be spd and the symmetric part of A is positive semidefinite. In the practical applications that we consider, M corresponds to a mass matrix and A to a second-order diffusion or diffusion-convection matrix. Hence, n is large. Under reasonable assumptions on the source function $\mathbf{f}$, such a system is stable for all t and its solution approaches a finite function, independent on the initial value $\mathbf{x}_0$, as $t \to \infty$.

Such stability results hold for more general problems, such as for a nonlinear parabolic problem,

$$\frac{\partial u}{\partial t} + F(t,u) = 0, \quad \text{where } F(t,u) = -\nabla\cdot\big(a(t,u,\nabla u)\nabla u\big) - f(t,u), x \in \Omega, t > 0, \tag{5}$$

where $f : (0,\infty) \times V \to V'$ and V is a reflexive Banach space.

For proper functions $a(\cdot)$ and $f(\cdot)$, then F is monotone, *i.e.*

$$\big(F(t,u) - F(t,v), u - v\big) \geq \rho(t)\|u - v\|^2, \quad \forall u, v \in V, t > 0. \tag{6}$$

Here, $\rho : (0,\infty) \to R$, $\rho(t) \geq 0$ and $(\cdot,\cdot)$, $\|\cdot\|$ denote the scalar product, and the corresponding norm in $L^2(\Omega)$, respectively. In this case, one can easily derive the bound

$$\frac{1}{2}\frac{d}{dt}\big(\|u - v\|^2\big) = -\big(F(t,u) - F(t,v), u - v\big) \leq -\rho(t)\|u - v\|^2,$$

where u, v are solution of (5) corresponding to different initial values. Consequently making use of the Gronwall lemma, we obtain

$$\big\|u(t) - v(t)\big\| \leq \exp\left(-\int_0^t \rho(s)\,ds\right)\big\|u(0) - v(0)\big\| \leq \big\|u(0) - v(0)\big\|, \quad t > 0.$$

Hence, (5) is stable in this case.

If F is strongly monotone (or dissipative), *i.e.* (6) is valid with $\rho(t) \geq \varrho_0 > 0$, then

$$\|u(t) - v(t)\| \leq \exp(-t\rho_0)\|u(0) - v(0)\| \to 0, \quad t \to \infty,$$

i.e. (5) is asymptotically stable. In particular, the above holds for the test problem considered in Section 6.

For large eigenvalues of $M^{-1}A$, such a system is stiff and can have fast decreasing and possibly oscillating components. This amounts to that the eigenvalues have large real part and possibly also large imaginary parts. To handle this, one needs stable numerical time-integration methods that do not contain corresponding increasing components. For $\sigma(t) = 1$, in (4), this amounts to proper approximations of the matrix exponential function $\exp(tE)$, $E = M^{-1}A$, by a rational function,

$$R_m(tE) = Q_m(tE)^{-1}P_m(tE),$$

where

$$\|R_m(tE)\| \leq 1, \quad t > 0, \text{ for } \mathrm{Re}\{\lambda_E\} > 0,$$

and λ_E denotes eigenvalues by E. Furthermore, to cope with problems where $\arg(\lambda_E) \leq \alpha < \frac{\pi}{2}$, but arbitrarily close to $\pi/2$, one needs A-stable methods; see *e.g.* [3, 7, 8]. To get stability for all times and time steps, one requires $\lim_{|\lambda|\to\infty} |R_m(\lambda)| \leq c < 1$ where preferably $c = 0$. Such methods are called L-stable (Lambert) and stiffly A-stable [3], respectively.

An important class of methods which are stiffly A-stable is a particular class of the implicit Runga-Kutta methods; see [1, 3, 5]. Such methods correspond to rational polynomial approximations of the matrix exponential function with denominator having a higher degree than the nominator. Examples of such methods are based on Radau quadrature where the quadrature points are zeros of $\tilde{P}_m(\xi) - \tilde{P}_{m-1}(\xi)$, where $\{\tilde{P}_k\}$ are the Legendre polynomials, orthogonal on the interval $(0,1)$, see *e.g.* [1] and references therein. Note that $\xi = 1$ is a root for all $m \geq 1$. The case $m = 1$ is identical to the implicit Euler method.

Following [5], we consider here the next simplest case, where $m = 2$, for the numerical solution of (4) over a time interval $[t, t + \tau]$.

In this case, the quadrature points (for a unit interval) are $\xi_1 = 1/3$, $\xi_2 = 1$ and the numerical solution $\mathbf{x}_1$, $\mathbf{x}_2$, at $t + \tau/3$ and $t + \tau$ satisfies

$$\begin{bmatrix} M + 5\sigma_1\tilde{A} & -\sigma_2\tilde{A} \\ 9\sigma_1\tilde{A} & M + 3\sigma_2\tilde{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} M\mathbf{x}_0 + \frac{\tau}{12}(5\mathbf{f}_1 - \mathbf{f}_2) \\ M\mathbf{x}_0 + \frac{\tau}{4}(3\mathbf{f}_1 + \mathbf{f}_2) \end{bmatrix}, \tag{7}$$

where $\mathbf{x}_0$ is the solution at time t, $\sigma_1 = \sigma(t + \tau/3)$, $\sigma_2 = \sigma(t + \tau)$, $\mathbf{f}_1 = \mathbf{f}(t + \tau/3)$, $\mathbf{f}_2 = \mathbf{f}(t + \tau)$, and $\tilde{A} = \frac{\tau}{12}A$. The global discretization error of the $\mathbf{x}_2$-component for this method is $O(\tau^3)$, *i.e.* it is a third-order method and it is stiffly A-stable even for arbitrary strong variations of the coefficient $\sigma(t)$. This can be compared with the trapezoidal or implicit midpoint methods which are only second order accurate and not stiffly stable.

The system in (7) can be solved *via* its Schur complement. Thereby, to avoid an inner system with matrix $M + 5\sigma_1\tilde{A}$, we derive a modified form of the Schur complement system,

that involves only an inner system with matrix M^{-1}. To this end, but only for the derivation of the method, we scale first the system with the block diagonal matrix $\left[\begin{smallmatrix} M^{-1} & 0 \\ 0 & M^{-1} \end{smallmatrix}\right]$ to get

$$\begin{bmatrix} I + 5\sigma_1 G & -\sigma_2 G \\ 9\sigma_1 G & I + 3\sigma_2 G \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}_0 + \frac{\tau}{12}(5\tilde{\mathbf{f}}_1 - \tilde{\mathbf{f}}_2) \\ \mathbf{x}_0 + \frac{\tau}{4}(3\tilde{\mathbf{f}}_1 + \tilde{\mathbf{f}}_2) \end{bmatrix},$$

where $G = \frac{\tau}{12} M^{-1}A$ and $\tilde{\mathbf{f}}_i = M^{-1}\mathbf{f}_i$, $i = 1, 2$. The Schur complement system for $\mathbf{x}_2$ is multiplied with $(I + 5\sigma_1 G)$. Using commutativity, we get then

$$\begin{aligned} &\left[(I + 5\sigma_1 G)(I + 3\sigma_2 G) + 9\sigma_1\sigma_2 G^2\right]\mathbf{x}_2 \\ &\quad = (I + 5\sigma_1 G)\left[\mathbf{x}_0 + \frac{\tau}{4}(3\tilde{\mathbf{f}}_1 + \tilde{\mathbf{f}}_2)\right] - 9\sigma_1 G\left[\mathbf{x}_0 + \frac{\tau}{12}(5\tilde{\mathbf{f}}_1 - \tilde{\mathbf{f}}_2)\right] \end{aligned}$$

or

$$\begin{aligned} &\left[I + (5\sigma_1 + 3\sigma_2)G + 24\sigma_1\sigma_2 G^2\right]\mathbf{x}_2 \\ &\quad = (I - 4\sigma_1 G)\mathbf{x}_0 + \frac{\tau}{4}(3\tilde{\mathbf{f}}_1 + \tilde{\mathbf{f}}_2) + 2\tau\sigma_1 G\tilde{\mathbf{f}}_2. \end{aligned}$$

Hence,

$$B\mathbf{x}_2 = \left(M - \frac{\tau}{3}\sigma_1 A\right)\mathbf{x}_0 + \frac{\tau}{4}M(3\tilde{\mathbf{f}}_1 + \tilde{\mathbf{f}}_2) + \frac{1}{6}\tau^2\sigma_1 A\tilde{\mathbf{f}}_2,$$

where

$$B = M + \frac{\tau}{12}(5\sigma_1 + 3\sigma_2)A + \frac{\tau^2}{6}\sigma_1\sigma_2 A M^{-1}A. \tag{8}$$

For higher order Radau quadrature methods, the corresponding matrix polynomial in $M^{-1}B$ is a mth order polynomial. By the fundamental theorem of algebra, one can factorize it in factors of at most second degree. They can be solved in a sequential order. Alternatively, using a method referred to in Remark 3.1, the solution components can be computed concurrently.

Each second-order factor can be preconditioned by the method in Section 2. The ability to factorize $Q_m(tE)$ in second-order factors and solve the arising systems as such two-by-two block matrix systems means that one only has to solve first-order systems. This is of importance if for instance M and A are large sparse bandmatrices, since then one avoids increasing bandwidths in matrix products and one can solve systems of linear combinations of M and A more efficiently than for higher order polynomial combinations. Furthermore, this enables one to keep matrices on element by element form (see, *e.g.* [9]) and it is in general not necessary to store the matrices M and A. The arising inner system can be solved by some inner iteration method.

The problem with a direct factorization in first order factors is that complex matrix factors appear. This occurs for the matrix in (8) for a ratio of $\frac{\sigma_1}{\sigma_2}$ in the interval

$$3\frac{11 - \sqrt{96}}{25} < \frac{\sigma_1}{\sigma_2} < 3\frac{11 + \sqrt{96}}{25}. \tag{9}$$

Therefore, it is more efficient to keep the second order factors and instead solve the corresponding systems by preconditioned iterations. Thereby, the preconditioner involves only first order factors. As shown in Section 2, a very efficient preconditioner for the matrix B in (8) is

$$C = C_\alpha = (M + \alpha\tau A)M^{-1}(M + \sigma\tau A), \tag{10}$$

where $\alpha > 0$ is a parameter. As already shown in [5], for the above particular application it holds.

Proposition 3.1 *Let B, C be as defined in* (8) *and* (10) *and assume that M is spd and A is spsd. Then letting*

$$\alpha = \max\left\{\sqrt{\sigma_1\sigma_2/6}, (5\sigma_1 + 3\sigma_2)/24\right\}$$

it holds

$$\kappa\left(C^{-1}B\right) \le \max_{i=1,2} \delta_i^{-1},$$

where

$$1 \ge \delta_1 = (5\sigma_1 + 3\sigma_2)/24\alpha \ge \sqrt{10}/4,$$
$$1 \ge \delta_2 = \sigma_1\sigma_2/6\alpha^2.$$

If $0.144 \le \frac{\sigma_1}{\sigma_2} \le 2.496$, *then* $\delta_2 = 1$ *and* $\delta_1 \ge \sqrt{\frac{5}{8}}$.
The spectral condition number is then bounded by

$$\kappa\left(C^{-1}B\right) \le \sqrt{\frac{8}{5}} \approx 1.265.$$

If $\sigma_1 = \sigma_2$, *then*

$$\kappa\left(C^{-1}B\right) \le \sqrt{\frac{3}{2}} \approx 1.225.$$

Proof Let $(\mathbf{u}, \mathbf{v})$ be the ℓ_2 product of $\mathbf{u}, \mathbf{v} \in \Re^n$. We have

$$(C\mathbf{x}, \mathbf{x}) - (B\mathbf{x}, \mathbf{x}) = 2\sigma\tau(1 - \delta_1)(A\mathbf{x}, \mathbf{x}) + \alpha^2\tau^2(1 - \delta_2)\left(AM^{-1}A\mathbf{x}, \mathbf{x}\right) \quad \forall \mathbf{x} \in \Re^n.$$

It follows that

$$(B\mathbf{x}, \mathbf{x}) \le (C\mathbf{x}, \mathbf{x}).$$

By the arithmetic-geometric means inequality, we have

$$\delta \ge \frac{1}{2}\sqrt{15\sigma_1\sigma_2}/\alpha \ge \frac{1}{\sqrt{2}}\sqrt{90} = \frac{\sqrt{10}}{4}. \tag{11}$$

a computation shows that

$$\sigma_1\sigma_2/6 \geq \left(\frac{5\sigma_1 + 3\sigma_2}{24}\right)^2$$

for $0.144 \lesssim \xi \lesssim 2.496$, where $\xi = \sigma_1/\sigma_2$. Further, a computation shows that $\delta_1 \geq \sqrt{\frac{5}{8}}$, which is in accordance with the lower bound in (11). Since

$$(C\mathbf{x}, \mathbf{x}) \geq 2\alpha\tau(A\mathbf{x}, \mathbf{x}) + \alpha^2\tau^2\big(AM^{-1}A\mathbf{x}, \mathbf{x}\big),$$

it follows that

$$1 - \frac{(B\mathbf{x}, \mathbf{x})}{(C\mathbf{x}, \mathbf{x})} \geq 1 - \delta_1$$

or

$$\frac{(B\mathbf{x}, \mathbf{x})}{(C\mathbf{x}, \mathbf{x})} \leq \delta_1 = \sqrt{\frac{5}{8}}.$$

For $\alpha_1 = \alpha_2$, a computation shows that

$$\delta_1 = \frac{1}{3}\sqrt{6} = \sqrt{\frac{2}{3}}.$$ □

We conclude that the condition number is very close to its ideal unit value 1, leading to very few iterations. For instance, it suffices with at most 5 conjugate gradient iterations for a relative accuracy of 10^{-6}.

Remark 3.1 High order implicit Runge-Kutta methods and their discretization error estimates can be derived using order tree methods as described in [1] and [10].

For an early presentation of implicit Runge-Kutta methods, see [2] and also [4], where the method was called global integration method to indicate its capability for large values of m to use few, or even just one, time discretization steps. It was also shown that the coefficient matrix, formed by the quadrature coefficients had a dominating lower triangular part, enabling the use of a matrix splitting and Richardson iteration method. It can be of interest to point out that the Radau method for $m = 2$ can be described in an alternative way, using Radau quadrature for the whole time step interval and combined with a trapezoidal method for the shorter interval.

Namely, let $\frac{du}{dt} + f(t, u) = 0$, $t_{k-1} < t < t_k$. Then Radau quadrature on the interval (t_{k-1}, t_k) has quadrature points $t_{k-1} + \tau/3$, t_k, and coefficients $b_1 = 3/4$, $b_2 = 1/4$, which results in the relation

$$\tilde{u}_1 - \tilde{u}_0 + \frac{3\tau}{4}f(\tilde{t}_{1/3}, \tilde{u}_{1/3}) + \frac{\tau}{4}f(\tilde{t}_1, \tilde{u}_1) = 0,$$

where $\tilde{u}_1$, $\tilde{u}_{1/3}$, $\tilde{u}_0$ denote the corresponding approximations of u at $\tilde{t}_1 \doteq t_{k-1} + \tau$ and $\tilde{t}_{1/3} = t_{k-1} + \tau/3$ and t_{k-1}, respectively.

This equation is coupled with an equation based on quadrature

$$u(t_{k-1}+\tau/3)-u(t_{k-1})+\int_{t_{k-1}}^{t_k} f(t,u)\,dt-\int_{t_{k-1}+\tau/3}^{t_k} f(t,u)\,dt=0,$$

which, using the stated quadrature rules, results in

$$\tilde{u}_{1/3}-\tilde{u}_0+\frac{3\tau}{4}f(\tilde{t}_{1/3},\tilde{u}_{1/3})+\frac{\tau}{4}f(\tilde{t}_1,\tilde{u}_1)-\frac{1}{2}\frac{2\tau}{3}\big[f(\tilde{t}_{1/3},\tilde{u}_{1/3})+f(\tilde{t}_1,\tilde{u}_1)\big]=0$$

that is,

$$\tilde{u}_{1/3}-\tilde{u}_0+\frac{5\tau}{12}f(\tilde{t}_{1/3},\tilde{u}_{1/3})-\frac{\tau}{12}f(\tilde{t}_1,\tilde{u}_1)=0.$$

Remark 3.2 The arising system in a high order method involving $q\geq 2$ quadratic polynomial factors, can be solved sequentially in the order they appear. Alternatively (see, *e.g.* [11], Exercise 2.31), one can use a method based on solving a matrix polynomial equation, $\mathcal{P}_{2q}(A)\mathbf{x}=\mathbf{b}$ as $\mathbf{x}=\sum_{k=1}^{q}\frac{1}{\mathcal{P}'_{2q}(r_k)}\mathbf{x}_k$, $\mathbf{x}_k=(A-r_kI)^{-1}\mathbf{b}$, where $\{r_k\}^{2q}$, is the set of zeros of the polynomial and it is assumed that A has no eigenvalues in this set. (This holds in our applications.) Then, combining pairs of terms corresponding to complex conjugate roots r_k, quadratic polynomials arise for the computation of the corresponding solution components. It is seen that in this method, the solution components can be computed concurrently.

Remark 3.3 Differential algebraic equations (DAE) arise in many important problems; see, for instance [10, 12]. The simplest example of a DAE takes the form

$$\begin{cases}\frac{du}{dt}=f(t,u,v),\\ g(t,u,v)=0,\quad t>0,\end{cases}$$

with $u(0)=u_0$, $v(0)=v_0$ and it is normally assumed that the initial values satisfy the constraint equation, *i.e.*

$$g(0,u_0,v_0)=0.$$

If $\det(\frac{\partial g}{\partial v})\neq 0$ in a sufficiently large set around the solution, one can formally eliminate the second part of the solution to form a differential equation in standard form.

$$\frac{du}{dt}=f\big(t,u,v(u)\big),\quad t>0,u(0)=u_0.$$

Such a DAE is said to have index one, see *e.g.* [13]. It can be seen to be a limit case of the system

$$\begin{cases}\frac{du}{dt}=f(t,u,v),\\ \frac{du}{dt}=\frac{1}{\varepsilon}g(t,u,v),\end{cases}$$

where $\varepsilon>0$ and $\varepsilon\to 0$.

Hence, such an DAE can be considered as an infinitely stiff differential equation problem. For strongly or infinitely stiff problems, there can occur an order reduction phenomenae. This follows since some high order error terms in the error expansion (*cf.* Section 5) are multiplied with (infinitely) large factors, leading to an order reduction for some methods. Heuristically, this can be understood to occur for the Gauss integration form of IRK but does not occur for the stiffly stable variants, such as based on the Radau quadrature. For further discussions of this, see, *e.g.* [10, 13].

4 High order integration methods for Hamiltonian systems

Another important application of high order time integration methods occurs for Hamiltonian systems. Such systems occur in mechanics and particle physics, for instance. As an introduction, consider the conservation of energy principle. To this end, consider a mechanical system of k point masses and its associated Lagrangian functional,

$$L = K - V = \sum_{i=1}^{k} \frac{1}{2} m_i |\dot{\mathbf{x}}_i|^2 - V(\mathbf{x}_1, \ldots, \mathbf{x}_k),$$

where K is the kinetic energy and V the potential energy. Here, $\mathbf{x}_i = (x_i, y_i, z_i)$ denote the Cartesian coordinate of the ith point mass m_i.

The configuration strives to minimize the total energy. The corresponding Euler-Lagrange equations become then $\frac{\partial L}{\partial \mathbf{x}_i} = 0$, that is,

$$m_i \ddot{\mathbf{x}}_i = -\frac{\partial V}{\partial \mathbf{x}_i}, \quad i = 1, 2, \ldots, k. \tag{12}$$

We consider conservative systems, *i.e.* mechanical systems for which the total force on the elements of the system are related to the potential $V : \Re^{3k} \Rightarrow \Re$ according to

$$\mathbf{F}_i = -\frac{\partial V}{\partial \mathbf{x}_i}.$$

This means that the Euler-Lagrange equation (12) is identical to the classical Newton's law

$$m_i \ddot{\mathbf{x}}_i = \mathbf{F}_i, \quad i = 1, 2, \ldots, k.$$

Let $p_i = m_i v_i$ be the momentum. Then

$$K = \sum_{1}^{k} \frac{1}{2} \frac{p_i^2}{m_i}.$$

A mechanical system can be described by general coordinates

$$\mathbf{q} = (q_1, \ldots, q_d)$$

i.e. not necessarily Cartesian, but angles, length along a curve, *etc.* The Lagrangian takes the form $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t)$. If $\mathbf{q}$ is determined to satisfy

$$\min_q \int_a^b \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t)\, dt,$$

then the motion of the system is described by the Lagrange equation,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}}(\mathbf{q},\dot{\mathbf{q}},t) = \frac{\partial L}{\partial \mathbf{q}}(\mathbf{q},\dot{\mathbf{q}},t). \tag{13}$$

Letting here

$$p_k = \frac{\partial L}{\partial \dot{q}_k}(\mathbf{q},\dot{\mathbf{q}}), \quad k = 1,2\ldots,n$$

be the momentum variable, and using the transformation $(\mathbf{q},\dot{\mathbf{q}}) = (\mathbf{q},\mathbf{p})$ we can write (13) as the Hamiltonian,

$$H(\mathbf{p},\mathbf{q},t) = \sum_{j=1}^{n} p_j\dot{q}_j - L\big(\mathbf{q},\dot{\mathbf{q}}(\mathbf{q},\mathbf{p},t),t\big).$$

For a mechanical system with potential energy a function of configuration only and kinetic energy K given by a quadratic form

$$K = \frac{1}{2}\dot{\mathbf{q}}^T G(\mathbf{q})\dot{\mathbf{q}},$$

where G is an spd matrix, possibly depending on $\mathbf{q}$, we get

$$\mathbf{p} = G(\mathbf{q})\dot{\mathbf{q}}, \quad \dot{\mathbf{q}} = G^{-1}(\mathbf{q})\mathbf{p} \tag{14}$$

and

$$\begin{aligned} H(\mathbf{p},\mathbf{q},t) &= \mathbf{p}^T G^{-1}(\mathbf{q})\mathbf{p} - \frac{1}{2}\mathbf{p}^T G^{-1}(\mathbf{q})\mathbf{p} + V(\mathbf{q}) \\ &= \frac{1}{2}\mathbf{p}^T G^{-1}(\mathbf{q})\mathbf{p} + V(\mathbf{q}) = K(\mathbf{p},\mathbf{q}) + V(\mathbf{q}), \end{aligned}$$

which equals the total energy of the system.

The corresponding Euler-Lagrange equations become now

$$\begin{cases} \dot{\mathbf{p}} = -\frac{\partial H}{\partial \mathbf{q}}, \\ \dot{\mathbf{q}} = \frac{\partial H}{\partial \mathbf{p}} \end{cases} \tag{15}$$

and are referred to as the Hamiltonian system. This follows from

$$\begin{aligned} \frac{\partial H}{\partial \mathbf{p}} &= \dot{\mathbf{q}}^T + \mathbf{p}^T\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{p}} - \frac{\partial L}{\partial \dot{\mathbf{q}}}\frac{\partial \dot{\mathbf{q}}}{\partial p} = \dot{\mathbf{q}}^T, \\ \frac{\partial H}{\partial \mathbf{q}} &= \mathbf{p}^T\frac{\partial \dot{\mathbf{q}}}{\partial q} - \frac{\partial L}{\partial \mathbf{q}} - \frac{\partial L}{\partial \dot{\mathbf{q}}}\frac{\partial \dot{\mathbf{q}}}{\partial \mathbf{q}} = -\frac{\partial L}{\partial \mathbf{q}}, \end{aligned}$$

which, since $\frac{d}{dt}(\frac{\partial L}{\partial \dot{\mathbf{q}}}) = \frac{\partial L}{\partial \mathbf{q}}$ implies $\dot{\mathbf{p}} = \frac{\partial L}{\partial \mathbf{q}}$, are hence equivalent to the Lagrange equations.

By (15), it holds

$$\frac{d}{dt}H(\mathbf{p},\mathbf{q}) = \frac{\partial H}{\partial \mathbf{p}}\dot{\mathbf{p}} + \frac{\partial H}{\partial \mathbf{q}}\dot{\mathbf{q}} = 0, \tag{16}$$

that is, the Hamiltonian function $H(\mathbf{p},\mathbf{q})$ is a first integral for the system (15).

The flow $\varphi_t : U \to \Re^{2n}$ of a Hamiltonian system is the mapping that describes the evolution of the solution by time, *i.e.* $\varphi_t(\mathbf{p}_0, \mathbf{q}_0) = (p(t, \mathbf{p}_0, \mathbf{q}_0), q(t, \mathbf{p}_0, \mathbf{q}_0))$, where $\mathbf{p}(t, \mathbf{p}_0, \mathbf{q}_0)$, $\mathbf{q}(t, \mathbf{p}_0, \mathbf{q}_0)$ is the solution of the system for the initial values $\mathbf{p}(0) = \mathbf{p}_0$, $\mathbf{q}(0) = \mathbf{q}_0$.

We consider now a Hamiltonian with a quadratic first integral in the form

$$H(\mathbf{y}) = \mathbf{y}^T C \mathbf{y}, \quad \mathbf{y} = (\mathbf{p}, \mathbf{q}), \tag{17}$$

where C is a symmetric matrix. For the solution of the Hamiltonian system (15), we shall use an implicit Runge-Kutta method based on Gauss quadrature.

The s-stage Runge-Kutta method applied to an initial value problem, $\dot{\mathbf{y}} = f(t, \mathbf{y})$, $\mathbf{y}(t_0) = \mathbf{y}_0$ is defined by

$$\begin{cases} k_i = f(t_0 + c_i\tau, y_0 + \tau \sum_{j=1}^{s} a_{ij} k_j), \quad i = 1, 2, \ldots, s, \\ y_1 = y_0 + \tau \sum_{i=1}^{s} b_i k_i, \end{cases} \tag{18}$$

where $c_i = \sum_{i=1}^{s} a_{ij}$, see *e.g.* [1, 4]. The familiar implicit midpoint rule is the special case where $s = 1$. Here, $c_1, \ldots, c_s$ are the zeros of the shifted Legendre polynomial $\frac{d^s}{dx^s}(x^s(1 - x)^s)$. For a linear problem, this results in a system which can be solved by the quadratic polynomial decomposition and the preconditioned iterative solution method, presented in Section 2.

If $u(t)$ is a polynomial of degree s, then (18) takes the form

$$\begin{aligned} &u(t_0) = y_0, \\ &\dot{u}(t + c_i\tau) = f\big(t_0 + c_0\tau, y(t_0 + c_i\tau)\big), \quad i = 1, \ldots, s \end{aligned} \tag{19}$$

and $u_1 = u(t_0 + \tau)$.

For the Hamiltonian (17), it holds

$$\frac{d}{dt} H\big(y(t)\big) = 2y^T(t) C y(t)$$

and it follows from (16) that

$$y_1^T C y_1 - y_0^T C y_0 = 2 \int_{t_0}^{t_0+\tau} u(t)^T C \dot{u}(t)\, dt.$$

Since the integrand is a polynomial of degree $2s - 1$, it is evaluated exactly by the s-stage Gaussian quadrature formula. Therefore, since

$$y(t_0 + c_i t)^T C \dot{y}(t_0 + c_i\tau) = u(t_0 + c_i\tau)^T C f\big(u(t_0 + c_i\tau)\big) = 0$$

it follows that the energy quadrature forms $y_i^T C_i y_i$ are conserved.

This is an important property in Hamiltonian systems and is referred to as being symplectic. For further references of symplectic integrators, see [10].

5 Discretization error estimates

Error estimation methods for parabolic and hyperbolic problems can differ greatly. Parabolic problems are characterized by the monotonicity property (6) while for hyperbolic problems a corresponding conservation property,

$$\big(F(t,u)-F(t,v),u-v\big)=0,\quad t>0\ \forall u,v\in V$$

holds, implying

$$\big\|u(t)-v(t)\big\|=\big\|u(0)-v(0)\big\|,\quad t\geq 0. \tag{20}$$

Hence, there is no decrease of errors occurring at earlier time steps. On the other hand, the strong monotonicity property for parabolic problems implies that errors at earlier time steps decrease exponentially as time evolves.

For a derivation of discretization errors for such parabolic type problems for a convex combination of the implicit Euler method and the midpoint method, referred to as the θ-method, the following holds (see [14]). Similar estimates can also be derived for the Radau quadrature method, see, *e.g.* [10].

The major result in [14] is the following.

Let $u_t^s=\frac{\partial^s(u(t))}{\partial t^s}$. Consider the problem $u_t'=F(t,u(t))$ where u belongs to some function space V and the corresponding truncation error,

$$\begin{aligned}R_\theta(t,u)&=F\big(\bar t,\bar u(t)\big)-\tau^{-1}\int_t^{t+\tau}u_t'(s)\,ds\\&=u'(\bar t)-\tau^{-1}\big[u(t+\tau)-u(t)\big]+F\big(\bar t,\bar u(t)\big)-F\big(\bar t,u(\bar t)\big),\end{aligned}$$

where $\bar t=\theta t+(1-\theta)(t+\tau)$, $\bar u(t)=\theta u(t)+(1-\theta)u(t+\tau)$, $0\leq\theta\leq 1$.

If $u\in C^3(V)$, then a Taylor expansion shows that

$$\begin{aligned}R_\theta(t,u)=&-\frac{1}{24}\tau^2u_t^{(3)}(t_3)+\left(\frac12-\theta\right)\tau u_t^{(2)}(t_2)\\&+\frac12\theta(1-\theta)\tau^2\frac{\partial F}{\partial y}\big(\bar t,\tilde u(\bar t)\big)u_t^{(2)}(t_1),\quad t<t_i<t+\tau,\ i=1,2,3,\end{aligned} \tag{21}$$

where $\tilde u(\bar t)$ takes values in a tube with radius $\|\overline{u}(t)-u(\bar t)\|$ about the solution $u(t)$.

It follows that if

$$\left\|\frac{\partial F}{\partial u}\big(\bar t,\tilde u(\bar t)\big)u_t^{(2)}\big(t_1(t)\big)\right\|\leq C_1 \tag{22}$$

and $\theta=\frac12-O(\tau)$, then

$$\big\|R_\theta(t,u)\big\|=O\big(\tau^2\big).$$

Under the above conditions, the discretization error $e(t)=u(t)-v(t)$, where

$$v(t+\tau)-v(t)+\tau F\big(\bar t,\bar v(t)\big)=0,\quad t=0,\tau,2\tau,\dots,$$

$v(0)=u(0)$, is the approximate solution, satisfies

(i) if F is strongly monotone and $\frac{1}{2} - |O(\tau)| \le \theta \le \theta_0$, then $\|e(t)\| \le \varrho_0^{-1} C' \tau^2$, $t > 0$;

(ii) if F is monotone (or conservative) and $\frac{1}{2} - |O(\tau)| \le \theta \le \frac{1}{2}$, then $\|e(t)\| \le tC'\tau^2$, $t > 0$.

Here, C' depends on $\|u_t^{(2)}\|$ and $\|u_t^{(3)}\|$, but is independent of the stiffness of the problem under the appropriate conditions stated above.

If the solution u is smooth so that $\frac{\partial F}{\partial u} u_t^{(2)}$ has also only smooth components, then $\|\frac{\partial F}{\partial u} u_t^{(2)}\|$ may be much smaller than $\|\frac{\partial F}{\partial u}\| \|u_t^{(2)}\|$, showing that the stiffness, *i.e.* factors $\|\frac{\partial F}{\partial u}\| \gg 1$, do not enter in the error estimate.

In many problems, we can expect that $\|\frac{\partial F}{\partial u} u_t^{(2)}\|$ is of the same order as $\|u_t^{(3)}\|$, *i.e.* the first and last forms in (21) have the same order. In particular, this holds for a linear problem $u_t + Au = 0$, where $u_t^{(3)} = A^3 u = \frac{\partial F}{\partial u} u_t^{(2)}$.

It is seen from (20) that for hyperbolic (conservative) problems like the Hamiltonian problem in Section 4, the discretization error grows at least linearly with t, but likely faster if the solution is not sufficiently smooth. It may then be necessary to control the error by coupling the numerical time-integration method with an adaptive time step control. We present here such a method based on the use of backward integration at each time-step using the adjoint operator. The use of adjoint operators in error estimates gives back to the classical Aubin-Nitsche L_2-lifting method used in boundary value problems to derive discretization error estimates in L_2 norm. It has also been used for error estimates in initial value problems, see *e.g.* [7].

Assume that the monotonicity assumption (20) holds. We show first a nonlinear (monotone) stability property, called B-stability, that holds for the numerical solution of implicit Runge-Kutta methods based on Gauss quadrature points. It goes back to a scientific note in [15]; see also [16].

Let $\tilde{u}$, $\tilde{v}$ be two approximate solutions to $u' = f(u,t)$, $t > 0$ extended to polynomials of degree m from their pointwise values at $t_{k,i}$ in the interval $[t_{k-1}, t_k]$. Let

$$\Psi(t) = \frac{1}{2} \left\| \tilde{u}(t) - \tilde{v}(t) \right\|^2.$$

Then, since by (18), $\tilde{u}'(t)$ and $\tilde{v}'(t)$ satisfy the differential equation at the quadrature points, and by (19) it holds

$$\Psi'(t_{k,i}) = \left(\tilde{u}'(t_{k,i}) - \tilde{v}'(t_{k,i}), \tilde{u}(t_{k,i}) - \tilde{v}(t_{k,i})\right) = \left(f\left(\tilde{u}(t_{k,i})\right) - f\left(\tilde{v}(t_{k,i})\right), \tilde{u}(t_{k,i}) - \tilde{v}(t_{k,i})\right) \le 0,$$

$i = 1, 2, \ldots, m$, where $\{t_{k,i}\}_{i=1}^m$ is the set of quadrature points. Since $\Psi'(t)$ is a polynomial of degree $2m-1$, Gauss quadrature is exact so

$$\Psi(t_k) - \Psi(t_{k-1}) = \int_{t_{k-1}}^{t_k} \Psi'(s)\, ds = \sum_{i=1}^{m} b_i \Psi'(t_{k,i}) \le 0.$$

Here, $b_i > 0$ are the quadrature coefficients.

Hence,

$$\left\| \tilde{u}(t_k) - \tilde{v}(t_k) \right\| \le \left\| \tilde{u}(t_{k-1}) - \tilde{v}(t_{k-1}) \right\| \le \cdots \le \left\| \tilde{u}(0) - \tilde{v}(0) \right\|, \quad k = 1, 2, \ldots.$$

Since $\Psi^{(2m)}(t) \ge 0$, this monotonicity property can be seen to hold also for the Radau quadrature method.

We present now a method for adaptive a posteriori error control for the initial value problem

$$\begin{aligned} &u'(t) = \sigma(t) f\big(u(t)\big), \quad t > 0, \\ &u(0) = u_0, \end{aligned} \tag{23}$$

where $u(t) \in \mathbb{R}^n$ and $f(u(t)) = Au(t) - \tilde{f}(t)$.

For the implicit Runge-Kutta method with approximate solution $\tilde{u}(t)$, it holds

$$\tilde{u}(t_k) = \tilde{u}(t_{k-1}) + \int_{t_{k-1}}^{t_k} \sigma(t) f\big(\tilde{u}(t)\big)\, dt, \quad k = 1, 2,$$

where $\tilde{u}(t)$ is a piecewise polynomial of degree m.

The corresponding residual equals

$$R\big(\tilde{u}(t)\big) = \tilde{u}'(t) - \sigma(t) f\big(\tilde{u}(t)\big).$$

By the property of implicit Runge-Kutta methods, it is orthogonal, *i.e.*

$$\int_{t_{k-1}}^{t_k} \big(\tilde{u}'(t) - \sigma(t) f\big(\tilde{u}(t)\big)\big) \cdot \underline{v}\, dt = 0, \quad k = 0, 1, \ldots \tag{24}$$

to all polynomials of degree m. Here, the 'dot' indicates a vector product in $\mathbb{R}^n$. The discretization error equals $e(t) = u(t) - \tilde{u}(t)$, $t > 0$. The error estimation will be based on the backward integration of the adjoint operator problem,

$$\begin{cases} \varphi'(t) = -\sigma(t) A^T \varphi(t), \quad t_{k-1} < t < t_k, \\ \varphi(t_k) = e(t_k). \end{cases} \tag{25}$$

Note that $\sigma(t) Ae(t) = \sigma(t)(f(u(t)) - f(\tilde{u}(t)))$. It holds

$$\big|e(t_k)\big|^2 = \big|e(t_k)\big|^2 + \int_{t_{k-1}}^{t_k} e \cdot \big(-\varphi' - \sigma(t) A^T \varphi\big)\, dt,$$

so by integration by parts, we get

$$\big|e(t_k)\big|^2 = \int_{t_{k-1}}^{t_k} \big(e' - \sigma(t) Ae\big) \cdot \varphi\, dt + e(t_{k-1}) \cdot \varphi(t_{k-1}).$$

Here

$$e' - \sigma(t) Ae = u' - \sigma(t)\big(Au - \tilde{f}(t)\big) - \big(\tilde{u}' - \sigma(t)\big(A\tilde{u} - \tilde{f}(t)\big)\big) = -\tilde{u}' + \sigma(t) f(\tilde{u}) = -R(\tilde{u}).$$

Hence,

$$\big|e(t_k)\big|^2 = -\int_{t_{k-1}}^{t_k} R(\tilde{u}) \cdot \varphi\, dt + e(t_{k-1}) \cdot \varphi(t_{k-1}).$$

Here, we can use the Galerkin orthogonality property (24) to get

$$\left|e(t_k)\right|^2 - \left|e(t_{k-1}) \cdot \varphi(t_{k-1})\right| \le \min_{\tilde{\varphi}} \left| \int_{t_{k-1}}^{t_k} R(\tilde{u}) \cdot (\varphi - \tilde{\varphi})\, dt \right|,$$

where $\tilde{\varphi}$ is a polynomial of degree m.

Since $\varphi(t_k) = e(t_k)$, it follows that

$$\left|e(t_k)\right| \le \left| \frac{\varphi(t_{k-1})}{\varphi(t_k)} \right| \left|e(t_{k-1})\right| + \min_{\tilde{\varphi}} \left| \int_{t_{k-1}}^{t_k} R(\tilde{u}) \frac{\varphi - \tilde{\varphi}}{\varphi(t_k)}\, dt \right|,$$

and from $\varphi'(t) = -\sigma(t) A^T \varphi(t)$ and $\mu(A^T) = \mu(A) = \max_i \operatorname{Re} |\lambda_i(A)| = 0$, it follows that

$$\varphi(t) = e^{\int_t^{t_k} \mu(t)\sigma(t)\, dt} \varphi(t_k) = \varphi(t_k).$$

Hence,

$$\left|e(t_k)\right| \le \left|e(t_{k-1})\right| + \min_{\tilde{\varphi}} \left| \int_{t_{k-1}}^{t_k} R(\tilde{u}) \frac{\varphi - \tilde{\varphi}}{\varphi(t_k)}\, dt \right|.$$

Under sufficient regularity assumptions the last term can be bounded by $C\tau^{2m}$. Hence, the discretization error grows linearly with time,

$$\left|e(t_k)\right| \le C t_k \tau^{2m}, \quad k = 0, 1, \ldots$$

i.e. the implicit Runge-Kutta method, based on Gaussian quadrature, applied for hyperbolic (conservative) problems has order $2m$.

6 A numerical test example

We consider the linear parabolic problem,

$$\frac{\partial u}{\partial t} + \sigma(t)(-\Delta u + \mathbf{b} \cdot \nabla u - f) = 0, \quad t > 0 \tag{26}$$

in the unit square domain $\Omega = [0,1]^2$ with boundary condition

$$\begin{cases} u = 0 & \text{on parts } y = 0, y = 1, \\ \frac{\partial u}{\partial v} + \ell u = g, \ell \ge 1 & \text{on parts } x = 0, x = 1. \end{cases} \tag{27}$$

As initial function u_0, we choose a tent-like function with $u_0 = 1$ at the center of Ω and $u_0 = 0$ on $\partial\Omega$; see Figure 1.

Here, $\sigma(t) = 1 + \frac{2}{5} \sin k\pi t$, where $k = 1, 2, \ldots, k \le \frac{1}{\tau}$, is a parameter used to test the stability of the method with respect to oscillating coefficients. Here, τ is the time step to be used in the numerical solution of (26). Note that this function $\sigma(t)$ satisfies the conditions of the ratio $\frac{\sigma_1}{\sigma_2}$ from (9). We let $f(x,y) \equiv 2e^{-\ell x}$.

Further $\mathbf{b}$ is a vector satisfying $\nabla \cdot \mathbf{b} \le 0$. We choose $\mathbf{b} = -[\ell, 0]$, where ℓ is a parameter, possibly $\ell \gg 1$.

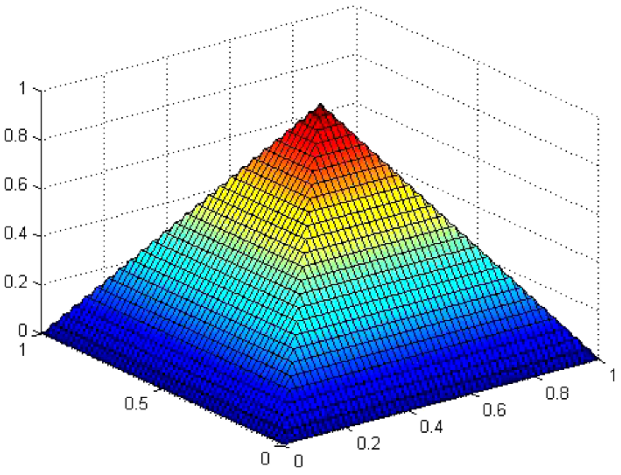

Figure 1 Initial function.

After a finite element or finite difference approximation, a system of the form (4) arises. For a finite difference approximation $M = I$, the identity matrix. The Laplacian operator is approximated with a nine-point difference scheme. We use an upwind discretization of the convection term. In the outer corner points of the domain, we use the boundary conditions $-u_x + \ell u = 0$ for $x = 0$ and $u_x + \ell u = 0$ for $x = 1$.

The time discretization is given by the implicit Runge-Kutta method with the Radau quadrature for $m = 2$; see Section 3. For comparison, we also consider $m = 1$, *i.e.* the implicit Euler method, in some experiments. For solving the time-discretized problems, we use the GMRES method with preconditioners from Section 2 and with the tolerance $1e - 10$. Let us note that GMRES needs 5-6 iteration for this tolerance. The problem is implemented in Matlab.

The primary aim is to show how the time-discretization errors decrease and how fast the numerical approximation of (26)-(27) approaches its stationary value, *i.e.* the corresponding numerical solution to the stationary problem

$$\begin{cases} -\Delta \hat{u} + \mathbf{b} \cdot \nabla \hat{u} = 2e^{-\ell x} & \text{in } \Omega, \\ u = 0 & \text{on parts } y = 0, y = 1, \\ \frac{\partial u}{\partial v} + \ell u = g & \text{on parts } x = 0, x = 1. \end{cases} \tag{28}$$

6.1 Experiments with a known and smooth stationary solution

If we let

$$g(y) = \begin{cases} 2\ell y(1-y) & \text{for } x = 0, \\ 0 & \text{for } x = 1 \end{cases}$$

then the solution to (28) satisfies

$$\hat{u}(x, y) = e^{-\ell x} y(1-y).$$

Table 1 The error estimates in dependence on ℓ and h

$\ell \setminus h$	1/10	1/20	1/50	1/100	1/150
1	1.2e–2	5.9e–3	2.3e–3	1.2e–3	7.7e–4
20	6.1e–1	4.5e–1	2.5e–1	1.4e–1	9.4e–2

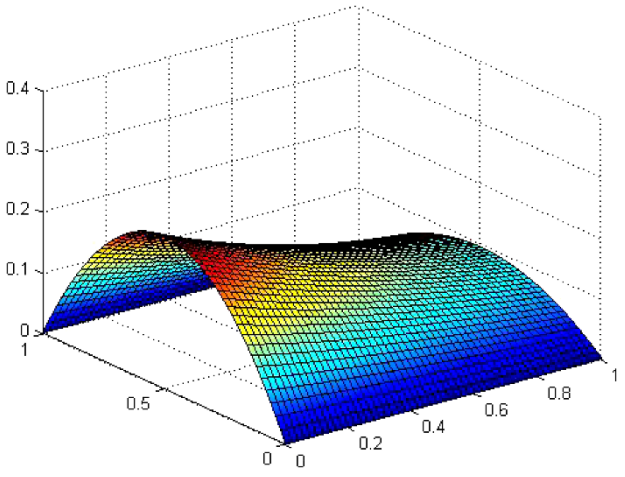

Figure 2 Numerical stationary solution for $\ell = 1$.

First, we will investigate the influence of the space discretization error on the stationary problem (28). To this end, we use the relative error estimate in the Euclidean norm:

$$e_h = \frac{\|\hat{\mathbf{u}}_h - \hat{\mathbf{u}}\|_2}{\|\hat{\mathbf{u}}\|_2}.$$

Here, $\hat{\mathbf{u}}$, $\hat{\mathbf{u}}_h$ denote the vectors representing the exact and numerical solutions to (28) at the nodal points, respectively. The error estimates in dependence on ℓ and h are found in Table 1. It is seen that the error decay is $O(h)$. This is caused by the use of first order upwind approximation of the convection term.

In Figures 2 and 3, there are depicted numerical stationary solutions for $\ell = 1$ and $\ell = 20$, respectively. The discretization parameter is $h = 1/50$.

Now we will investigate how fast the numerical solution to (26)-(27) approaches the numerical solution to (28) in dependence on τ. We fix $k = 10$ and we search the smallest time T for which

$$\frac{\|\mathbf{u}_h(T) - \hat{\mathbf{u}}_h\|_2}{\|\hat{\mathbf{u}}_h\|_2} < 10^{-6},$$

where the vectors $\hat{\mathbf{u}}_h$ and $\mathbf{u}_h(T)$ represent the numerical solution to (28) and the numerical solution to (26)-(27) at time T, respectively. The results for various ℓ and h are in Table 2. We can observe that the dependence of the results on h is small. For smaller ℓ, the final time does not depend on τ, while for larger ℓ, the dependence on τ is more significant.

Finally, we investigate how the time-discretization error decrease in dependence on τ at a fixed, relatively small, time $T = 1/8$. We consider five different time-discretization parameters: $\tau_1 = T$, $\tau_2 = T/2$, $\tau_3 = T/4$, $\tau_4 = T/8$, and $\tau_5 = T/16$. We will compare the max-

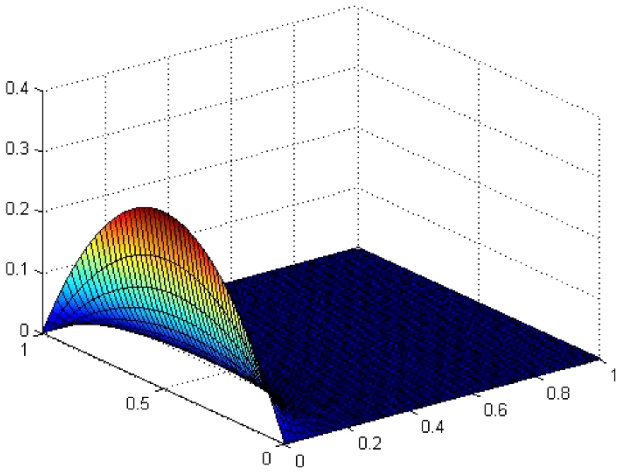

Figure 3 Numerical stationary solution for $\ell = 20$.

Table 2 Values of time T in dependence on h and τ

$h \setminus \tau$	1/5	1/10	1/20	1/40
1/20	1.40	1.30	1.25	1.25
1/50	1.40	1.30	1.25	1.25

$h \setminus \tau$	1/5	1/10	1/20	1/40
1/20	1.40	0.70	0.35	0.18
1/50	1.60	0.80	0.40	0.18

We use $\ell = 1$ (left) and $\ell = 20$ (right).

Table 3 Time discretization error at time $T = 1/8$ in dependence on h and τ

$h \setminus i$	1	2	3	4
1/20	1.7e−1	1.6e−2	3.0e−4	9.5e−6
1/50	1.8e−1	2.0e−2	5.6e−4	4.5e−6
1/100	1.8e−1	2.1e−2	6.8e−4	3.0e−6

Table 4 Time discretization error at time $T = 1/8$ in dependence on ℓ and τ

$\ell \setminus i$	1	2	3	4
1	7.1e−2	3.9e−3	2.1e−4	2.8e−5
20	1.8e−1	2.0e−2	5.6e−4	4.5e−6

Table 5 Time discretization error at time $T = 1/8$ in dependence on k and τ

$k \setminus i$	1	2	3	4
0	1.1e−1	2.4e−2	4.3e−4	2.4e−5
10	1.8e−1	2.0e−2	5.6e−4	4.5e−6

imal differences between the vectors $\mathbf{u}_i(T)$ and $\mathbf{u}_{i+1}(T)$, $i = 1, \dots, 4$, where $\mathbf{u_i}(T)$ represents the numerical solution to (26)-(27) at time T for the time-discretization parameter τ_i, $i = 1, \dots, 5$. So, we investigate the following error:

$$e_i = \left\| \mathbf{u}_{i+1}(T) - \mathbf{u}_i(T) \right\|_\infty, \quad i = 1, \dots, 4,$$

which values are found in Tables 3-5. If we let $k = 10$, $\ell = 20$ and use various h, we obtain results written in Table 3. It is seen that the influence of h on the time discretization error

Table 6 Time discretization error at time $T = 1/8$ in dependence on ℓ and τ for the implicit Euler method

$k \setminus i$	1	2	3	4
0	7.8e–2	4.2e–2	1.7e–2	4.6e–3
10	2.5e–2	2.5e–2	4.8e–2	2.2e–2

Table 7 Values of stabilized time T in dependence on ℓ and τ

$\ell \setminus \tau$	1/5	1/10	1/20	1/40
1	1.40	1.30	1.25	1.23
20	1.60	0.80	0.45	0.20

We let $h = 1/50$.

Table 8 Time discretization error at time $T = 1/8$ in dependence on ℓ and τ

$l \setminus i$	1	2	3	4
1	7.4e–2	4.0e–2	2.0e–4	2.6e–5
20	1.8e–1	1.9e–2	5.6e–4	4.5e–6

is small for the larger time steps but more noticeable for the smaller time steps when the time and space discretization errors are of the same order.

If we let $k = 10$, $h = 1/50$, $\ell = 1$ and $\ell = 20$, we obtain results in Table 4. We can see that the investigated time-discretization error decreases faster for $\ell = 20$ than for $\ell = 1$.

If we let $k = 0$, $k = 10$, $h = 1/50$, and $\ell = 20$, we obtain results in Table 5.

The error estimates from Tables 3-5 indicate that the expected error estimate $O(\tau^3)$ holds.

For comparison, we perform the same experiment as in Table 5 for the implicit Euler time discretization. The results are in Table 6.

The error estimates are here significantly influenced by the oscillation parameter k. For the larger value $k = 10$, we do not observe convergence. In case $k = 0$, the convergence is first order $O(\tau)$, that is, much slower than for the Runge-Kutta method with the two-point Radau quadrature.

6.2 Experiments with an unknown and less smooth stationary solution

Here, we replace the above defined function g with the following one:

$$g(y) = \begin{cases} \left.\begin{matrix} y(1-y), & y < 1/4 \text{ or } y > 3/4, \\ e^{2|y-1/2|}, & 1/4 \le y \le 3/4 \end{matrix}\right\} & \text{for } x = 0, \\ 0 & \text{for } x = 1 \end{cases}$$

and prepare Tables 7 and 8 correspondingly to Tables 2 and 4, respectively. The results in Tables 7 and 8 are very similar to the results from Tables 2 and 4. It means that less smoothness in space of the solution to (26)-(27) do not significantly influence the time-discretization error.

In Figures 4 and 5, there are depicted numerical stationary solutions for $\ell = 1$ and $\ell = 20$, respectively. The discretization parameter is $h = 1/50$.

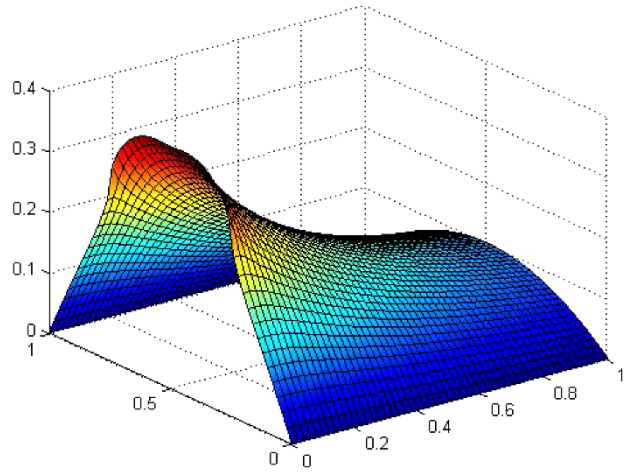

Figure 4 Numerical stationary solution for $\ell = 1$.

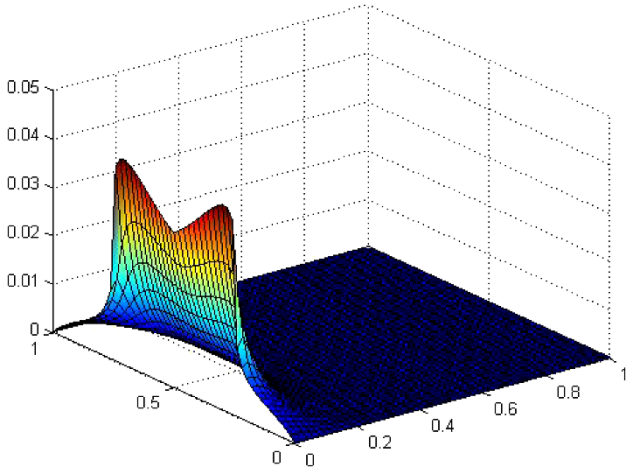

Figure 5 Numerical stationary solution for $\ell = 20$.

7 Concluding remarks

There are several advantages in using high order time integration methods. Clearly, the major advantage is that the high order of discretization errors enables the use of larger, and hence fewer timesteps to achieve a desired level of accuracy. Some of the methods, like Radau integration, are highly stable, *i.e.* decrease unwanted solution components exponentially fast and do not suffer from an order reduction, which is otherwise common for many other methods. The disadvantage with such high order methods is that one must solve a number of quadratic matrix polynomial equations. For this reason, much work has been devoted to development of simpler methods, like diagonally implicit Runge-Kutta methods; see *e.g.* [10]. Such methods are, however, of lower order and may suffer from order reduction.

In the present paper, it has been shown that the arising quadratic matrix system polynomial factors can be handled in parallel and each of them can be solved efficiently with a preconditioning method, resulting in very few iterations. Each iteration involves just

two first order matrix real valued factors, similar to what arises in the diagonal implicit Runge-Kutta methods. An alternative, stabilized explicit Runge-Kutta methods, *i.e.* methods where the stability domain has been extended by use of certain forms of Chebyshev polynomials; see, *e.g.* [17] can only be competitive for modestly stiff problems.

It has also been shown that the methods behave robustly with respect to oscillations in the coefficients in the differential operator. Hence, in practice, high order methods have a robust performance and do not suffer from any real disadvantage.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
Each of the authors, OA, RB, SS and BA, contributed to each part of this work equally and read and approved the final version of the manuscript.

Author details
[1]King Abdulaziz University, Jeddah, Saudi Arabia. [2]IT4 Innovations Department, Institute of Geonics AS CR, Ostrava, Czech Republic.

Acknowledgements
This paper was funded by King Abdulaziz University, under grant No. (35-3-1432/HiCi). The authors, therefore, acknowledge technical and financial support of KAU.

Received: 15 February 2013 Accepted: 9 April 2013 Published: 26 April 2013

References

1. Butcher, JC: Numerical Method for Ordinary Differential Equations, 2nd edn. Wiley, Chichester (2008)
2. Butcher, JC: Implicit Runge-Kutta processes. Math. Comput. **18**, 50-64 (1964)
3. Axelsson, O: A class of *A*-stable methods. BIT **9**, 185-199 (1969)
4. Axelsson, O: Global integration of differential equations through Lobatto quadrature. BIT **4**, 69-86 (1964)
5. Axelsson, O: On the efficiency of a class of *A*-stable methods. BIT **14**, 279-287 (1974)
6. Axelsson, O, Kucherov, A: Real valued iterative methods for solving complex symmetric linear systems. Numer. Linear Algebra Appl. **7**, 197-218 (2000)
7. Varga, RS: Functional Analysis and Approximation Theory in Numerical Analysis. SIAM, Philadelphia (1971)
8. Gear, CW: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice Hall, New York (1971)
9. Fried, I: Optimal gradient minimization scheme for finite element eigenproblems. J. Sound Vib. **20**, 333-342 (1972)
10. Hairer, E, Wanner, G: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (1996)
11. Axelsson, O: Iterative Solution Methods. Cambridge University Press, Cambridge (1994)
12. Hairer, E, Lubich, Ch, Roche, M: The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods. Lecture Notes in Mathematics, vol. 1409. Springer, Berlin (1989)
13. Petzold, LR: Order results for implicit Runge-Kutta methods applied to differential/algebraic systems. SIAM J. Numer. Anal. **23**(4), 837-852 (1986)
14. Axelsson, O: Error estimates over infinite intervals of some discretizations of evolution equations. BIT **24**, 413-424 (1984)
15. Wanner, G: A short proof on nonlinear *A*-stability. BIT **16**, 226-227 (1976)
16. Frank, R, Schneid, J, Ueberhuber, CW: The concept of *B*-convergence. SIAM J. Numer. Anal. **18**, 753-780 (1981)
17. Hundsdorfer, W, Verwer, JG: Numerical Solution of Time Dependent Advection-Diffusion-Reaction Equations. Springer, Berlin (2003)

doi:10.1186/1687-2770-2013-108
Cite this article as: Axelsson et al.: On the solution of high order stable time integration methods. *Boundary Value Problems* 2013 **2013**:108.