Energy-conserving Runge-Kutta methods for the incompressible Navier-Stokes equations

This paper was published in Journal of Computational Physics 233 (2013), p. 100-131

B. Sanderse

November 2012
ECN-W--12-046

Journal of Computational Physics 233 (2013) 100–131

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Energy-conserving Runge–Kutta methods for the incompressible Navier–Stokes equations

B. Sanderse *

Energy research Centre of the Netherlands (ECN), Petten, The Netherlands
Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Received 6 December 2011
Received in revised form 6 April 2012
Accepted 25 July 2012
Available online 17 August 2012

Keywords:
Energy conservation
Time reversibility
Runge–Kutta method
Additive Runge–Kutta method
Incompressible Navier–Stokes equations
Differential–algebraic equations
Algebraic stability
L stability
Stiffness

ABSTRACT

Energy-conserving methods have recently gained popularity for the spatial discretization of the incompressible Navier–Stokes equations. In this paper implicit Runge–Kutta methods are investigated which keep this property when integrating in time. Firstly, a number of energy-conserving Runge–Kutta methods based on Gauss, Radau and Lobatto quadrature are constructed. These methods are suitable for convection-dominated problems (such as turbulent flows), because they do not introduce artificial diffusion and are stable for any time step. Secondly, to obtain robust time-integration methods that work also for stiff problems, the energy-conserving methods are extended to a new class of additive Runge–Kutta methods, which combine energy conservation with *L*-stability. In this class, the Radau IIA/B method has the best properties. Results for a number of test cases on two-stage methods indicate that for pure convection problems the additive Radau IIA/B method is competitive with the Gauss methods. However, for stiff problems, such as convection-dominated flows with thin boundary layers, both the higher order Gauss and Radau IIA/B method suffer from order reduction. Overall, the Gauss methods are the preferred method for energy-conserving time integration of the incompressible Navier–Stokes equations.

1. Introduction

1.1. Advantages of energy-conserving methods

In this work we address the time integration of flows governed by the incompressible Navier–Stokes equations. In case of inviscid flow with periodic or no-slip boundary conditions these continuous equations possess a number of properties, also called *symmetries* or *invariants*, see e.g. [1]. Such inviscid flows are of interest because many flows of practical importance are convection-dominated. In this paper we focus on one important invariant of inviscid incompressible flows, namely the kinetic energy. Upon discretizing the continuous equations in space and/or time this invariant is often not conserved. There are, however, several (related) reasons to have a discretization conserving energy in a discrete sense. Firstly, from a physical point of view, an energy-conserving scheme is free of numerical diffusion. This is important for turbulent flow simulations with DNS or LES, because it prevents numerical diffusion from overwhelming the molecular diffusion (in case of DNS) or the effect of the sub-grid model (in case of LES), so that the energy spectrum is not affected. Energy-conserving discretizations guarantee that all diffusion is modeled (laminar and/or turbulent), and not artificial. This is why energy-conserving schemes

* Address: Energy research Centre of the Netherlands (ECN), Petten, The Netherlands. Tel.: +31 205924161.
E-mail address: sanderse@ecn.nl

0021-9991/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jcp.2012.07.039

are seen as a necessity for DNS and LES by different researchers, see e.g. [2–8]. Energy-conserving methods necessitate the use of central schemes for the convective terms. Upwind schemes, typically used in RANS simulations of turbulent flows, are robust because they introduce numerical diffusion, but should for this reason not be used in LES or DNS. Even high-order upwind methods can damp turbulence fluctuations and mask the effects of the sub-grid scale models used in LES [9–11]. Although central schemes do not have numerical diffusion, they introduce dispersive errors; these were found to be less detrimental than diffusive errors, at least in the simulation of turbulent channel flow [12]. Secondly, from a more mathematical point of view, discrete energy conservation provides a non-linear stability bound to the solution (see e.g. [13]). Flow simulations are then stable for any mesh and any time step, so that these parameters can be chosen purely based on accuracy requirements. This is especially important for simulating turbulent flows that involve large time and/or length scales, like weather prediction [14]. Lastly, energy-conserving methods are important when dealing with coarse grids and large time steps. Simulations of turbulence with DNS and LES are computationally very expensive and mesh sizes are kept as large as possible in practice, even under-resolved. On the other hand, the order of a discretization scheme is defined for vanishing mesh sizes and time steps, and on coarse grids it is not obvious whether the order of a method is still a good measure of accuracy and whether a formally higher order method is preferred over a formally lower order method [15,16]. Energy-conserving methods are of particular interest then, because they lead to well-posed discrete operators and as a consequence to well-behaved global errors.

One application of energy-conserving methods that we have in mind is wind-turbine wake aerodynamics, a situation in which turbulent flow structures travel over large distances while mixing with the atmospheric boundary layer [17]. This is an example in which one cannot resolve the small scale turbulent fluctuations in detail; we believe that preserving their total energy and dissipation rate is then of utmost importance.

Discrete energy conservation requires an appropriate spatial discretization and an appropriate temporal discretization method. The focus of this paper is on the latter, but in order to provide a background for the reader, we will summarize existing work on both spatial and temporal energy conservation.

1.2. Overview of spatially energy-conserving schemes

The first energy-conserving scheme for the incompressible Navier–Stokes equations is probably Harlow and Welch's staggered grid method [18]. The staggering of the variables leads to a method that conserves mass, momentum, energy and vorticity, and strongly couples pressure and velocity, making it the method of choice for simulating incompressible flows on Cartesian grids [19]. Ham et al. [20] extended the method to retain these properties on non-uniform grids. In order to simulate flows around complex geometries, Wesseling et al. [21,19] extended the method to general structured grids that can be described by a Cartesian grid in computational space. On unstructured meshes staggered methods have been investigated by Perot and co-workers [3,22]. Mahesh et al. [2] also consider energy-conserving methods for unstructured meshes and formulate a second order staggered method for tetrahedral elements, but propose a non-staggered formulation for elements of more general shape. This results in a formulation that is not fully energy-conserving, because the pressure gradient contributes to the kinetic energy.

In fact, the contribution of the pressure gradient to the kinetic energy is intrinsic to non-staggered layouts [23,12]. Felten and Lund [12] show that this energy error makes staggered schemes superior to collocated schemes in case of inviscid simulations and in case of viscous simulations with relatively coarse meshes. However, operators on collocated meshes can be 'shifted' to obtain operators for staggered meshes, as shown by Hicken et al. [24]. This elegant approach can be seen as a generalization of the work of Perot to general meshes [3], including locally refined ones. Numerical experiments show that on anisotropic Cartesian grids, where the local truncation error of the gradient is inconsistent (zeroth order), it is still possible to have a first order accurate solution, emphasizing the positive influence of energy-conserving methods on global discretization errors.

High-order energy-conserving methods have been addressed for finite difference methods by Morinishi et al. [23] and Vasilyev [25], who indicate how to obtain any (even) order of accuracy on uniform grids based on Richardson extrapolation. On non-uniform grids strict conservation and (local) order of accuracy cannot be obtained simultaneously. Verstappen and Veldman [8,7] employ a finite volume method and construct a fourth order accurate method on non-uniform grids that retains all properties of the Harlow and Welch scheme. They call their method 'symmetry-preserving', because it is based on mimicking symmetry properties of continuous operators in a discrete sense, instead of on minimizing the local truncation error. Preserving these symmetries leads to energy conservation. Again, it should be stressed that the global order of accuracy cannot be derived from the local truncation error alone; the discretized operator is at least of equal importance, and it is indeed the energy-conserving schemes that lead to a discretized operator that is well-posed, even on coarse meshes.

Another way to obtain higher order methods is to use compact (implicit) schemes, either in finite difference or finite volume context (see e.g. [26,27] for references). Implementation of boundary conditions is easier due to their smaller computational stencil, and furthermore they have better resolution of high wave numbers than explicit schemes. Knikker [26] obtains energy conservation with a fourth order compact finite difference scheme with the nonlinear terms in skew-symmetric form, Hokpunna and Manhart [27] mention that energy conservation for compact finite volume methods is still an open issue.

Lastly, this section would be incomplete without mentioning the recent review article of Perot [28] and the work on 'mimetic' methods, like the support-operator method by Shashkov [29], Hyman et al. [30], the work on summation-by-parts

operators initiated by Kreiss and Scherer [31] and the dual-mesh methods of Nicolaides [32], Subramanian and Perot [33] and Mullen et al. [34].

1.3. Energy-conserving time-integration methods

Most of the above cited work focuses on energy conservation for the spatial discretization only. Time-integration methods used in conjunction with the energy-conserving methods from the previous paragraph are typically explicit methods, like multi-stage Runge–Kutta (RK) or multi-step Adams–Bashforth (AB), possibly with an implicit Crank–Nicolson (CN) scheme for the diffusive term in case of wall-bounded flows. The divergence-free constraint is imposed with a fractional step approach, that decouples the treatment of velocity and pressure when marching in time. Examples are Wray's RK3 method (with or without CN) [23,26,35,36], AB2 (with or without CN) [2,37] or a generalization of AB2 [7]. The AB2 method is unstable for pure convection problems, and the explicit Runge–Kutta methods of order higher than 2 usually add a small amount of dissipation, because their amplification factors are smaller than unity inside the linear stability domain on the imaginary axis. In general such explicit methods are efficient if the time step is dictated by accuracy, and not by stability. In that case the errors associated with the spatial discretization are typically much larger than the errors associated with the temporal discretization [16].

Relatively little research has been performed on the use of energy-conserving methods for the Navier–Stokes equations. One example is the work of Ham et al. [20], who employ the implicit midpoint method and solve pressure and velocity coupledly. They find that the statistics of the DNS of turbulent channel flow are much less sensitive to the time step than results obtained by Choi and Moin [38] (who use a Crank–Nicolson scheme with a fractional step method), although the CFL number was still limited to approximately 4 in order to have solutions that were accurate enough. Mullen et al. [34] also apply the implicit midpoint rule and indicate how to accelerate the solution of the non-linear system. It is found that the Crank–Nicolson method is not exactly energy-conserving, but oscillates around the initial energy. Simo and Armero [39] recognized that it is possible to conserve energy in time (and therefore obtain non-linear stability) with a linear implicit scheme, since the time level of the convecting quantity does not affect the secondary conservation properties of the convected quantity. For the laminar flow over a circular cylinder it is shown that the linear implicit scheme is as accurate as the non-linear implicit scheme. Mullen et al. [34] take a similar approach in their hybrid time integration, taking vorticity evaluated at the previous time level. Turek [40] also discusses such linear implicit schemes, but not in conjunction with energy conservation.

A possible drawback of these energy-conserving methods is that they are necessarily implicit, leading to a higher computational cost per time step as compared to explicit methods (namely the solution of a non-linear saddle point problem at each time step instead of a Poisson equation for the pressure). A common view is that the time step for the simulation of turbulent flows is dictated by accuracy, and not by (convective) stability, so that an implicit treatment of the convective terms is not efficient. This, however, depends on the specific flow problem [16]. For example, Verstappen et al. [41] argue that in many turbulent flows the time step imposed by the convective stability limit can be much smaller than the characteristic time-scale of the smallest resolved eddies. Similarly, Vreman [42] mentions that the time step determined by the stability restriction of the numerical scheme is considerably smaller than the shortest turbulent time-scale. Consequently, the truncation errors from the spatial discretization method are more important than truncation errors resulting from the discretization in time, when using explicit methods. This opens the perspective for efficient application of implicit methods. Turek [40] also suggests to use implicit methods, for both convection and diffusion, to allow for larger time steps.

Another potential drawback of energy-conserving methods is that they are not well-suited for stiff problems. Methods like implicit midpoint and Crank–Nicolson are not L-stable, meaning that they hardly damp stiff components (arising for example from initial and/or boundary conditions), and oscillatory solutions can result. One of the matters we address in this paper is the construction of new methods that are both energy-conserving and L-stable.

Related to energy conservation is the time-reversibility property of the inviscid equations, since both properties are destroyed as soon as viscosity is introduced. These properties can therefore be used to assess if a discretization introduces numerical diffusion, as suggested by Duponcheel et al. [43], although in fact no fully reversible time integration methods are used in that work. A time-reversibility test was performed by Ham et al. [20], showing that the implicit midpoint method is time-reversible in the inviscid limit, until round-off errors contaminate the solution. However, questions such as 'are there energy-conserving methods which are not time-reversible?', 'are there time-reversible methods which are not energy-conserving?', and 'is one of these properties more important than the other for fluid simulations?' have apparently not yet been assessed for the incompressible Navier–Stokes equations.

1.4. Outline of paper

In this paper we will investigate the use of Runge–Kutta methods for high-order, energy-conserving time integration of the incompressible Navier–Stokes equations. To the author's knowledge, the application of these methods to the incompressible Navier–Stokes equations is new. Runge–Kutta methods can combine high-order with good stability and efficiency, they allow easy adaptive step size selection and are self-starting; multi-step methods, although less costly, do not have these properties, and it is questionable if they can be energy-conserving [44]. We start with the general framework for Runge–Kutta methods for incompressible flows as developed in [45]. We investigate higher order methods that are energy-conserving and/or time-reversible. Subsequently we propose a new class of additive Runge–Kutta methods that combine energy-conservation

properties with L-stability. A unified approach for obtaining a high-order accurate pressure then concludes the theoretical analysis. Numerical experiments on different test cases are carried out to show the performance of the methods.

2. Spatial discretization

The governing equations for incompressible flow are those describing conservation of mass and momentum:

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \nu \nabla^2 \boldsymbol{u}, \tag{2}$$

defined on a domain Ω and supplemented with suitable initial conditions and boundary conditions. Sometimes we will make an explicit distinction between the *convecting* quantity $\boldsymbol{c}(\boldsymbol{x}, t)$ and the *convected* quantity $\boldsymbol{u}(\boldsymbol{x}, t)$ (see e.g. [23,27]), and write the non-linear convective term as $(\boldsymbol{c} \cdot \nabla)\boldsymbol{u}$. The equation for kinetic energy $k = \frac{1}{2}|\boldsymbol{u}|^2$ is, for incompressible flows, derived from the momentum equation and as such expresses secondary conservation. Upon integration over the entire domain one obtains, with periodic or no-slip boundary conditions,

$$\frac{\mathrm{d}K}{\mathrm{d}t} = -\nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{u}), \tag{3}$$

where the total kinetic energy is defined by $K = \int_\Omega k \,\mathrm{d}\Omega = \frac{1}{2}(\boldsymbol{u}, \boldsymbol{u})$, and the inner product is denoted by $(\boldsymbol{a}, \boldsymbol{b}) = \int_\Omega \boldsymbol{a} \cdot \boldsymbol{b} \,\mathrm{d}\Omega$. For tensors this is extended by defining $(\boldsymbol{A}, \boldsymbol{B}) = \int_\Omega \mathrm{tr}[\boldsymbol{A}^T \boldsymbol{B}] \,\mathrm{d}\Omega$. Since $(\nabla \boldsymbol{u}, \nabla \boldsymbol{u}) \geqslant 0$, this equation shows that the total energy of the flow can only decrease, due to viscous dissipation:

$$K(t) \leqslant K(t_0), \quad t \geqslant t_0. \tag{4}$$

If Eq. (3) or (4) can be satisfied in a discrete sense, then the discrete solution is bounded, independent of grid layout or time step.

In deriving the continuous energy equation in integral form, Eq. (3), a number of mathematical operations are performed, such as application of Gauss' divergence theorem and application of the product rule. For a method to conserve energy these operations should be mimicked on a *discrete* level. An example of such a method is the second-order method of Harlow and Welch in finite volume form on staggered Cartesian grids [18], which we will employ here; other examples of suitable schemes are e.g. [7,20,23]. This leads to the semi-discrete representation of Eqs. (1) and (2) (method of lines):

$$Mu(t) = r_1(t), \tag{5}$$

$$\Omega \dot{u}(t) = -C(c(t), u(t)) + \nu D u(t) - G p(t) + r_2(u(t), t). \tag{6}$$

The explicit t-dependence of $u(t)$ (=$c(t)$) and $p(t)$ will be left out in what follows. For more information on notation see [45]. Ω is a matrix with the finite volume sizes on its diagonal. We take the discrete inner product $(u, v)_\Omega = v^T \Omega u$, with a corresponding energy norm $K = \frac{1}{2}(u, u)_\Omega = \frac{1}{2}\|u\|_\Omega^2$. This choice of inner product is consistent with the numerical method for the convective terms. The discrete operators satisfy (when $r_1(t) = 0$ and $r_2(t) = 0$)

$$(Gp, u)_\Omega = -(p, Mu)_\Omega, \tag{7}$$

$$(C(c, u), u)_\Omega = 0, \tag{8}$$

$$(Du, v)_\Omega = (u, Dv)_\Omega, \quad (Du, u)_\Omega \leqslant 0. \tag{9}$$

The first property is the relation between the divergence and gradient operator. The second property indicates that the convective terms do not change the total energy of the flow (if $Mc = 0$). The third property shows that the diffusive terms dissipate energy. Note that in the remainder we will leave out the subscript Ω when denoting the inner product. The time-reversibility of the incompressible, inviscid Navier–Stokes equations (defined here as invariance under the transformation $(t, \boldsymbol{x}, \boldsymbol{u}, p) \to (-t, \boldsymbol{x}, -\boldsymbol{u}, p)$) is kept for the spatial discretization if the semi-discrete equations are invariant when changing u to $-u$ and t to $-t$. The discrete convection operator arising from the discretization of Harlow and Welch is indeed time-reversible because $C(-c, -u) = C(c, u)$. This property is easier to satisfy than the energy conservation property: the class of discrete time-reversible convective operators is much larger than the class of discrete energy-conserving convective operators.

3. Runge–Kutta methods for the incompressible Navier–Stokes equations

3.1. Introduction

We shortly summarize the introduction given in [45]. Eqs. (5) and (6) represent a differential–algebraic equation (DAE) system of index 2:

$$0 = Mu - r_1(t), \tag{10}$$

$$\dot{u} = F(u,t) - Gp, \tag{11}$$

where $F = -C(c,u) + \nu Du + r_2(u,t)$ and Ω^{-1} is absorbed in the definition of C, D, G and r_2. Applying the divergence-free constraint to the momentum equations results in the pressure Poisson equation:

$$Lp = MF(u,t) - \dot{r}_1(t), \tag{12}$$

where $L = MG$ is the Laplace operator, which should be invertible for a proper index 2 system. The initial conditions at $t = t_0$ should be consistent with Eqs. (10) and (11):

$$Mu_0 = r_1(t_0), \tag{13}$$

$$Lp_0 = MF(u_0,t_0) - \dot{r}_1(t_0). \tag{14}$$

Eq. (14) expresses that the initial pressure cannot be chosen freely, but has to be calculated based on u_0.

The exact solution to the semi-discrete equations at time t_n, $u(t_n)$, $p(t_n)$, will be approximated by the fully discrete solution u_n, p_n by means of a Runge–Kutta method. A general Runge–Kutta method applied to the index 2 DAE (10) and (11) reads [46,45]:

$$U_i = u_n + \Delta t \sum_{j=1}^{s} a_{ij}(F_j - G\psi_j), \quad MU_i = r_1(t_i), \tag{15}$$

$$u_{n+1} = u_n + \Delta t \sum_{i=1}^{s} b_i(F_i - G\psi_{n+1}), \quad Mu_{n+1} = r_1(t_{n+1}), \tag{16}$$

with $t_i = t_n + c_i\Delta t$ and s is the number of stages. 'The classical' order of the method (when applied to non-stiff ODEs) is called p (from the context confusion with the pressure will be avoided). The set of coefficients a_{ij}, b_i and c_i will be denoted collectively by $\mathcal{A}$ and can be written in compact form using a Butcher tableau:

$$\begin{array}{c|ccc} c_1 & a_{11} & a_{12} & \dots & a_{1s} \\ c_2 & a_{21} & a_{22} & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \dots & \dots & a_{ss} \\ \hline & b_1 & \dots & \dots & b_s \end{array} \tag{17}$$

with the convention

$$c_i = \sum_{j=1}^{s} a_{ij}, \tag{18}$$

not to be confused with the convecting velocity in (6). The pressure-like variable ψ is introduced to explicitly distinguish it from p, since ψ generally does not satisfy (12).

Similar to [45], we rewrite (15) and (16) as

$$U = u_n + \Delta t AF - \Delta t C^p G\phi, \tag{19}$$

with ϕ defined by

$$C^p L\phi = AMF - \frac{r_1 - r_1(t_n)}{\Delta t}, \tag{20}$$

and $C^p = \mathrm{diag}(c_1, \dots, c_s)$. This gives $\phi = A(C^p)^{-1}\psi$. We have used the following notation:

$$U = \begin{pmatrix} U_1 \\ \vdots \\ U_s \end{pmatrix}, \quad F = \begin{pmatrix} F_1 \\ \vdots \\ F_s \end{pmatrix}, \quad \phi = \begin{pmatrix} \phi_1 \\ \vdots \\ \phi_s \end{pmatrix}, \quad r_1 = \begin{pmatrix} r_1(t_1) \\ \vdots \\ r_1(t_s) \end{pmatrix}, \tag{21}$$

and we have written M instead of $I_s \otimes M$, G instead of $I_s \otimes G$, etc. The size of these matrices should be clear from the context. In this way Eq. (20) resembles Eq. (12), and the total gradient matrix is block-diagonal. In terms of ϕ a single Lagrange multiplier ϕ_i makes each velocity field U_i divergence-free, while in terms of ψ a sum of Lagrange multipliers would be required.

We note that, in contrast to the explicit methods discussed in [45], implicit methods can have $c_1 = 0$ (e.g. the methods based on Lobatto quadrature) so that C^p is singular. In that case we define a new pressure-like variable $\rho = C^p\phi$ and Eqs. (19) and (20) can be written without the c coefficients.

Once the stage values U_i are determined, u_{n+1} follows by introducing the Lagrange multiplier ϕ_{n+1} and rewriting Eq. (16) as the following equivalent uncoupled system:

$$\hat{u}_{n+1} = u_n + \Delta t \sum_{i=1}^{s} b_i F_i, \tag{22}$$

$$L\phi_{n+1} = \sum_{i=1}^{s} (b_i M F_i) - \frac{r_1(t_{n+1}) - r_1(t_n)}{\Delta t}, \tag{23}$$

$$u_{n+1} = \hat{u}_{n+1} - \Delta t G \phi_{n+1}. \tag{24}$$

This projection of $\hat{u}_{n+1}$ onto the space of divergence-free velocity fields is known as a projected Runge–Kutta method [47,46]. It leads to the 'classical' order of accuracy of the Runge–Kutta method (the accuracy for ODEs), as will be detailed in the next section. An alternative to this projection step is described by Jay [48], but requires the solution of a larger system of coupled non-linear equations, and is therefore not considered here.

There are two important cases for which the additional projection step can be avoided, making the solution of the Poisson equation (23) unnecessary:

- If the implicit method is stiffly accurate, i.e., $a_{si} = b_i$ and $c_s = 1$, then $u_{n+1} = U_s$, $\phi_{n+1} = \phi_s$ and the last projection step is redundant. This is for example the case for the Radau IIA, Lobatto IIIA and IIIC methods.
- If A is invertible and the boundary conditions $r_1(t)$ are not a function of time, ϕ_{n+1} can be expressed as a linear combination of ϕ's, so that a Poisson solve for ϕ_{n+1} (Eq. (23)) is not necessary. The equation for u_{n+1} then reads

$$u_{n+1} = u_n + b^T A^{-1}(U - u_n), \tag{25}$$

which satisfies the divergence-free constraint. In practice we therefore use formulation (25) instead of (22) and in case of unsteady boundary conditions Eq. (25) should be read for $\hat{u}_{n+1}$ and the additional projection is performed. This formulation also saves an additional evaluation of $F(U_i)$ when the nonlinear iterative process has converged.

Well-known examples of implicit Runge–Kutta methods are methods based on high-order quadrature formulas (like the Gauss, Lobatto and Radau methods, see e.g. [49,46]), and (single) diagonally implicit Runge–Kutta ((S)DIRK) methods, which have zeros in the upper triangle of the Butcher tableau. Each of these methods has certain advantages for integrating the incompressible Navier–Stokes equations. The Gauss methods, for example, have the highest possible order ($p = 2s$) given a certain number of stages; a well-known member is the implicit midpoint method ($s = 1$, $p = 2$). On the other hand, certain Lobatto and Radau methods have the advantage that they are stiffly accurate ($a_{si} = b_i$), which is an important property for differential–algebraic equations [47,46]: (i) it avoids the additional projection step mentioned above, and (ii) it leads to L-stable methods (provided that A is non-singular). L-stable methods are robust, give stable results for problems at any Reynolds number, and damp perturbations that originate from rough initial or boundary conditions. The (S)DIRK methods have the advantage that the stages of the Runge–Kutta method can be solved sequentially. The advantage of energy-conserving (and time-reversible) methods were already outlined in Section 1.1. In our opinion, an ideal method would combine all these properties, resulting in a high-order, energy-conserving, time-reversible, stiffly-accurate/L-stable (S)DIRK scheme.

In the next sections we formulate the order conditions for implicit Runge–Kutta methods and the conditions for energy conservation, time-reversibility and L-stability. The W-transformation of Hairer and Wanner [46] will appear frequently in this context and is therefore summarized in Appendix A.

3.2. Order considerations

The order conditions for the *local error* of Runge–Kutta methods can be obtained by using Butcher's analysis based on trees [50,51]. For high-order methods, say order higher than 4, it becomes cumbersome to derive methods that satisfy these (non-linear) conditions. For implicit Runge–Kutta methods one can fortunately make use of the following 'simplifying conditions':

$$B(p): \quad \sum_{i=1}^{s} b_i c_i^{q-1} = \frac{1}{q}, \quad q = 1, \ldots, p; \tag{26}$$

$$C(\eta): \quad \sum_{j=1}^{s} a_{ij} c_j^{q-1} = \frac{c_i^q}{q}, \quad i = 1, \ldots, s, \;\; q = 1, \ldots, \eta; \tag{27}$$

$$D(\zeta): \quad \sum_{i=1}^{s} b_i c_i^{q-1} a_{ij} = \frac{b_j}{q}(1 - c_j^q), \quad j = 1, \ldots, s, \;\; q = 1, \ldots, \zeta, \tag{28}$$

and a theorem of Butcher [52]: if $\mathcal{A}$ satisfies $B(p)$, $C(\eta)$, $D(\zeta)$ with $p \leqslant \eta + \zeta + 1$ and $p \leqslant 2\eta + 2$, then the method is of order p. However, there are cases in which this 'classical' or 'ODE' order of accuracy is not obtained. We group these in three categories.

Firstly, in case of very stiff ODEs order reduction occurs if a method is not stiffly accurate [46,51,53]. The stage order η of the method then determines the convergence of the error, instead of the classical order, and the value of the linear stability

function $R(z)$ at infinity is of importance. If $R(\infty) = 0$ (L-stable methods) or $R(\infty) = -1$ (e.g. Gauss with s odd), the global error behaves as $\mathcal{O}(\Delta t^{\eta+1})$, whereas for methods with $R(\infty) = 1$ (e.g. Gauss with s even) the global error behaves as $\mathcal{O}(\Delta t^{\eta})$ [51]. This effect will be illustrated in Section 7.3.

Secondly, order reduction can occur if index 2 DAEs are considered instead of ODEs. In some sense this is related to the order reduction for very stiff ODEs, by regarding the divergence-free constraint (5) as an infinitely stiff equation. For explicit methods we showed, based on an analysis with trees, that no order reduction occurs for the velocity, up to and including order 4, if the mesh is not moving in time [45]. For the implicit methods under consideration here such an analysis is not necessary; no order reduction occurs according to a number of theorems for which we refer to Hairer and Wanner [46] (Theorems 4.12, 5.10, and 5.13). On the other hand, the pressure suffers in general from order reduction. This will be detailed in Section 5. Convergence of the *global error* follows from Theorems 4.4 and 4.9 in [54], which require invertibility of the A-matrix and consistency of the initial values.

We note here that recently implicit Runge–Kutta methods have been applied to the incompressible Navier–Stokes equations by Montlaur et al. [55]. In that work the Gauss methods are dismissed as useful 'because they present higher order reduction [than Radau] when applied to DAEs with respect to ODEs'. *This is a misconception: the Gauss methods keep their ODE accuracy when projected Runge–Kutta methods are used* [46]. In Section 7 this will be supported by numerical experiments.

Thirdly, order reduction can occur upon simultaneously refining mesh and time-step combined with time-dependent inflow boundary conditions, due to so-called 'space–time errors'. This phenomenon will not be investigated here (for the same reasons mentioned in [45]).

3.3. Energy conservation

In this section we derive conditions for the coefficients of the Butcher tableau such that the resulting Runge–Kutta method is energy-conserving. In all cases we consider inviscid flow ($\nu = 0$) and periodic boundary conditions. The monotonicity property of the fully continuous equations, Eq. (4), is shared by the semi-discrete system of Eqs. (10) and (11):

$$\|u(t)\| \leqslant \|u(t_0)\|, \quad t \geqslant t_0, \tag{29}$$

due to properties (7)–(9). Equality holds in case $\nu = 0$. For the fully discrete equations, this monotonicity or energy-conservation property is investigated by taking the inner product of the fully discrete momentum equation, Eq. (16), with itself. First we write Eq. (16) with $r_1 = 0$ as

$$u_{n+1} = u_n + \Delta t \sum_{i=1}^{s} b_i P F_i, \tag{30}$$

where $P = I - GL^{-1}M$ is a projection operator that projects velocity fields onto the space of divergence-free velocity fields. Taking the norm of u_{n+1} and substituting the equation for the stages leads to

$$\|u_{n+1}\|^2 = \|u_n\|^2 + 2\Delta t \sum_{i=1}^{s} b_i (PF_i, U_i) - \Delta t^2 \sum_{i,j=1}^{s} (b_i a_{ij} + b_j a_{ji} - b_i b_j)(PF_i, PF_j). \tag{31}$$

Conservation of energy requires that the two last terms vanish. The term (PF_i, U_i), is zero if $(C(U_i, U_i), U_i) = 0$; the presence of the projection operator P does not influence the energy as long as $MU_i = 0$. For the last term to be zero the coefficients A and b from the Runge–Kutta tableau should satisfy

$$e_{ij} \equiv b_i a_{ij} + b_j a_{ji} - b_i b_j = 0. \tag{32}$$

In the literature Runge–Kutta methods that satisfy $E = (e_{ij}) = 0$ are known as *symplectic*, since this condition is obtained when applying a Runge–Kutta method to symplectic systems and requiring that the symplectic structure is preserved. In general, symplectic methods are *not* conserving energy, but in the case of the incompressible Navier–Stokes equations the energy is a quadratic invariant. Every Runge–Kutta method that preserves quadratic invariants is a symplectic method [44]. If a method satisfies Eq. (32), $(C(U_i, U_i), U_i) = 0$ and $MU_i = 0$, then the total kinetic energy is exactly conserved for inviscid flows:

$$\|u_{n+1}\|^2 = \|u_n\|^2. \tag{33}$$

For viscous flows the second term in (31) also contains the viscous contribution $(DU_i, U_i) \leqslant 0$, so Eq. (31) reads (with $E = 0$)

$$\|u_{n+1}\|^2 = \|u_n\|^2 + 2\Delta t \sum_{i=1}^{s} b_i (DU_i, U_i), \tag{34}$$

assuming $MU_i = 0$. This is the discrete equivalent of Eq. (3). Note that we define the energy 'pointwise' in time; at n, $n+1$, etc. Other definitions, such as the average value over a time step, can also be employed and are the subject of future work. Eq. (31) reveals that methods which have a nonnegative definite E and nonnegative b coefficients lead to energy-stable time-integration, for *any time step* (and mesh size). Using terminology from [46] methods that satisfy these conditions are called *algebraically stable*; algebraic stability is equivalent to B-stability as long as the c-coefficients are all distinct.

Table 1
Examples of energy-conserving methods.

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

(a) Gauss, $p = 2$

$$\begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

(b) Gauss, $p = 4$

$$\begin{array}{c|cc} \frac{1}{3} & \frac{3}{8} & -\frac{1}{24} \\ 1 & \frac{7}{8} & \frac{1}{8} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

(c) Radau IIB, $p = 3$

$$\begin{array}{c|ccc} 0 & \frac{1}{12} & -\frac{1}{6} & \frac{1}{12} \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{12} & \frac{5}{6} & \frac{1}{12} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

(d) Lobatto IIIE, $p = 4$

Table 2
Examples of time-reversible methods.

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

(a) Lobatto IIIA, $p = 2$

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{6} & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} & 0 \\ 1 & \frac{1}{6} & \frac{5}{6} & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

(b) Lobatto IIIB, $p = 4$

The energy-conservation condition (32) cannot be satisfied by explicit methods. Implicit energy-conserving Runge–Kutta methods can be constructed with the *W*-transformation of Hairer and Wanner [46], explained in Appendix A, in particular condition (A.6). This transformation shows that the Gauss methods are the methods with the highest possible order ($p = 2s$) that satisfy (32). For one order lower, $p = 2s - 1$, a one-parameter family of methods results, of which the Radau IB and IIB methods [56] are two examples. For $p = 2s - 2$ a two-parameter family of methods results, which includes the Lobatto IIIE methods (also denoted by IIID) [57,58]. Some examples are shown in Table 1. Of particular interest are diagonally-implicit energy-conserving methods. For $s = p = 2$ there is a one-parameter family which reads

$$\begin{array}{c|cc} c_1 & c_1 & 0 \\ c_1+\frac{1}{2} & 2c_1 & \frac{1}{2}-c_1 \\ \hline & 2c_1 & 1-2c_1 \end{array} \tag{35}$$

with $c_1 \in (0, \frac{1}{2})$ to keep the b coefficients positive (for $c_1 = 0$ or $\frac{1}{2}$ we have implicit midpoint). This family is recognized as the implicit midpoint applied twice over intervals of size $2c_1$ and $1 - 2c_1$, and is limited to second order; we call it DIRK E. As is indicated in Appendix B, it is not possible to construct algebraically stable or *A*-stable symplectic DIRK methods with $p > 2$.

An alternative for constructing energy-conserving methods with the *W*-transformation is to use the following transformation [59]

$$\mathcal{A}^* = \mathcal{E}(\mathcal{A}), \tag{36}$$

where $\mathcal{E}$ is such that

$$a^*_{ij} = \frac{1}{2}\left(a_{ij} + b_j\left(1 - \frac{a_{ji}}{b_i}\right)\right), \quad b^*_i = b_i, \quad c^*_i = \sum_j a^*_{ij}. \tag{37}$$

In this way an energy-conserving method can be obtained from *any* existing Runge–Kutta method. This does not necessarily lead to useful methods, because $\mathcal{E}$ is a non-linear transformation, which in general changes the order of the method. However, in case the original method $\mathcal{A}$ satisfies simplifying conditions $B(p)$, $C(\eta)$ and $D(\zeta)$, then the transformed method $\mathcal{E}(\mathcal{A})$ satisfies $B(p)$, $C(\xi)$ and $D(\xi)$, with $\xi = \min(\eta, \zeta)$ [59]. The transformation $\mathcal{E}$ relates certain quadrature methods. For example, the Radau IIB methods follow by applying $\mathcal{E}$ on the Radau IIA methods and the Lobatto IIIE methods follow from the Lobatto IIIC methods. These relations will be used to construct additive Runge–Kutta methods in Section 4.

In Table 3 we list a number of energy-conserving methods, together with other properties that will be discussed in subsequent sections.

Table 3
Properties of implicit Runge–Kutta methods investigated in this work.

Type	Energy-conserving	Time-reversible	Stiffly accurate	L-stable	Algebraically stable	Order (ODE & DAE)	η	ζ
Gauss	Y	Y	N	N	Y	$2s$	s	s
Radau IIA	N	N	Y	Y	Y	$2s-1$	s	$s-1$
Radau IIB	Y	N	N	N	Y	$2s-1$	$s-1$	$s-1$
Lobatto IIIA	N	Y	Y	N	N	$2s-2$	s	$s-2$
Lobatto IIIB	N	Y	N	N	N	$2s-2$	$s-2$	s
Lobatto IIIC	N	N	Y	Y	Y	$2s-2$	$s-1$	$s-1$
Lobatto IIIE(D)	Y	Y	N	N	Y	$2s-2$	$s-1$	$s-1$
DIRK E	Y	Y	N	N	Y	2	1	1

3.4. Time reversibility

Time reversibility is investigated by changing u_{n+1} to $-u_n$, u_n to $-u_{n+1}$ and $U_i = u_{n+c_i}$ to $-u_{n+1-c_i}$ in Eqs. (15) and (16) and checking if the resulting equations are equivalent to the original ones. This leads to the following conditions [60,44]:

$$a_{ij} + a_{s+1-i,s+1-j} = b_j, \tag{38}$$

$$b_i = b_{s+1-i}, \tag{39}$$

$$c_i = 1 - c_{s+1-i}. \tag{40}$$

Runge–Kutta methods satisfying these conditions are called *symmetric*. In terms of the permutation matrix $\widehat{P}$ they are written as $A + \widehat{P}A\widehat{P} = eb^T$, $\widehat{P}b = b$, $\widehat{P}c = e - c$, where $e = (1,\ldots,1)^T$. Like the symplecticity condition, explicit Runge–Kutta methods cannot satisfy the symmetry conditions.

Implicit methods, on the other hand, can satisfy the symmetry conditions. For $s = 1$ there is one solution to these conditions, being the 1-stage Gauss method (better known as the implicit midpoint method), see Table 1(a). For $s = 2$ condition (39) fixes the b-coefficients: $b_1 = b_2 = \frac{1}{2}$. Examples are the Gauss, Lobatto IIIA, IIIB and IIIE methods, shown in Tables 1(a), (b), (d) and 2(a), (b). The 2-stage Lobatto IIIA method is better known as the trapezoidal rule or Crank–Nicolson method. General (higher order) symmetric methods can be constructed with the W-transformation in a similar way as the energy-conserving methods, now by satisfying condition (A.8). Table 3 lists for some methods whether they are symmetric or not.

The symmetry conditions do not imply stability, so it is of interest which symmetric methods are algebraically stable, i.e., have a nonnegative definite E matrix and $b \geqslant 0$. By using the X-matrix, introduced in Appendix A, algebraic stability requires that

$$Y = X + X^T - e_1 e_1^T \quad \text{is nonnegative definite.} \tag{41}$$

From the W-transformation we know that symmetric methods have an X-matrix with a zero diagonal (except for $x_{1,1} = \frac{1}{2}$). As a result Y is a symmetric matrix with a zero diagonal, and consequently it cannot be nonnegative definite, except if it is completely zero. Since $Y = W^T E W$ this implies that *only energy-conserving* ($E = 0$) *symmetric methods are algebraically stable*. A similar conclusion was drawn in [43], where it was stated that energy conservation is a practical requirement for schemes to be time-reversible. However, algebraic stability is a strong stability requirement and in many situations A-stability suffices, so this 'practical' requirement might be too restrictive. An important example of an A-stable time-reversible method which is not energy-conserving is the Crank–Nicolson method (member of the Lobatto IIIA family). In fact this method is conserving another quantity (see e.g. [61]),

$$\|u_{n+1}\|^2 + \frac{1}{4}\Delta t^2 \|PF_{n+1}\|^2 = \|u_n\|^2 + \frac{1}{4}\Delta t^2 \|PF_n\|^2, \tag{42}$$

with $PF_n = F_n - GL^{-1}MF_n = F_n - Gp_n$ (and similarly for $n+1$). This statement is however not as strong as energy conservation, because $\|u_{n+1}\|^2$ can grow in time due to the contribution of $\|PF_n\|^2$.

3.5. Other properties

In the previous sections we derived the conditions for energy conservation and time reversibility and we mentioned some high-order methods that satisfy these conditions, see Table 3. Other desirable properties, such as L-stability and algebraic stability have also been listed in this table. The table reveals that all energy-conserving or time-reversible method are not stiffly accurate or L-stable (with the exception of the Lobatto IIIA methods which have singular A). This is not a coincidence. It can be shown that energy-conserving and time-reversible methods have $R = 1$ on the imaginary axis of the stability domain by applying them on a linear test equation. Since the stability function $R(z)$ is a rational function, it attains the same value when approaching infinity on either the negative real axis or the imaginary axis, and consequently $|R(\infty)| = 1$. Energy-conserving or time-reversible methods can therefore not satisfy the condition $R(\infty) = 0$, necessary for L-stability.

An alternative way of seeing this, which provides insight in how to construct L-stable methods, is to write the stability function as

$$R(z) = \frac{\mathrm{Det}(I - zQ)}{\mathrm{Det}(I - zA)}, \tag{43}$$

where $Q = A - eb^T$. L-stable methods require Q to be singular. Energy-conserving and time-reversible methods have $Q = -B^{-1}A^TB$ and $Q = -\widehat{P}A\widehat{P}$ ($B = \mathrm{diag}(b_1, \ldots, b_s)$), respectively, meaning that in both cases the eigenvalues of Q are $-1\times$ the eigenvalues of A. The degree of the numerator and denominator of $R(z)$ are therefore the same, so such methods cannot be L-stable.

This negative result leads us to consider *additive* Runge–Kutta methods, where the convective and diffusive terms are treated with different tableaux, in order to arrive at methods that possess energy-conservation, time-reversibility and L-stability.

4. Additive Runge–Kutta methods

The conditions for energy conservation and time reversibility are derived by considering time integration of the non-linear convective terms. As such, it is not surprising that these methods do not have ideal properties for integrating the diffusive terms. It seems therefore logical to take two different Runge–Kutta methods for the convective and the diffusive terms. We propose to use *additive Runge–Kutta methods* (ARK), a class of methods that includes the well-known implicit-explicit (IMEX) Runge–Kutta methods [62]. In implicit-explicit methods stiff terms (such as diffusion terms) are handled implicitly, whereas other terms (such as non-linear convective terms) are handled explicitly, both with a different Butcher tableau. Here we use the idea of two different Butcher tableaux, $\mathcal{A}$ and $\widehat{\mathcal{A}}$, but we take all terms implicitly, resulting in an 'implicit-implicit' method:

$$U_i = u_n + \Delta t \sum_{j=1}^{s} (a_{ij} P F_j + \hat{a}_{ij} P \widehat{F}_j) + GL^{-1}(r_1(t_i) - r_1(t_n)), \tag{44}$$

$$u_{n+1} = u_n + \Delta t \sum_{i=1}^{s} (b_i P F_i + \hat{b}_i P \widehat{F}_i) + GL^{-1}(r_1(t_{n+1}) - r_1(t_n)). \tag{45}$$

The diffusive terms are denoted by F and the convective terms by $\widehat{F}$. We require that the Butcher tableau for the convective terms, $\hat{\mathcal{A}}$, is energy-conserving, so that in case $\nu = 0$ no artificial viscosity is introduced. The Butcher tableau for the diffusive terms, $\mathcal{A}$, is chosen such that an L-stable method results, so that stiff problems are damped sufficiently. A natural way to obtain such properties is to use an algebraically stable and stiffly accurate method for $\mathcal{A}$ and its energy-transformed counterpart $\hat{\mathcal{A}} = \mathcal{E}(\mathcal{A})$ for the convective terms. From an implementation point of view, the solution of Eqs. (44), (45) is not more expensive than for the 'standard' Runge–Kutta method (which is a special case of an ARK with $A = \hat{A}$), except that in case of steady boundary conditions for the continuity equation we cannot employ formulation (25); we always need an additional Poisson solve to make u_{n+1} divergence-free.

In the next sections we will investigate the order conditions and stability properties of such ARK methods, and subsequently we will propose three new classes of methods: one based on Radau quadrature, one based on Lobatto quadrature, and one which is of DIRK type (see Table 4).

4.1. Order conditions

The transformation $\mathcal{E}$ leaves the b-coefficients unchanged, leading to $b = \hat{b}$. It is also desirable that the c-coefficients remain unchanged. Firstly, having $b = \hat{b}$ and $c = \hat{c}$ means that no additional order conditions (so-called coupling conditions) appear for orders less than four [63,64], and for higher orders the number of additional order conditions is much smaller than for the case $c \neq \hat{c}$ and $b \neq \hat{b}$. Secondly, $c = \hat{c}$ is required to evaluate the divergence-free constraint $MU_i = r_1(t_i)$. We use the following theorem to show when $c = \hat{c}$.

Theorem 4.1. *If a Runge–Kutta method is transformed into an energy-conserving method according to (37), then the c-coefficients remain unchanged if the original method satisfies $D(1)$.*

Table 4
Properties of some implicit additive Runge–Kutta methods.

Type	Energy-conserving	Time-reversible	L-stable	Algebraically stable	Order
Gauss	Y	Y	N	Y	$2s$
Radau IIA/B	Y	N	Y	N	$2s-1$
Lobatto IIIC/E	Y	Y	Y	N	$2s-2$
DIRK L/E	Y	Y	Y	N	2

Proof. Let the original method be given by $\mathcal{A}$, and the transformed method by $\widehat{\mathcal{A}} = \mathcal{E}(\mathcal{A})$. Then we have

$$\hat{c}_i = \sum_{j=1}^{s} \hat{a}_{ij} = \frac{1}{2}\sum_{j=1}^{s} a_{ij} + \frac{1}{2}\sum_{j=1}^{s} b_j\left(1 - \frac{a_{ji}}{b_i}\right) = \frac{1}{2}c_i + \frac{1}{2} - \frac{1}{2}\frac{1}{b_i}\sum_{j=1}^{s} b_j a_{ji}, \tag{46}$$

which can be written as

$$\sum_{j=1}^{s} b_j a_{ji} = b_i(1 + c_i - 2\hat{c}_i). \tag{47}$$

Condition $D(1)$ can be written as

$$\sum_{j=1}^{s} b_j a_{ji} = b_i(1 - c_i). \tag{48}$$

Therefore, if a method satisfies both (47) and (48), then $c_i - 2\hat{c}_i = -c_i$, so $\hat{c}_i = c_i$. □

By considering an additive Runge–Kutta method as a partitioned Runge–Kutta method (see [62]), the order of the pair is [46,59]

$$\min(p, 2\xi + 1), \tag{49}$$

where $\xi = \min(\eta, \zeta)$, and η and ζ correspond to method $\mathcal{A}$. It remains to be proven that the order of the additive Runge–Kutta method is not affected when the method is applied to DAEs instead of ODEs. This seems to be an open question, but theoretical [65] and numerical results, e.g. [36], indicate that, like for the standard Runge–Kutta methods, no order reduction occurs for the velocity in case of the incompressible Navier–Stokes equations. We will also confirm this with numerical experiments in Section 7.

4.2. Stability

The algebraic stability properties of this new class of methods are investigated by assessing the energy conservation properties of the method, similar to Eq. (31). Although both $\mathcal{A}$ and $\hat{\mathcal{A}}$ are chosen to be algebraically stable, their combination in an ARK method requires additional conditions to be satisfied. The expression for $\|u_{n+1}\|^2$ reads, after employing Eqs. (44) and (45) with $b = \hat{b}$:

$$\|u_{n+1}\|^2 = \|u_n\|^2 + 2\Delta t\sum_{i=1}^{s} b_i((U_i, PF_i) + (U_i, P\widehat{F}_i)) - \Delta t^2\left(\sum_{i,j=1}^{s} e_{ij}(PF_i, PF_j) + \sum_{i,j=1}^{s}\hat{e}_{ij}(P\widehat{F}_i, P\widehat{F}_j) + 2\sum_{i,j=1}^{s}\tilde{e}_{ij}(PF_i, P\widehat{F}_j)\right), \tag{50}$$

where e_{ij} is given by (32), $\hat{e}_{ij}$ is given by (32) with a replaced by $\hat{a}$, and $\tilde{e}_{ij}$ is given by

$$\tilde{e}_{ij} = b_i\hat{a}_{ij} + b_j a_{ji} - b_i b_j. \tag{51}$$

Since $\hat{\mathcal{A}}$ is energy conserving we have $\widehat{E} = 0$. In Appendix C.1 we proof that as a result the additive Runge–Kutta method cannot be algebraically stable – even though the two constituents that form the method are in itself algebraically stable.

We therefore investigate a weaker (linear) stability concept by applying an additive Runge–Kutta method to the linear scalar test equation [62]

$$\dot{u} = (\lambda + \hat{\lambda})u, \tag{52}$$

where λ is an eigenvalue of the diffusion operator ($\lambda \in \mathbb{R}$, $\lambda \leqslant 0$), and $\hat{\lambda}$ an eigenvalue of the (linearized) convection operator ($\hat{\lambda} \in i\mathbb{R}$). When an additive Runge–Kutta method is applied to (52), the expression for the amplification factor R becomes [63,66]:

$$R(z, \hat{z}) \equiv \frac{u_{n+1}}{u_n} = \frac{\mathrm{Det}(I - zA - \hat{z}\widehat{A} + (z + \hat{z})eb^T)}{\mathrm{Det}(I - zA - \hat{z}\widehat{A})} = \frac{N(z, \hat{z})}{D(z, \hat{z})}, \tag{53}$$

where $z = \lambda\Delta t$, $\hat{z} = \hat{\lambda}\Delta t$, I is the $s \times s$ identity matrix and e is $(1, \ldots, 1)^T$. The requirement for linear stability, which we will loosely call 'A-stability', is

$$|R(z, \hat{z})| \leqslant 1 \quad \text{for } z \leqslant 0. \tag{54}$$

As for algebraic stability, A-stability cannot be concluded from the A-stability of the two methods, and has to investigated for each method separately. On the other hand, the L-stability of $\mathcal{A}$ is inherited by the additive Runge–Kutta method, see Appendix C.2.

4.3. Radau IIA/B pair

The Radau IIA methods are a suitable choice for $\mathcal{A}$, because they give the highest possible order $(2s-1)$ combined with stiff accuracy (and therefore L-stability). For $\hat{\mathcal{A}}$ we then use $\mathcal{E}(\mathcal{A})$, the Radau IIB methods, and their combination results in a method that has many of the required properties (Table 4 and 5) : it is of order $2s-1$, energy-conserving, L-stable, and has $c=\hat{c}$ and $b=\hat{b}$. We note that, instead of using Eq. (49) it is also possible to evaluate the additional order conditions originating from the use of additive methods; for $s=2$ there are no additional order conditions, for $s=3$ there are 16 additional conditions, and they are indeed satisfied by the Radau IIA/B pair.

Evaluating (53) for the new scheme results in

$$R(z,\hat{z})=\frac{12+6\hat{z}+4z+z\hat{z}+\hat{z}^2}{12-8z-6\hat{z}+2z^2+3z\hat{z}+\hat{z}^2}. \tag{55}$$

This amplification factor is shown in Fig. 1(b). A-stability can be proven as follows. Let $\hat{z}=\alpha i$, $z=\beta$, then the requirement for A-stability can be written as

$$|R(\beta,\alpha i)|^2\leqslant 1 \quad \text{for } \beta\leqslant 0, \tag{56}$$

which becomes

$$-\beta(\beta-6)(\beta^2-2\beta+\alpha^2+12)<0. \tag{57}$$

This expression is indeed satisfied for all α and $\beta<0$ and we can conclude that the two-stage Radau IIA/B method is A-stable. For pure convection problems we have $z=0$ and (55) reduces to

$$R(z=0,\hat{z})=\frac{12+6\hat{z}+\hat{z}^2}{12-6\hat{z}+\hat{z}^2}, \tag{58}$$

so $|R(z=0,\hat{z})|=1$, and there is no damping on the imaginary axis, as expected. One recognizes the stability function for Radau IIB which is in fact the same as the stability function for the two-stage Gauss method. Consequently, the Radau IIB method is fourth order accurate for linear equations, instead of third order. For diffusion dominated problems where $\hat{z}\to-\infty$, we find (for fixed $\hat{z}$):

$$\lim_{z\to-\infty} R(z,\hat{z})=0. \tag{59}$$

For the fifth order, 3-stage method, A-stability can be proven in a similar way.

Table 5
Energy-conserving, L-stable, 2-stage, 3rd order Radau IIA/B pair.

$\frac{1}{3}$	$\frac{5}{12}$	$-\frac{1}{12}$
1	$\frac{3}{4}$	$\frac{1}{4}$
	$\frac{3}{4}$	$\frac{1}{4}$

(a) $\mathcal{A}_{\text{RadIIA}}$

$\frac{1}{3}$	$\frac{3}{8}$	$-\frac{1}{24}$
1	$\frac{7}{8}$	$\frac{1}{8}$
	$\frac{3}{4}$	$\frac{1}{4}$

(b) $\mathcal{A}_{\text{RadIIB}}=\mathcal{E}(\mathcal{A}_{\text{RadIIA}})$

Table 6
Energy-conserving, time-reversible, L-stable, 2-stage, 2nd order, Lobatto IIIC/E pair.

0	$\frac{1}{2}$	$-\frac{1}{2}$
1	$\frac{1}{2}$	$\frac{1}{2}$
	$\frac{1}{2}$	$\frac{1}{2}$

(a) $\mathcal{A}_{\text{LobIIIC}}$

0	$\frac{1}{4}$	$-\frac{1}{4}$
1	$\frac{3}{4}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

(b) $\mathcal{A}_{\text{LobIIIE}}=\mathcal{E}(\mathcal{A}_{\text{LobIIIC}})$

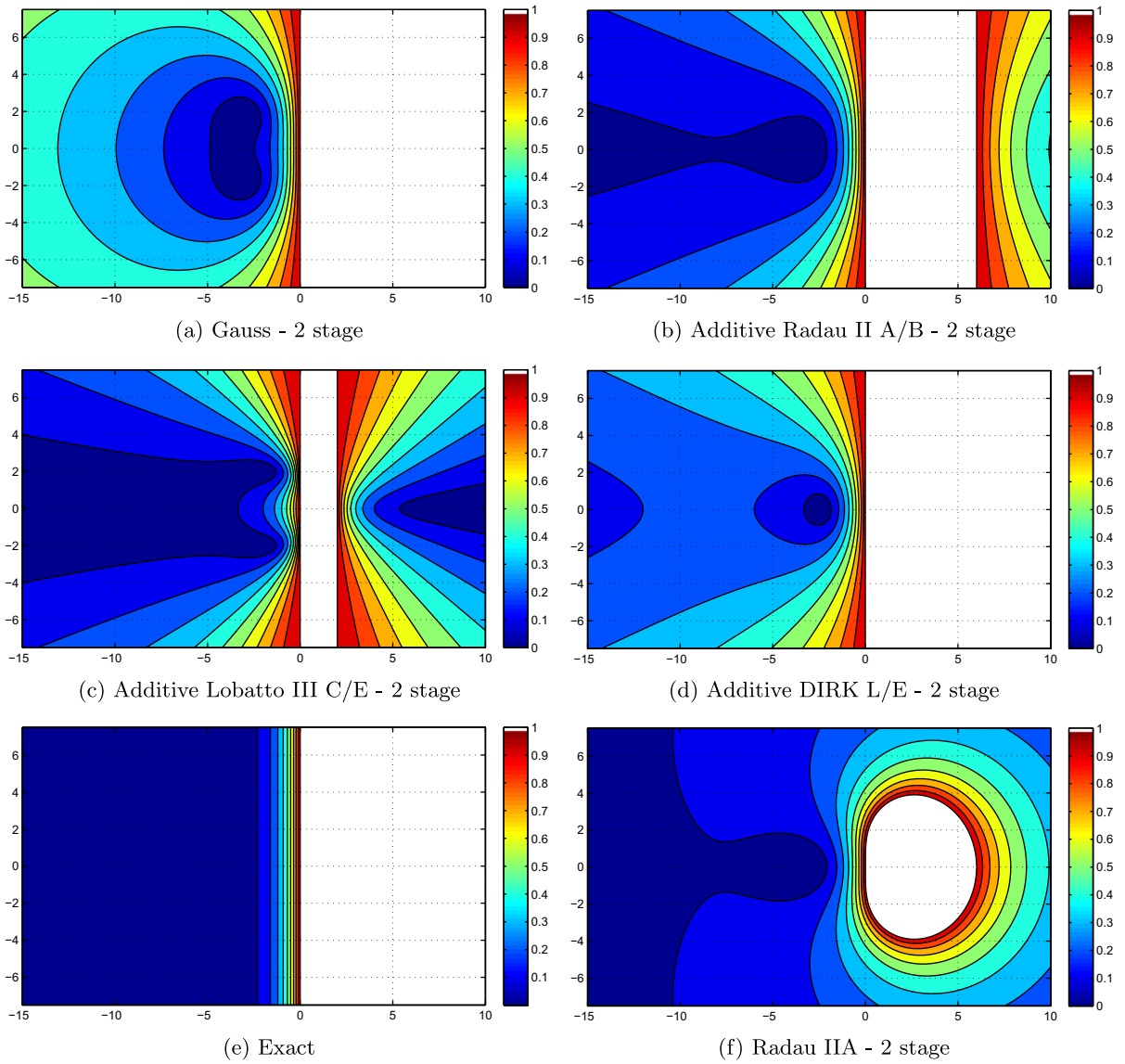


Fig. 1. Stability regions $|R(\beta, \alpha i)| \leqslant 1$.

Table 7
Energy-conserving, time-reversible, L-stable, 2-stage, 2nd order DIRK pair.

$\frac{1}{4}$	$\frac{1}{4}$	0
$\frac{3}{4}$	$\frac{5}{12}$	$\frac{1}{3}$
	$\frac{1}{2}$	$\frac{1}{2}$

(a) $\mathcal{A}_{\text{DIRK L}}$

$\frac{1}{4}$	$\frac{1}{4}$	0
$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

(b) $\mathcal{A}_{\text{DIRK E}}$

4.4. Lobatto IIIC/E pair

Another additive Runge–Kutta method combining stiff accuracy and energy conservation can be constructed by applying the transform $\mathcal{E}$ on the Lobatto IIIC methods, leading to the IIIE methods. These are of lower order $(2s - 2)$ than the Radau pair, but keep the time-reversibility property for inviscid flows. We will investigate numerically if this additional property

outweighs the loss in order compared to the Radau pair. From the one-parameter family of stiffly accurate Lobatto methods that are of order $2s-2$ [58], the Lobatto IIIC method is the only one that satisfies $D(1)$ for $s=2$ (necessary for $c=\hat{c}$ according to Theorem 4.1). The degree of the numerator of the stability function R is at most $s-2$ for the Lobatto IIIC method [46], leading to a very fast decay along the negative real axis of the stability domain, and making them very suitable for stiff problems. Table 6 shows the Butcher tableaux for the 2-stage methods.

A-stability for the 2-stage method follows by calculating the stability function as before:

$$R(z,\hat{z}) = \frac{4+2\hat{z}+\hat{z}^2+z\hat{z}}{4-4z-2\hat{z}+2z^2+3z\hat{z}+\hat{z}^2}, \tag{60}$$

shown in Fig. 1(c). $|R(\beta,\alpha i)|^2 \leqslant 1$ leads to

$$-\beta(-2+\beta)(\beta^2-2\beta+\alpha^2+4) \leqslant 0, \tag{61}$$

which is indeed satisfied for $\beta \leqslant 0$. For the 3-stage method, A-stability follows in a similar fashion.

We note that, apart from the Lobatto IIIC/E method, other combinations of Lobatto methods can lead to interesting methods. For example, the Lobatto IIIA/B pair has the property that $\widetilde{E}=0$, but is not energy-conserving in the inviscid limit. We refer to [67] for an extensive discussion on additive Lobatto methods.

4.5. DIRK pair

We mentioned before that the maximum order for an energy-conserving DIRK method is two, and for a two-stage method there is one free parameter c_1, see Eq. (35). We combine this method with a two stage, second order L-stable DIRK method to arrive at an additive Runge–Kutta method of DIRK type which combines energy conservation, time reversibility and L-stability. Again, we look for a method with $c=\hat{c}$, which implies $b=\hat{b}$ due to the second order coupling condition. When we furthermore adhere to convention (18), the condition for L-stability (namely a singular Q-matrix), leads to the following family of L-stable DIRK methods:

$$\begin{array}{c|cc} c_1 & c_1 & 0 \\ c_1+\frac{1}{2} & \frac{1}{2}\frac{c_1(2c_1-3)}{c_1-1} & \frac{1}{2}\frac{2c_1-1}{c_1-1} \\ \hline & 2c_1 & 1-2c_1 \end{array} \tag{62}$$

In contrast to the previous tableaux for the diffusive terms, $\mathcal{A}$ is not algebraically stable, because E is not nonnegative definite. It is not possible to choose c_1 such that both the convective and diffusive tableau are of SDIRK type. In the sequel we take $c_1=\frac{1}{4}$, so that for pure convection problems the method *is* of SDIRK type; the tableaux are then given in Table 7. Note that here $\hat{\mathcal{A}} \neq \mathcal{E}(\mathcal{A})$. The amplification factor is given by

$$R(z,\hat{z}) = \frac{48+20z+24\hat{z}+3z\hat{z}+3\hat{z}^2}{(-4+z+\hat{z})(-12+4z+3\hat{z})}, \tag{63}$$

and is shown in Fig. 1(d). The method is A-stable, because

$$-\beta(-12+\beta)(\beta^2-2\beta+24+\alpha^2) \leqslant 0 \tag{64}$$

is satisfied for $\beta \leqslant 0$.

4.6. Summary

To summarize, we have a hierarchy of methods, listed in Table 4. The Gauss methods have highest possible order, $p=2s$, but lack L-stability. Sacrificing one order, $p=2s-1$, leads to the additive Radau IIA/B methods, which have both energy conservation and L-stability. Sacrificing another order, $p=2s-2$, leads to the additive Lobatto IIIC/E methods, which have energy conservation, time reversibility and L-stability. Dropping the requirement on algebraic stability of the diffusive tableau leads to an additive DIRK L/E method, also energy-conserving, time-reversible and L-stable. These methods will be numerically tested in Section 7, together with the Lobatto IIIA (Crank–Nicolson) method, because this is a widely used method and an example of a method which is time-reversible but not energy-conserving.

All these methods are two-stage methods, with the exception of the 1-stage Gauss method. The stability domains of the methods, $|R(\beta,\alpha i)| \leqslant 1$, are shown in Fig. 1(a)–(d). The stability domain of the 2-stage Lobatto IIIA method (Crank–Nicolson) coincides with the Gauss 1-stage method (implicit midpoint) and can be found in any standard textbook, e.g. [46]. One can observe that the proposed methods satisfy both $|R|=1$ on the imaginary axis and $|R|=0$ for $\beta \to -\infty$. However, when compared with the exact stability function (Fig. 1(e)), or with a stiffly accurate method such as Radau IIA (Fig. 1(f)), the proposed methods possess a much slower decay of R to zero along the negative real axis.

We note that all additive methods considered here have a region $|R|<1$ in the right half of the complex plane. For example, Eq. (57) is also satisfied for $\beta \geqslant 6$, and Eq. (64) for $\beta > 12$. This is an undesirable property when one wants to integrate physically 'unstable' phenomena, such as the transition from laminar to turbulent flow. However, compared to the 'original'

stiffly accurate/L-stable methods, such as the Radau IIA method depicted in Fig. 1(f), the behavior is much better. From the methods considered here the Gauss methods are the only ones that have $|R| > 1$ in the entire right half of the complex plane.

5. The accuracy of the pressure

The velocity at the new time step follows from the sequence (22)–(24). The 'pressure' ϕ_{n+1} necessary to obtain a divergence-free u_{n+1} is only a first order approximation to p_{n+1} [45]. A higher order accurate p_{n+1} can be computed by solving an additional Poisson equation, Eq. (12), at t_{n+1}:

$$Lp_{n+1} = MF_{n+1} - \dot{r}_1(t_{n+1}). \tag{65}$$

The resulting pressure has the same temporal accuracy as the velocity field u_{n+1}. If $\dot{r}_1(t)$ is not available or not well-defined, or to avoid the additional Poisson equation, one can construct higher order approximations by making linear combinations of the stage values ϕ_i. For the Runge–Kutta methods based on Gauss, Radau and Lobatto quadrature we apply the techniques mentioned in [46,54] to the Navier–Stokes equations, and subsequently we extend and unify these techniques in a new approach for additive Runge–Kutta methods proposed in [45]. In contrast to what is reported for explicit methods in that work, for implicit methods and steady boundary conditions for the continuity equation it is not possible to obtain a pressure of the same order as the velocity field without the additional Poisson solve mentioned above.

5.1. Radau IIA, Lobatto IIIC

If A is non-singular one can write [46]:

$$p_{n+1} = p_n + \sum_{i,j=1}^{s} b_i \omega_{ij} (\psi_j - p_n) = R(\infty) p_n + \sum_{i,j=1}^{s} b_i \omega_{ij} \psi_j, \tag{66}$$

where $\psi = A^{-1} C^p \phi$ and ω_{ij} are the entries of A^{-1}. For the Radau IIA and Lobatto IIIC methods this gives an approximation of order s and $s-1$, respectively, corresponding to their stage orders. They both feature stiff accuracy and a non-singular A, leading to $R(\infty) = 0$, so p_{n+1} is not depending on p_n and the global error equals the local error. Furthermore

$$b^T = (0 \quad \dots \quad 0 \quad 1) A, \tag{67}$$

and we have

$$p_{n+1} = \psi_s = \sum_{i=1}^{s} \omega_{si} c_i \phi_i. \tag{68}$$

In case of Lobatto IIIC $c_i \phi_i$ is replaced by ρ_i since $c_1 = 0$.

5.2. Gauss

For the Gauss methods $|R(\infty)| = 1$ and (66) gives an approximation of order $s-1$ (s odd) or $s-2$ (s even), which is unsatisfactory, especially for the 1- and 2-stage methods under consideration. An alternative is to construct a polynomial $v(t)$ that interpolates the ψ_i values. Evaluating $v(t)$ at t_{n+1} then gives an s-th order approximation to the pressure [46,68]:

$$p_{n+1} \approx v(t_{n+1}) = \sum_{i=1}^{s} \psi_i \prod_{j=1, j\neq i}^{s} \frac{1 - c_j}{c_i - c_j}. \tag{69}$$

With this expression the pressure p_{n+1} is independent of p_n, like for L-stable methods, and the global error is therefore the same as the local error.

5.3. Lobatto IIIA

Although the Lobatto IIIA methods are collocation methods, they have $c_1 = 0$, so the approach for the Gauss methods cannot be applied. Furthermore, A is singular, hence the approach for the Radau IIA and Lobatto IIIC methods also fails. However, since the method is stiffly accurate and satisfies $c_1 = 0$, we can take $p_{n+1} = \psi_s$. By defining $\tilde{A} = A_{2:s,2:s}$ (which is non-singular) and $\tilde{\omega}_{ij} = (\tilde{A}^{-1})_{ij}$ the entries of its inverse, ψ can be written in terms of ρ as

$$\psi_i = \sum_{j=2}^{s} \tilde{\omega}_{ij} (\rho_j - a_{j1} \psi_1), \tag{70}$$

where $\psi_1 = p_n$. In this case p_{n+1} depends on p_n and the order is s (s even) or $s-1$ (s odd) [54].

5.4. Additive Runge–Kutta methods: a unified approach

Applying any of the approaches outlined above to an additive Runge–Kutta method leads to the central question 'which matrix to choose for A^{-1}?'. Here we circumvent this question by using the approach outlined for explicit methods in Sanderse and Koren [45]. This method is based on the reconstruction of the point value p_{n+1} from the *average* pressure values ϕ_i. If a Runge–Kutta method has at least η stages that satisfy $C_i(\eta)$, then it is possible to obtain an $\mathcal{O}(\Delta t^\eta)$ accurate approximation to $p(t_{n+1})$ by combining the ϕ_i values. For the implicit methods under consideration we have $C(\eta)$ (holding for all stages), with $\eta = s$ or $s-1$ (except the DIRK methods), so such a construction is always possible. p_{n+1} then follows as [45]:

$$p_{n+1} = \sum_{k\in K} \phi_k c_k \Delta t \dot{\ell}_k(t_{n+1}), \tag{71}$$

where

$$\ell_k(t) = \prod_{j\in \widetilde{K}, j\neq k} \frac{t-t_j}{t_k - t_j}, \tag{72}$$

and $\widetilde{K} = \{0, K\}$. The set K consists of (at least) η stages, which have to be chosen such that the c_k are distinct and nonzero.

Eq. (71) appears to be equivalent to the formulations above for the Radau and Gauss methods:

- The Gauss methods satisfy $C(s)$, so $\eta = s$ and $K = \{1, 2, \ldots, s\}$. First we evaluate our approach, Eq. (71), for $s = 2$:

$$p_{n+1} = \phi_1 \frac{2-c_2}{c_1 - c_2} + \phi_2 \frac{c_2 - 2}{c_1 - c_2}. \tag{73}$$

Secondly, we rewrite (69) by interpreting $C(s)$, see Eq. (27), as an equation for the a-coefficients in terms of the c-coefficients. $\psi = A^{-1}C^p\phi$ can then be completely expressed in terms of the c-coefficients, and when evaluating the resulting expression, Eq. (73) is obtained. This also holds for $s = 3$, and we conjecture that our approach, Eq. (71), is equivalent to Eq. (69) for any s.

- For the Radau methods we proceed in a similar way. We evaluate (73) for $c_s = 1$ and compare with (68), evaluated with $C(s)$ and $c_s = 1$. Both formulations are again identical for $s = 2$ and $s = 3$.

For the Lobatto IIIC methods our approach (71) differs from (68). Since $C(s-1)$ and $c_1 = 0$ are satisfied, we take $K = \{2, \ldots, s\}$, and (71) boils down to the following first order approximation:

$$p_{n+1} = \frac{\rho_2}{c_2}, \tag{74}$$

while (68) gives

$$p_{n+1} = \omega_{21}\rho_1 + \omega_{22}\rho_2, \tag{75}$$

which is also of first order. It turns out, somewhat surprisingly, that $\omega_{22} = \frac{1}{c_2}$. The presence of the ρ_1 term in (75) is therefore not necessary to obtain the required order of accuracy. For $s = 3$ a similar phenomenon occurs.

Lastly, we compare the pressure update for the Lobatto IIIA method with our approach. Eq. (70) reads, for $s = 2$:

$$p_{n+1} = \frac{1}{a_{22}}(\rho_2 - a_{21}p_n), \tag{76}$$

which is a second-order accurate approximation. This is the only method where p_n appears in the formulation, and the only method for which our approach gives a lower order: first order instead of second order for $s = 2$.

5.5. Extension to ARK methods

With our unified approach, (71) can be directly applied to the additive Runge–Kutta methods, because it only depends on the c-coefficients, which are the same for the convective and diffusive tableaux. Table 8 summarizes the different methods. Although the Radau IIA/B method satisfies only $C(s-1)$ instead of $C(s)$, we still take the same approach as for Radau IIA, so

Table 8
Computation of the pressure for implicit (additive) Runge–Kutta methods.

Type	$s = 2$	General s	Order
Gauss, Radau IIA	(73)	(71)	s
Radau IIA/B	(73)	(71)	$s-1$
Lobatto IIIC/IIIE	(74)	(71)	$s-1$
Lobatto IIIA	(76)	(70)	$s/s-1$ (even/odd)
DIRK	(77)	–	1

that the Radau IIA method is simply a special case of the Radau IIA/B method. For the Lobatto IIIC/E pair we can directly use the formulation for Lobatto IIIC. The DIRK pair satisfies $C(1)$ and the pressure is therefore limited to first order; we simply take

$$p_{n+1} = \phi_2. \tag{77}$$

6. Implementation issues

6.1. System of equations

Eq. (15) represents a full, implicit system of nonlinear equations. Introducing the iteration counter k we write for the stage values

$$U^{k+1} = U^k + \Delta U^k. \tag{78}$$

One iteration of Newton's method leads to the following linear system of equations:

$$\begin{bmatrix} T^k & \Delta t C^p G \\ M & 0 \end{bmatrix} \begin{bmatrix} \Delta U^k \\ \Delta \phi^k \end{bmatrix} = d^k \quad \text{where } d^k = \begin{bmatrix} d_u^k \\ d_\phi^k \end{bmatrix} = \begin{bmatrix} u_n - U^k + \Delta t A F^k - \Delta t C^p G \phi^k \\ -MU^k + r_1 \end{bmatrix}, \tag{79}$$

and

$$T^k = I - \Delta t A J^k, \quad J^k = \frac{\partial F}{\partial U}(U^k). \tag{80}$$

J represents the Jacobian of the system. With full Newton the exact Jacobian is evaluated at each iteration step. We stress again that by using ϕ instead of ψ, M and G are block-diagonal matrices ($I_s \otimes M$, $I_s \otimes G$). At each time step the non-linear system is solved until an absolute tolerance δ_a or relative tolerance δ_r on the residual is satisfied:

$$\|d^k\|_\infty \leqslant \delta_a \quad \text{or} \quad \|d^k\|_\infty \leqslant \delta_r \|d^1\|_\infty. \tag{81}$$

The presence of the convective terms in the Jacobian leads to an asymmetric system, which moreover changes every iteration, making the solution of the system expensive. A possible method to solve the system is the Newton–Krylov method proposed by Pereira et al. [69]; a general overview of methods is given in [70]. In the results presented in this work we have used small test problems for which the use of a direct solver was most efficient. However, to make the approach suitable for large-scale computations, we will give some directions for improvement.

One can use simplified Newton instead of full Newton: take the Jacobian at the start of each time step and keep it constant during the iterative process. If a direct method is used, one can even compute the LU-decomposition of $I - \Delta t A J(U^0)$ and use it for all iterations. The convergence of the residual, given a sufficiently accurate starting value, is linear instead of quadratic. A starting value can be obtained by fitting a polynomial of degree $s+1$ through the values of U_i of the previous time step and then extrapolate this to the stages of the new time step [46]. This provides in general an $\mathcal{O}(\Delta t^n)$ approximation to U^k.

6.2. Splitting

Another technique which can significantly reduce computational costs is splitting the solution of velocity and pressure. Such a splitting can be seen as a block LU-decomposition of matrix (79) [71]:

$$\begin{bmatrix} T^k & G \\ M & 0 \end{bmatrix} = \begin{bmatrix} T^k & 0 \\ M & -M(T^k)^{-1}G \end{bmatrix} \begin{bmatrix} I & (T^k)^{-1}G \\ 0 & I \end{bmatrix}, \tag{82}$$

so one iteration of the nonlinear method is equivalent to the following sequence:

$$T^k \Delta \widehat{U}^k = d_u^k, \tag{83}$$

$$M(T^k)^{-1} G \Delta \phi^k = M \Delta \widehat{U}^k - d_\phi^k, \tag{84}$$

$$\Delta U^k = \Delta \widehat{U}^k - (T^k)^{-1} G \Delta \phi^k. \tag{85}$$

This factorization does not introduce a splitting error yet. Although the large original matrix is not present in the formulation, the explicit computation of $(T^k)^{-1}$, necessary in the second equation (note that M and G are not square), makes the approach unattractive in practice. When $(T^k)^{-1}$ in (84) and (85) is replaced by an approximation $(\widehat{T}^k)^{-1}$ one obtains an approximate factorization as proposed by Perot [71]. The simplest choice is to take the first order approximation $T^k = I$, which leads to a scheme similar to fractional step or projection methods. Higher order methods follow by more accurate approximations of $(T^k)^{-1}$, but result in a matrix $M(T^k)^{-1}G$ which is not a Laplacian anymore. In any case, a splitting error

is introduced, which destroys the energy-conserving property of the method (if present). This is due to the fact that the intermediate velocity field $\widehat{U}$ does not satisfy the divergence constraint and therefore contributes to the kinetic energy. Furthermore, the unconditional stability of methods associated with the energy conservation property is lost (although a weaker type of stability can still be proven [72]).

To retrieve the original conservation and stability properties we can proceed as follows. Taking $(T^k)^{-1} = I$ one can repeatedly solve (83)–(85) until (81) is satisfied – the iterative method is now used to remove both linearization and splitting errors. From our experience it is most efficient to solve the linear problem (83) at each iteration k and to not introduce a subiteration to solve the nonlinear system at each k. A slightly better approximation to $(T^k)^{-1}$ is to take $(\mathrm{diag}(T^k))^{-1}$ [34], but this only leads to a positive definite matrix $M(T^k)^{-1}G$ if the time step is small enough. In any case, the quadratic or linear convergence of 'full' or 'simplified' Newton is lost.

A completely different approach is the exact fractional step method developed in Chang et al. [73] for staggered mesh methods, by applying a discrete rotation operator to the system of equations. This leads to a system of size approximately $N_p \times N_p$ (2D) or $N_u \times N_u$ (3D). Since there is no splitting error involved, the same number of iterations as for the original system is required. The matrix to be solved has similar properties as matrix T^k, so the same (iterative) technique normally applied to that matrix can be used. The solution of the saddle-point system is avoided. For future work on large-scale problems this will be the method of choice.

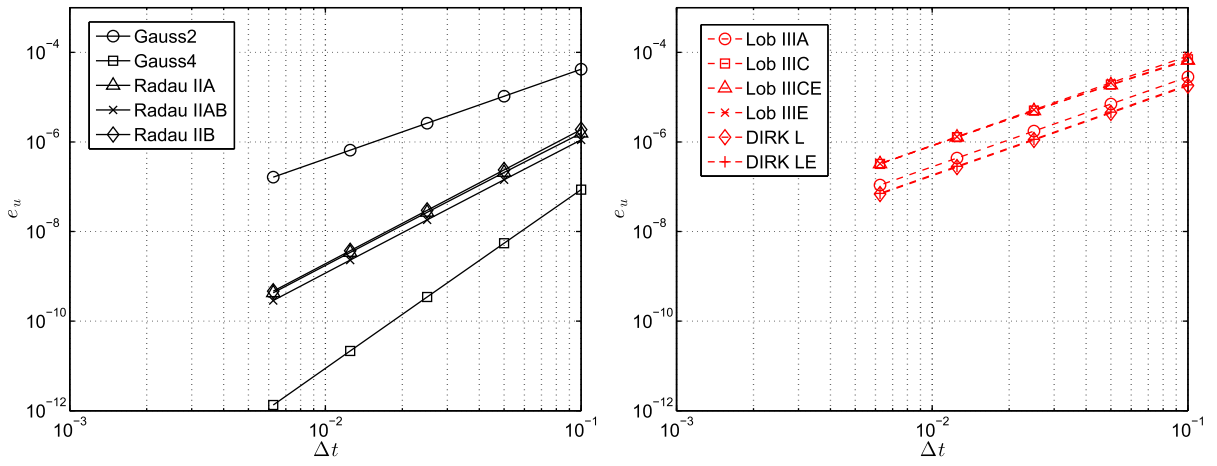
6.3. Linearization

The energy-conservation property (33) has been derived for Runge–Kutta methods assuming the convective terms are given by $C(U_i, U_i)$. However, property (8) shows that $(C(c_i, U_i), U_i) = 0$ holds independent of the time level of c as long as $Mc = 0$ (see e.g. [39,7,74]): the time level of the *convecting* quantity is not important for the conservation of the norm of the *convected* quantity. This means that instead of taking $c_i = U_i$ one can approximate c_i by using previous time levels, and then only a *linear* system has to be solved in order to achieve conservation of energy. For example, for the 1-stage Gauss method, a second order approximation is $\tilde{c}_1 = \frac{3}{2}c_n - \frac{1}{2}c_{n-1}$. The extension to higher order methods is straightforward.

If linearization is combined with the splitting of Chang et al. [73] no iteration is required. Although the total work involved at each time step is obviously higher than the solution of a simple Poisson equation (as done in fully explicit methods), the reduction in number of necessary iterations [73] and increase in allowable time step can make this approach competitive with explicit methods.

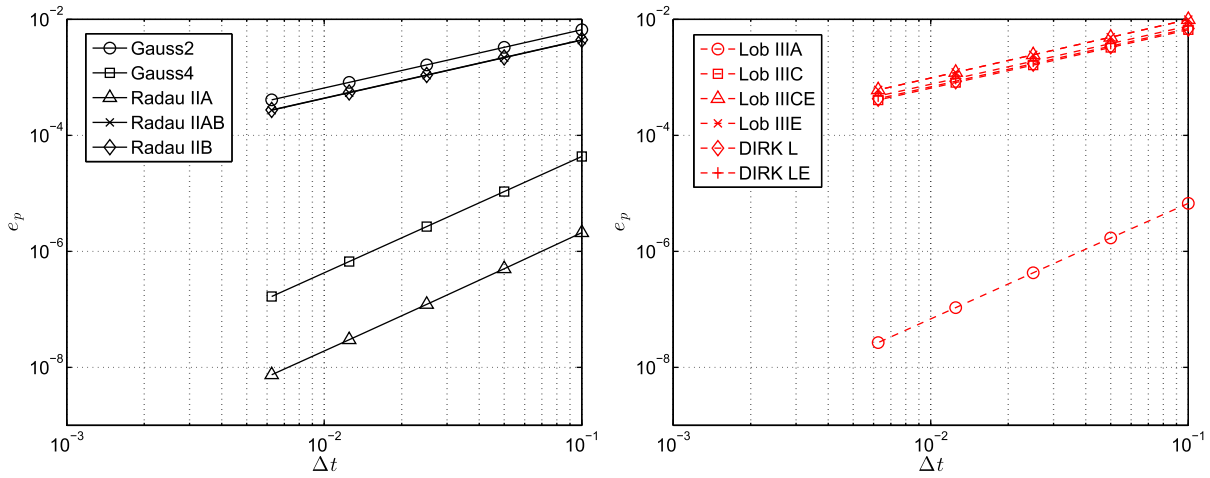
7. Results

A number of test cases are considered to evaluate the practical performance of the (additive) Runge–Kutta methods presented in this paper, which were summarized in Section 4.6. It is interesting to compare the additive methods with the two methods by which they are formed (such as Radau IIA and IIB in case of IIA/B). In the last test we leave out the Lobatto IIIC/E method, because it will turn out not to be competitive. We consider 1-stage (Gauss 2) and 2-stage methods (all others). We have also numerically verified the orders of accuracy of the 3-stage variants of these methods. These will not be presented here, because the 2-stage methods are a better compromise between order of accuracy and computational cost. Note that we leave out the DIRK E method, because it is simply the Gauss 2 method but then applied to a time step twice as large.



(a) Gauss, Radau (b) Lobatto, DIRK

Fig. 2. Error in velocity as function of time step.



(a) Gauss, Radau (b) Lobatto, DIRK

Fig. 3. Error in pressure as function of time step.

7.1. Order of accuracy study: Taylor–Green vortex

We use the Taylor–Green vortex to numerically verify the order of accuracy of the developed methods. This is an exact solution to the Navier–Stokes equations, given by:

$$u(x,y,t) = -\sin(\pi x)\cos(\pi y)\, e^{-2\pi^2 t/\mathrm{Re}}, \tag{86}$$

$$v(x,y,t) = \cos(\pi x)\sin(\pi y)\, e^{-2\pi^2 t/\mathrm{Re}}, \tag{87}$$

$$p(x,y,t) = \frac{1}{4}(\cos(2\pi x) + \cos(2\pi y)) e^{-4\pi^2 t/\mathrm{Re}}. \tag{88}$$

The domain on which we define the solution is the square $[\frac{1}{4}, 2\frac{1}{4}] \times [\frac{1}{4}, 2\frac{1}{4}]$ with time-dependent Dirichlet boundary conditions. The initial condition at $t = 0$ for the velocity field is given by the exact solution (86) and (87), and the initial condition for the pressure is obtained by solving (14). The Reynolds number is $\mathrm{Re} = 1/\nu = 100$, the number of volumes is 20×20 and we integrate from $t = 0$ to $t = 1$. For details on the spatial accuracy of the method and results obtained with explicit Runge–Kutta methods we refer to [75,45]. The nonlinear system of equations is solved with full Newton, with a trivial initial guess and $\delta_a = \delta_r = 10^{-14}$. For the largest time step, $\Delta t = 1/10$, this requires 2 or 3 nonlinear iterations (depending on the method); for the smallest time steps 1 iteration suffices.

Figs. 2 and 3 show the temporal error in the u-velocity and pressure for all methods:

$$e_u = \|u - u_{\Delta t \to 0}\|_\infty, \quad e_p = \|p - p_{\Delta t \to 0}\|_\infty. \tag{89}$$

The error in the v-velocity component and the error measured in the L_2-norm give almost identical results and are not shown here. The methods are split over two figures to keep the presentation as clear as possible. All methods achieve their theoretical order of accuracy. The third order methods of the Radau family are almost indistinguishable, and similarly the second order methods of Lobatto and DIRK type are close. The results for the pressure error show that only Gauss 4, Radau IIA and Lobatto IIIA reach second order convergence, as predicted by Table 8. The Lobatto IIIA stands out since it is the only method that achieves the same order in velocity and pressure, and its pressure error is much lower than Gauss 2. Although this corresponds to the theoretical results, it is still surprising considering that both methods are so closely related: implicit midpoint (Gauss 2) and trapezoidal rule (Lobatto IIIA), which are identical for linear equations. We remark that in all cases a pressure with the same temporal order as the velocity can be obtained at the expense of solving the additional Poisson equation (65).

7.2. Energy conservation and time reversibility: inviscid shear layer roll-up

7.2.1. Introduction

In order to study the energy conservation and time reversibility properties of the different Runge–Kutta methods we simulate the roll-up of a shear layer governed by the inviscid incompressible Navier–Stokes equations, see e.g. [76,26,27]. The domain is $[0, 2\pi] \times [0, 2\pi]$ with periodic boundary conditions. The initial condition is

$$u = \begin{cases} \tanh\left(\frac{y-\pi/2}{\delta}\right), & y \leqslant \pi, \\ \tanh\left(\frac{3\pi/2-y}{\delta}\right), & y > \pi, \end{cases} \qquad v = \varepsilon \sin(x), \tag{90}$$

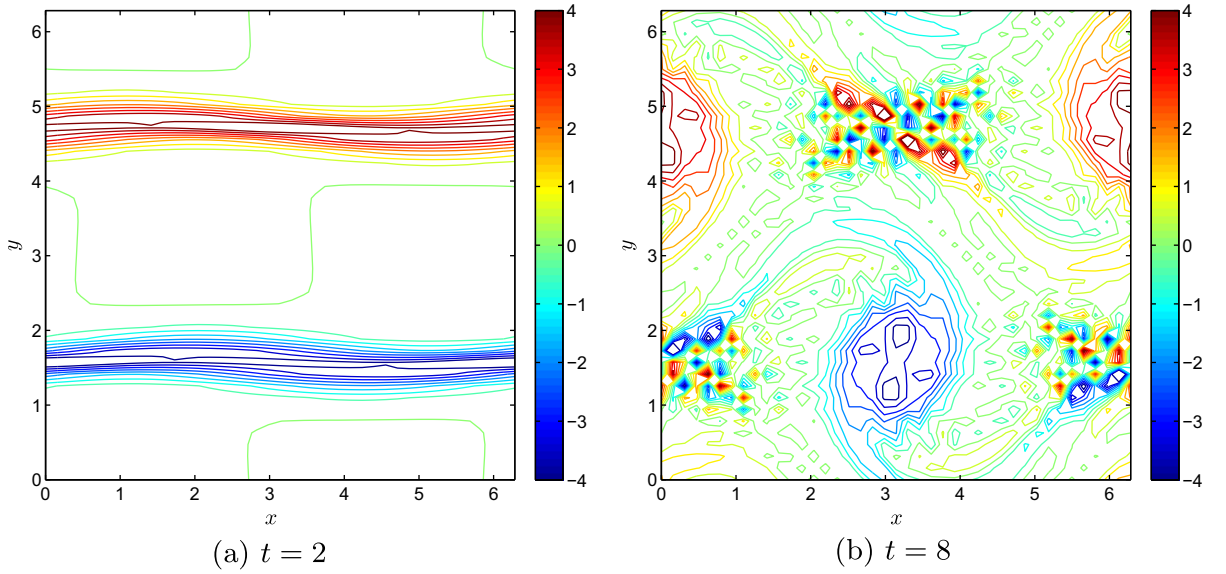


Fig. 4. Vorticity field of shear layer roll-up at different time instances, computed with a uniform grid with 40^2 volumes. Iso-contours $-4 \ldots 4$ with steps 0.5.

with $\delta = \pi/15$, $\varepsilon = 0.05$ taken from [26]. Due to the perturbation of the v-velocity the shear layer starts to roll-up. Impressions of the flow field at $t = 2$ and $t = 8$ are given in Fig. 4, which have been computed with 40^2 volumes and $\Delta t = 10^{-3}$. For qualitative flow field pictures at finer meshes we refer to [26]. Significant noise ('wiggles') can be observed in the solution at $t = 8$, which is due to dispersive and aliasing errors and the absence of viscosity to damp the smallest scales. The associated 'bursts' in vorticity lead to an increase in the enstrophy. However, in contrast to what is mentioned in Brown and Minion [76], such oscillations do *not* lead to 'catastrophic blowup' with the energy-conserving time-integration methods under investigation (in fact, they *cannot* blow up due to their unconditional stability properties). Time-reversibility errors will be investigated by reverting the simulation at $t = 2$ or $t = 8$: we change Δt to $-\Delta t$ and march back to $t = 0$. Since we solve the inviscid Navier–Stokes equations, the additive methods reduce to their energy-conserving part: Radau IIA/B is Radau IIB, Lobatto IIIC/E is IIIE and DIRK L/E is DIRK E.

We define the following error measures. The error in the energy of the flow at any time instant t is defined as

$$e_k(t) = \frac{K(t) - K(t=0)}{K(t=0)}, \tag{91}$$

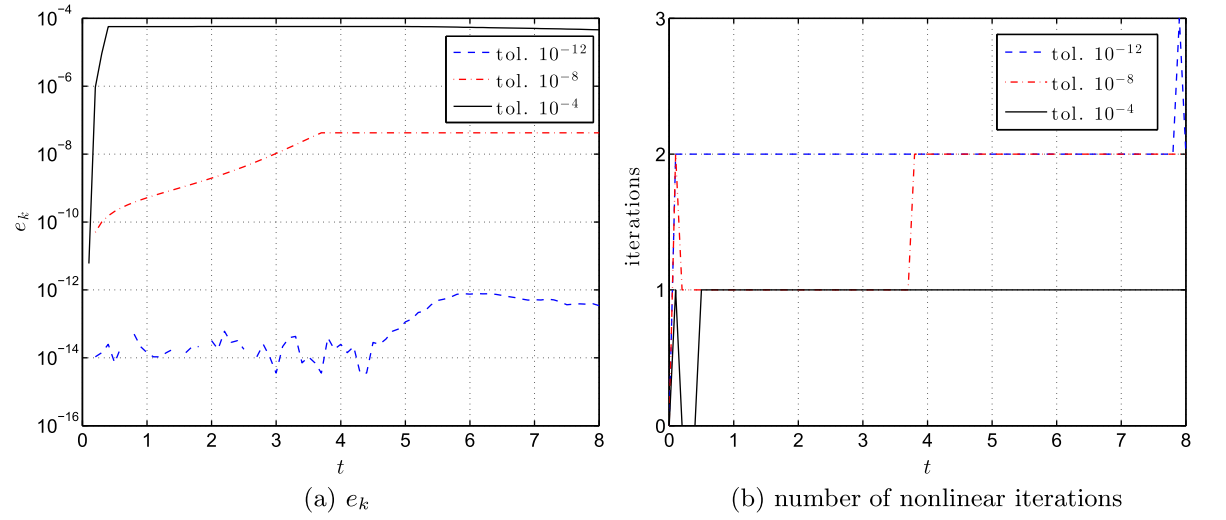


Fig. 5. Influence of solver tolerance, Gauss 2, $\Delta t = 10^{-1}$.

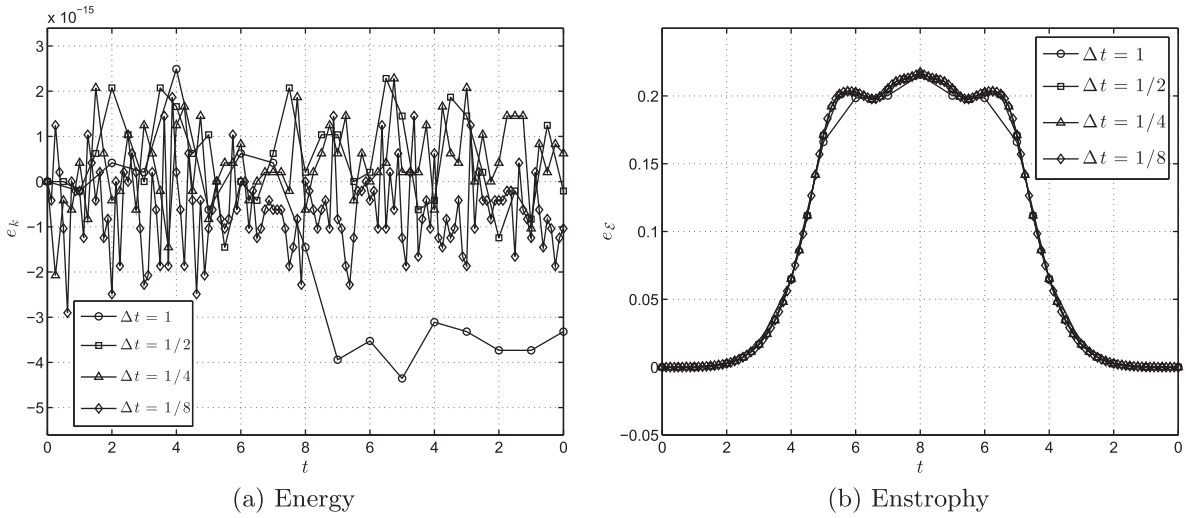


Fig. 6. Energy and enstrophy errors for Gauss 4 applied to shear-layer problem.

where K is the total discrete energy,

$$K = \frac{1}{2}\sum_{ij}\Omega^u_{ij}u^2_{ij} + \frac{1}{2}\sum_{ij}\Omega^v_{ij}v^2_{ij}. \tag{92}$$

Ω^u and Ω^v are the sizes of the u- and v-centered finite volumes; $\sum_{ij}\Omega^u_{ij} = \sum_{ij}\Omega^v_{ij} = (2\pi)^2$. To express the error in time reversibility of a method we compare the velocity fields at the start and the end of the simulation:

$$e_t = \|V(t_{\text{end}}) - V(t=0)\|_\infty. \tag{93}$$

Finally, it is common for this test case to study the total enstrophy of the flow,

$$\mathcal{E} = \frac{1}{2}\sum_{ij}\Omega^\omega_{ij}\omega^2_{ij}, \tag{94}$$

with Ω^ω the sizes of the vorticity-centered finite volumes. The enstrophy is a quadratic quantity which is conserved by the continuous inviscid, incompressible, two-dimensional Navier–Stokes equations, but not by the spatial discretization that we employ [18]. The corresponding error,

$$e_\varepsilon(t) = \frac{\mathcal{E}(t) - \mathcal{E}(t=0)}{\mathcal{E}(t=0)}, \tag{95}$$

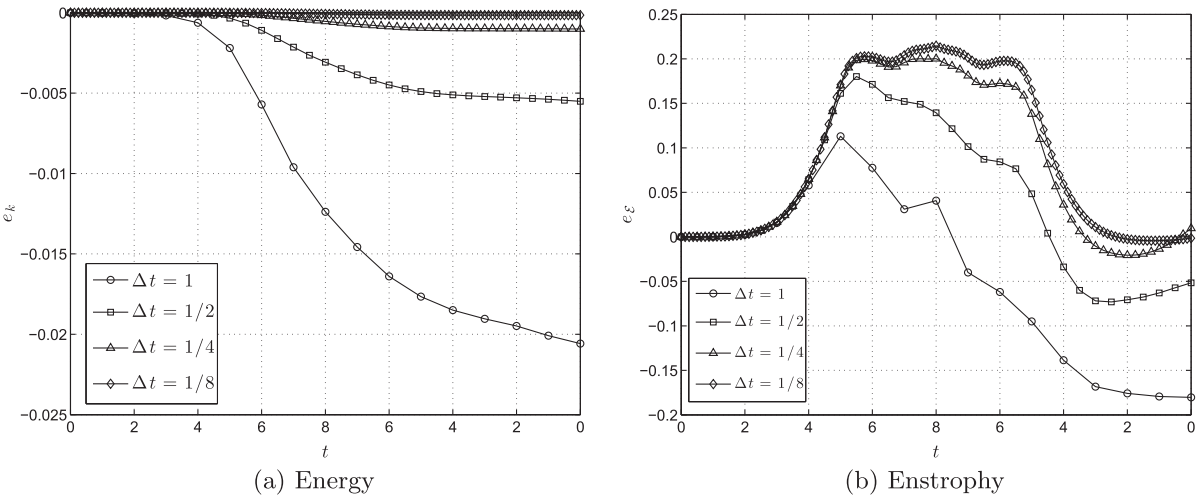
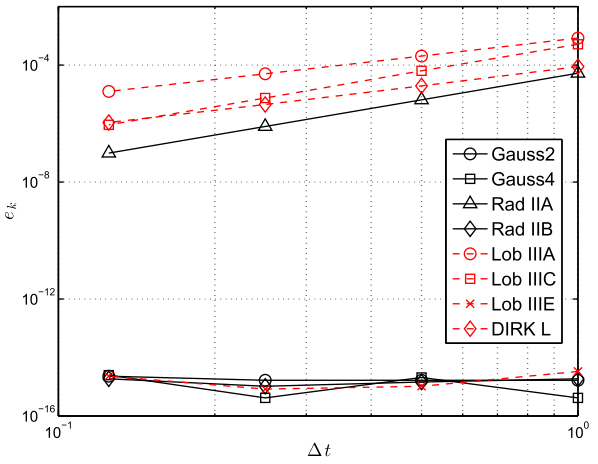
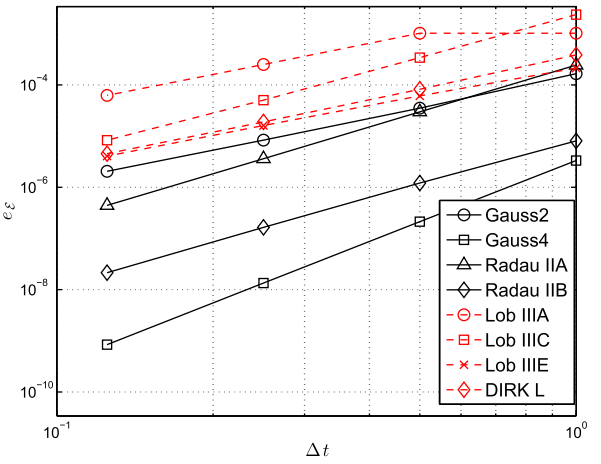


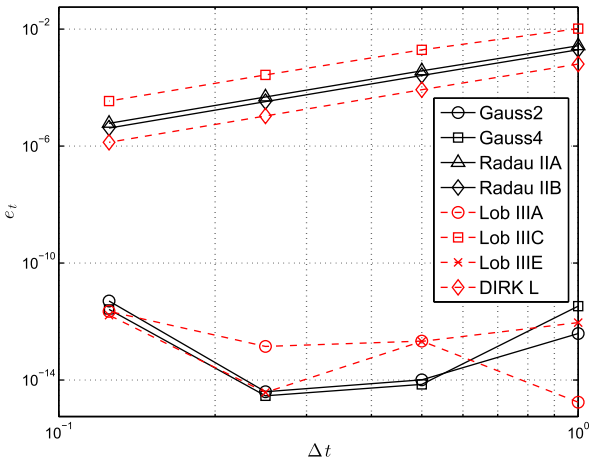
Fig. 7. Energy and enstrophy errors for Radau IIA applied to shear-layer problem.



(a) Energy



(b) Enstrophy



(c) Time-reversibility

Fig. 8. Energy, enstrophy and time-reversibility errors at $t = 2$.

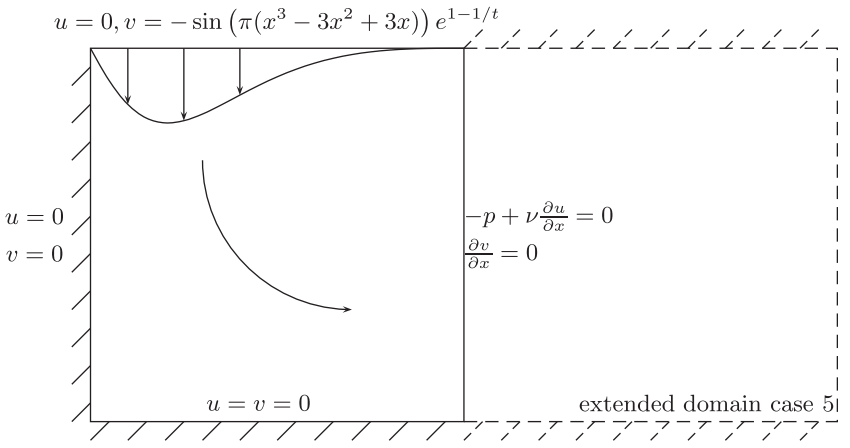


Fig. 9. Corner flow.

can be used to investigate temporal errors of the methods that have no energy and time-reversibility errors. It gives an indication of how well the flow field is resolved. The enstrophy error defined by (95) consists of both spatial and temporal errors. When performing order studies this spatial error is removed by subtracting a simulation at a very fine time step.

All simulations are performed with full Newton and $\delta_a = 10^{-14}$. We do not perform the additional pressure solve (65), but noticed that it helps to obtain a better initial guess for the next time step. However, extrapolating the velocities from the previous time step as prescribed in Section 6 leads to more iterations of the Newton process if $t > 4$, especially for higher order methods ($p > 2$). This is attributed to the under-resolution of the simulation: the flow field becomes noisy when the shear layer starts to roll up (see Fig. 4(b)). For well-resolved simulations this issue will not occur, but in any case it can be solved by using the trivial initial guess. In order to investigate the sensitivity of energy conservation to the tolerance of the nonlinear method we performed a simulation with different tolerance levels, $\delta_a = 10^{-12}, 10^{-8}$ and 10^{-4}. In Fig. 5 the behavior of the energy error and the number of iterations are shown for the case of Gauss 2 with $t = 10^{-1}$. The fact that the flow field is more 'difficult' after $t = 4$ manifests itself in the number of iterations (for tol $= 10^{-8}$) or in an increase in energy error (tol $= 10^{-12}$). After an initial increase the error stays approximately constant and remains on the order of the specified tolerance. This is important, since it shows that the energy conserving methods are robust with respect to convergence of the nonlinear iteration, and keep their stability properties for larger tolerances. For smaller time steps or higher order methods the error levels are lower.

7.2.2. Comparison

To show the beneficial effect of energy conservation we compare the behavior of e_k and e_ε for the Gauss 4 method (Fig. 6(a) and (b)) and the Radau IIA method (Fig. 7(a) and (b)). Time reversal occurs at $t = 8$. The Gauss methods behave as expected: energy is conserved until machine precision and the simulation is exactly time-reversible. Enstrophy is not conserved by the spatial discretization, as can be seen in Fig. 6(b). The enstrophy error is dominated by the spatial errors, even for large time steps; at $\Delta t = 1$ the CFL number is almost 10. Radau IIA has a small energy error (1–2% of the initial energy), but for longer time periods it will continue to grow in time; eventually smoothing and damping the entire flow. The sign of the error is negative, so the energy of the flow strictly decreases, as was expected based on the theoretical analysis from Section 3.3. When the time step is reversed to $-\Delta t$, expression (31) still predicts a decrease in the total energy ($\|u_{n+1}\|^2 \leqslant \|u_n\|^2$), which is in accordance with the results presented here. In contrast to Gauss 4, the (temporal) enstrophy error of Radau IIA, Fig. 7(b), is approximately the same order as the spatial error. A much smaller time step is necessary to achieve results that are of the same quality as the Gauss methods. The errors in energy, enstrophy and time reversibility of all methods have been collected in Fig. 8(a) and (b) (time reversal at $t = 2$). Fig. 8(a) shows that all energy-conserving methods (Gauss, Radau IIB, Lobatto IIIE) have an error in energy which stays at machine precision, as expected. The error in the non-energy-conserving schemes (Radau IIA, Lobatto IIIC and DIRK) decreases upon time-step refinement according to the order of the method, except for Lobatto IIIC, which shows third order instead of second order. Similarly, Fig. 8(b) shows that the enstrophy error at $t = 2$ converges according to the order of the method, except for Lobatto IIIC, which shows again a higher order convergence rate. Most importantly, the energy-conserving methods have a much lower error constant than their non-energy-conserving counterparts. The order of accuracy alone is therefore not decisive. Fig. 8(c) shows the time-reversibility error, which decreases with third order for all time-irreversible methods (Radau IIA, Radau IIB, Lobatto IIIC and DIRK) – this can probably be explained by fortuitous error cancellations in the backward simulation of the second order methods. Comparing Fig. 8(a) to (c) reveals that Radau IIB is energy-conserving but time-irreversible, and Lobatto IIIA (Crank–Nicolson) is time-reversible, but not energy-conserving. Although Crank–Nicolson is conceptually close to the implicit midpoint method, we see here that the energy-conserving properties of the implicit midpoint method make it clearly superior in predicting the enstrophy of the flow. Furthermore we see that the time-reversibility property of Lobatto IIIE (which was obtained by sacrificing one order compared to Radau IIB) is unimportant for accurate enstrophy prediction. Therefore, we conclude that *energy conservation is more important than time-reversibility*. We disagree with the conclusions stated by Duponcheel et al. [43], who mention that a 'crucial factor for time-reversibility is the accuracy of the time-stepping scheme and its interaction with the space-discretization'. The time-reversibility of a method is *not* depending on its accuracy; we clearly see that second-order methods like Crank–Nicolson or Gauss 2 can be time-reversible, while a third-order method like Radau IIA is not. For Runge–Kutta methods, time-reversibility is completely determined by satisfying conditions (38) and (40). The only relation between time-reversibility and order is that time-reversible schemes are of even order.

At $t = 8$ similar conclusions can be drawn. The absolute error values are much higher than at $t = 2$, which is attributed to the lack of smoothness of the flow field compared to $t = 2$. Furthermore, the enstrophy errors at $t = 8$ are more irregular than at $t = 2$, and not all methods are in the range of asymptotic convergence. Still, the energy-conserving methods show the best performance.

In practice, the choice for a method depends on efficiency: accuracy versus computational costs. Gauss 2 and Lobatto IIIA are the 'cheapest' methods, having roughly the same cost, DIRK is about twice as expensive, and all other methods are much more expensive. For this test case the Gauss 2 or DIRK methods are favorable if second-order accuracy is sufficient. For highly accurate computations the Gauss 4 method is to be preferred, followed closely by the Radau IIB method. The Lobatto IIIC/E methods are not competitive for this test case due their high computational costs and relatively low accuracy.

7.3. Corner flow

In this section we investigate the performance of the different time-integration schemes for problems with varying stiffness and the role of L-stability for such problems. We consider a test case from van Kan [72]: unsteady flow through a corner,

featuring unsteady inflow conditions on the top, no-slip conditions on the left and bottom, and outflow conditions on the right (Fig. 9). Although the flow in the lower-left corner is similar to the flow in lid-driven cavities, we deliberately do not compute that test case here, because it features steady boundary conditions for the continuity equation. Unsteady inflow conditions are a more severe test for methods to show the correct convergence order in both velocity and pressure (see the discussion in [45]). Boundary layers develop on the solid walls, and the stiffness of the problem can be controlled by varying the mesh size and Reynolds number (and as such the thickness of the boundary layers).

We perform the five different test cases listed in Table 9. In each case one of the parameters (N_x/N_y, Re, domain or grid) is varied with respect to the previous case. The cosine grid is given by transforming a uniform grid ξ according to

$$x = L_x\left(1 - \cos\left(\frac{\pi}{2}\frac{\xi}{L_x}\right)\right), \tag{96}$$

and similarly in y-direction. This transformation results in a fine grid near the solid walls $x = 0$ and $y = 0$. For case 5 the transformation in y-direction is changed to $y = L_y/2\left(1 - \cos\left(\pi\frac{\xi}{L_y}\right)\right)$ in order to have a fine grid at both $y = 0$ and $y = 1$.

Qualitative pictures of the flow field are shown in Fig. 11(a) for case 1 (2 and 3 are similar), Fig. 11(b) for case 4, and Fig. 12 for case 5. In cases 1–3 the flow is diffusion-dominated and the mesh Péclet condition is satisfied in the entire domain. In cases 4 and 5 the flow is convection-dominated and the mesh Péclet condition is satisfied only near the walls, where the largest gradients appear. Inspection of the flow field did not show any significant wiggles.

Fig. 13(a)–(d) show the error in velocity for the different time-integration methods. The error is calculated with respect to a simulation with a very small time step, and the infinity norm is displayed. The error in pressure is left out, because it does not provide new insights with respect to the Taylor–Green test case. The Lobatto IIIC, C/E and E methods have been left out because their accuracy was again comparable to Gauss 2 and Lobatto IIIA, but at much higher computational costs. To interpret the convergence plots we will use the eigenvalues of the linearized operator (Jacobian) $-C(u) + vD$ at $t = 1$. Fig. 10 shows for each case the largest magnitude eigenvalue in the complex plane (they lie on the real axis) and the amplification factor of each method for $\Delta t = 1$. These eigenvalues scale as $\lambda \sim \frac{1}{\mathrm{Re}}\frac{1}{h^2}$, with h the smallest mesh size and $\mathrm{Re} = 1/v$ the Reynolds number. If h is (locally) of the order as required by the mesh Péclet condition ($h \sim 2/\mathrm{Re}$), the eigenvalues scale as $\lambda \sim \mathrm{Re}$.

In case 1 the mesh is coarse and the position of the eigenvalues reveals that the problem is not very stiff. All methods obtain the theoretical rate of convergence, as was the case for the Taylor–Green vortex. Radau IIA/B is, somewhat surprisingly, less accurate than Radau IIB. Intuitively one might think that Radau IIA/B should lie in between Radau IIA and IIB, but this is not necessarily the case. The local truncation error of Radau IIA/B can be higher due to the presence of error terms that result from the additional order conditions.

Upon refining the mesh, case 2, the problem becomes stiffer and the eigenvalues shift away from the imaginary axis. Gauss 4 still attains its theoretical convergence order, but shows a reduced order (namely 2) at large time steps, and is only more accurate than Radau IIA for sufficiently small time steps. Radau IIA/B is slowly converging towards its asymptotic rate, but is not more accurate than the second order methods over the range of time steps considered here.

Table 9
Settings for corner flow.

Case	$N_x \times N_y$	t_{end}	Re	Domain ($L_x \times L_y$)	Grid
1	20×20	1	10	1×1	Uniform
2	80×80	1	10	1×1	Uniform
3	80×80	1	10	1×1	Cosine
4	80×80	1	1000	1×1	Cosine
5	200×100	40	1000	2×1	Cosine

	case 1	case 2	case 3	case 4
Gauss 2, Lob IIIA	-0.98859	-0.99928	-1.00000	-0.99996
Gauss 4, Radau IIB	0.96617	0.99784	1.00000	0.99989
Radau IIA, IIA/B	$-5.6 \cdot 10^{-3}$	$-3.6 \cdot 10^{-4}$	$-1.8 \cdot 10^{-7}$	$-1.8 \cdot 10^{-5}$
DIRK L, L/E	$-1.4 \cdot 10^{-2}$	$-9.0 \cdot 10^{-4}$	$-4.4 \cdot 10^{-7}$	$-4.4 \cdot 10^{-5}$

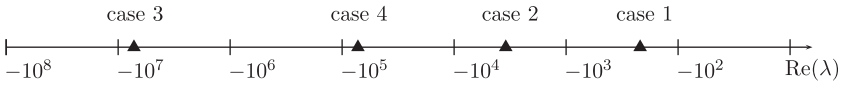


Fig. 10. Largest magnitude eigenvalues on the real axis and corresponding amplification factors for cases 1–4.

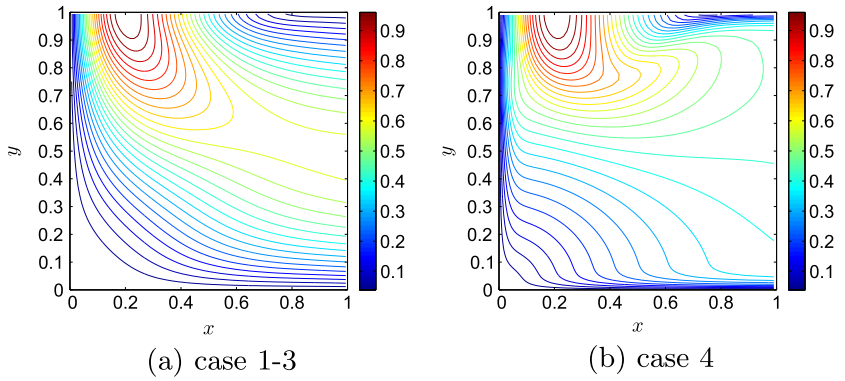


Fig. 11. Contour lines of velocity magnitude at $t = 1$.

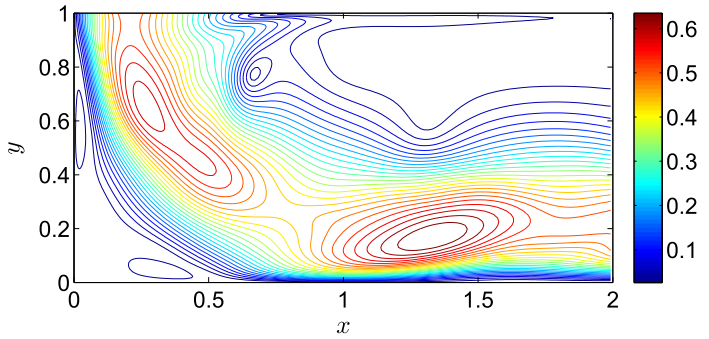


Fig. 12. Contour lines of velocity magnitude, case 5 at $t = 40$.

In case 3, the mesh is refined near the walls and the problem becomes very stiff. The ratio of the smallest mesh size between the uniform and cosine grid is approximately 65, and the eigenvalues shift with this factor from case 2 to case 3 as is observed in Fig. 10. All methods now converge according to the orders for stiff problems mentioned in Section 3.2; Gauss 2: $\mathcal{O}(\Delta t^{\eta+1}) = \mathcal{O}(\Delta t^2)$, Gauss 4: $\mathcal{O}(\Delta t^{\eta}) = \mathcal{O}(\Delta t^2)$, Radau IIB: $\mathcal{O}(\Delta t^{\eta}) = \mathcal{O}(\Delta t)$, Radau IIA/B: $\mathcal{O}(\Delta t^{\eta+1}) = \mathcal{O}(\Delta t^2)$, DIRK L or L/E: $\mathcal{O}(\Delta t^{\eta+1}) = \mathcal{O}(\Delta t^2)$. Radau IIA and Lobatto IIIA are stiffly accurate and obtain their classical order p. Not surprisingly, Radau IIA is superior to Gauss 4. Radau IIA/B is now more accurate than Radau IIB (which has very poor behavior due to $R(\infty) = 1$ and $\eta = 1$), but it is not more accurate than Gauss 4 for the range of time steps considered here. Although Radau IIA/B is L-stable, its lower stage order compared to Gauss 4 prevents it to be more accurate. The second order methods all behave very similarly, making Gauss 2 and Lobatto IIIA most attractive due to the lowest computational cost. We stress once more that the observed order reduction is *not* a consequence of the fact that we are dealing with DAEs instead of ODEs, but a result of 'ordinary' stiffness that would also lead to order reduction if ODEs were considered.

A practically more important example of a convection-dominated flow, where the mesh spacing is close to 'physical' requirements, is obtained by increasing the Reynolds number to 1000, while keeping the grid fixed: case 4. The eigenvalues shift back towards the imaginary axis by a factor of 100. Lobatto IIIA loses its advantage over Gauss 2 compared to cases 1–3, where it was more accurate than Gauss 2. Fig. 13(d) indicates that Gauss 4 and Radau IIA/B perform much better than in case 3. A 'hump' in the curves of Radau IIB and Gauss 4 around $\Delta t = 10^{-2}$ indicates a transition from stage order to classical order (a similar hump appeared in case 2 at a roughly ten times larger Δt, corresponding to the ratio of eigenvalues of case 2 and 4). When increasing the Reynolds number further, with the mesh locally satisfying the Péclet condition, the eigenvalues will shift again away from the imaginary axis, increasing the stiffness of the problem. We expect therefore that for high Reynolds number flow the order reduction as observed in case 3 is likely to occur again. For practical computations it might be worthwhile to detect stiffness during running. For example, one can estimate the largest eigenvalue $\lambda_{\max}$ of the Jacobian by Gershgorin's theorem and compute the associated value of the stability function $|R(\lambda_{\max}\Delta t)|$. If this value is larger than some predefined threshold $R_{\max}$, e.g. $R_{\max} = 0.999$, then a code could switch to a stiffly-accurate method such as Radau IIA.

Finally we consider a similar grid as in case 4, but change the inflow condition to a periodic one:

$$v(t) = -\sin(\pi(x^3 - 3x^2 + 3x))e^{1-1/t^4}\left(\frac{1}{2} + \sin^2(t)\right)/2e, \quad 0 \leqslant x \leqslant 1. \tag{97}$$

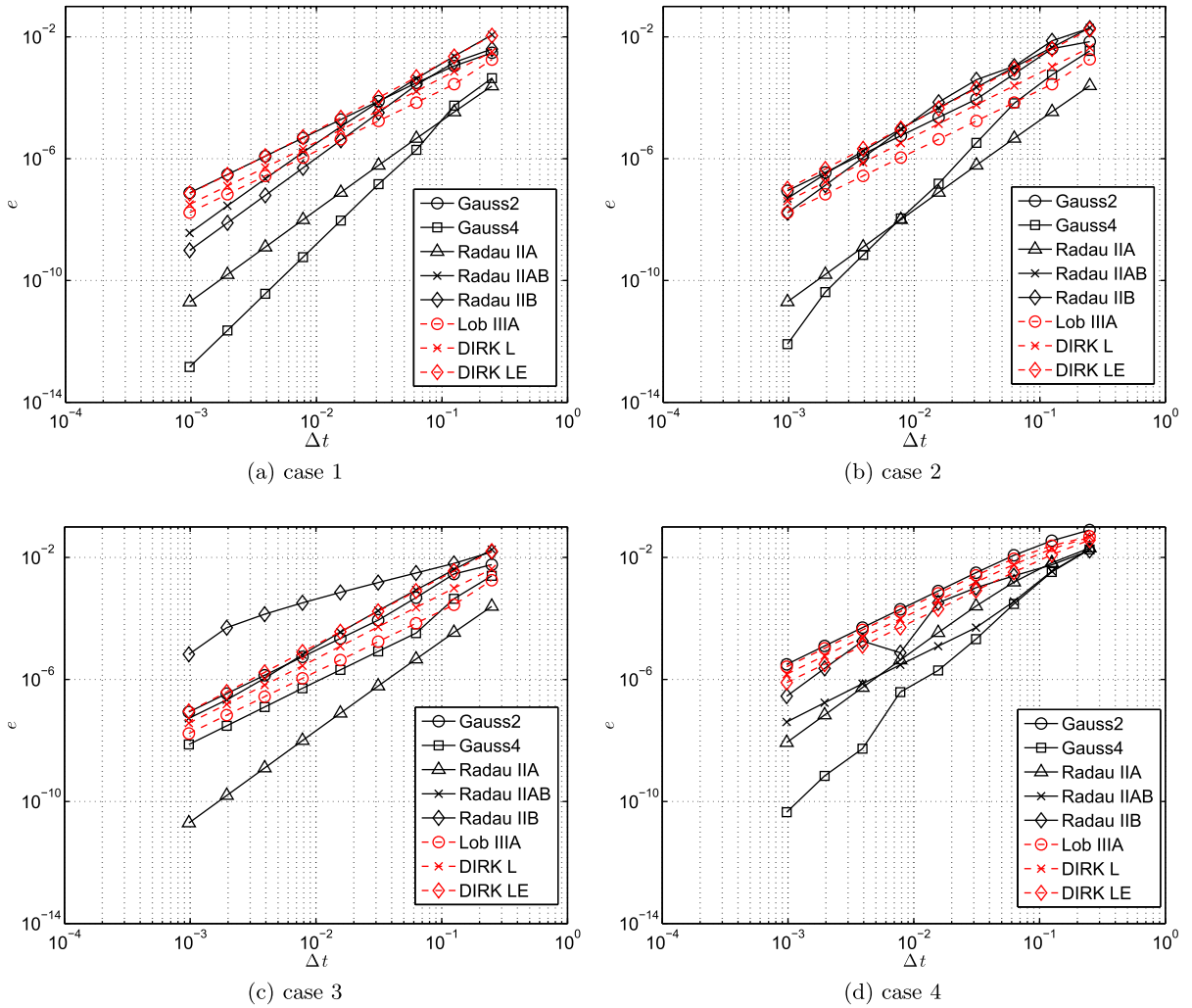
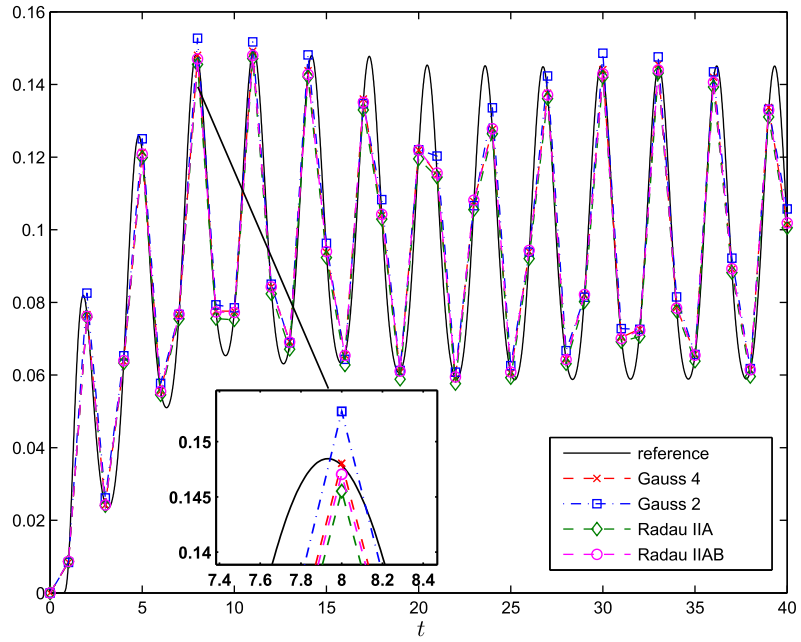
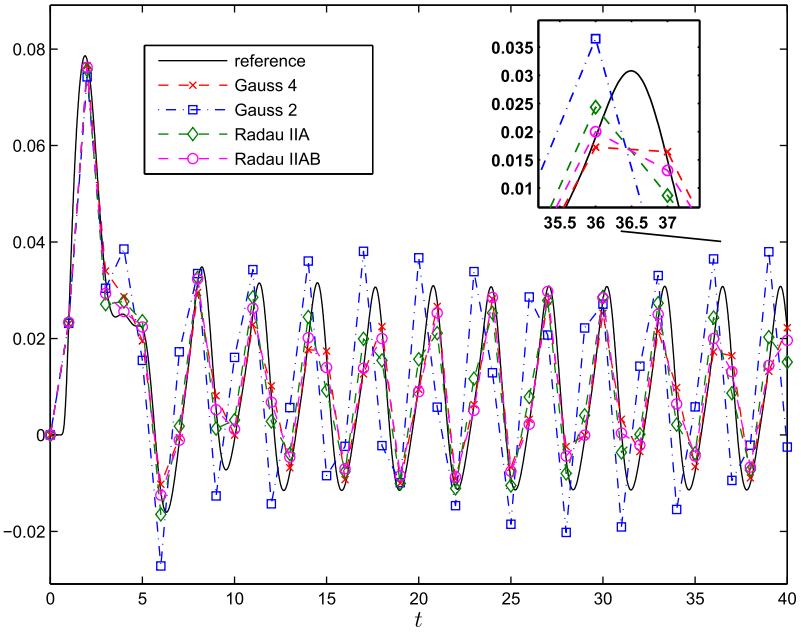


Fig. 13. Velocity errors corner flow for cases 1–4.

After an initial transient of approximately 5 time units, the inflow becomes periodic with period π, average 0.5 and amplitude 0.25. The domain is increased in size (see Fig. 9) to allow vortices to travel for some distance without being affected by the outflow boundary conditions. Based on the results of case 4 we select the Gauss 2, 4, Radau IIA and Radau IIA/B methods. In order to study the performance of the different methods, we look at the global kinetic energy K and the u-velocity component in point $(0.2, 0.2)$ as function of time. The latter point lies inside an oscillating vortex and its value gives an indication how well the small scales of the flow are resolved in time. In all cases we use a very large time step, $\Delta t = 1$, meaning that *there are only 3 time steps to resolve a period of the oscillation*. Fig. 14(a) and (b) shows K and $u(0.2, 0.2)$ for these methods, together with reference results obtained at very small Δt. Surprisingly, even at this large time step, all methods agree reasonably well with the reference solution. When zooming in on the results of Fig. 14(a), one finds that Gauss 4 is following the exact solution perfectly, followed closely by Radau IIA/B and Radau IIA. The good behavior of Gauss 4 can be explained by the fact that the problem is smooth in time, so that higher-order methods are effective. Gauss 2 is the least accurate, but this is not surprising since it is only second order in time. These conclusions become more pronounced when studying the small scale oscillations, Fig. 14(b), where Gauss 2 starts to deviate strongly from the reference solution, while Gauss 4 and Radau IIA/B still perform very well. To obtain qualitatively similar results with Gauss 2 as with Gauss 4 the time step had to be reduced to approximately $\Delta t = 1/4$, increasing the number of time steps with a factor 4. Since the size of the matrix of the non-linear problem to be solved for Gauss 4 is $(2(N_u + N_p))^2$ instead of $(N_u + N_p)^2$, a factor of 4 is generally not enough to offset the increase in computational costs, because the solution of the system is more expensive than $\mathcal{O}(N_u + N_p)$. Gauss 2 is then to be preferred. However, for a higher accuracy and smaller time steps this factor will rapidly increase and make Gauss 4 competitive.



(a) K



(b) $u(0.2, 0.2)$

Fig. 14. Periodic inflow, case 5.

8. Conclusions

In this paper we have investigated energy-conserving Runge–Kutta methods for the incompressible Navier–Stokes equations. Energy-conserving schemes are devoid of numerical viscosity, which is a favorable property for DNS and LES computations, where the Reynolds numbers are typically high and any viscosity should be as physical as possible. LES sub-grid models can be tested without the interference of any artificial viscosity. Furthermore, the non-linear stability properties associated with energy conservation make these methods very appropriate for integrating over long periods of time, possibly

with large time steps. The Gauss methods stand out since they are energy conserving, time reversible and have the highest possible order given the number of stages. Methods based on Radau and Lobatto quadrature can also be transformed into energy-conserving methods (such as Radau IIB and Lobatto IIIE), but are of lower order.

The beneficial effect of energy conservation was shown by simulating the roll-up of an inviscid shear layer. The energy-conserving methods gave an accurate prediction of the enstrophy of the flow; in particular the two-stage Gauss and Radau IIB methods gave excellent results at very large time steps. Apparently, these methods have small error constants, which is at least as important as the order of accuracy of the method. Classical methods, such as explicit Runge–Kutta or multi-step methods, cannot combine high order with stability and inevitably add some artificial viscosity to the flow. The time-reversibility property of a time integration method was found to be unimportant by comparing Gauss 2 (time-reversible and energy-conserving) with Crank–Nicolson (time-reversible but not energy-conserving) and Radau IIB (energy-conserving but not time-reversible).

A disadvantage of all energy-conserving methods is that they lack L-stability, which makes them unsuitable if stiff components (such as boundary layers) are present in the semi-discrete equations. This leads to order reduction for energy-conserving methods of order higher than 2. We have therefore proposed a new class of high-order Runge–Kutta methods, that combine energy-conservation in the inviscid limit with L-stability in the stiff limit. We construct such methods by starting from a stiffly accurate (L-stable) Runge–Kutta method, transforming it with the method of Sun [59], and using these together to form a fully implicit additive Runge–Kutta (ARK) method. Although these new ARK methods lack the algebraic stability properties of their two constituents, they possess A-stability, which was sufficient for the test cases considered here. The highest possible order ($p = 2s - 1$) ARK method with these properties is the Radau IIA/B method. Other additive methods derived in this paper, being diagonally implicit or based on Lobatto quadrature, were not found to be competitive in terms of accuracy versus computational cost. The Radau IIA/B method gave excellent results for long time integration of a convection-dominated problem with considerable stiffness, employing large time steps. However, order reduction still appears for very stiff problems – not due to a lack of L-stability, but due to a lower stage order – and the asymptotic convergence rate is only reached for sufficiently small time steps. As a consequence, the Radau IIA/B method does not outperform the Gauss method in the test cases considered here. For very stiff problems the issue of order reduction persists, and only stiffly-accurate methods perform well on such problems. The construction of an 'all-round' high-order (additive) Runge–Kutta method, which is both energy-conserving and not suffering from order reduction for stiff problems, remains an open question.

In the end, the choice for a method depends on the problem at hand and on the desired accuracy. We believe the Gauss 4 method is very suitable for sensitive flow phenomena that require high accuracy, such as DNS studies of the transition from laminar to turbulent flow, especially when long time intervals are considered. On the other hand, for many engineering problems second order is generally sufficient and then the Gauss 2 method is the best candidate for turbulent flow simulations in which artificial viscosity should be avoided. This method has the additional advantage that it does not suffer from order reduction when applied to stiff problems.

For future work on large turbulent flow computations it is important to solve the non-linear saddle-point problem efficiently, which can be achieved by applying proper splitting and linearization techniques. Furthermore, it is crucial to perform adaptive time-stepping in order to obtain efficient methods which adjust to the dynamics of the flow. This is not trivial, since variable step sizes can considerably deteriorate the good properties of symplectic methods (see e.g. [77] for an example, and [44] for remedies). Eventually, our goal is to combine the methods presented here with high-order spatial discretizations such as [7]. It is expected that this leads to accurate simulation of turbulence, even with large time steps and coarse grids. In future work we will address such turbulent flows and compare the efficiency of the methods analyzed in this paper with the current state-of-the-art methods, such as the IMEX Runge–Kutta method from [35].

Acknowledgments

The author thanks B. Koren and W. Hundsdorfer for stimulating discussions and proofreading of the manuscript.

Appendix A. *W*-transformation

Hairer and Wanner introduced the W-transformation in the study of the construction of high-order implicit Runge–Kutta methods [46]. The $s \times s$ matrix W is a Vandermonde matrix defined by

$$w_{ij} = \widetilde{P}_{j-1}(c_i), \quad i = 1, \ldots, s, \ j = 1, \ldots, s, \tag{A.1}$$

where $\widetilde{P}_k(x)$ are the normalized shifted Legendre polynomials defined by

$$\widetilde{P}_k(x) = \sqrt{2k+1} \sum_{j=0}^{k} (-1)^{j+k} \binom{k}{j} \binom{j+k}{j} x^j, \quad k = 0, 1, \ldots. \tag{A.2}$$

Application of W on the coefficient matrix A (the W-transformation) leads to a matrix, X^*, with a special structure:

$$X^* = W^{-1} A W. \tag{A.3}$$

This matrix can be related to the simplifying conditions $C(\eta)$ and $D(\zeta)$ and is therefore linked to the order of the method ([46], Theorem 5.11). The matrix X, defined by

$$X = W^T BAW = JX^*, \quad \text{where } J = W^T BW, \tag{A.4}$$

is closely resembling X^* and can be used to characterize symmetry, symplecticity and stability properties. In case the quadrature formula is of order $\geqslant 2s-1$, then $J = I$ and $X = X^*$. The special structure of matrix X is especially evident for the Gauss methods, for which it is denoted by X_G:

$$X_G = \begin{pmatrix} 1/2 & -\xi_1 & & & \\ \xi_1 & 0 & -\xi_2 & & \\ & \xi_2 & \ddots & \ddots & \\ & & \ddots & 0 & -\xi_{s-1} \\ & & & \xi_{s-1} & 0 \end{pmatrix}, \tag{A.5}$$

where $\xi_k = \frac{1}{2\sqrt{4k^2-1}}$. The conditions for symplecticity, algebraic stability and symmetry in terms of the X matrix read [57,46]:

$$X + X^T - e_1 e_1^T = 0 \qquad \text{symplectic}, \tag{A.6}$$

$$X + X^T - e_1 e_1^T \geqslant 0 \qquad \text{algebraically stable}, \tag{A.7}$$

$$X_{kl} = 0 \text{ for } k + l \neq 2 \text{ is even}, \ \widehat{P}c = e - c, \ \widehat{P}b = b \qquad \text{symmetric}, \tag{A.8}$$

where $\widehat{P}$ is the permutation matrix with elements $\hat{p}_{ij} = \delta_{i,s+1-j}$, $e = (1,\ldots,1)^T$ and $e_1 = (1,0,\ldots,0)^T$.

Appendix B. Energy-conserving (S)DIRK methods

We are interested in algebraically stable energy-conserving (S)DIRK methods, i.e., having $E = 0$ and $b \geqslant 0$. Energy-conserving DIRK methods can be derived by employing the W-transform of the previous section, but for 'low' order methods (say lower than 4) it is somewhat easier to take the following approach. Evaluating the symplecticity condition

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0, \tag{B.1}$$

for DIRK methods ($a_{ij} = 0$ for $j > i$) leads to a Butcher tableau with the following structure:

$$\begin{array}{c|cccc} b_1/2 & b_1/2 & & & \\ b_1 + b_2/2 & b_1 & b_2/2 & & \\ b_1 + b_2 + b_3/2 & b_1 & b_2 & b_3/2 & \\ \vdots & \vdots & \vdots & \vdots & \ddots \\ b_1 + \ldots + b_{s-1} + b_s/2 & b_1 & b_2 & b_3 & \ldots\ b_s/2 \\ \hline & b_1 & b_2 & b_3 & \ldots\ b_s \end{array} \tag{B.2}$$

A particular family which satisfies this form is given by $b_i = 1/s$ ($j < i$), but it is limited to second order (it is simply repeated application of the midpoint method). All high-order energy-conserving DIRK methods obtained in literature, e.g. [78,79,56,80], have negative b-coefficients and are therefore not algebraically stable – they are only useful for pure convective-type problems, where all eigenvalues lie on the imaginary axis. This is confirmed by Chan et al. [81], who showed by using theory on order stars that *all algebraically stable energy-conserving DIRK methods are limited to second order*, independent of s. This negative result also holds for the weaker concept of A-stability. For energy-conserving DIRK methods the stability function reads (see Eq. (43))

$$R(z) = \frac{\prod_{i=1}^{s} 1 + \frac{1}{2} b_i z}{\prod_{i=1}^{s} 1 - \frac{1}{2} b_i z}, \tag{B.3}$$

and its poles are given by $z = 2/b_i$. A-stability requires that all poles of the stability function $R(z)$ lie in the positive half plane $\mathbb{C}^+$ [46]. The condition $b_i \geqslant 0$ is therefore not only necessary for algebraic stability, but also for A-stability, when energy-conserving DIRK methods are considered.

Appendix C. Stability of additive Runge–Kutta methods

C.1. Algebraic stability

Consider Eq. (50), repeated here for convenience:

$$\|u_{n+1}\|^2 = \|u_n\|^2 + 2\Delta t\sum_{i=1}^{s} b_i((U_i, PF_i) + (U_i, P\widehat{F}_i)) - \Delta t^2\left(\sum_{i,j=1}^{s} e_{ij}(PF_i, PF_j) + \sum_{i,j=1}^{s}\hat{e}_{ij}(P\widehat{F}_i, P\widehat{F}_j) + 2\sum_{i,j=1}^{s}\tilde{e}_{ij}(PF_i, P\widehat{F}_j)\right). \tag{C.1}$$

The second term on the right-hand side of this equation, $2\Delta t(\ldots)$, is zero for pure convection problems and negative if diffusion is present, as long as $b_i \geqslant 0$. The sign of the third term, $-\Delta t^2(\ldots)$, depends on the matrices $E, \widehat{E}$ and $\widetilde{E}$. The entire term can be written as

$$-\Delta t^2\sum_{i,j=1}^{2s}\bar{e}_{ij}(x_i, x_j), \tag{C.2}$$

by defining $x = (PF_1, \ldots, PF_s, P\widehat{F}_1, \ldots, P\widehat{F}_s)^T$, and

$$\overline{E} = \begin{pmatrix} E & \widetilde{E} \\ \widetilde{E} & \widehat{E} \end{pmatrix}. \tag{C.3}$$

Unconditional algebraic stability requires that $\overline{E}$ is nonnegative definite (nonnegative definiteness of the individual blocks is not sufficient, see [82]). We now encounter the following negative result.

Theorem C.1 (*Algebraic stability of additive Runge–Kutta methods*). *Consider an additive Runge–Kutta method (44) and (45) with $b = \hat{b}$ and the Butcher tableau $\widehat{A}$ for the convective terms such that $\widehat{E} = 0$. Then the method cannot be algebraically stable, except if it is trivial (i.e. $A = \widehat{A}$).*

Proof. For algebraic stability $\overline{E}$ should be nonnegative definite. Given that $\widehat{E} = 0$ this leads to the requirement $\widetilde{E} = 0$. However, $\widehat{E} = B\widehat{A} + \widehat{A}^TB - bb^T = 0$ leads to $\widetilde{E} = B\widehat{A} + A^TB - bb^T = B(\widehat{A} - A)$, which is zero only in case $\widehat{A} = A$. □

In [82] it is shown (for general additive Runge–Kutta methods) how the term

$$2\Delta t\sum_{i=1}^{s} b_i((U_i, PF_i) + (U_i, P\widehat{F}_i)) \tag{C.4}$$

can be used to obtain *conditional* algebraic stability, by requiring a more strict condition than $(F_i, U_i) \leqslant 0$, namely

$$(F_i, U_i) \leqslant \mu_i\|F_i\|^2, \quad \mu_i < 0. \tag{C.5}$$

Such a condition permits one to add diagonal matrices to the diagonal blocks of $\overline{E}$, and with a condition on Δt this can lead to a nonnegative definite $\overline{E}$. In our case, (C.5) cannot hold for the convective terms because $(C(U_i, U_i), U_i) = 0$, and consequently the zero block in $\overline{E}$ caused by $\widehat{E}$ will remain zero. Conditional algebraic stability can therefore not be proven either.

C.2. Linear stability

Consider the expression for the amplification factor for an additive Runge–Kutta method, Eq. (53):

$$R(z, \hat{z}) \equiv \frac{u_{n+1}}{u_n} = \frac{\mathrm{Det}(I - zA - \hat{z}\widehat{A} + (z + \hat{z})eb^T)}{\mathrm{Det}(I - zA - \hat{z}\widehat{A})} = \frac{N(z, \hat{z})}{D(z, \hat{z})}, \tag{C.6}$$

Both $N(z, \hat{z})$ and $D(z, \hat{z})$ can be expressed as polynomials of the form:

$$N(z, \hat{z}) = \sum_{i=0}^{s}\left(\sum_{j=0}^{s-i} n_{ij}\hat{z}^j\right)z^i, \quad D(z, \hat{z}) = \sum_{i=0}^{s}\left(\sum_{j=0}^{s-i} d_{ij}\hat{z}^j\right)z^i, \tag{C.7}$$

where the coefficients n_{ij} and d_{ij} associated with the highest order terms are

$$n_{0s} = (-1)^s\mathrm{Det}(\widehat{Q}), \quad n_{s0} = (-1)^s\mathrm{Det}(Q), \tag{C.8}$$

$$d_{0s} = (-1)^s\mathrm{Det}(\widehat{A}), \quad d_{s0} = (-1)^s\mathrm{Det}(A). \tag{C.9}$$

If the original method $\mathcal{A}$ is L-stable (i.e., it has $\mathrm{Det}(Q) = 0$ and $\mathrm{Det}(A) \neq 0$), then these expressions show that the resulting additive method is still L-stable, i.e., we have for fixed $\hat{z}$:

$$\lim_{z\to-\infty} R(z, \hat{z}) = 0, \tag{C.10}$$

provided that the additive method is A-stable. Unfortunately, as in the case of algebraic stability, A-stability of the additive method does not follow from the A-stability of the two separate methods due to the coupling terms in $N(z,\hat{z})$ and $D(z,\hat{z})$.

References

[1] C. Foias, O. Manley, R. Rosa, R. Temam, Navier-Stokes Equations and Turbulence, Cambridge University Press, 2001.
[2] K. Mahesh, G. Constantinescu, P. Moin, A numerical method for large-eddy simulation in complex geometries, J. Comput. Phys. 197 (2004) 215–240.
[3] B. Perot, Conservation properties of unstructured staggered mesh schemes, J. Comput. Phys. 159 (2000) 58–89.
[4] R. Mittal, P. Moin, Suitability of upwind-biased finite difference schemes for large-eddy simulation of turbulent flows, AIAA J. 35 (1997) 1415–1417.
[5] S. Nagarajan, S. Lele, J. Ferziger, A robust high-order compact method for large eddy simulation, J. Comput. Phys. 191 (2003) 392–419.
[6] F. Ham, K. Mattsson, G. Iaccarino, P. Moin, Towards time-stable and accurate LES on unstructured grids, in: S. Kassinos, C. Langer, G. Iaccarino, P. Moin (Eds.), Complex Effects in Large Eddy Simulations, Lecture Notes in Computational Science and, Engineering, vol. 56, 2007, pp. 235–249.
[7] R. Verstappen, A. Veldman, Symmetry-preserving discretization of turbulent flow, J. Comput. Phys. 187 (2003) 343–368.
[8] R. Verstappen, A. Veldman, Spectro-consistent discretisation of Navier-Stokes: a challenge to RANS and LES, J. Eng. Math. 34 (1998) 163–179.
[9] N. Kampanis, J. Ekaterinaris, A staggered, high-order accurate method for the incompressible Navier-Stokes equations, J. Comput. Phys. 215 (2006) 589–613.
[10] P. Sagaut, Large Eddy Simulation for Incompressible Flows, Springer, 2002.
[11] D. Tafti, Comparison of some upwind-biased high-order formulations with a second-order central difference scheme for time integration of the incompressible Navier-Stokes equations, Comput. Fluids 25 (1996) 647–665.
[12] F. Felten, T. Lund, Kinetic energy conservation issues associated with the collocated mesh scheme for incompressible flow, J. Comput. Phys. 215 (2006) 465–484.
[13] R. Sadourny, The dynamics of finite-difference models of the shallow-water equations, J. Atmos. Sci. 32 (1975) 680–689.
[14] A. Arakawa, Computational design for long-term numerical integration of the equations of fluid motion: two-dimensional incompressible flow. Part I, J. Comput. Phys. 1 (1966) 119–143.
[15] J. Ferziger, M. Perić, Computational Methods for Fluid Dynamics, third revised ed., Springer, 2002.
[16] B. Geurts, Elements of Direct and Large-Eddy Simulation, Edwards, 2004.
[17] B. Sanderse, S. van der Pijl, B. Koren, Review of computational fluid dynamics for wind turbine wake aerodynamics, Wind Energ. 14 (2011) 799–819.
[18] F. Harlow, J. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluids 8 (1965) 2182–2189.
[19] P. Wesseling, Principles of Computational Fluid Dynamics, Springer, 2001.
[20] F. Ham, F. Lien, A. Strong, A fully conservative second-order finite difference scheme for incompressible flow on nonuniform grids, J. Comput. Phys. 177 (2002) 117–133.
[21] P. Wesseling, A. Segal, C. Kassels, Computing flows on general three-dimensional nonsmooth staggered grids, J. Comput. Phys. 149 (1999) 333–362.
[22] X. Zhang, D. Schmidt, B. Perot, Accuracy and conservation properties of a three-dimensional unstructured staggered mesh scheme for fluid dynamics, J. Comput. Phys. 175 (2002) 764–791.
[23] Y. Morinishi, T. Lund, O. Vasilyev, P. Moin, Fully conservative higher order finite difference schemes for incompressible flows, J. Comput. Phys. 143 (1998) 90–124.
[24] J. Hicken, F. Ham, J. Militzer, M. Koksal, A shift transformation for fully conservative methods: turbulence simulation on complex, unstructured grids, J. Comput. Phys. 208 (2005) 704–714.
[25] O. Vasilyev, High order finite difference schemes on non-uniform meshes with good conservation properties, J. Comput. Phys. 157 (2000) 746–761.
[26] R. Knikker, Study of a staggered fourth-order compact scheme for unsteady incompressible viscous flow, Int. J. Numer. Meth. Fluids 59 (2009) 1063–1092.
[27] A. Hokpunna, M. Manhart, Compact fourth-order finite volume method for numerical solutions of Navier-Stokes equations on staggered grids, J. Comput. Phys. 229 (2010) 7545–7570.
[28] J.B. Perot, Discrete conservation properties of unstructured mesh schemes, Annu. Rev. Fluid. Mech. 43 (2011) 299–318.
[29] M. Shashkov, Conservative Finite-Difference Methods on General Grids, CRC Press, 1996.
[30] J. Hyman, M. Shashkov, S. Steinberg, The numerical solution of diffusion problems in strongly heterogeneous non-isotropic materials, J. Comput. Phys. 132 (1997) 130–148.
[31] H.-O. Kreiss, G. Scherer, Finite element and finite difference methods for hyperbolic partial differential equations, in: C. de Boor (Ed.), Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, New York, 1974.
[32] R. Nicolaides, The covolume approach to computing incompressible flow, in: M. Gunzburger, R. Nicolaides (Eds.), Incompressible Computational Fluid Dynamics. Trends and Advances, Cambridge University Press, 1993, pp. 295–333.
[33] V. Subramanian, J. Perot, Higher-order mimetic methods for unstructured meshes, J. Comput. Phys. 219 (2006) 68–85.
[34] P. Mullen, K. Crane, D. Pavlov, Y. Tong, M. Desbrun, Energy-preserving integrators for fluid animation, in: Proceedings of ACM SIGGRAPH, vol. 28, 2009.
[35] P. Spalart, R. Moser, M. Rogers, Spectral methods for the Navier-Stokes equations with one infinite and two periodic directions, J. Comput. Phys. 96 (1991) 297–324.
[36] H. Le, P. Moin, An improvement of fractional step methods for the incompressible Navier-Stokes equations, J. Comput. Phys. 92 (1991) 369–379.
[37] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier-Stokes equations, J. Comput. Phys. 59 (1985) 308–323.
[38] H. Choi, P. Moin, Effect of the computational time step on numerical simulation of turbulent flow, J. Comput. Phys. 113 (1994) 1–4.
[39] J. Simo, F. Armero, Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations, Comput. Meth. Appl. Mech. Eng. 111 (1994) 111–154.
[40] S. Turek, A comparative study of some time-stepping techniques for the incompressible Navier-Stokes equations. From fully implicit nonlinear schemes to semi-implicit projection methods, Int. J. Numer. Meth. Fluids 20 (1996) 987–1011.
[41] R. Verstappen, J. Wissink, A. Veldman, Direct numerical simulation of driven cavity flows, Appl. Sci. Res. 51 (1993) 377–381.
[42] B. Vreman, Direct and large-eddy simulation of the compressible turbulent mixing layer, Ph.D. thesis, University of Twente, 1993.
[43] M. Duponcheel, P. Orlandi, G. Winckelmans, Time-reversibility of the Euler equations as a benchmark for energy conserving schemes, J. Comput. Phys. 227 (2008) 8736–8752.
[44] E. Hairer, C. Lubich, G. Wanner, Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations, second ed., Springer, 2006.
[45] B. Sanderse, B. Koren, Accuracy analysis of explicit Runge-Kutta methods applied to the incompressible Navier-Stokes equations, J. Comput. Phys. 231 (2011) 3041–3063.
[46] E. Hairer, G. Wanner, Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, second ed., Springer, 1996.
[47] U. Ascher, L. Petzold, Projected implicit Runge-Kutta methods for differential-algebraic equations, SIAM J. Numer. Anal. 28 (1991) 1097–1120.
[48] L. Jay, Specialized Runge-Kutta methods for index 2 differential-algebraic equations, Math. Comput. 75 (2005) 641–654.
[49] K. Dekker, J. Verwer, Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations, North-Holland, 1984.
[50] J. Butcher, Coefficients for the study of Runge-Kutta integration processes, J. Aust. Math. Soc. 3 (1963) 185–201.
[51] J. Butcher, Numerical Methods for Ordinary Differential Equations, Wiley, 2003.
[52] J. Butcher, Implicit Runge-Kutta processes, Math. Comput. 18 (1964) 50–64.
[53] U. Ascher, L. Petzold, Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, SIAM, 1998.

[54] E. Hairer, C. Lubich, M. Roche, The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods, Springer, 1989.
[55] A. Montlaur, S. Fernandez-Mendez, A. Huerta, High-order implicit time integration for unsteady incompressible flows, Int. J. Numer. Meth. Fluids (2011), http://dx.doi.org/10.1002/fld.2703.
[56] G. Sun, Construction of high order symplectic Runge-Kutta methods, J. Comput. Math. 11 (1993) 250–260.
[57] R. Chan, On symmetric Runge-Kutta methods of high order, Computing 45 (1990) 301–309.
[58] H. Liu, G. Sun, Implicit Runge-Kutta methods based on Lobatto quadrature formula, Int. J. Comput. Math. 82 (2005) 77–88.
[59] G. Sun, A simple way constructing symplectic Runge-Kutta methods, J. Comput. Math. 18 (2000) 61–68.
[60] G. Wanner, Runge-Kutta methods with expansion in even powers of h, Computing 11 (1973) 81–85.
[61] W. Hundsdorfer, J. Verwer, Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations, Springer, 2007.
[62] U. Ascher, S. Ruuth, R. Spiteri, Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, Appl. Numer. Math. 25 (1997) 151–167.
[63] C. Kennedy, M. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, Technical Report TM-2001-211038, NASA, 2001.
[64] L. Pareschi, G. Russo, Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation, J. Sci. Comput. 25 (2005) 129–155.
[65] L. Jay, Symplectic partitioned Runge-Kutta methods for constrained Hamiltonian systems, SIAM J. Numer. Anal. 33 (1996) 368–387.
[66] M. Calvo, J. de Frutos, J. Novo, Linearly implicit Runge-Kutta methods for advection-reaction-diffusion equations, Appl. Numer. Math. 37 (2001) 535–549.
[67] J. Laurent, Structure preservation for constrained dynamics with super partitioned additive Runge-Kutta methods, SIAM J. Sci. Comput. 20 (1998) 416–446.
[68] U. Ascher, On numerical differential algebraic problems with application to semiconductor device simulation, SIAM J. Numer. Anal. 26 (1989) 517–538.
[69] J. Pereira, M. Kobayashi, J. Pereira, A fourth-order-accurate finite volume compact method for the incompressible Navier-Stokes equations, J. Comput. Phys. 167 (2001) 217–243.
[70] M. Benzi, G. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. (2005) 1–137.
[71] J. Perot, An analysis of the fractional step method, J. Comput. Phys. 180 (1993) 51–58.
[72] J. van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Stat. Comput. 7 (1986) 870–891.
[73] W. Chang, F. Giraldo, B. Perot, Analysis of an exact fractional step method, J. Comput. Phys. 180 (2002) 183–199.
[74] J. van Os, R. Uittenbogaard, Towards the ultimate variance-conserving convection scheme, J. Comput. Phys. 197 (2004) 197–214.
[75] B. Sanderse, ECNS: Energy-Conserving Navier-Stokes Solver. Verification of steady laminar flows, Technical Report ECN-E–11-042, Energy research Centre of the Netherlands, 2011.
[76] D. Brown, M. Minion, Performance of under-resolved two-dimensional incompressible flow simulations, II, J. Comput. Phys. 138 (1997) 734–765.
[77] E. Hairer, S. Nørsett, G. Wanner, Solving Ordinary Differential Equations I. Nonstiff Problems, third ed., Springer, 2008.
[78] G. Cooper, Stability of Runge-Kutta methods for trajectory problems, IMA J. Numer. Anal. 7 (1987) 1–13.
[79] M.-Z. Qin, M.-Q. Zhang, Symplectic Runge-Kutta algorithms for Hamiltonian systems, J. Comput. Math. (Suppl.) (1992) 205–215.
[80] J. Franco, I. Gómez, Fourth-order symmetric DIRK methods for periodic stiff problems, Numer. Algorithms 32 (2003) 317–336.
[81] R. Chan, H. Liu, G. Sun, Efficient symplectic Runge-Kutta methods, Appl. Math. Comput. 172 (2006) 908–924.
[82] B. García-Celayeta, I. Higueras, T. Roldán, Contractivity/monotonicity for additive Runge-Kutta methods: inner product norms, Appl. Numer. Math. 56 (2006) 862–878.