A unified discontinuous Galerkin framework for time integration

Sigal Gottlieb[1], G. W. Wei[2,3] and Shan Zhao[4]
[1]Department of Mathematics
University of Massachusetts Dartmouth, North Dartmouth, MA 02747, USA
[2]Department of Mathematics,
Michigan State University, East Lansing, MI 48824, USA
[3]Department of Electrical and Computer Engineering,
Michigan State University, East Lansing, MI 48824, USA
[4]Department of Mathematics,
University of Alabama, Tuscaloosa, AL 35487, USA

June 14, 2010

Abstract

We introduce a new discontinuous Galerkin approach for time integration. Based on the method of weighted residual, numerical quadratures are employed in the finite element time discretization to account for general nonlinear ordinary differential equations. Many different conditions, including explicit, implicit, and symplectic conditions, are enforced for the test functions in the variational analysis in order to obtain desirable features of the resulting time-stepping scheme. The proposed discontinuous Galerkin approach provides a unified framework to derive various time-stepping schemes, such as low order one-step methods, high order Runge-Kutta methods, and multistep methods. Based on the proposed framework, various novel explicit Runge-Kutta methods of different orders are constructed. The derivation of symplectic Runge-Kutta methods has also been realized. The proposed framework allows the optimization of new schemes in terms of several characteristics, such as accuracy, sparseness and stability. The accuracy optimization is performed based on an analytical form of the error estimation function for a linear test initial value problem. Schemes with higher formal order of accuracy are found to provide more accurate solutions. We have also explored the optimization potential of sparseness, which is related to the general compressive sensing in signal/imaging processing. Two critical dimensions of the stability region, i.e., maximal intervals along the imaginary and negative real axes, are employed as the criteria for stability optimization. This gives the largest Courant-Friedrichs-Lewy (CFL) time steps in solving hyperbolic and parabolic partial differential equations, respectively. Numerical benchmarking experiments are conducted to validate the proposed optimal Runge-Kutta schemes.

1 Introduction

As a dominant approach for solving partial differential equations, the finite element method can also be used for solving ordinary differential equations in time. Early developments on variational formulation for time integration have been introduced by Argyris and Scharpf [1], and Fried [16] in 1969. Since then, various finite element methods in time have been proposed

in the literature and the majority of recent approaches are based on the discontinuous Galerkin (DG) finite element method. The first DG method was introduced by Reed and Hill [32] in 1973 as a technique to solve neutron transport equation. The first error analysis of the DG method was carried out in 1974 by LeSaint and Raviart [28]. In their work [28], the first application of the DG method for time stepping has also been considered and the connection between the DG time method and certain Runge-Kutta schemes of Gauss-Radau type has been established. Comparing with the usual finite element method, the DG method has a compact formulation, i.e., the solution within each element is weakly connected to neighboring elements. Such a feature is of the same spirit of the one-step time-stepping methods for solving ordinary differential equations. This perhaps explains the popularity of the DG methods in time.

A discontinuous approximation is usually assumed at time steps t_n in the DG time discretizations. A α-averaging DG method for time stepping was proposed by Delfour, Hager and Trochu [10]. In their DG method, the approximated solution U at t_n is taken as an average across the jump: $\alpha_n U\left(t_n^-\right) + (1-\alpha_n) U\left(t_n^+\right)$. By choosing special α values, one can obtain the original DG scheme of LeSaint and Raviart [28] and Euler's explicit, improved, and implicit schemes. The use of Gauss quadratures is discussed and implicit Runge-Kutta schemes of Gauss-Legendre, Gauss-Radau, and Gauss-Lobatto types are derived [10]. A more general framework was introduced by Delfour and Dubeau [11]. In such a DG variational formulation, one-step methods such as implicit Runge-Kutta and Crank-Nicholson schemes, multistep methods, such as Adams-Bashforth and Adams-Moulton schemes, and hybrid methods can be attained. A priori error estimates for a class of implicit one-step methods generated by the DG time discretization were given by Johnson [23]. Quasi-optimal a priori error bound and posterior error bound of the DG time stepping were studied by Estep [14]. Based on error analysis, both Johnson [23] and Estep [14] considered automatic step-size control and global error control. A finite element variational approach has been suggested for solving time ordinary differential equations via the application of Hamilton's law for dynamics [4]. Nevertheless, such a method is essentially equivalent to the DG time scheme as noticed in [4].

The derivation of implicit Runge-Kutta methods based on the DG formulation has been considered by many authors. In general, the DG approximations are not equivalent to any standard difference scheme. Only after some appropriate quadratures are chosen, the Runge-Kutta schemes can be derived based on the DG approximation. Such a connection has been initially explored in the classical work [28, 10, 11] and has been further studied in the recent work [5, 15]. In Ref. [5], discontinuous approximations are permitted at either t_n or both ends for each time interval $[t_n, t_{n+1}]$. The corresponding DG methods are referred to as singly discontinuous and bi-discontinuous DG approaches. For both DG approaches, after a quadrature is chosen, shortcut expressions for computing the resulting Runge-Kutta schemes are derived by Bottasso [5]. Moreover, by selecting the Gauss quadrature in the bi-discontinuous DG approach, the classical Gauss-Legendre Runge-Kutta schemes can be produced. Such schemes are known to be symplectic when applied to the Hamiltonian problems. Estep and Stuart considered two types of quadratures to the DG formulation [15]. The regular one that approximates the integral is referred to as an external quadrature. They also discussed the internal quadrature, in which a polynomial approximation is introduced to the integrand and one can integrate the integral consequently. Runge-Kutta schemes obtained from both quadratures are considered. Moreover, these authors showed that the choice of quadrature can have a strong effect on the dissipativity properties of the resulting difference schemes and discussed how to

preserve desired dissipativity properties [15].

Various DG methods of different scopes have been proposed for time integration. Although each DG method may be able to generate only a few time stepping schemes, the combination of different DG methods has led to the derivations of most standard time integration schemes in numerical analysis, including explicit Euler, implicit Euler, improved Euler, Crank-Nicholson, leapfrog, Guass-Legendre Runge-Kutta, Guass-Radau Runge-Kutta, Gauss-Lobatto Runge-Kutta, symplectic Runge-Kutta, Adams-Bashforth and Adams-Moulton multistep methods, and hybrid multistep-multistage methods. Nevertheless, we note that all explicit Runge-Kutta schemes do not fit into any existing DG formulation. Clearly, there is a pressing need to have a unified framework that is able to render standard time integration schemes that have been derived by different DG formulations. The construction of such a unified framework enables a better understanding of the structure and capability of DG methods. Academically, it is more attractive to endow a DG framework with the ability to derive standard explicit Runge-Kutta schemes that have not been derived by other existing DG approaches. Such a new capability is certainly a progress in the enrichment of the DG formalism. Finially, what would be the most desirable DG framework is the one that is empowered with the creation of new time integration schemes.

Generally speaking, the construction of new time integration schemes is still an active topic of the numerical analysis, even though many standard procedures are available. There is a constant drive from real applications to develop new schemes with optimized features in either accuracy, stability, or efficiency. For example, it is well known that the generation of ultra high order Runge-Kutta methods is algebraically formidable, because the number of order conditions increases exponentially. Deferred correction Runge-Kutta methods have been developed to bypass this difficulty [13, 8, 9]. Arbitrarily high order Runge-Kutta methods can be efficiently generated with exact coefficients via a deferred correction procedure which involves an integral form of the error equation [8, 9]. Instead of pushing the algebraic orders, an alternative approach to optimize the accuracy is to minimize the dispersive and dissipative errors in the Fourier space for low wavenumbers [39, 29, 3, 36]. However, Mead and Renaut noted that the maximizations of orders of dissipation and dispersion are competing processes, i.e., maximizing one may minimize the other [29].

Additionally, the optimization of time integration stability is another major theme in developing new time discretization methods. Based on eigenvalue analysis, the optimization in the complex domain to extend the stability region has been advanced by many researchers [38, 7, 31, 29, 37]. The resulting optimized methods usually permit a larger Courant-Friedrichs-Lewy (CFL) number for certain problems so that the time integration can be more efficient. Another stability concern is the preservation of the symplectic structure of a Hamiltonian system in the time discretization. Being able to preserve the canonical or symplectic map of the discretized phase flow after a long time integration, the symplectic time integrators [33, 27, 34, 35, 30] are commonly used for Hamiltonian systems, dynamical dynamics and other systems that are integrable in the Liouville sense. A closely related approach in solving partial differential equations (PDEs) is the time splitting method [40] which is frequently employed as a means to construct higher order symplectic integrators in both spatial and temporal domains. In solving the Maxwell's equations with discontinuous coefficients or material interfaces, the temporal stability is often controlled by the spatial discretization as analyzed in an earlier work [41]. In solving PDEs with discontinuous solutions, a stronger measure of

stability is usually required. For instance, a class of strong stability preserving (SSP) time discretization schemes [18, 19, 20, 25] have been developed for the time integration of semi-discretizations of hyperbolic conservation law equations. The SSP schemes are also known as the total variation diminishing (TVD), contractivity preserving, or monotonicity preserving methods. These schemes preserve any convex functional bound (such as positivity) as the forward Euler scheme does. However, the forward Euler scheme is of first-order in accuracy, while SSP schemes are usually higher-order accurate numerical methods. When solving the elliptic and parabolic PDEs by the finite difference method, the stability constrain may be removed or loosen by using the alternating-direction implicit (ADI) type methods [12, 2].

Finnaly, there are other issues in developing optimized time integration schemes, such as low storage requirement and sparseness. In discretizations of PDEs by the method of lines, fast memory for temporary storage is often a limiting factor of computation. This motivates the development of various minimum storage methods since 1950 [17]. The reader is referred to some major achievements in the last decade [24, 6, 26] and references therein for this field. In constructing multistage time integration methods, when possible, we prefer to a choice such that the resulting coefficient matrix being as sparse as possible. This is somehow related to the compressive sensing problem in signal/imaging processing. We will further explore the sparseness issue in the present paper.

The objective of the present paper is threefold. First, we introduce a discontinuous Galerkin (DG) procedure as a unified framework to reconstruct all standard time stepping schemes that have been derived by a combinatiion of many DG approaches. Additionally, we utilize the unified framework to reproduce many standard explicit Runge-Kutta methods that have not been derived by any existing DG approach. Finally, based on the proposed DG framework, we further explore its utility of constructing new time integration schemes. Various optimization criteria are considered for this purpose. Benchmark numerical tests are carried out to validate the newly developed time stepping schemes.

The rest of this paper is organized as the follows. In Section 2, the mathematical formulation of the DG time discretization is laid out and the controllable modeling features of the proposed DG formulation are elaborated. Extensive examples are considered in Section 3 to demonstrate that the proposed DG approach offers a unified framework for deriving all classical one-step, multistep, and multistage time integration schemes. In Section 4, we show for the first time how to derive explicit Runge-Kutta methods based on the DG variational analysis. Meanwhile, novel explicit Runge-Kutta schemes of various stages are developed. Section 5 is devoted to the generation of novel symplectic Runge-Kutta methods. In Section 6, the optimization of explicit DG based Runge-Kutta methods is carried out. Both three stage and fourth stage Runge-Kutta methods are examined. Different criteria such as accuracy, sparseness, and stability are investigated to develop some new optimized Runge-Kutta schemes. These schemes are validated numerically in Section 7. Finally, this paper ends with some concluding remarks.

2 A novel discontinuous Galerkin approach

In this section, we develop a novel discontinuous Galerkin approach for solving the following initial value problem of a nonlinear differential equation

$$\begin{cases} \dot{u}(t) &= f(u(t),t), \quad 0 < t \le T, \\ u(0) &= u_0. \end{cases} \tag{1}$$

Consider a partition of a closed interval $[0,T]$: $0 = t_0 < t_1 < \cdots < t_N = T$. Let denote I_n an open interval $I_n := (t_n, t_{n+1})$ and h_n its length $h_n := t_{n+1} - t_n$.

2.1 Discontinuous Galerkin variational formulation

We first establish a general discontinuous Galerkin (DG) finite element formulation for (1). Denote $P^{(q)}(I_n)$ the space of polynomials of degree q or less on the interval I_n. We define the finite element space containing the piecewise polynomials to be $\mathbb{U} = \{U : U|_{I_n} \in P^{(q)}(I_n)\}$. In the DG method, the trial function or finite element solution $U(t) \in \mathbb{U}$ is continuous within each time element I_n, but can be discontinuous across the interface of the elements, i.e., at time instants $t_1, \ldots, t_{N-1}$. Denote the restriction of $U(t)$ as $U_n(t) = U|_{I_n}$ for $n = 0, 1, \ldots, N-1$, which is continuous in time element $I_n = (t_n, t_{n+1})$. At each node t_n, the limiting values of finite element approximations from the left $(U_{n-1}(t_n^+))$ and the right $(U_n(t_n^-))$ are usually different. The jump is denoted as $[U]_n = U_n(t_n^+) - U_{n-1}(t_n^-)$.

The problem of global variational formulation of (1) is given as: find $U \in \mathbb{U}$ and N trace values $u^n \in R$ such that

$$\begin{cases} \int_0^T \dot{U}(t)w(t)\,dt = \int_0^T f(U(t),t)w(t)\,dt, \\ U_0(t_0^+) = u^0 = u_0 \text{ and some additional conditions on } U_n(t) \end{cases} \tag{2}$$

for all $w(t) \in \mathbb{U}$ and $n = 1, \ldots, N$. Since $U(t)$ is piecewisely continuous, its time derivative at a point of discontinuity is an appropriately scaled delta function. Thus U solves the global problem

$$\sum_{n=0}^{N-1} \int_{I_n} \dot{U}(t)w(t)\,dt + \sum_{n=0}^{N-1} [U]_{n+1} w(t_{n+1}) = \sum_{n=0}^{N-1} \int_{I_n} f(U(t),t)w(t)\,dt \tag{3}$$

for all $w(t) \in \mathbb{U}$. In practice, U can be computed locally for each interval

$$\int_{I_n} \dot{U}_n(t)w(t)\,dt + U_{n+1}(t_{n+1}^+)w(t_{n+1}) = U_n(t_{n+1}^-)w(t_{n+1}) + \int_{I_n} f(U_n(t),t)w(t)\,dt \tag{4}$$

for $n = 0, 1, 2, \ldots$. Moreover, $U_n(t)$ should satisfy some additional conditions that relate time node values $U_n(t_i)$ with nearby trace values u^i, $i = n, n \pm 1, n \pm 2, \ldots$. These conditions also are called as boundary conditions for $U_n(t)$, and we assume the total number of such conditions being L. Finally, by using the integration by parts, we have the following local variational problem: to find $U_n(t) \in P^{(q)}(I_n)$ such that

$$\begin{cases} U_{n+1}(t_{n+1}^+)w(t_{n+1}) = U_n(t_n^+)w(t_n) + \int_{I_n} U_n(t)\dot{w}(t)\,dt + \int_{I_n} f(U_n(t),t)w(t)\,dt \\ L \text{ additional conditions on } U_n(t) \end{cases} \tag{5}$$

for all $w(t) \in P^{(q)}(I_n)$ and $n = 1, \ldots, N$.

2.2 Three controllable components in the DG formulation

There are three controllable components in the DG formulation (5). First, appropriate L additional conditions need to be imposed for $U_n(t)$ in (5). Second, a numerical quadrature is required to discretize the integrals involved in (5). Finally, a set of test functions $w(t)$ should to be specified in the practical computation of (5). By considering different choices of these components, one derives different time stepping schemes. In the following, we will first make some comments on each component's role in the DG formulation. We then present our choices of these components for the proposed DG framework. These special choices facilitate our derivation of explicit Runge-Kutta methods.

- **L boundary conditions**

  We note that depending on a different choice of L additional conditions, one attains different families of approximation schemes. When $L = 0$, we obtain the so-called completely discontinuous DG method [11] or doubly discontinuous DG method [5], i.e., the trace values at two ends of I_n are distinct from the values assumed by $U_n(t)$ at t_n and t_{n+1}. Most DG methods in time usually employ $L = 1$. In such a case, while an obvious choice is to set u^n to be either $U_{n-1}(t_n^-)$ or $U_n(t_n^+)$, an α-weighted scheme with $u^n = \alpha_n U_{n-1}(t_n^-) + (1-\alpha_n)U_n(t_n^+)$ has also been considered [10]. If one enforces two boundary conditions of the form: $U_n(t_n) = u^n$ and $U_n(t_{n+1}) = u^{n+1}$, one actually deals with a continuous finite element method instead of a DG method, while $L > 2$ is essential for deriving multistep methods [11].

  In the proposed DG framework, our default choice regarding to L conditions is as the follows: we take $L = 1$ in Eq. (5) and specify one additional condition for $U_n(t)$. In particular, we set $U_n(t_n) = u^n$ but leave value of $U_n(t_{n+1})$ open so that in general $U_n(t_{n+1}) \neq u^{n+1}$. In other words, we enforce initial conditions rigorously and permit a jump "in the future value". Nevertheless, we note that $U_{n+1}(t_{n+1}) = u^{n+1}$ is the initial condition for the next time element. Consequently, the local variational equation further reduces to

$$u^{n+1}w(t_{n+1}) = u^n w(t_n) + \int_{I_n} U_n(t)\dot{w}(t)\,dt + \int_{I_n} f(U_n(t),t)w(t)\,dt \tag{6}$$

  for all $w(t) \in P^{(q)}(I_n)$ and $n = 1, \ldots, N$.

- **Numerical quadrature**

  In order to convert the DG time integration equation (6) into a commonly used time-advancing scheme, which is typically of collocation nature, a numerical quadrature is essential. When $f(u,t)$ is linear and simple, Eq. (6) may be evaluated analytically after specifying particular expansion basis for both trial function $U_n(t)$ and test function $w(t)$. However, for general applications, we have $f(u,t)$ being a nonlinear function. Thus, the last integral of Eq. (6), i.e., $\int_{I_n} f(U(t),t)w(t)\,dt$, can not be integrated analytically in practice. Consequently, numerical approximation involving quadratures has to be introduced at this point as in the literature. Both internal and external quadratures have been considered by Estep and Stuart [15] to this end.

  In the present study, we employ a regular external quadrature for both integrals in Eq. (6). We often utilize Newton-Cotes quadrature rules generated via the Lagrange

polynomials. To illustrate the idea, we consider here a quadrature rule with s abscissae c_j and weights b_j

$$\int_{I_n} g(t)\,dt \approx h_n \sum_{j=1}^{s} b_j g(t_n + c_j h_n). \tag{7}$$

Without the loss of generality, we assume $b_j \neq 0$ for $j = 1, 2, \ldots, s$ in the quadrature rule. By using such a quadrature rule, Eq. (6) is approximated to be

$$\begin{aligned} u^{n+1}w(t_{n+1}) &= u^n w(t_n) + h_n \sum_{j=1}^{s} b_j U_n(t_n + c_j h_n)\dot{w}(t_n + c_j h_n) \\ &\quad + h_n \sum_{j=1}^{s} b_j f(U_n(t_n + c_j h_n), t_n + c_j h_n) w(t_n + c_j h_n) \end{aligned} \tag{8}$$

To simplify notation, we denote $U^j := U_n(t_n + c_j h_n)$. This gives rise to

$$u^{n+1}w(t_{n+1}) = u^n w(t_n) + h_n \sum_{j=1}^{s} b_j U^j \dot{w}(t_n + c_j h_n) + h_n \sum_{j=1}^{s} b_j f(U^j, t_n + c_j h_n) w(t_n + c_j h_n) \tag{9}$$

- **Test function**

  It is obvious that by selecting a different test function $w(t)$ in Eq. (9), a different time-stepping scheme can be derived. Thus, the test function $w(t)$ plays a important role in the proposed DG variational approach for time integration. As the guideline for calculating $w(t)$, we will propose various different conditions for $w(t)$ so that the desired new schemes can be attained. A particular example for Runge-Kutta methods is discussed in the next subsection.

2.3 Runge-Kutta methods and their DG formulation

We present in this subsection a general procedure for deriving Runge-Kutta methods based on the proposed DG framework through appropriately choosing the L boundary conditions, numerical quadrature, and test functions. In numerical analysis, a general s-stage Runge-Kutta method can be given by

$$u^{n+1} = u^n + h_n \sum_{j=1}^{s} b_j f(y^j, t_n + c_j h_n), \tag{10}$$

$$y^j = u^n + h_n \sum_{k=1}^{s} a_{jk} f(y^k, t_n + c_k h_n), \quad k = 1, \ldots, s, \tag{11}$$

where y^i are internal stages. It has become customary to represent the free parameters of the Runge-Kutta method by using a Butcher tableau, consisting of an $s \times s$ matrix $\mathbf{A}$ and two $s \times 1$ vectors $\mathbf{b}$ and $\mathbf{c}$

$$\begin{array}{c|ccc} c_1 & a_{11} & \ldots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \ldots & a_{ss} \\ \hline & b_1 & \ldots & b_s \end{array}$$

Here b_i and c_i are the weights and abscissae of the method, respectively. A Runge-Kutta method is said to be irreducible [22], if $b_j \neq 0$, $j = 1, \ldots, s$. We assume the irreducibility in this work.

We choose the three controllable components of the proposed DG framework as the follows: First, we make use of the default choice of boundary conditions, i.e., $L = 1$ and $U_n(t_n) = u^n$. Second, the choice of numerical quadrature is very natural. To derive a targeted Runge-Kutta method, one can simply take c_i and b_i given in the Butcher tableau as the numerical quadrature rule. Alternatively, if one desires to derive a unknown Runge-Kutta method, a Newton-Cotes quadrature rule could be utilized. Finally, since the Runge-Kutta methods are multistage time integration schemes, a different test function $w(t)$ has to be chosen in different stage so that the different updating equation can be attained. In other words, multiple test functions shall be employed in deriving one-step multistage methods.

We propose the following rules in selecting test function $w(t)$ so that a Runge-Kutta method can be constructed. The trivial choice $w(t) = 1$ will always be assumed. In particular, by taking $w(t) = 1$ in Eq. (9), we have

$$u^{n+1} = u^n + h_n \sum_{j=1}^{s} b_j f(U^j, t_n + c_j h_n), \tag{12}$$

which obviously becomes (10) provided $U^i = y^i$. Moreover, in order to determine U^i in the present DG method, we shall consider s independent test functions $w_i(t)$, $i = 1, \ldots, s$, for Eq. (9). Furthermore, for simplicity, we assume throughout $w_i(t_{n+1}) = 0$. This gives rise to s algebraic equations

$$u^n w_i(t_n) + h_n \sum_{j=1}^{s} b_j U^j \dot{w}_i(t_n + c_j h_n) + h_n \sum_{j=1}^{s} b_j f(U^j, t_n + c_j h_n) w_i(t_n + c_j h_n) = 0, \quad i = 1, \ldots, s. \tag{13}$$

By symbolically solving U^i from these equations, we have

$$U^i = \alpha_i u^n + h_n \sum_{j=1}^{s} \beta_{ij} f(U^j, t_n + c_j h_n), \quad i = 1, \ldots, s, \tag{14}$$

where the coefficients α_i and β_{ij} depend on quadrature and time node values of $w_i(t)$ and $\dot{w}_i(t)$. As in the Runge-Kutta literature, we refer two conditions

$$\alpha_i = 1, \quad \text{for } i = 1, \ldots s \tag{15}$$

$$\sum_{j=1}^{s} \beta_{ij} = c_i, \quad \text{for } i = 1, \ldots s \tag{16}$$

as the consistent conditions.

Remark If two consistent conditions (15) and (16) are satisfied in the proposed DG framework, Eqs. (12) and (14) represent a novel Runge-Kutta method. We note that in general such a DG deduced Runge-Kutta method is different from the Runge-Kutta method given by Eqs. (11) and (10), even though they have the same weights b_i and abscissae c_i. However, under a special choice of $w_i(t)$ with $i = 1, \ldots, s$, such two Runge-Kutta methods could be identical.

3 A unified framework

In this section, we illustrate the unified feature of the proposed DG formulation. Various classical one-step, multistep and multistage schemes will be studied.

3.1 Euler and Crank-Nicholson schemes

Euler and Crank-Nicholson schemes can be derived by using a simple choice of $w(t) = 1$ in the proposed DG formulation. With $w(t) = 1$, Eq. (9) reduces to

$$u^{n+1} = u^n + h_n \sum_{j=1}^{s} b_j f(U^j, t_n + c_j h_n). \tag{17}$$

By using a one-point support quadrature with $c_1 = 0$ and $b_1 = 1$ in Eq. (17), we have particularly

$$u^{n+1} = u^n + h_n f(U^1, t_n), \tag{18}$$

where $U^1 = U_n(t_n + c_1 h_n) = U_n(t_n) = u^n$. This gives rise to the explicit Euler scheme

$$u^{n+1} = u^n + h_n f(u_n, t_n). \tag{19}$$

By considering different boundary conditions, the implicit Euler scheme,

$$u^{n+1} = u^n + h_n f(u_{n+1}, t_{n+1}), \tag{20}$$

and the Crank-Nicholson scheme

$$u^{n+1} = u^n + \frac{h_n}{2} f(u^n, t_n) + \frac{h_n}{2} f(u^{n+1}, t_{n+1}), \tag{21}$$

can be similarly derived from Eq. (17). See Appendix A.1 for more details.

3.2 Multistep and hybrid schemes

Delfour and Dubeau [11] proposed a general DG framework for deriving multistep and hybrid methods. These schemes can be derived based on the present DG model as well, after adopting two extensions as in the original work [11], i.e., additional conditions with $L > 2$ and non-standard quadrature with some abscissae c_j outside of the integral interval $(0, 1)$.

For instance, let us consider the Adams-Moulton scheme of order 3

$$u^{n+1} = u^n + \frac{h}{12}[-f(u^{n-1}, t_{n-1}) + 8f(u^n, t_n) + 5f(u^{n+1}, t_{n+1})], \tag{22}$$

where for simplicity, we take $h = h_{n-1} = h_n$. Since this scheme refers to time node t_{n-1}, the abscissae are chosen as $c_1 = -1$, $c_2 = 0$, and $c_3 = 1$. Computational formulas of corresponding weights b_j are given in the appendix of Ref [11]: $b_1 = -\frac{1}{12}$, $b_2 = \frac{2}{3}$, and $b_3 = \frac{5}{12}$. This is actually a generalized Newton-Cotes rule. By taking $L = 3$, additional conditions for $U_n(t)$ are imposed as $U_n(t_{n-1}) = u^{n-1} = U^1$, $U_n(t_n) = u^n = U^2$, and $U_n(t_{n+1}) = u^{n+1} = U^3$. Again, by taking $w(t) = 1$, one obtains Eq. (17), which further reduces to the Adams-Moulton scheme (22). Other Adams-Moulton schemes, Adams-Bashforth schemes, and hybrid schemes can be similarly derived by taking $w(t) = 1$.

3.3 Leapfrog scheme

For the time stepping schemes derived so far in this section, only two components of our DG framework, i.e., L additional constraints and numerical quadrature, play the essential role in the derivation. The third component is fixed to be trivially $w(t) = 1$. Here, we demonstrate a case that requires the use of a nontrivial test function $w(t)$.

Consider the leapfrog multistep scheme

$$u^{n+1} = u^{n-1} + 2hf(u^n, t_n). \tag{23}$$

Similar to the Adams-Moulton scheme of order 3, the quadrature shall be naturally chosen as $(c_1, c_2, c_3) = (-1, 0, 1)$ and $(b_1, b_2, b_3) = \left(-\frac{1}{12}, \frac{2}{3}, \frac{5}{12}\right)$. Also, additional conditions with $L = 3$ are $U_n(t_{n-1}) = u^{n-1} = U^1$, $U_n(t_n) = u^n = U^2$, and $U_n(t_{n+1}) = u^{n+1} = U^3$. Now, Eq. (9) takes a particular form

$$\begin{aligned} u^{n+1}w(t_{n+1}) &= u^n w(t_n) - \frac{h}{12}U^1\dot{w}(t_{n-1}) + \frac{2h}{3}U^2\dot{w}(t_n) + \frac{5h}{12}U^3\dot{w}(t_{n+1}) \\ &\quad -\frac{h}{12}f(U^1, t_{n-1})w(t_{n-1}) + \frac{2h}{3}f(U^2, t_n)w(t_n) + \frac{5h}{12}f(U^3, t_{n+1})w(t_{n+1}). \end{aligned} \tag{24}$$

We then set the test function to be

$$w(t) = -\frac{27}{10}\frac{(t-t_n)^5}{h^5} + \frac{9}{5}\frac{(t-t_n)^4}{h^4} + \frac{21}{5}\frac{(t-t_n)^3}{h^3} - \frac{14}{5}\frac{(t-t_n)^2}{h^2} - \frac{3}{2}\frac{(t-t_n)}{h} + 1$$

In fact, by noting that on time nodes, $w(t_{n-1}) = 0$, $w(t_n) = 1$, $w(t_{n+1}) = 0$, $\dot{w}(t_{n-1}) = -\frac{4}{h}$, $\dot{w}(t_n) = -\frac{3}{2h}$, and $\dot{w}(t_{n+1}) = -\frac{4}{5h}$, it is easy to see that Eq. (24) becomes Eq. (23).

3.4 Special Runge-Kutta schemes

We consider the derivation of the midpoint rule

$$u^{n+1} = u^n + h_n f\left(\frac{u^{n+1} + u^n}{2}, t_n + \frac{h_n}{2}\right), \tag{25}$$

and the improved Euler scheme or the predictor-corrector method

$$u^{n+1} = u^n + \frac{h_n}{2}\left[f(u^n, t_n) + f(u^n + h_n f(u^n, t_n), t_{n+1})\right]. \tag{26}$$

Instead of the compact form, both schemes can be rewritten as Runge-Kutta type methods. For example, the midpoint rule is actually an implicit Runge-Kutta method with one stage

$$y^1 = u^n + \frac{h_n}{2}f\left(y^1, t_n + \frac{h_n}{2}\right), \tag{27}$$

$$u^{n+1} = u^n + h_n f\left(y^1, t_n + \frac{h_n}{2}\right). \tag{28}$$

Thus, both schemes can be derived based on the guidelines given in Section. 2.3. See Appendix A.2 for more details.

4 Developing new explicit Runge-Kutta methods

In this section, we will construct some new explicit Runge-Kutta schemes based on the proposed DG framework. We note that no explicit Runge-Kutta scheme has been derived based on discontinuous Galerkin finite element approach in the literature.

Remark Explicit Runge-Kutta methods can be obtained in the present DG framework if the consistent conditions (15) and (16) and the following conditions are satisfied

$$w_i(t_{n+1}) = 0, \quad w_i(t_n + c_j h_n) = 0, \quad \text{for} \quad j = i, \ldots, s \tag{29}$$

$$\dot{w}_i(t_n + c_j h_n) = 0, \quad \text{for} \quad j = i+1, \ldots, s. \tag{30}$$

Moreover, the determination of $w_i(t)$ satisfying (15), (16), (29) and (30) is not mathematically ill-posed.

In fact, by substituting explicit conditions (29) and (30) into Eq. (13), one can guarantee that U^i given in (14) only depends on the previously computed values so that the resulting Runge-Kutta scheme is an explicit one. The proposed procedure is not mathematically ill-posed. When $i = 1$, a concern is that the number of unknowns might be smaller than the number of conditions. We study this issue by considering two cases: $c_1 = 0$ and $c_1 \neq 0$. When $c_1 = 0$ in the abscissae, we actually do not need to impose any condition, but simply have that $U^1 = U_n(t_n + c_1 h_n) = U_n(t_n) = u^n$. If $c_1 \neq 0$, the test function $w_1(t)$ satisfying Eqs. (29) and (30) can still allow that $w_1(t_n) \neq 0$. Thus, Eq. (13) becomes

$$u^n w_1(t_n) + h_n b_1 U^1 \dot{w}_1(t_n + c_1 h_n) = 0, \tag{31}$$

The values of $w_1(t_n)$ and $\dot{w}_1(t_n + c_1 h_n)$ can be uniquely determined by consistent conditions (15) and (16). Consequently, we actually also have $U^1 = u^n$. When $i = 2$, there are three degree of freedom remaining after explicit conditions (29) and (30) being satisfied. Thus, the scheme construction is well-posed when consistent conditions (15) and (16) are imposed. When $i > 2$, one actually has far more unknowns than the conditions.

A systematic procedure is employed in the present study to derive explicit Runge-Kutta schemes. For $i \geq 2$, we propose to further reduce the degree of freedom by assuming that nodal values of $w_i(t)$ to be a kronecker delta, i.e., instead of explicit condition (29), we impose a stronger explicit condition

$$w_i(t_{n+1}) = 0, \quad w_i(t_n + c_{i-1} h_n) = 1, \quad \text{and} \quad w_i(t_n + c_j h_n) = 0, \ \text{ if } \ j \neq i-1. \tag{32}$$

In the following, we refer conditions (30) and (32) as the explicit conditions.

Remark Novel explicit Runge-Kutta methods will be developed in the proposed DG framework by imposing consistent conditions (15) and (16) and explicit conditions (30) and (32) for test function $w_i(t)$.

4.1 Two stage explicit Runge-Kutta methods

We consider two typical numerical quadratures with two points support:

$$\text{RK2-1:} \quad (c_1, c_2, b_1, b_2) = (0, 1, \frac{1}{2}, \frac{1}{2}), \qquad \text{RK2-2:} \quad (c_1, c_2, b_1, b_2) = (0, \frac{2}{3}, \frac{1}{4}, \frac{3}{4}),$$

Based on these quadratures, explicit Runge-Kutta methods can be uniquely generated by enforcing conditions (15), (16), (30) and (32).

We consider the scheme RK2-1 here. We have first $U^1 = u^n$ since $c_1 = 0$. We then consider determine $w(t) = w_2(t)$ for U^2. From the conditions, we have $w(t_n) = 1$ and $w(t_{n+1}) = 0$ due to (32) and $\dot{w}(t_n) = -\frac{1}{h_n}$ and $\dot{w}(t_{n+1}) = -\frac{1}{h_n}$ due to (15) and (16). This uniquely determines the test function $w(t) = -\frac{1}{h_n}(t - t_n) + 1$. By substituting such a $w(t)$ into Eq. (9), one attains

$$0 = u^n + h_n \cdot \left(-\frac{1}{h_n}\right) \cdot \left(\frac{1}{2}U^1 + \frac{1}{2}U^2\right) + h_n \frac{1}{2} f\left(U^1, t_n\right).$$

By noting that $U^1 = u^n$, this gives rise to

$$U^2 = u^n + h_n f(U^1, t_n)$$

With U^1 and U^2, u^{n+1} is updated according to Eq. (12)

$$u^{n+1} = u^n + \frac{h_n}{2} f(U^1, t_n) + \frac{h_n}{2} f(U^2, t_{n+1}).$$

We thus obtain a two stage explicit Runge-Kutta scheme, i.e., Scheme RK2-1, whose Butcher tableau is given below.

Scheme RK2-1:

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

Scheme RK2-2:

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

Based on the other quadrature rule, the Scheme RK2-2 can be constructed, see Appendix A.3 for more details. These two schemes are some well known second order Runge-Kutta methods [21].

4.2 Three stage explicit Runge-Kutta method

We next derive a novel three stage explicit Runge-Kutta method based on the Simpson's rule $(c_1, c_2, c_3) = (0, \frac{1}{2}, 1)$ and $(b_1, b_2, b_3) = (\frac{1}{6}, \frac{2}{3}, \frac{1}{6})$. The enforcement of conditions (15), (16), (30) and (32) will not uniquely determine the Runge-Kutta coefficients so that one free parameter C ($C \neq 0$) will be presented in the final scheme (see Appendix A.4 for more details)

Scheme RK3:

0			
$\frac{1}{2}$	$\frac{1}{2}$		
1	$\frac{C-4}{C}$	$\frac{4}{C}$	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

Remark The classical third order Runge-Kutta scheme [21] is a special case of the propose explicit RK3 scheme by taking $C = 2$. On the other hand, we note that the proposed RK3 scheme can be greatly simplified if we take $C = 4$. Other novel three stage Runge-Kutta methods with optimal C values will be considered in Section 6.

4.3 Four stage explicit Runge-Kutta methods

As one of the most commonly used four stage Runge-Kutta methods, the classical fourth order Runge-Kutta method employs a non-standard quadrature [21], i.e., with the abscissae $(c_1, c_2, c_3, c_4) = (0, \frac{1}{2}, \frac{1}{2}, 1)$ and weights $(b_1, b_2, b_3, b_4) = (\frac{1}{6}, \frac{1}{3}, \frac{1}{3}, \frac{1}{6})$. We consider the derivation of novel four stage Runge-Kutta methods based on such a quadrature.

A modified DG approach has to be utilized, because two collocation nodes involved in the present quadrature are identical. To overcome this difficulty, we introduce an internal discontinuity in $U(t)$ at time instant $t_{n+\frac{1}{2}}$. Consequently, the integrand is discontinuous at the same place. Taking $g(t)$ as the general integrand to illustrate the idea, the integral should be carried out piecewisely

$$\int_{t_n}^{t_{n+1}} g(t)\,dt = \int_{t_n}^{t_{n+\frac{1}{2}}} g(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} g(t)\,dt. \tag{33}$$

Now, the present abscissae and weights can be regarded as composite quadrature rule for the piecewise integrals:

$$\int_{t_n}^{t_{n+\frac{1}{2}}} g(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} g(t)\,dt \approx \frac{h_n}{6} g(t_n) + \frac{h_n}{3} g\left(t^-_{n+\frac{1}{2}}\right) + \frac{h_n}{3} g\left(t^+_{n+\frac{1}{2}}\right) + \frac{h_n}{6} g(t_{n+1}). \tag{34}$$

With an internal discontinuity, the DG framework should also be modified. Fortunately, the DG method is a very flexible variational formulation so that this change can be easily handled. In particular, Eq. (4) now becomes

$$\begin{aligned} &\int_{t_n}^{t_{n+\frac{1}{2}}} \dot{U}(t) w^-(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} \dot{U}(t) w^+(t)\,dt + U_{n+1}(t^+_{n+1}) w^+(t_{n+1}) + U_n\left(t^+_{n+\frac{1}{2}}\right) w^+\left(t_{n+\frac{1}{2}}\right) \qquad (35)\\ = &\int_{t_n}^{t_{n+\frac{1}{2}}} f(U(t),t) w^-(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} f(U(t),t) w^+(t)\,dt + U_n(t^-_{n+1}) w^+(t_{n+1}) + U_n\left(t^-_{n+\frac{1}{2}}\right) w^-\left(t_{n+\frac{1}{2}}\right), \end{aligned}$$

where a piecewise definition is assumed in this subsection

$$w(t) = \begin{cases} w^-(t), & \text{if} \quad t \in \left(t_n, t_{n+\frac{1}{2}}\right) \\ w^+(t), & \text{if} \quad t \in \left(t_{n+\frac{1}{2}}, t_{n+1}\right). \end{cases} \tag{36}$$

Similar to the original DG method, after applying the integration by parts, Eq. (6) is modified to be

$$\begin{aligned} u^{n+1} w^+(t_{n+1}) &= u^n w^-(t_n) + \int_{t_n}^{t_{n+\frac{1}{2}}} U(t) \dot{w}^-(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} U(t) \dot{w}^+(t)\,dt \qquad (37)\\ &\quad + \int_{t_n}^{t_{n+\frac{1}{2}}} f(U(t),t) w^-(t)\,dt + \int_{t_{n+\frac{1}{2}}}^{t_{n+1}} f(U(t),t) w^+(t)\,dt. \end{aligned}$$

Also, we still use the notation $U^j = U(t_n + c_j h_n)$. Consequently, $U^1 = u^n$, $U^2 = U\left(t^-_{n+\frac{1}{2}}\right)$, $U^3 = U\left(t^+_{n+\frac{1}{2}}\right)$, and $U^4 = U(t_{n+1})$. The detailed derivation of U^2, U^3, and U^4 in the proposed Runge-Kutta scheme is given in Appendix A.5.

Remark A new four stage explicit Runge-Kutta method is developed in the present DG framework

Scheme RK4

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & \frac{C_1-2}{2C_1} & \frac{1}{C_1} & & \\ 1 & 1-\frac{2}{C_3}+\frac{2C_2}{C_1C_3} & -\frac{2C_2}{C_1C_3} & \frac{2}{C_3} & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

The RK4 scheme involves three independent parameters. By taking $C_1 = 2$, $C_2 = 0$, and $C_3 = 2$, one attains the classical fourth order Runge-Kutta scheme. Other optimal choices of parameters will be consider in Section 6.

5 Developing new symplectic Runge-Kutta methods

The first symplectic integrator was introduced by Ruth in 1983 for Hamiltonian systems of differential equations [33]. Symplectic schemes are ideally-suited to long time integrations, due to their ability in preserving the canonical or symplectic map of continuous systems in discretizations. Around 1988, it has been discovered independently by several authors [27, 34] that Runge-Kutta schemes could be symplectic if and only if the following condition is satisfied

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0, \quad i, j = 1, \ldots, s, \tag{38}$$

in the Butcher tableau. The classical Gauss-Legendre Runge-Kutta scheme is one example satisfying condition (38) and symplectic Runge-Kutta schemes are all implicit [35]. A complete list of symplectic Runge-Kutta methods with up to 6 stages is presented in [30].

5.1 Derivation procedure

To derive new symplectic Runge-Kutta schemes based on the proposed DG framework, a different set of conditions for test function $w(t)$ should be employed. We first note that the Gaussian quadratures are typically employed to generate symplectic Runge-Kutta methods, in which the starting point t_n and ending point t_{n+1} are skipped in the abscissae. Thus, we assume in this subsection that $w(t_n) = 1$ and $w(t_{n+1}) = 0$ for simplicity. Moreover, the symplectic condition (38) originally implies s^2 constrains for a Butcher matrix A. However, due to symmetry, half of off-diagonal constrains can be dropped in (38). Therefore, in the present DG formulation, only a lower triangular part (including diagonal) of (38) shall be enforced. Finally, we note that explicit conditions (29), (30), or (32) should not be imposed, since symplectic Runge-Kutta schemes are implicit. Instead, we propose to enforce the following implicit condition:

$$\dot{w}(t_n + c_j h_n) = 0, \quad \text{for } j = 1, \ldots, s \quad j \neq i. \tag{39}$$

Remark New symplectic Runge-Kutta methods can be obtained by using the consistence conditions (15), (16), the symplectic condition (38), and the implicit condition (39) in the present DG framework.

5.2 Two stage symplectic Runge-Kutta method

We consider a two-points support Gaussian quadrature with $c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}$, $c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}$, $b_1 = \frac{1}{2}$, and $b_2 = \frac{1}{2}$. With $w(t_n) = 1$ and $w(t_{n+1}) = 0$ being fixed, we can have four undetermined coefficients, i.e., nodal and derivative values of $w(t)$ at $t_n + c_1 h_n$ and $t_n + c_2 h_n$. They can be uniquely determined according to conditions (15), (16), (38), and (39), see Appendix A.6 for more details. The resulting symplectic Runge-Kutta method is a fourth order implicit method [21]

$$\text{Scheme SRK2:} \quad \begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

5.3 Three stage symplectic Runge-Kutta method

We next derive a novel three stage symplectic Runge-Kutta method based on the Gaussian quadrature with $c_1 = \frac{1}{2} - \frac{1}{2}\sqrt{\frac{3}{5}}$, $c_2 = \frac{1}{2}$, $c_3 = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}}$, $b_1 = \frac{5}{18}$, $b_2 = \frac{4}{9}$, and $b_3 = \frac{5}{18}$. The enforcement of conditions (15), (16), (38) and (39) will not uniquely determine the Runge-Kutta coefficients so that one free parameter A will be presented in the proposed scheme (see Appendix A.7 for more details)

$$\text{Scheme SRK3:} \quad \begin{array}{c|ccc} \frac{1}{2} - \frac{1}{2}\sqrt{\frac{3}{5}} & \frac{5}{36} & \frac{4A}{9} & \frac{5}{18}\frac{13-18\sqrt{\frac{3}{5}}-16A}{10} \\ \frac{1}{2} & \frac{5}{18}(1-A) & \frac{2}{9} & \frac{5}{18}A \\ \frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}} & \frac{5}{18}\frac{16A+18\sqrt{\frac{3}{5}}-3}{10} & \frac{4}{9}(1-A) & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18}. \end{array}$$

By taking $A = \frac{1}{2} - \frac{3}{20}\sqrt{15}$, one attains the classical three stage six order symplectic Runge-Kutta scheme [21]. Higher stage symplectic Runge-Kutta schemes can be similarly constructed based on the present DG formulation and are not illustrated further.

6 Optimized Runge-Kutta methods

As presented previously, we have derived two novel explicit Runge-Kutta schemes

$$\text{RK3:} \quad \begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & \frac{C-4}{C} & \frac{4}{C} & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} \tag{40}$$

$$\text{RK4:} \quad \begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & \frac{C_1-2}{2C_1} & \frac{1}{C_1} & & \\ 1 & 1 - \frac{2}{C_3} + \frac{2C_2}{C_1 C_3} & -\frac{2C_2}{C_1 C_3} & \frac{2}{C_3} & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}. \end{array} \tag{41}$$

In this section, we examine the optimization of these two DG deduced Runge-Kutta methods. The development of optimized Runge-Kutta methods has attracted numerous research interests. Three major optimization criteria are studied in the present study, i.e., accuracy, stability and sparseness. The optimization procedures proposed in the present work can be applied to other DG deduced Runge-Kutta methods, and as well as non-DG based Runge-Kutta methods.

6.1 Optimized three stage explicit Runge-Kutta methods

6.1.1 Sparseness

Remark The sparsest RK3 scheme can be attained by taking $C = 4$ in (40).

6.1.2 Accuracy

Without of the loss of generality, we consider a linear system of ordinary differential equations (ODEs) obtained via a semi-discretization process of a partial differential equation (PDE):

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y}, \tag{42}$$

where $\mathbf{A}$ is a $n \times n$ matrix. The general solution to (42) can be written symbolically as

$$\mathbf{y}(t) = \sum_{i=1}^{n} K_i e^{\lambda_i t} \xi_i, \tag{43}$$

where λ_i are the eigenvalues, ξ_i are the corresponding eigenvectors, and K_i are expanding coefficients. It is known that the accuracy and stability of a particular Runge-Kutta scheme applied to (42) can be determined by its approximation to eigenvalues. Therefore, a scalar test equation

$$\dot{u} = \lambda u, \quad u(0) = 1, \quad \lambda \in \mathbb{C}, \tag{44}$$

is commonly used in the literature [29, 8] to investigate the accuracy and stability of Runge-Kutta methods. This is equivalent to take $f(u,t) = \lambda u$ and $u_0 = 1$ in the general ODE (1).

To analyze the accuracy, we consider the integration of (44) from 0 to t by using only one step, i.e., $h = t$. The RK3 scheme with a free parameter C is employed.

$$\begin{aligned}
U^1 &= u^0 \\
U^2 &= u^0 + \frac{h}{2}\lambda u^0 = \left(1 + \frac{\lambda h}{2}\right) u^0 \\
U^3 &= u^0 + \frac{C-4}{C} h\lambda u^0 + \frac{4}{C} h\lambda \left(1 + \frac{\lambda h}{2}\right) u^0 = \left(1 + \lambda h + \frac{2}{C}\lambda^2 h^2\right) u^0 \\
u^1 &= u^0 + \frac{h}{6}\lambda u^0 + \frac{2h}{3}\lambda \left(1 + \frac{\lambda h}{2}\right) u^0 + \frac{h}{6}\lambda \left(1 + \lambda h + \frac{2}{C}\lambda^2 h^2\right) u^0 \\
&= \left(1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{3C}\right) u^0 = 1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{3C}.
\end{aligned}$$

Here the initial condition $u^0 = u_0 = 1$ has been applied.

Following Ref. [13], we analyze the accuracy of the RK3 numerical solution by investigating the error function $e(t) = u(t) - U(t)$, where $u(t)$ is the analytical solution and $U(t)$ is the numerical solution. It is noted that we do not need to know the analytical solution $u(t)$ in the present analysis. Instead, an analytical form of the error function is utilized [8, 9]

$$e'(t) + \epsilon(t) = f(U(t) + e(t), t) - f(U(t), t), \tag{45}$$

where $\epsilon(t) = U'(t) - f(U(t), t)$. In the present study, the numerical solution $U(t)$ of the RK3 scheme can be given as $U(t) = 1 + \lambda t + \frac{\lambda^2 t^2}{2} + \frac{\lambda^3 t^3}{3C}$. With such a $U(t)$, we have

$$\begin{aligned} \epsilon\left(t\right) &= \left(1 + \lambda t + \frac{\lambda^2 t^2}{2} + \frac{\lambda^3 t^3}{3C}\right)' - \lambda\left(1 + \lambda t + \frac{\lambda^2 t^2}{2} + \frac{\lambda^3 t^3}{3C}\right) \\ &= \left(\frac{\lambda^3}{C} - \frac{\lambda^3}{2}\right) t^2 - \frac{\lambda^4}{3C} t^3. \end{aligned}$$

Consequently, Eq. (45) becomes

$$e'\left(t\right) - \lambda e\left(t\right) = \frac{\lambda^4}{3C} t^3 + \left(\frac{\lambda^3}{2} - \frac{\lambda^3}{C}\right) t^2. \tag{46}$$

For the current scalar linear problem, the error equation (46) is analytically solvable, while this is not true for general nonlinear problems [8, 9]. By solving (46) exactly, one obtains

$$e(t) = \frac{\int \left(\frac{\lambda^4}{3C} t^3 + \left(\frac{\lambda^3}{2} - \frac{\lambda^3}{C}\right) t^2\right) e^{-\lambda t}\, dt + K}{e^{-\lambda t}} = -1 - \lambda t - \frac{\lambda^2 t^2}{2} - \frac{\lambda^3 t^3}{3C} + K e^{\lambda t}. \tag{47}$$

The arbitrary constant K can be fixed by noting the fact that $e(0) = 0$ initially. This gives rise to $K = 1$. Therefore, the error of the RK3 scheme for the test ODE (44) is given as

$$e(t) = e^{\lambda t} - 1 - \lambda t - \frac{\lambda^2 t^2}{2} - \frac{\lambda^3 t^3}{3C}. \tag{48}$$

The accuracy optimization problem can thus be formulated to be minimizing the the error $e(t)$ via choosing an optimal value of C. The C value such that $e(t) = 0$ is given to be

$$C = \frac{\lambda^3 h^3}{3} \cdot \frac{1}{e^{\lambda h} - 1 - \lambda h - \frac{\lambda^2 h^2}{2}}, \tag{49}$$

where we have set $t = h$. For a finite step size h, (49) gives a discretization dependent choice of C to attain the best accuracy. A uniform, or discretization independent optimal C can be obtained by passing the limit $h \to 0$. By using the L'Hôpital's Rule, we have

$$C = \lim_{h \to 0} \frac{\lambda^3 h^3}{3} \cdot \frac{1}{e^{\lambda h} - 1 - \lambda h - \frac{\lambda^2 h^2}{2}} = 2. \tag{50}$$

Remark Without resorting to the analytical solution, we have found the optimized C value such that the error function $e(t) = 0$ or being minimized. By taking $C = 2$, the RK3 scheme (40) actually achieves the highest possible order of accuracy, i.e., the third order.

6.1.3 Stability

Dependent on different PDEs, the spectrum of the matrix $\mathbf{A}$ in (42) has different features. To design a stable time stepping scheme, such features should be taken into account [29]. Without the loss of generality, in the present study, we consider two types of one-dimensional PDEs, i.e., a hyperbolic system

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0, \tag{51}$$

and a parabolic system

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}. \tag{52}$$

The proposed stability optimization procedure can be generalized to other types of PDEs.

Mathematically, the analytical eigenvalues of the hyperbolic equation (51) are all pure imaginary numbers, while those of the heat equation (52) are all negative real numbers [43]. In the present study, the spatial derivatives in both equations are discretized using the standard higher order central finite differences [43]. Due to the use of symmetrical spatial discretization, the numerical spectra of the semi-discretized equations are also locate along the axes of the corresponding spectra of the analytical operators, subject to a negligible roundoff error. However, we note that if a severe asymmetric approximation is involved, the discrete spectra might significantly deviate from the analytical ones, giving rise to the so-called spurious modes. See Ref. [42] for more details on the spurious solutions and how to suppress them numerically.

In the present study, we consider the optimization of the stability of the RK3 scheme (40) for these two types of PDEs. As discussed previously, the stability of a particular Runge-Kutta method in solving the semi-discretized ODEs (42) is essentially determined by the stability of the same Runge-Kutta method applying to a single scalar ODE (44) with λ being an eigenvalue of $\mathbf{A}$. The stability of a Runge-Kutta method in solving (44) can be fully characterized by its stability region. However, for more complicated PDEs, such as those involving discontinuous coefficients, the discretization of the spatial operators also affect the stability analysis [41].

Definition The stability region of the RK3 scheme for solving (44) is defined to be the set of all $h\lambda \in \mathbb{C}$ such that [29]

$$\left|1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{3C}\right| \leq 1. \tag{53}$$

Moreover the boundary of the stability region is given by the set of all $h\lambda \in \mathbb{C}$ such that

$$\left|1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{3C}\right| = 1. \tag{54}$$

Based on the above definition, the semi-discretized ODEs (42) are stable if all its eigenvalues are within the stability region of the RK3 scheme. Thus, the special features of the spectral of a PDE can be utilized as the criterion for optimizing the Runge-Kutta methods. Various different types of such criteria have been studied in the literature [29, 37]. In present study, we explore the optimization of the RK3 scheme for hyperbolic and parabolic equations, via extending the stability region along the imaginary axis and negative real axis, respectively. To simplify the notation, we let $z = \lambda h = a+bi$. The stability regions of the RK3 scheme for $C = 2$ and $C = 4$ are depicted in Fig. 1 (a) and (b). It is clear that by using a different C, the interval being covered on the imaginary axis varies. So does the interval on the negative real axis. We

(a) (b)

(c) (d)

Figure 1: The stability region of the three stage explicit Runge-Kutta method. (a) $C = 2$; (b) $C = 4$; (c) $C = \frac{4}{3}$; (d) $C = \frac{16}{3}$.

denote the interval on the imaginary axis and negative real axis to be, respectively, $[-\beta i, \beta i]$ and $[-\alpha, 0]$. For a given C, the α and β values can be analytically solved. In particular, by setting $a = 0$ in (54), we have equation for the imaginary axis

$$\left|1 + bi + \frac{(bi)^2}{2} + \frac{(bi)^3}{3C}\right|^2 = 1 + \frac{b^4}{4} - \frac{2b^4}{3C} + \frac{b^6}{9C^2} = 1. \tag{55}$$

When $C = 2$, the nonzero roots of (55) are $\pm\sqrt{3}$, while when $C = 4$, there are no nonzero real roots. Thus we have $\beta = \sqrt{3}$ and $\beta = 0$, respectively, for $C = 2$ and $C = 4$. Similarly, by setting $b = 0$ in (54), we have equation for the real axis

$$\left|1 + a + \frac{a^2}{2} + \frac{a^3}{3C}\right|^2 = 1 + 2a + 2a^2 + a^3 + \frac{2a^3}{3C} + \frac{a^4}{4} + \frac{2a^4}{3C} + \frac{a^5}{3C} + \frac{a^6}{9C^2} = 1. \tag{56}$$

When $C = 2$, the only nonzero real root of (56) is $-(4 + \sqrt{17})^{1/3} + (4 + \sqrt{17})^{-1/3} - 1$, while when $C = 4$, the only nonzero real root is $-2^{4/3} - 2$. Thus, we have $\alpha = (4 + \sqrt{17})^{1/3} - (4 + \sqrt{17})^{-1/3} + 1 \approx 2.512745327$ and $\alpha = 2^{4/3} + 2 \approx 4.519842100$, respectively, for $C = 2$ and $C = 4$.

We next consider the maximization of the imaginary interval $[-\beta i, \beta i]$, i.e., we explore a optimal C value such that β attains the maximum. Such a choice permits the largest Courant-Friedrichs-Lewy (CFL) time step in solving the hyperbolic equation (51). For this purpose, we solve (55) symbolically

$$b = \pm\frac{1}{2}\sqrt{24C - 9C^2}, \tag{57}$$

and differentiate this function with respect to C

$$\frac{\delta b}{\delta C} = \pm\frac{1}{4}\frac{24 - 18C}{\sqrt{24C - 9C^2}}. \tag{58}$$

The derivative in (58) is zero if $C = \frac{4}{3}$ and is undefined at $C = 0$ and $C = \frac{8}{3}$. By plugging the nonzero C values into (57), we have $b = 2$ and $b = 0$, respectively, for $C = \frac{4}{3}$ and $C = \frac{8}{3}$. Thus, when $C = \frac{4}{3}$, the RK3 scheme attains the maximal imaginary interval with $\beta = 2$. By taking $C = \frac{4}{3}$ in (56), we have $\alpha = 2$ in the present case. See Fig. 1 (c).

We then consider the maximization of the real interval $[-\alpha, 0]$. The maximal α value allows the largest CFL time step in solving the parabolic equation (52). By solving (56) symbolically,

(a) (b)

Figure 2: The perturbed stability region of the three stage explicit Runge-Kutta method with $C = \frac{16}{3} + \epsilon$. (a) $\epsilon = 0.001$; (b) $\epsilon = -0.001$.

we have three nonzero real roots

$$\begin{aligned} a_1 &= -\frac{3C}{4} + \frac{1}{4}\sqrt{9c^2 - 48C}, \quad a_2 = -\frac{3C}{4} - \frac{1}{4}\sqrt{9c^2 - 48C} \\ a_3 &= \frac{1}{2}\left(6C^2 - 24C - C^3 + 2\sqrt{-56C^3 + 9C^4 + 144C^2}\right)^{1/3} \\ &\quad - \frac{2(C - C^2/4)}{\left(6C^2 - 24C - C^3 + 2\sqrt{-56C^3 + 9C^4 + 144C^2}\right)^{1/3}} - \frac{C}{2}. \end{aligned}$$

Because the analytical forms are complicated, an alternative optimization procedure involving both numerical and analytical arguments is conducted. We first solve the optimization problem numerically. A search of local maximal α values is conducted by considering a sequence of increasing C values with increment being 0.1. The numerical results indicate that the global maximal value of α is achieved when $C \approx 5.3$. We then confirm this result analytically by noting that the derivatives of a_1 and a_2 with respective to C are undefined at $C = \frac{16}{3}$. By taking $C = \frac{16}{3}$ in (56), the maximal α value is

$$\alpha = \frac{2}{3}\frac{\left(46 - 6\sqrt{57}\right)^{2/3} + 4 + 4\left(46 - 6\sqrt{57}\right)^{1/3}}{\left(46 - 6\sqrt{57}\right)^{1/3}} \approx 6.260790890.$$

Moreover, by taking $C = \frac{16}{3}$ in (55), there is no nonzero real root. Thus, we have $\beta = 0$ for $C = \frac{16}{3}$. See Fig. 1 (d).

We note that the boundary of stability region for the case $C = \frac{16}{3}$ intersects itself at the point $(-4, 0)$, see Fig. 1 (d). Theoretically, a solution with negative real eigenvalues located at that point has a unitary amplification factor so that the solution could be stable in time integration. However, due to numerical roundoff errors, the actual boundary of stability region might differ from the analytical one by a small perturbation. Such a situation can be illustrated analytically by considering a small perturbation ϵ in C value too. For example, by taking $C = \frac{16}{3} + \epsilon$ where we choose $\epsilon = 0.001$, the perturbed stability region is shown in Fig. 2 (a). It can be seen that the stability regions disconnect at point $(-4, 0)$ so that the scheme is unstable if λh locates at that point. Instead if we choose $\epsilon = -0.001$, the stability region is well connected so that the negative real axis is fully covered, see Fig. 2 (b). Therefore, numerically, instead of choosing $C = \frac{16}{3}$, we treat $C = \frac{16}{3} - 0.001$ as the maximum for the extended negative real axis. The corresponding dimension of the stability region is calculated to be $\alpha = -6.25941410$ and $\beta = 0$.

Remark: The proposed three stage explicit Runge-Kutta schemes are listed in Table 1.

Table 1: New three stage explicit Runge-Kutta schemes developed via optimizations. Here we take $\epsilon = 0.001$.

Criterion	C	α	β
Highest order of accuracy	2	2.512745327	$\sqrt{3}$
Sparsest	4	4.519842100	0
Max stability interval $[\beta i, \beta i]$	$\frac{4}{3}$	2	2
Max stability interval $[-\alpha, 0]$	$\frac{16}{3} - \epsilon$	6.259414105	0

6.2 Optimized four stage explicit Runge-Kutta methods

We consider the optimization of the four stage explicit RK4 scheme (41), which involves three parameters C_1, C_2, and C_3. When taking $C_1 = 2$, $C_2 = 0$, and $C_3 = 2$, one attains the classical fourth order Runge-Kutta scheme.

6.2.1 Sparseness

Remark The sparsest RK4 scheme can be attained by taking $C_1 = 2$, $C_2 = 0$, and $C_3 = 2$ in (41), i.e., the classical fourth order Runge-Kutta scheme.

6.2.2 Accuracy

For general values of C_1, C_2, and C_3, the DG deduced RK4 schemes are at least second order accurate. Through the solution the test equation (44) as discussed above for the RK3 schemes, it can be similarly shown that the highest order of accuracy of the DG deduced RK4 schemes can be achieved when $C_1 = 2$, $C_2 = 0$, and $C_3 = 2$.
Remark The classical fourth order Runge-Kutta scheme is the most accurate DG deduced RK4 scheme with $C_1 = 2$, $C_2 = 0$, and $C_3 = 2$.

6.2.3 Stability

We consider the optimization of the stability by studying the extended real and imaginary intervals. However, since we have three free parameters for the RK4 schemes, this allows us to develop optimized RK4 schemes with combined criteria. In particular, we manage to utilize three parameters to optimize three features, i.e., sparseness, accuracy and stability, simultaneously.

We first consider the accuracy optimization by consuming only one degree of freedom. As discussed above, we consider the integration of (44) from 0 to h by using the RK4 scheme with

one step

$$
\begin{aligned}
U^1 &= u^0 \\
U^2 &= u^0 + \frac{h}{2}\lambda u^0 = \left(1 + \frac{\lambda h}{2}\right) u^0 \\
U^3 &= u^0 + \frac{C_1 - 2}{2C_1} h\lambda u^0 + \frac{1}{C_1} h\lambda \left(1 + \frac{\lambda h}{2}\right) u^0 = \left(1 + \frac{\lambda h}{2} + \frac{\lambda^2 h^2}{2C_1}\right) u^0 \\
U^4 &= u^0 + \left(1 - \frac{2}{C_3} + \frac{2C_2}{C_1 C_3}\right) h\lambda u^0 - \frac{2C_2}{C_1 C_3} h\lambda \left(1 + \frac{\lambda h}{2}\right) u^0 + \frac{2}{C_3} h\lambda \left(1 + \frac{\lambda h}{2} + \frac{\lambda^2 h^2}{2C_1}\right) u^0 \\
&= \left(1 + \lambda h + \frac{C_1 - C_2}{C_1 C_3}\lambda^2 h^2 + \frac{1}{C_1 C_3}\lambda^3 h^3\right) u^0 \\
u^1 &= u^0 + \frac{h}{6}\lambda u^0 + \frac{h}{3}\lambda \left(1 + \frac{\lambda h}{2}\right) u^0 + \frac{h}{3}\lambda \left(1 + \frac{\lambda h}{2} + \frac{\lambda^2 h^2}{2C_1}\right) u^0 \\
&\quad + \frac{h}{6}\lambda \left(1 + \lambda h + \frac{C_1 - C_2}{C_1 C_3}\lambda^2 h^2 + \frac{1}{C_1 C_3}\lambda^3 h^3\right) u^0 \\
&= \left(1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{C_3 + C_1 - C_2}{6C_1 C_3}\lambda^3 h^3 + \frac{1}{6C_1 C_3}\lambda^4 h^4\right) u^0 \\
&= 1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{C_3 + C_1 - C_2}{6C_1 C_3}\lambda^3 h^3 + \frac{1}{6C_1 C_3}\lambda^4 h^4
\end{aligned}
$$

Here the initial condition $u^0 = u_0 = 1$ has been applied. It is obvious that by exploiting only one degree of freedom, the highest order of accuracy can be attained by the RK4 schemes is three, which corresponds the choice of

$$\frac{C_3 + C_1 - C_2}{6C_1 C_3} = \frac{1}{6} \quad \text{or} \quad C_2 = C_3 + C_1 - C_1 C_3. \tag{59}$$

Remark By fixing one parameter to be $C_2 = C_3 + C_1 - C_1 C_3$, the optimized DG deduced RK4 schemes are guaranteed to be the third order of accurate.

We next consider the sparseness. With a fixed $C_2 = C_3 + C_1 - C_1 C_3$, the RK4 scheme becomes

$$\text{RK4:} \quad \begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & \frac{C_1-2}{2C_1} & \frac{1}{C_1} & & \\ 1 & \frac{2-C_1}{C_1} & \frac{2C_1C_3-2C_1-2C_3}{C_1C_3} & \frac{2}{C_3} & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array} \tag{60}$$

With C_1 and C_3 being finite, one cannot eliminate the diagonal line of the Butcher matrix. Thus, at most, three lower triangular coefficients could be zero by considering different C_1 and C_3 values. Thus, the optimized scheme in terms of the sparseness for the present RK4 methods is obviously obtained by setting $C_1 = 2$, because in doing so, one degree of freedom can be exchanged for two zero coefficients in the Runge-Kutta matrix.

Remark The third order DG deduced RK4 schemes are as sparse as possible when taking $C_1 = 2$,

We finally consider the stability. With two free parameters being fixed, we have only one degree of freedom left, i.e., C_3. Alternatively, we introduce a new parameter D defined to be

$D = C_1C_3 = 2C_3$ for stability optimization. Based on the previous discussions, the stability region of the RK4 schemes is given as:

Definition The stability region of the DG deduced RK4 scheme for solving (44) is defined to be the set of all $h\lambda \in \mathbb{C}$ such that [29]

$$\left|1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{6} + \frac{\lambda^4 h^4}{6D}\right| \leq 1. \tag{61}$$

Moreover the boundary of the stability region is given by the set of all $h\lambda \in \mathbb{C}$ such that

$$\left|1 + \lambda h + \frac{\lambda^2 h^2}{2} + \frac{\lambda^3 h^3}{6} + \frac{\lambda^4 h^4}{6D}\right| = 1. \tag{62}$$

In the present notation, the classical fourth order Runge-Kutta method which has been shown to be the most accurate and sparsest RK4 scheme, has $D = 4$. The corresponding stability region is depicted in Fig. 3 (a). As usual, the dimension of the stability region can be analytically solved. As discussed above, we take $z = \lambda h = a + bi$. By setting $b = 0$ in (62), we have the equation for the real axis

$$\left|1 + a + \frac{a^2}{2} + \frac{a^3}{6} + \frac{a^4}{6D}\right|^2 = 1, \tag{63}$$

or equivalently

$$1 + 2a + 2a^2 + \frac{4a^3}{3} + \frac{a^6}{36} + \frac{7a^4}{12} + \frac{a^5}{6} + \frac{a^8}{36D^2} + \frac{a^7}{18D} + \frac{a^5}{3D} + \frac{a^4}{3D} + \frac{a^6}{6D} = 1. \tag{64}$$

If we choosing $D = 4$, Eq. (64) reduces to

$$1 + 2a + 2a^2 + \frac{4a^3}{3} + \frac{2a^4}{3} + \frac{a^5}{4} + \frac{5a^6}{72} + \frac{a^7}{72} + \frac{a^8}{576} = 1. \tag{65}$$

The only nonzero real root is

$$-\alpha = -\frac{1}{3}\left(172 + 36\sqrt{29}\right)^{1/3} + \frac{20}{3}\left(172 + 36\sqrt{29}\right)^{-1/3} - \frac{4}{3} \approx -2.785293563. \tag{66}$$

By taking $a = 0$ in (62), we have equation for the imaginary axis

$$\left|1 + bi + \frac{(bi)^2}{2} + \frac{(bi)^3}{6} + \frac{(bi)^4}{6D}\right|^2 = 1 - \frac{b^4}{12} + \frac{b^6}{36} - \frac{b^6}{6D} + \frac{b^8}{36D^2} + \frac{b^4}{3D} = 1. \tag{67}$$

With $D = 4$, (67) reduces to be $1 - \frac{b^6}{72} + \frac{b^8}{576} = 1$ and there are two nonzero real roots: $\pm 2\sqrt{2}$. Thus, we have $\beta = 2\sqrt{2}$. There results are consistent with our previous study [43].

We next consider the maximal real stability interval $[-\alpha, 0]$. The corresponding optimal D value provides the largest CFL time step in solving the parabolic equation (52). To this end, we first symbolically solve Eq. (64). The only nonzero root is given as

$$\begin{aligned} a \;=\; & -\frac{D}{3} + \frac{1}{6}\left(108D^2 - 648D - 8D^3 + 36\sqrt{-72D^3 + 5D^4 + 324D^2}\right)^{1/3} \\ & -6(D - \frac{D^2}{9})\left(108D^2 - 648D - 8D^3 + 36\sqrt{-72D^3 + 5D^4 + 324D^2}\right)^{-1/3}. \end{aligned}$$

(a) (b)

(c) (d)

Figure 3: The stability region of the fourth stage explicit Runge-Kutta method. (a) $D = 4$; (b) $D = 9$; (c) $D = 9.05$; (d) $D = 2^{2/3} + 2$.

Due to the complicated expression, a combined analytical and numerical optimization is conducted. A search of local maximal α values is carried out by testing a sequence of increasing D values with increment being 0.1. The numerical search indicates that the global maximal value is assumed when $D \approx 9$. We then confirm this result analytically by noting that $D = 9$ is the only nonzero root such that

$$108D^2 - 648D - 8D^3 + 36\sqrt{-72D^3 + 5D^4 + 324D^2} = 0.$$

Thus, such a D value renders the derivative of a with respect to D, $\frac{\delta a}{\delta D}$, undefined.

The stability region of the DG based RK4 scheme with $D = 9$ is plotted in Fig. 3 (b). With $D = 9$, the analytical α value is $\alpha = 6$, which is slightly less than that of the optimal RK3 scheme. This means that the three stage Runge-Kutta method could provide a larger CFL time step than the four stage Runge-Kutta method when solving the parabolic equation (52). With $D = 9$, Eq. (67) has two nonzero real roots: $\pm\frac{1}{2}\sqrt{-54 + 6\sqrt{141}}$. Thus we have $\beta = \frac{1}{2}\sqrt{-54 + 6\sqrt{141}} \approx 2.076418342$. Furthermore, it can be observed from Fig. 3 (b) that the stability region has a tendency to be broken into to two separate regions when D is increasing. We numerically confirm this by plotting the stability region for $D = 9.05$, see Fig. 3 (c). Based on the shapes of the stability region in Fig. 3 (b) and (c), it is suspected that there exists another critical number of $\frac{\delta a}{\delta D}$ in the interval $D \in [9, 9.05]$, such that at that value, the boundary of the stability region intersects with itself as in Fig. 1 (d). However, should we find such D value, we still had to adopt a perturbed value $D - \epsilon$ so that the numerical optimal D value is very close to $D = 9$. The difference between them is negligible. Thus, in the present study, we treat $D = 9$ as the optimal value such that the negative real axis achieves the maximum.

We finally consider the maximization of the imaginary interval $[-\beta i, \beta i]$. Such a choice allows the largest CFL time step in solving the hyperbolic equation (51). For this purpose, we solve (67) analytically. This gives rise to four nonzero real roots

$$\begin{aligned} b_1 &= \frac{1}{2}\sqrt{-2D^2 + 12D + 2\sqrt{D^4 - 12D^3 + 48D^2 - 48D}}, \quad b_2 = -b_1 \\ b_3 &= \frac{1}{2}\sqrt{-2D^2 + 12D - 2\sqrt{D^4 - 12D^3 + 48D^2 - 48D}}, \quad b_4 = -b_3. \end{aligned}$$

Essentially, we need to test the critical numbers for b_1 and b_3, while those for b_2 and b_4 are the same. We first consider the derivative of b_3 with respect to D, $\frac{\delta b_3}{\delta D}$. We have found that $\frac{\delta b_3}{\delta D}$ is zero at two imaginary roots of D, while $\frac{\delta b_3}{\delta D}$ is undefined at $D = 4$ and $D = 0$. Note again that with $D = 4$ the DG deduced RK4 scheme becomes the classical fourth order Runge-Kutta scheme. On the other hand, the derivative of b_1 with respect to D, $\frac{\delta b_1}{\delta D}$, is zero

(a) (b)

Figure 4: The boundary of stability region of the DG based RK4 method. (a) $D = 4$; (b) $D = 2^{2/3} + 2$.

Table 2: The optimized fourth stage explicit Runge-Kutta schemes.

Criterion	D	α	β
Highest order of accuracy	4	2.785293563	$2\sqrt{2}$
Sparsest	4	2.785293563	$2\sqrt{2}$
Accuracy + Sparse + Max stability interval $[\beta i, \beta i]$	4	2.785293563	$2\sqrt{2}$
Accuracy + Sparse + Max stability interval $[-\alpha, 0]$	9	6	2.076418342

if $D = 2^{2/3} + 2$ and is undefined only when $D = 0$. Thus, overall, we find two nonzero real critical numbers $D = 4$ and $D = 2^{2/3} + 2$. As shown before, the β value for $D = 4$ is $\beta = 2\sqrt{2} \approx 2.828427124$. By substituting $D = 2^{2/3} + 2$ into b_1, we have the corresponding β to be $\beta = \sqrt{2^{2/3} + 2^{4/3} + 4} \approx 2.847322102$, see Fig. 3 (d). This seems to suggest that $D = 2^{2/3} + 2$ is the global maximum.

However, we have found that by using $D = 2^{2/3} + 2$, the RK4 scheme is actually quite unstable numerically. By comparing chart (a) and (d) in Fig. 3, $D = 2^{2/3} + 2$ seems to give a slight larger interval along the imaginary axis than $D = 4$ does. Nevertheless, if we look at the coverage of eigenvalues along the imaginary axis, especially eigenvalues near 0, there is a loss of coverage for some interval inside $[-\beta i, \beta i]$, see Fig. 4. It can be seen that for $D = 2^{2/3} + 2$, a small deviation of the boundary of stability region towards left has been found. We note that such a deviation is so small that one has to zoom in sufficiently to see the problem, while in a normal scale, one has the impression that the boundaries of stability regions for both $D = 4$ and $D = 2^{2/3} + 2$ locate on the imaginary axis. Due to this loss of coverage, even though $D = 2^{2/3} + 2$ gives a larger β value, the corresponding RK4 scheme is unstable for small eigenvalues. Thus, we have to treat the effective β for $D = 2^{2/3} + 2$ to be $\beta = 0$ and rule out this choice as our optimized scheme. Therefore, it turns out that the classical fourth order Runge-Kutta scheme with $D = 4$ is again the winner for the maximal stability along the imaginary axis.

Remark: The optimized fourth stage explicit Runge-Kutta schemes are list in Table 2. The present study shows that the classical fourth order Runge-Kutta scheme is the best scheme in three categories. This perhaps explains why this scheme is the most widely used method in the Runge-Kutta family.

7 Numerical experiments

In this section, we numerically validate the proposed three stage and four stage explicit Runge-Kutta schemes, i.e., RK3 and RK4 schemes, in terms of accuracy and stability. To this end, two boundary initial value problems with analytical solutions are considered. The first one is

a hyperbolic equation [43]

$$\begin{aligned} \frac{\partial u}{\partial t}+\frac{\partial u}{\partial x} &= 0, \quad 0 \le x \le 1, \quad t \ge 0, \\ u(x,0) &= \sin(2\pi x), \\ u(0,t) &= \sin(2\pi(-t)), \quad u(1,t) = \sin(2\pi(1-t)). \end{aligned} \tag{68}$$

with analytical solution $u(x,t) = \sin(2\pi(x-t))$. The another one is a parabolic equation [43]

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{\partial^2 u}{\partial x^2}, \quad 0 \le x \le 1, \quad t \ge 0, \\ u(x,0) &= C\sin x, \\ u(0,t) &= 0, \quad \left.\frac{\partial u}{\partial x}\right|_{x=1} = C\cos(1)e^{-t}, \end{aligned} \tag{69}$$

where $C = e^{10}$. The analytical solution is $u(x,t) = C\sin(x)e^{-t}$. Since the optimized three stage Runge-Kutta schemes are usually of second order of accuracy in time, we also consider an explicit two stage second order Runge-Kutta, i.e., the RK2-1 scheme in Section 4, for a comparison. Here we rename this scheme to be RK2 scheme. The dimension of the stability region of this scheme is given as $\alpha = 2$ and $\beta = 0$.

In both examples, we consider time integration on an interval $t \in [0,T]$ with a time increment being $h = \Delta t$. A uniform grid with N nodes is employed spatially. The spatial discretization is carried out by using either the central finite difference (FD) or high order central FD methods [43]. We denote the bandwidth of the high order central FD to be $2M+1$. The corresponding spatial order of accuracy for such a high order scheme is $(2M)$th order in approximating first and second derivatives. With $M = 1$, one attains the regular central difference with the order being two. Since we focus only on the temporal discretization in the present study, the boundary closure of the FD discretization is simply treated by using the analytical solutions. For real physical problems where analytical solution is unknown, some advanced boundary closure methods, such as the matched interface and boundary (MIB) method [43], have to be employed to implement the high order central FD discretization. Denoting u_h as the numerical solution, we use the following measures to estimate errors in numerical examples:

$$L_\infty = \frac{\max|u-u_h|}{\max|u|}, \qquad L_2 = \sqrt{\frac{\sum_{i=1}^N |u-u_h|^2}{\sum_{i=1}^N |u|^2}}.$$

We first consider the hyperbolic problem. The order of convergence in time is first examined. A high order central FD with a large bandwidth $M = 10$ is employed for spatial discretization. By using $N = 51$, this essentially guarantees the spatial error being negligible in the present test. By taking the end time $T = 1$, three time increment values are considered, i.e., $h = 2\times 10^{-3}$, $h = 1\times 10^{-3}$, and $h = 5\times 10^{-4}$. The corresponding errors and orders of convergence are shown in Table 3. It can be seen from the table that the theoretical orders of all studied schemes are numerically verified. In particular, the DG deduced RK3 schemes with $C \neq 2$ are all of second order in time, which is the same as the RK2 scheme. However, the RK3 schemes are more accurate than the RK2 scheme. Similarly, the DG deduced RK4 schemes with $D \neq 4$ are all of third order in time and the RK3 scheme with $C = 2$ achieves that order too. Moreover, the four stage Runge-Kutta schemes are more accurate than the

Table 3: Temporal convergence tests for the hyperbolic problem. Here $N = 51$, $M = 10$, and $T = 1$.

			L_∞		L_2	
Scheme	Parameter	h	Error	Order	Error	order
RK2	none	2×10^{-3}	2.40×10^{-4}		7.93×10^{-5}	
		1×10^{-3}	5.99×10^{-5}	2.00	1.98×10^{-5}	2.00
		5×10^{-4}	1.50×10^{-5}	2.00	4.95×10^{-6}	2.00
RK3	$C = \frac{4}{3}$	2×10^{-3}	1.20×10^{-4}		4.20×10^{-5}	
		1×10^{-3}	3.00×10^{-5}	2.00	1.05×10^{-5}	2.00
		5×10^{-4}	7.52×10^{-6}	2.00	2.62×10^{-6}	2.00
RK3	$C = 4$	2×10^{-3}	1.20×10^{-4}		4.20×10^{-5}	
		1×10^{-3}	3.01×10^{-5}	2.00	1.05×10^{-5}	2.00
		5×10^{-4}	7.52×10^{-6}	2.00	2.62×10^{-6}	2.00
RK3	$C = \frac{16}{3}$	2×10^{-3}	1.50×10^{-4}		5.25×10^{-5}	
		1×10^{-3}	3.76×10^{-5}	2.00	1.31×10^{-5}	2.00
		5×10^{-4}	9.40×10^{-6}	2.00	3.28×10^{-6}	2.00
RK3	$C = \frac{16}{3} - \epsilon$	2×10^{-3}	1.50×10^{-4}		5.25×10^{-5}	
		1×10^{-3}	3.76×10^{-5}	2.00	1.31×10^{-5}	2.00
		5×10^{-4}	9.40×10^{-6}	2.00	3.28×10^{-6}	2.00
RK3	$C = 2$	2×10^{-3}	6.25×10^{-7}		2.87×10^{-7}	
		1×10^{-3}	7.80×10^{-8}	3.00	3.59×10^{-8}	3.00
		5×10^{-4}	9.74×10^{-9}	3.00	4.48×10^{-9}	3.00
RK4	$D = 9$	2×10^{-3}	3.47×10^{-7}		1.59×10^{-7}	
		1×10^{-3}	4.33×10^{-8}	3.00	1.99×10^{-8}	3.00
		5×10^{-4}	5.41×10^{-9}	3.00	2.49×10^{-9}	3.00
RK4	$D = 2^{2/3}+2$	2×10^{-3}	7.28×10^{-8}		3.30×10^{-8}	
		1×10^{-3}	9.03×10^{-9}	3.01	4.12×10^{-9}	3.00
		5×10^{-4}	1.12×10^{-9}	3.00	5.15×10^{-10}	3.00
RK4	$D = 4$	2×10^{-3}	6.28×10^{-9}		1.28×10^{-9}	
		1×10^{-3}	3.96×10^{-10}	3.99	8.00×10^{-11}	4.00
		5×10^{-4}	2.43×10^{-11}	4.02	4.91×10^{-12}	4.02

three stage Runge-Kutta scheme, even though having the same order. This perhaps suggests that DG deduced Runge-Kutta schemes with more stages tend to be more accurate, although their orders may not be higher. Finally, the RK4 with $D = 4$ attains the highest order in Table 3.

We next investigate the stability of DG deduced Runge-Kutta schemes for the hyperbolic process. The theoretical stability analysis of high order central FD discretization for hyperbolic equation has been conduced in [43]. In the present study, we only consider the central FD with $M = 1$ for simplicity. The corresponding spectral radius ρ of central FD matrix is known to

Table 4: Numerical CFL numbers for the hyperbolic problem. Here $N = 51$, $M = 1$, and $T = 10000$. The analytical CFL number equals to β, while the numerical CFL numbers given in the last column are computed based on the critical h.

Scheme	Parameter	β	Critical h	CFL
RK2	none	0	1.12×10^{-3}	0.0560
RK3	$C = 4$	0	1.61×10^{-3}	0.0805
RK3	$C = \frac{16}{3}$	0	1.41×10^{-3}	0.0705
RK3	$C = \frac{16}{3} - \epsilon$	0	1.41×10^{-3}	0.0705
RK3	$C = 2$	$\sqrt{3}$	3.47×10^{-2}	1.7350
RK3	$C = \frac{4}{3}$	2	4.00×10^{-2}	2.0000
RK4	$D = 2^{2/3} + 2$	0	3.38×10^{-3}	0.1690
RK4	$D = 9$	2.076418342	4.16×10^{-2}	2.0800
RK4	$D = 4$	$2\sqrt{2}$	5.66×10^{-2}	2.8300

be $\rho = \frac{1}{\Delta x}$ and a numerical discretization is stable if

$$\rho \Delta t = \frac{\Delta t}{\Delta x} = \frac{h}{\Delta x} \leq \beta, \tag{70}$$

where β is the dimension of the imaginary interval of the time stepping scheme. Consequently, we have the analytical Courant-Friedrichs-Lewy (CFL) number for the hyperbolic problem being β. We next study the numerical CFL numbers of the proposed Runge-Kutta schemes. By fixing $N = 51$, we consider an extremely large end time $T = 10000$ in the present study. We then numerically search for the critical $\Delta t = h$ value such that the computation is still stable. The numerical CFL number can then be computed as $h/\Delta x$. The critical h and numerical CFL numbers of the DG deduced Runge-Kutta schemes are reported in Table 4. These results can be grouped into two categories, i.e., with $\beta = 0$ and $\beta \neq 0$. The critical h for $\beta \neq 0$ is easy to test. Basically, before that value, the computation is still quite accurate, but after that value, the error quickly goes to infinity. It can be seen from Table 4 that the numerical CFL numbers for $\beta \neq 0$ agree with the analytical ones very well. On the other hand, the critical h for $\beta = 0$ is somewhat vague. Around certain critical value, the error deteriorates slowly when h increases. The error blows up only when a quite larger h is chosen. In our computation, we print out ten sample errors in the interval $t \in [0, 10000]$. Consequently, the reported critical h value is determined to be the h value such that the error in the last step remains to be of the same magnitude as other errors in the previous nine steps. It can be observed from Table 4 that such critical h values are not zero, although $\beta = 0$. Instead, they are some quite small numbers. Furthermore, it has been found that when T is larger, such critical h numbers become smaller. Thus, the numerical CFL numbers of these schemes become 0 only under a limit sense. We also note that the numerical CFL number for the RK4 scheme with $D = 2^{2/3} + 2$ is 0.1690 due to the effective $\beta = 0$. Furthermore, we note that the proposed RK3 scheme with $C = \frac{4}{3}$ attains the largest CFL number among all three stage schemes, while the classical fourth order Runge-Kutta method with $D = 4$ attains the largest CFL number among all four stage schemes. This confirms our theoretical stability analysis.

Table 5: Temporal convergence tests for the parabolic problem. Here $N = 51$, $M = 10$, and $T = 1$.

			L_∞		L_2	
Scheme	Parameter	h	Error	Order	Error	order
RK2	none	1×10^{-4}	2.69×10^{-9}		3.59×10^{-9}	
		5×10^{-5}	6.63×10^{-10}	2.02	7.06×10^{-10}	2.35
		2.5×10^{-5}	1.65×10^{-10}	2.00	1.65×10^{-10}	2.09
RK3	$C = \frac{4}{3}$	1×10^{-4}	9.25×10^{-10}		1.25×10^{-9}	
		5×10^{-5}	2.28×10^{-10}	2.02	2.64×10^{-10}	2.25
		2.5×10^{-5}	5.67×10^{-11}	2.01	5.91×10^{-11}	2.16
RK3	$C = 4$	1×10^{-4}	9.07×10^{-10}		9.28×10^{-10}	
		5×10^{-5}	2.26×10^{-10}	2.00	2.20×10^{-10}	2.07
		2.5×10^{-5}	5.65×10^{-11}	2.00	5.37×10^{-11}	2.04
RK3	$C = \frac{16}{3}$	1×10^{-4}	1.14×10^{-9}		1.25×10^{-9}	
		5×10^{-5}	2.83×10^{-10}	2.01	2.83×10^{-10}	2.14
		2.5×10^{-5}	7.07×10^{-11}	2.00	6.79×10^{-11}	2.06
RK3	$C = \frac{16}{3} - \epsilon$	1×10^{-4}	1.14×10^{-9}		1.25×10^{-9}	
		5×10^{-5}	2.83×10^{-10}	2.01	2.83×10^{-10}	2.14
		2.5×10^{-5}	7.07×10^{-11}	2.00	6.79×10^{-11}	2.06
RK3	$C = 2$	1×10^{-4}	5.24×10^{-11}		2.23×10^{-10}	
		5×10^{-5}	5.72×10^{-12}	3.20	2.45×10^{-11}	3.19
		2.5×10^{-5}	6.43×10^{-13}	3.15	2.79×10^{-12}	3.14
RK4	$D = 9$	1×10^{-4}	2.43×10^{-11}		1.06×10^{-10}	
		5×10^{-5}	2.88×10^{-12}	3.08	1.25×10^{-11}	3.08
		2.5×10^{-5}	3.41×10^{-13}	3.08	1.48×10^{-12}	3.07
RK4	$D = 2^{2/3}+2$	1×10^{-4}	2.28×10^{-11}		9.54×10^{-11}	
		5×10^{-5}	1.15×10^{-12}	4.17	5.91×10^{-12}	4.01
		2.5×10^{-5}	1.22×10^{-13}	3.66	4.90×10^{-13}	3.59
RK4	$D = 4$	1×10^{-4}	1.83×10^{-11}		5.84×10^{-11}	
		5×10^{-5}	8.29×10^{-13}	4.46	2.74×10^{-12}	4.41
		2.5×10^{-5}	4.94×10^{-14}	4.07	1.52×10^{-13}	4.17

Next, we examine the order of convergence in time for the parabolic problem. Again, a high order central FD with a large bandwidth $M = 10$ is employed for spatial discretization so that the spatial error is vanishing. We also fix $N = 51$ and $T = 1$. Here we test the orders by considering $h = 1\times10^{-4}$, $h = 5\times10^{-5}$, and $h = 2.5\times10^{-5}$. Note that these h values are smaller due to the nature of the parabolic process. The numerical errors and orders of convergence are listed in Table 5. The theoretical orders of all schemes are numerically validated. Moreover, the DG deduced RK4 scheme with $D = 2^{2/3}+2$ actually experiences some over performance so that its numerical order is higher than three. Furthermore, we observe a similar pattern in the Table 5, i.e., the DG based Runge-Kutta schemes with more stages tend to be more accurate, even though their orders are not higher. The double precision limit is almost achieved by using

Table 6: Numerical CFL numbers for the parabolic problem. Here $N = 51$, $M = 1$, and $T = 100$. The analytical CFL number equals to $\frac{\alpha}{4}$, while the numerical CFL numbers given in the last column are computed based on critical h.

Scheme	Parameter	α	$\frac{\alpha}{4}$	Critical h	CFL
RK2	none	2	0.5	2.00×10^{-4}	0.5000
RK3	$C = \frac{4}{3}$	2	0.5	2.00×10^{-4}	0.5000
RK3	$C = 2$	2.512745327	0.628186332	2.51×10^{-4}	0.6275
RK3	$C = 4$	4.519842100	1.129960525	4.52×10^{-4}	1.1300
RK3	$C = \frac{16}{3}$	6.260790890	1.565197717	5.93×10^{-4}	1.4825
RK3	$C = \frac{16}{3} - \epsilon$	6.259414105	1.564853525	6.26×10^{-4}	1.5650
RK4	$D = 2^{2/3} + 2$	2.617454426	0.654363607	2.61×10^{-4}	0.6525
RK4	$D = 4$	2.785293563	0.696323391	2.78×10^{-4}	0.6950
RK4	$D = 9$	6	1.5	6.00×10^{-4}	1.5000

the RK4 scheme with $D = 4$.

At last, we test the stability of the proposed Runge-Kutta schemes for the parabolic process. The theoretical stability analysis of high order central FD approaches for parabolic equation has been considered in [43]. Again, we only test the central FD with $M = 1$ for simplicity. The spectral radius ρ of the central FD approximation to the second order derivative is known to be $\rho = \frac{4}{\Delta x^2}$. Thus, the numerical scheme is stable if

$$\rho \Delta t = \frac{4\Delta t}{\Delta x^2} = \frac{4h}{\Delta x^2} \leq \alpha, \tag{71}$$

where α is the dimension of the real interval of the time stepping scheme. Consequently, the analytical CFL number for the parabolic problem is $\frac{\alpha}{4}$. The numerical CFL numbers of the proposed Runge-Kutta schemes are calculated by using $N = 51$. Because we have a term e^{-t} in the analytical solution to the parabolic problem, a too large T introduces computational vanishing solutions. To avoid that, we select the end time $T = 100$ for stability analysis. We again numerically search for the critical $\Delta t = h$ value such that the computation is still stable. The numerical CFL number can then be computed as $h/\Delta x^2$. The critical h and numerical CFL numbers of the DG based Runge-Kutta schemes are given in Table 6. The present test of critical h is similar to the previous case with $\beta \neq 0$, i.e., the cutoff value can be easily found through a linear scanning. We note that the numerical results in Table 6 confirm our analysis that for the RK3 schemes, the largest CFL number is not achieved at $C = \frac{16}{3}$, because the intersection point on the real axis is not covered (see Fig. 2). In fact, $C = \frac{16}{3}$ is the only exception in Table 6 that the numerical CFL number does not agree with the theoretical one. Instead, when we perturb this value by a small $\epsilon = 0.001$, the RK3 scheme attains the largest CFL number. For four stage schemes, the proposed RK4 with $D = 9$ is the most stable one as shown in Table 6. Among all schemes studied, the RK3 scheme with $C = \frac{16}{3} - \epsilon$ is the best one in terms of stability for the parabolic process.

8 Concluding Remarks

In this work, we introduce a unified discontinuous Galerkin (DG) finite element framework for solving ordinary differential equations (ODEs) in time. With each element being weakly connected to its neighboring elements, the present DG framework provides a flexible way to generate time stepping schemes. We particularly investigate different choices of three main components in the present DG formulation, i.e., boundary conditions, numerical quadrature, and nontrivial test functions, so that many different time integration methods can be derived. The boundary conditions are shown to be essential to the switch of one step or multistep time integrators. Based on the method of weighted residual, numerical quadratures are indispensable in our finite element time discretization to account for general nonlinear ODEs. Many different conditions, e.g. explicit conditions, implicit conditions, and symplectic conditions, are proposed for the test functions to control the features of the resulting time-stepping schemes. With these modeling issues being well taken care, the proposed DG formulation provides a unified framework to rederive all standard time-stepping schemes, such as low order one-step methods, high order Runge-Kutta methods, and multistep methods. Additionally, by using the proposed explicit conditions, explicit Runge-Kutta methods can be generated via a DG variational analysis for the first time in the literature.

Moreover, we have explored the potential of the proposed DG framework for constructing new time integration schemes. The successful derivation of novel explicit Runge-Kutta methods and symplectic implicit Runge-Kutta methods has also been realized. The generated high order Runge-Kutta methods often involve a set of essential parameters based on the proposed DG formalism. In order to render such new schemes being useful, we develop many optimization strategies. Three optimization criteria, i.e., accuracy, sparseness and stability, are considered. First, we consider the accuracy optimization of a linear initial value problem in terms of an error function without using the analytical solution. The most accurate scheme is found to be the one with the highest formal order of accuracy. Additionlly, We have also explored the optimization of the sparseness, which is related to the compressive sensing problem in signal/imaging processing. Finally, the stability optimization is considered based on the special features of the semi-discretized matrix obtained from solving partial differential equations. Both the maximizations of stability region along the real and imaginary axes are studied. Various novel three stage and four stage Runge-Kutta methods have been constructed. Numerical experiments have been carried out to validate the proposed optimized time integration schemes and to compare among them.

One feature of the proposed optimization procedure is that it is analytical in nature and thus is free of numerical error. A different general optimization procedure was proposed in the literature [29] to optimize the stability regions of Runge-Kutta schemes. Via a numerical search, optimized Runge-Kutta schemes with the maximal dimension along the imaginary axis, and along both imaginary and real axes were studied up to six stages. This algorithm could be extended to obtain extended interval along the negative real axis too. It is interesting to compare some results obtained from this approach with those of the present work. For the three stage third order Runge-Kutta scheme, the reported β value obtained via their numerical procedure was $\beta \approx 1.7871$ (see Table IV of Ref. [29]), while the analytical value found in the present work is $\beta = \sqrt{3} \approx 1.7321$. The numerical error in this case is about 0.055. Moreover, in Table V of Ref. [29], the expansion coefficient of 0.25 was reported for the location of the optimal three-stage second-order Runge-Kutta method. This actually agrees with $C = \frac{4}{3}$ of

the present study. Nevertheless, the numerical β was given as $\beta \approx 2.0696$ [29], while the analytical value found in the present work is $\beta = 2$. Thus, the numerical error is about 0.0696. Finally, the optimal imaginary interval scheme for the four-stage third-order Runge-Kutta methods was reported to be located at an expansion coefficient 0.03812 with a numerical $\beta \approx 2.8521$, see Table IV of Ref. [29]. Such an expansion coefficient 0.03812 corresponds to $D = 4.3722$ in our description. While in the present study, it is found analytically that the true optimal scheme to be $\beta = 2\sqrt{2}$ (i.e., $\beta \approx 2.8284$) and $D = 4$. In fact, it is easy to verify that $\beta \approx 2.8521$ cannot be a true maximum. At $D = 4.3722$, the analytical β value is $\beta = 2.78339056$, which is smaller than our maximum of $\beta \approx 2.8284$. This inconsistence might be explained by examining the numerical error. In fact, the numerical error of their estimate is $|2.8521 - 2.78339056| \approx 0.0687$, which is consistent with their errors in the other case. Nevertheless, we note that even though involving numerical errors, their optimization procedure is quite flexible to be applied to optimizations involving more stages, while the present analytical optimization procedure becomes cumbersome when dealing with Runge-Kutta schemes with more stages.

Finally, we like to point out that the proposed DG formulation encounters difficulties to recover certain time integration schemes. One such example is the reducible Runge-Kutta scheme [22]. The irreducible Runge-Kutta schemes have weights $b_j \neq 0$, $j = 1, \ldots, S$. If one b_j is zero, one deals with a reducible Runge-Kutta scheme. See Appendix A.8 for more details about this. We also experience difficulties in generating a popular strong stability-preserving (SSP) Runge-Kutta method, i.e., the third order Shu-Osher scheme [18, 19, 20, 25]. The essential difficulty of the proposed DG approach for deriving the Shu-Osher scheme and the attempts to cover other SSP Runge-Kutta methods can be found in Ref [21].

Acknowledgment

The research of S. Gottlieb was supported in part by . The research of G. W. Wei was supported in part by NIH grants CA-127189 and GM-090208, and NSF grants DMS-0616704 and CCF-0936830. The research of S. Zhao was supported in part by NSF grants DMS-0731503 and DMS-0616704, and by a UA Research Grants Committee Award.

References

[1] J.H. Argyris and D.W. Scharpf, Finite elements in time and space, *Aer. J. Royal Aer. Soc.*, **73**, 1041-1044 (1969).

[2] P. Bates, Z. Chen, Y. Sun, G.W. Wei and S. Zhao, Geometric and potential driving formation and evolution of biomolecular surfaces, *Journal of Mathematical Biology*, **59**, 193-231 (2009).

[3] C. Bogey and C. Bailly, A family of low dispersive and low dissipative explicit schemes for flow and noise computations, *J. Comput. Phys.*, **194**, 194-214 (2004).

[4] M. Borri and C.L. Bottasso, A general framework for interpreting time finite element formulations, *Comput. Mech.*, **13**, 133-142 (1993).

[5] C.L. Bottasso, A new look at finite elements in time: a variational interpretation of Runge-Kutta methods, *Applied Numerical Mathematics*, **25**, 355-368 (1997).

[6] M. Calvo, J. Franco, and L. Randez, A new minimum storage Runge-Kutta scheme for computational acoustics, *Journal of Computational Physics*, **201**, 1-12 (2004).

[7] C. Chiu and D.A. Kopriva, An optimal Runge-Kutta method for steady-state solutions of hyperbolic systems, *SIAM J. Numer. Anal.*, **29**, 425-438 (1992).

[8] A. Christlieb, B. Ong and J.-M. Qiu, Comments on high order integrators embedded within integral deferred correction methods, *Communications of Applied and Computational Mathematics*, **4**, 27-56 (2009).

[9] A. Christlieb, B. Ong and J.-M. Qiu, Integral Deferred Correction Methods constructed with High Order Runge-Kutta Integrators, *Mathematics of Computation*, **79**, 761-783 (2010).

[10] M. Delfour, W. Hager and F. Trochu, Discontinuous Galerkin Methods for Ordinary Differential Equations, *Math. Comput.*, **36**, 455-473 (1981).

[11] M. Delfour and F. Dubeau, Discontinuous polynomial approximations in the theory of one-step, hybrid and multistep methods for nonlinear ordinary differential equations, *Math. Comput.*, **47**, 169-189 (1986).

[12] J. Douglas, Jr., Alternating direction methods for three space variables, *Numerische Mathematik*, **4**, 41-63 (1962).

[13] A. Dutt, L. Greengard, and V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, *BIT*, **40**, 241-266 (2000).

[14] D. Estep, A posteriori error bounds and global error control for approximation of ordinary differential equations, *SIAM J. Numer. Anal.*, **32**, 1-48 (1995).

[15] D. Estep and A. Stuart, The dynamical behavior of the discontinuous Galerkin method and related difference schemes, *Math. Comput.*, **71**, 1075-1103 (2001).

[16] I. Fried, Finite element analysis of time dependent phenomena, *AIAA J.*, **7**, 1170-1173 (1969).

[17] S. Gill, A process for the step-by-step integration of differential equations in an automatic digital computing machine, *Proceedings of the Cambridge Philosophical Society*, **47**, 96-108 (1950).

[18] S. Gottlieb, C.W. Shu, and E. Tadmor, Strong stability-preserving high order time discretization methods, *SIAM Rev.*, **43**, 89-112 (2001).

[19] S. Gottlieb, On high order strong stability preserving Runge-Kutta and multi step time discretizations, *J. Sci. Comput.*, **25**, 105-128 (2005).

[20] S. Gottlieb and S.J. Ruuth, Optimal strong-stability-preserving time-stepping schemes with fast downwind spatial discretizations, *J. Sci. Comput.*, **27**, 289-304 (2006).

[21] S. Gottlieb, G.W. Wei and S. Zhao A unified discontinuous Galerkin approach for ordinary differential equations, to be uploaded to ArXiv, (2010)

[22] P. Gortz and R. Scherer, Reducibility and characterization of symplectic Runge-Kutta methods, *Electronic Transactions on Numerical Analysis*, **2**, 194-204 (1994).

[23] C. Johnson, Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations, *SIAM J. Numer. Anal.*, **25**, 908-926 (1988).

[24] C.A. Kennedy, M.H. Carpenter, and R.M. Lewis, Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations, *Applied Numerical Mathematics*, **35**, 177-219 (2000).

[25] D.I. Ketcheson, C.B. Macdonald, and S. Gottlieb, Optimal implicit strong stability preserving Runge-Kutta methods, *Applied Numerical Mathematics*, **59**, 373-392 (2009).

[26] D.I. Ketcheson, Runge-Kutta methods with minimum storage implementations, *Journal of Computational Physics*, **229**, 1763-1773 (2010).

[27] F. Lasagni, Canonical Runge-Kutta methods, *Z. Angew. Math. Phys.*, **39**, 952-953 (1988).

[28] P. Lesaint and P.A. Raviart, On a finite element method for solving the neutron transport equation, *in Mathematical Aspects of Finite Elements in Partial Differential Equations*, , C. de Boor, ed., Academic Press, New York, 89-145 (1974).

[29] J.L. Mead and R.A. Renaut, Optimal Runge-Kutta methods for first order pseudospectral operators, *J. Comput. Phys.*, **152**, 404-419 (1999).

[30] W. Oevel and M. Sofroniou Symplectic Runge-Kutta schemes II: Classification of symmetric methods, unpublished, (1997)

[31] S.C. Reddy and L.N. Trefethen, Stability of the method of lines, *Numer. Math.*, **62**, 235-267 (1992).

[32] W.H. Reed and T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, *Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory*, (1973).

[33] R.D. Ruth, A canonical integration technique, *IEEE Trans. Nucl. Sci.*, **30**, 3669-2671 (1983).

[34] J.M. Sanz-Serna, Runge-Kutta schemes for Hamiltonian systems, *BIT*, **28**, 877-883 (1988).

[35] J.M. Sanz-Serna, Symplectic integrators for Hamiltonian problems: An overview, *Acta Numerica*, **1**, 243-286 (1992).

[36] K. Tselios and T.E. Simos, Runge-Kutta methods with minimal dispersion and dissipation for problems arising from computational acoustics, *Journal of Computational and Applied Mathematics*, **175**, 173-181 (2005).

[37] M. Torrilhon and R. Jeltsch, Essentially optimal explicit Runge-Kutta methods with applications to hyperbolic-parabolic equations, *Numer. Math.*, **106**, 303-334 (2007).

[38] P.J. Van Der Houwen and B.P. Sommeijer, On the internal stability of explicit m-stage Runge-Kutta methods for large m-values, *Z. Angew. Math. Mech.*, **60**, 479-485 (1980).

[39] P.J. Van Der Houwen and B.P. Sommeijer, Explicit Runge-Kutta (-Nystrom) methods with reduced phase errors for computing oscillating solutions, *SIAM J. Numer. Anal.*, **24**, 595-617 (1987).

[40] S.N. Yu, S. Zhao, and G.W. Wei, Local spectral time-splitting method for first and second order partial differential equations, *Journal of Computational Physics*, **206**, 727-780 (2005).

[41] S. Zhao, and G.W. Wei, High order FDTD methods via derivative matching for Maxwell's equations with material interfaces, *J. Comput. Phys.*, **200**, 60-103 (2004).

[42] S. Zhao, On the spurious solutions in the high-order finite difference methods for eigenvalue problems, *Computer Methods in Applied Mechanics and Engineering*, **196**, 5031-5046 (2007).

[43] S. Zhao and G.W. Wei, Matched interface and boundary (MIB) for the implementation of boundary conditions in high order central finite differences, *International Journal for Numerical Methods in Engineering*, **77**, 1690-1730 (2009).

A Derivation details

A.1 Euler and Crank-Nicholson schemes

The implicit Euler scheme can be obtained by consider a different boundary condition. In particular, we still have $L = 1$, but take $U_n(t_{n+1}) = u^{n+1}$. We note that under such a condition, the discretized variational equation is still of the form (9) after some similar derivations. Then, by taking $w(t) = 1$, one attains Eq. (17) as well. Now by employing a quadrature with $c_1 = 1$ and $b_1 = 1$, we have

$$u^{n+1} = u^n + h_n f(U^1, t_{n+1}), \tag{72}$$

where $U^1 = U_n(t_n + c_1 h_n) = U_n(t_{n+1}) = u^{n+1}$. We thus have the implicit Euler scheme (20).

Likewise, the Crank-Nicholson scheme obviously demands a two points support quadrature: $(c_1, c_2) = (0, 1)$ and $b_1 = b_2 = \frac{1}{2}$. Moreover, we need to set $L = 2$ and enforce boundary conditions rigorously at both ends of time element I_n: $U_n(t_n) = u^n$ and $U_n(t_{n+1}) = u^{n+1}$. The resulting DG scheme

$$u^{n+1} = u^n + \frac{h_n}{2} f(U^1, t_n) + \frac{h_n}{2} f(U^2, t_{n+1}), \tag{73}$$

is in fact the Crank-Nicholson scheme (21), because $U^1 = U_n(t_n + c_1 h_n) = U_n(t_n) = u^n$ and $U^2 = U_n(t_n + c_2 h_n) = U_n(t_{n+1}) = u^{n+1}$.

A.2 Midpoint rule and improved Euler scheme

We first consider the midpoint rule. Following the guidelines given in Sec. 2.3, the numerical quadrature is taken as $c_1 = 1/2$ and $b_1 = 1$. Here, we need to determine only one nontrial test function $w(t)$ with $w(t_{n+1}) = 0$ for a one stage scheme. For the present case, Eq. (9) becomes

$$u^n w(t_n) + h_n U^1 \dot{w}\left(t_{n+\frac{1}{2}}\right) + h_n f\left(U^1, t_{n+\frac{1}{2}}\right) w\left(t_{n+\frac{1}{2}}\right) = 0. \tag{74}$$

By taking $w(t_n) = 1$, the consistent conditions (15) and (16) imply that $w\left(t_{n+\frac{1}{2}}\right) = \frac{1}{2}$ and $\dot{w}\left(t_{n+\frac{1}{2}}\right) = -\frac{1}{h_n}$. This gives rise to a linear function

$$w(t) = -\frac{1}{h_n}(t - t_n) + 1.$$

With this test function, Eq. (74) becomes

$$U^1 = u^n + \frac{h_n}{2} f\left(U^1, t_n + \frac{h_n}{2}\right),$$

which is the same as Eq. (27), while Eq. (28) can be simply attained by assuming $w(t) = 1$. We thus derived the midpoint rule (25) based on the DG finite element method.

The improved Euler scheme correspondes to a two stage Runge-Kutta method

$$y^1 = u^n \tag{75}$$

$$y^2 = u^n + h_n f\left(y^1, t_n\right), \tag{76}$$

$$u^{n+1} = u^n + \frac{h_n}{2}\left[f(y^1, t_n) + f(y^2, t_{n+1})\right] \tag{77}$$

With $s = 2$, the abscissae and weights of the DG method are taken as $c_1 = 0$, $c_2 = 1$, $b_1 = \frac{1}{2}$, and $b_2 = \frac{1}{2}$. We note that since $c_1 = 0$, we have simply $U^1 = U_n(t_n + C_1 h_n) = U_n(t_n) = u^n$, according to the boundary condition. In fact, Eq. (75) can always be trivially satisfied if the abscissae starts from $c_1 = 0$. No test function $w(t)$ is required for calculating U^1.

To compute U^2, we first have $w(t_{n+1}) = 0$ as usual and we take $w(t_n) = 1$ for simplicity. Then, two consistent conditions (15) and (16) can be employed to determine $\dot{w}(t)$ values on abscissae. We thus have $\dot{w}(t_n) = -\frac{1}{h_n}$ and $\dot{w}(t_{n+1}) = -\frac{1}{h_n}$. Therefore, the test function is the same linear polynomial

$$w(t) = -\frac{1}{h_n}(t - t_n) + 1.$$

and U^2 is given as

$$U^2 = u^n + h_n f\left(U^1, t_n\right),$$

which is identical to (76). Finally, Eq. (77) can be simply derived by taking $w(t) = 1$.

A.3 Two stage explicit Runge-Kutta method

We note that last point t_{n+1} is missing in the quadrature for the Scheme RK2-2, so that the current derivation is slightly different that of the Scheme RK2-1. For scheme RK2-2, U^1 is

again given as $U^1 = u^n$. To compute U^2, one needs to fix $w(t) = w_2(t)$. According to avaiable conditions, we have $w(t_n) = 1$, $w\left(t_{n+\frac{2}{3}}\right) = 0$, and $w\left(t_{n+1}\right) = 0$ due to (32) and $\dot{w}(t_n) = -\frac{5}{2h_n}$ and $\dot{w}\left(t_{n+\frac{2}{3}}\right) = -\frac{1}{2h_n}$ due to (15) and (16). Hence, at $t = t_{n+1}$, function value $w(t)$ is zero, but the derivative value is unknown. In the present study, we optimize the unfixed derivative value at t_{n+1} so that the expression for $w(t)$ could be simpler. The optimized test function is given as

$$w(t) = \frac{3}{2h_n^2}(t - t_n)^2 - \frac{5}{2h_n}(t - t_n) + 1.$$

By substituting $w(t)$ into Eq. (9), one attains

$$U^2 = u^n + \frac{2h_n}{3} f(U^1, t_n).$$

With U^1 and U^2, u^{n+1} is updated according to Eq. (12)

$$u^{n+1} = u^n + \frac{h_n}{4} f(U^1, t_n) + \frac{3h_n}{4} f\left(U^2, t_{n+\frac{2}{3}}\right).$$

A.4 Three stage explicit Runge-Kutta method

Following the previous discussions, we have $U^1 = u^n$ first. Here U^2 can be uniquely determined based on the proposed procedure. In particular, we have $w(t_n) = 1$, $w\left(t_{n+\frac{1}{2}}\right) = 0$, and $w(t_{n+1}) = 0$ according to (32) and $\dot{w}(t_{n+1}) = 0$ following from (30). The remaining coefficients can be solved from (15) and (16) to be $\dot{w}(t_n) = -\frac{4}{h_n}$ and $\dot{w}\left(t_{n+\frac{1}{2}}\right) = -\frac{1}{2h_n}$. Consequently, the weight function is

$$w(t) = -\frac{2}{h_n^3}(t - t_n)^3 + \frac{5}{h_n^2}(t - t_n)^2 - \frac{4}{h_n}(t - t_n) + 1.$$

By substituting $w(t)$ into Eq. (9), one attains

$$U^2 = u^n + \frac{h_n}{2} f(U^1, t_n). \tag{78}$$

One degree of freedom is involved in calculating U^3. From the explicit condition (32), we have $w(t_n) = 0$, $w\left(t_{n+\frac{1}{2}}\right) = 1$, and $w(t_{n+1}) = 0$. Now, the second explicit condition (30) is not applicable so that three nodal values of $\dot{w}(t)$ are unknowns. Taking into account the previous results, we assume these three numbers being $\dot{w}(t_n) = -\frac{A}{h_n}$, $\dot{w}\left(t_{n+\frac{1}{2}}\right) = -\frac{B}{h_n}$, and $\dot{w}(t_{n+1}) = -\frac{C}{h_n}$. By plugging in these coefficients into Eqs. (15) and (16), consistent conditions suggest that one can express A and B in terms of C. Thus, the general form of $\dot{w}(t)$ is $\dot{w}(t_n) = \frac{8-C}{h_n}$, $\dot{w}(t_{n+\frac{1}{2}}) = \frac{C-4}{2h_n}$, and $\dot{w}(t_{n+1}) = -\frac{C}{h_n}$ for $C \neq 0$. Thus, the weight function is

$$w(t) = \frac{8 - 2C}{h_n^3}(t - t_n)^3 + \frac{3C - 16}{h_n^2}(t - t_n)^2 + \frac{8 - C}{h_n}(t - t_n).$$

By substituting $w(t)$ into Eq. (9), the general formula for U^3 is

$$U^3 = u^n + \frac{C - 4}{C} h_n f(U^1, t_n) + \frac{4}{C} h_n f\left(U^2, t_{n+\frac{1}{2}}\right). \tag{79}$$

With U^2 and U^3 from Eqs. (78) and (79), the general explicit Runge-Kutta scheme can be given as

$$u^{n+1} = u^n + \frac{h_n}{6} f(U^1, t_n) + \frac{2h_n}{3} f\left(U^2, t_{n+\frac{1}{2}}\right) + \frac{h_n}{6} f(U^3, t_{n+1}).$$

A.5 Four stage explicit Runge-Kutta method

To derive U^2, we set $w^+(t) = 0$ and need to determine $w^-(t)$. According to (32), one has $w^-(t_n) = 1$ and $w^-\left(t_{n+\frac{1}{2}}\right) = 0$, while $\dot{w}^-(t_n) = -\frac{4}{h_n}$ and $\dot{w}^-\left(t_{n+\frac{1}{2}}\right) = -\frac{1}{h_n}$ follow from (15) and (16). Thus, the weight function is taken as

$$w(t) = \begin{cases} -\frac{4}{h_n^3}(t-t_n)^3 + \frac{6}{h_n^2}(t-t_n)^2 - \frac{4}{h_n}(t-t_n) + 1, & \text{if} \quad t \in \left(t_n, t_{n+\frac{1}{2}}\right) \\ 0, & \text{if} \quad t \in \left(t_{n+\frac{1}{2}}, t_{n+1}\right). \end{cases} \tag{80}$$

This weight function gives rise to

$$U^2 = u^n + \frac{h_n}{2} f(U^1, t_n).$$

For U^3, we have first $w^-(t_n) = 0$, $w^-\left(t_{n+\frac{1}{2}}\right) = 1$, $w^+\left(t_{n+\frac{1}{2}}\right) = 0$, $w^+(t_{n+1}) = 0$, and $\dot{w}^+(t_{n+1}) = 0$, based on (30) and (32). One free constant has to be introduced here and let us denote it as C_1. By using (15) and (16), we then have $\dot{w}^-(t_n) = \frac{4}{h_n}$, $\dot{w}^-\left(t_{n+\frac{1}{2}}\right) = \frac{C_1-2}{h_n}$, and $\dot{w}^+\left(t_{n+\frac{1}{2}}\right) = -\frac{C_1}{h_n}$. Thus, the weight function is taken as

$$w(t) = \begin{cases} \frac{4C_1-8}{h_n^3}(t-t_n)^3 - \frac{2C_1}{h_n^2}(t-t_n)^2 + \frac{4}{h_n}(t-t_n), & \text{if} \quad t \in \left(t_n, t_{n+\frac{1}{2}}\right) \\ -\frac{4C_1}{h_n^3}(t-t_n)^3 + \frac{10C_1}{h_n^2}(t-t_n)^2 - \frac{8C_1}{h_n}(t-t_n) + 2C_1, & \text{if} \quad t \in \left(t_{n+\frac{1}{2}}, t_{n+1}\right). \end{cases} \tag{81}$$

This weight function gives rise to

$$U^3 = u^n + \frac{C_1 - 2}{2C_1} h_n f(U^1, t_n) + \frac{1}{C_1} h_n f\left(U^2, t_{n+\frac{1}{2}}\right).$$

For U^4, only nodal values can be fixed $w^-(t_n) = 0$, $w^-\left(t_{n+\frac{1}{2}}\right) = 0$, $w^+\left(t_{n+\frac{1}{2}}\right) = 1$, and $w^+(t_{n+1}) = 0$, due to (32). Two free constants C_2 and C_3 are introduced for nodal derivative values of $w(t)$. By using (15) and (16), we then have $\dot{w}^-(t_n) = \frac{4-C_3}{h_n}$, $\dot{w}^-\left(t_{n+\frac{1}{2}}\right) = \frac{C_2+C_3-2}{h_n}$, $\dot{w}^+\left(t_{n+\frac{1}{2}}\right) = -\frac{C_2}{h_n}$, and $\dot{w}^+(t_{n+1}) = -\frac{C_3}{h_n}$. Consequently, $w^-(t)$ and $w^+(t)$ are taken as

$$\begin{aligned} w^-(t) &= \frac{4C_2+8}{h_n^3}(t-t_n)^3 + \frac{2C_3-2C_2-12}{h_n^2}(t-t_n)^2 + \frac{4-C_3}{h_n}(t-t_n), \\ w^+(t) &= \frac{16-4C_2-4C_3}{h_n^3}(t-t_n)^3 + \frac{10C_2+8C_3-36}{h_n^2}(t-t_n)^2 \\ &\quad + \frac{24-8C_2-5C_3}{h_n}(t-t_n) + 2C_2 + C_3 - 4. \end{aligned} \tag{82}$$

It is interesting to note that C_1 is not involved in the current weight function, while it is involved in the updating equation

$$U^4 = u^n + \left(1 - \frac{2}{C_3} + \frac{2C_2}{C_1C_3}\right) h_n f(U^1, t_n) - \frac{2C_2}{C_1C_3} h_n f\left(U^2, t_{n+\frac{1}{2}}\right) + \frac{2}{C_3} h_n f\left(U^3, t_{n+\frac{1}{2}}\right).$$

Finally, by taking $w^-(t) = w^+(t) = 1$, we have the last step of the Runge-Kutta method

$$u^{n+1} = u^n + \frac{h_n}{6} f(U^1, t_n) + \frac{h_n}{3} f\left(U^2, t^-_{n+\frac{1}{2}}\right) + \frac{h_n}{3} f\left(U^3, t^+_{n+\frac{1}{2}}\right) + \frac{h_n}{6} f(U^4, t_{n+1}). \quad (83)$$

A.6 Two stage symplectic Runge-Kutta method

Since the symplect Runge-Kutta methods are implicit, U^1 is not simply given by the boundary condition. Here to compute U^1, one symplectic condition from (38) shall be imposed, i.e., $a_{11} = \frac{1}{2}b_1$. In addition to that, we have two consistence conditions (15) and (16). Also, the implicit condition (39) suggests $\dot{w}(t_n + c_2h_n) = 0$. Thus, the rest three coefficients can be solved to be $w(t_n + c_1h_n) = \frac{1}{2}$, $w(t_n + c_2h_n) = \frac{1}{2} - \frac{\sqrt{3}}{3}$, and $\dot{w}(t_n + c_1h_n) = -\frac{h_n}{2}$. Consequently, the weight function is

$$w(t) = \frac{\sqrt{3}}{h_n^2}(t - t_n)^2 - \frac{\sqrt{3}+1}{h_n}(t - t_n) + 1.$$

By substituting $w(t)$ into Eq. (9), one attains

$$U^1 = u^n + \frac{h_n}{4} f\left(U^1, t_n + \left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right) h_n\right) + \left(\frac{1}{4} - \frac{\sqrt{3}}{6}\right) h_n f\left(U^2, t_n + (\frac{1}{2} + \frac{\sqrt{3}}{6})h_n\right). \quad (84)$$

We next consider U^2. We first assume four undetermined coefficients to be $w(t_n + c_1h_n) = A$, $w(t_n + c_2h_n) = B$, $\dot{w}(t_n + c_1h_n) = C$, and $\dot{w}(t_n + c_2h_n) = D$. Equation (9) thus becomes

$$0 = u^n + \frac{h_n}{2} CU^1 + \frac{h_n}{2} DU^2 + \frac{h_n}{2} Af(U^1, t_n + c_1h_n) + \frac{h_n}{2} Bf(U^2, t_n + c_2h_n). \quad (85)$$

By substituting (84) into (85), one attains

$$\begin{aligned} -\frac{h_n}{2} DU^2 &= \left(1 + \frac{Ch_n}{2}\right) u^n + \left(\frac{Ah_n}{2} + \frac{Ch_n^2}{8}\right) f(U^1, t_n + c_1h_n) \\ &\quad + \left(\frac{Bh_n}{2} + \left(\frac{1}{8} - \frac{\sqrt{3}}{12}\right) Ch_n^2\right) f(U^2, t_n + c_2h_n). \end{aligned}$$

We now switch to necessary conditions. There are two conditions involved in symplectic condition (38), i.e.,

$$a_{22} = \frac{1}{2}b_2, \quad b_2a_{21} + b_1a_{12} - b_1b_2 = 0. \quad (86)$$

However, these two conditions and two consistence conditions (15) and (16) are not independent. In fact, one can derive (16) from other three in the present context. Therefore, we

also need to impose the implicit condition (39) $C = \dot{w}(t_n + c_1 h_n) = 0$. Then the rest undetermined coefficients can be solved based on (86) and (15): $A = w(t_n + c_1 h_n) = \frac{1}{2} + \frac{\sqrt{3}}{3}$, $B = w(t_n + c_2 h_n) = \frac{1}{2}$, and $D = \dot{w}(t_n + c_2 h_n) = -\frac{2}{h_n}$. Consequently, the test function is

$$w(t) = -\frac{\sqrt{3}}{h_n^2}(t - t_n)^2 + \frac{\sqrt{3}+1}{h_n}(t - t_n) + 1.$$

By substituting $w(t)$ into Eq. (9), one attains

$$U^2 = u^n + \left(\frac{1}{4} + \frac{\sqrt{3}}{6}\right) h_n f\left(U^1, t_n + \left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right) h_n\right) + \frac{h_n}{4} f\left(U^2, t_n + \left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right) h_n\right). \tag{87}$$

After U^1 and U^2 being computed, the symplectic Runge-Kutta scheme is completed by taking $w(t) = 1$ in Eq. (9)

$$u^{n+1} = u^n + \frac{h_n}{2} f\left(U^1, t_n + \left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right) h_n\right) + \frac{h_n}{2} f\left(U^2, t_n + \left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right) h_n\right). \tag{88}$$

A.7 Three stage symplectic Runge-Kutta method

Similarly, we first fix $w(t_n) = 1$ and $w(t_{n+1}) = 0$. Moreover, the implicit condition (39) indicates that the derivative $\dot{w}(t)$ will be zero at two of three Gaussian nodes. This leaves one undetermined coefficient in terms of derivative. Together with three nodal values of $w(t)$, we have a total of four degrees of freedom in each stage.

For the purpose of computing U^1, only one diagonal condition shall be used in the symplectic condition (38), $a_{11} = \frac{1}{2} b_1$. Together with two consistent conditions (15) and (16), one degree of freedom is left. We let $A_1 = w(t_n + c_1 h_n)$, $A_2 = w(t_n + c_2 h_n)$, $A_3 = w(t_n + c_3 h_n)$, and $B = \dot{w}(t_n + c_1 h_n)$. Equation (9) now becomes

$$0 = u^n + \frac{5h_n}{18} B U^1 + \frac{5h_n}{18} A_1 f(U^1, t_n + c_1 h_n) + \frac{4h_n}{9} A_2 f(U^2, t_n + c_2 h_n) + \frac{5h_n}{18} A_3 f(U^3, t_n + c_3 h_n). \tag{89}$$

The consistent condition (15) can be satisfied if $\frac{5h_n B}{18} = -1$ or $B = -\frac{18}{5h_n}$. By enforcing $a_{11} = \frac{1}{2} b_1$, we have $\frac{5h_n A_1}{18} = \frac{1}{2} \cdot \frac{5h_n}{18}$. This gives rise to $A_1 = \frac{1}{2}$. Then, the consistent condition (16) implies $16A_2 + 10A_3 = 13 - 18\sqrt{\frac{3}{5}}$. We treat A_2 as the remaining degree of freedom and represent A_3 in terms of it, i.e., $A_3 = \frac{1}{10}\left(13 - 18\sqrt{\frac{3}{5}} - 16A_2\right)$. Consequently, Eq. (89) reduces to

$$U^1 = u^n + \frac{5h_n}{36} f(U^1, t_n + c_1 h_n) + \frac{4h_n}{9} A_2 f(U^2, t_n + c_2 h_n) + \frac{5h_n}{18} \frac{13 - 18\sqrt{\frac{3}{5}} - 16A_2}{10} f(U^3, t_n + c_3 h_n). \tag{90}$$

We now continue on U^2 with the assumption that A_2 is still unfixed. Now, there are two constrains involved in symplectic condition (38)

$$a_{22} = \frac{1}{2} b_2, \quad b_2 a_{21} + b_1 a_{12} - b_1 b_2 = 0. \tag{91}$$

Thus, four unknowns at this stage are uniquely determined from these two constrains and two consistent conditions. The unknown A_2 is passed on to the next stage. We let $C_1 = w(t_n + c_1 h_n)$, $C_2 = w(t_n + c_2 h_n)$, $C_3 = w(t_n + c_3 h_n)$, and $D = \dot{w}(t_n + c_2 h_n)$. Similarly, we have first

$$u^n + \frac{4h_n}{9} DU^2 + \frac{5h_n}{18} C_1 f(U^1, t_n + c_1 h_n) + \frac{4h_n}{9} C_2 f(U^2, t_n + c_2 h_n) + \frac{5h_n}{18} C_3 f(U^3, t_n + c_3 h_n) = 0. \tag{92}$$

The consistent condition (15) can be satisfied if $\frac{4h_n D}{9} = -1$ or $D = -\frac{9}{4h_n}$. By enforcing $a_{22} = \frac{1}{2} b_2$, we have $\frac{4h_n C_2}{9} = \frac{1}{2} \cdot \frac{4h_n}{9}$. This gives rise to $C_2 = \frac{1}{2}$. Then, the consistent condition (16) implies $\frac{5h_n}{18} C_1 + \frac{4h_n}{18} + \frac{5h_n}{18} C_3 = \frac{h_n}{2}$, which reduces to $C_1 + C_3 = 1$. The other symplectic condition $b_2 a_{21} + b_1 a_{12} - b_1 b_2 = 0$ becomes

$$\frac{5h_n}{18} \frac{4h_n}{9} A_2 + \frac{4h_n}{9} \frac{5h_n}{18} C_1 - \frac{5h_n}{18} \frac{4h_n}{9} = 0.$$

Thus, we have simply $A_2 + C_1 = 1$. Therefore, both C_1 and C_3 can be solved in terms of A_2: $C_1 = 1 - A_2$ and $C_3 = A_2$. Then, the U^2 is given as

$$U^2 = u^n + \frac{5h_n}{18}(1 - A_2) f(U^1, t_n + c_1 h_n) + \frac{4h_n}{18} f(U^2, t_n + c_2 h_n) + \frac{5h_n}{18} A_2 f(U^3, t_n + c_3 h_n). \tag{93}$$

Finally, we consider U^3. The undetermined coefficients of this stage are $E_1 = w(t_n + c_1 h_n)$, $E_2 = w(t_n + c_2 h_n)$, $E_3 = w(t_n + c_3 h_n)$, and $F = \dot{w}(t_n + c_3 h_n)$. Together with the left degree of freedom A_2 from the previous stages, we have totally five unknowns. Now the symplectic condition introduces three conditions:

$$a_{33} = \frac{1}{2} b_3, \quad b_3 a_{31} + b_1 a_{13} - b_1 b_3 = 0, \quad b_3 a_{32} + b_2 a_{23} - b_2 b_3 = 0. \tag{94}$$

Together with two consistent conditions, it seems that one can uniquely determine five unknowns. However, we show that out of these five conditions, only four are independent, so that A_2 is still remained as one degree of freedom. We have first

$$0 = u^n + \frac{5h_n}{18} FU^3 + \frac{5h_n}{18} E_1 f(U^1, t_n + c_1 h_n) + \frac{4h_n}{9} E_2 f(U^2, t_n + c_2 h_n) + \frac{5h_n}{18} E_3 f(U^3, t_n + c_3 h_n). \tag{95}$$

The consistent condition (15) can be satisfied if $\frac{5h_n F}{18} = -1$ or $F = -\frac{18}{5h_n}$. By enforcing $a_{33} = \frac{1}{2} b_3$, we have $\frac{5h_n E_3}{18} = \frac{1}{2} \cdot \frac{5h_n}{18}$. This gives rise to $E_3 = \frac{1}{2}$. The symplectic condition $b_2 a_{21} + b_1 a_{12} - b_1 b_2 = 0$ implies

$$\frac{5h_n}{18} \cdot \frac{5h_n}{18} \cdot \frac{13 - 18\sqrt{\frac{3}{5}} - 16A_2}{10} + \frac{5h_n}{18} \cdot E_1 \cdot \frac{5h_n}{18} - \frac{5h_n}{18} \cdot \frac{5h_n}{18} = 0.$$

From which, one solves $E_1 = \frac{16A_2 + 18\sqrt{\frac{3}{5}} - 3}{10}$. The last symplectic condition $b_3 a_{32} + b_2 a_{23} - b_2 b_3 = 0$ suggests

$$\frac{4h_n}{9} \cdot \frac{5h_n}{18} \cdot A_2 + \frac{5h_n}{18} \cdot E_2 \cdot \frac{4h_n}{9} - \frac{5h_n}{18} \cdot \frac{4h_n}{9} = 0.$$

Thus, we have $E_2 = 1 - A_2$. Finally, the consistent condition (16) is given as

$$\frac{5h_n}{18} \cdot \frac{16A_2 + 18\sqrt{\frac{3}{5}} - 3}{10} + \frac{4h_n}{9}(1 - A_2) + \frac{5h_n}{36} = \left(\frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}}\right) h_n.$$

It can be verified that this equation is trivially true for any A_2 value. Therefore, A_2 is the free parameter involved in all stages of current DG version of symplectic Runge-Kutta scheme. Now, U^3 is computed as

$$U^3 = u^n + \frac{5h_n}{18}\frac{16A_2 + 18\sqrt{\frac{3}{5}} - 3}{10} f(U^1, t_n + c_1 h_n) + \frac{4h_n}{9}(1 - A_2) f(U^2, t_n + c_2 h_n) + \frac{5h_n}{36} f(U^3, t_n + c_3 h_n). \tag{96}$$

After U^1, U^2, and U^3 being computed, we can take $w(t) = 1$ as usual to complete the derivation of the three stage symplectic Runge-Kutta scheme. For simplicity, we denote A_2 to be A in the final scheme.

A.8 Reducible Runge-Kutta scheme

We consider an explicit two stage reducible Runge-Kutta method to illustrate this issue.

Scheme RRK

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

The present quadrature rule $c_1 = 0$, $c_2 = \frac{1}{2}$, $b_1 = 0$, and $b_2 = 1$ is an incomplete one because the first node c_1 is of no use in numerical approximation. Let us consider the DG derivation. First, as usual we have $U^1 = u^n$ since $c_1 = 0$. To computer U^2, we would like to explore all possible degrees of freedom. Thus, we consider undetermined coefficients not only for $t_n + c_1 h_n$ and $t_n + c_2 h_n$, but also the end point t_{n+1}. We let $A = w(t_n + c_1 h_n)$, $B = w(t_n + c_2 h_n)$, $C = w(t_{n+1})$, $D = \dot{w}(t_n + c_1 h_n)$, $E = \dot{w}(t_n + c_2 h_n)$, and $F = \dot{w}(t_{n+1})$. Then Eq. (9) becomes

$$Cu^{n+1} = Au^n + Eh_n U^2 + Bh_n f(U^2, t_n + C_2 h_n). \tag{97}$$

Obviously, no matter what values are used for undetermined coefficients, Eq. (97) can not be rewritten into the form

$$U^2 = u^n + \frac{h_n}{2} f(U^1, t_n), \tag{98}$$

simply because the term $f(U^1, t_n)$ of Eq. (98) is missing in Eq. (97), due to the incomplete quadrature. Similar difficulties have been found in deriving other reducible Runge-Kutta schemes of different order.